Riemannian optimal system identification method of linear continuous-time systems with symmetry
Kazuhiro Sato, Hiroyuki Sato, and Tobias Damm

Abstract—This paper develops identification methods of linear continuous-time systems with symmetry. Such systems are important, because they include various examples such as electrical network systems, multi-agent network systems, and temperature dynamics in buildings. To this end, we formulate a novel prediction error problem of the corresponding discrete-time systems. More specifically, the problem is the minimization problem of the squared sum of differences between the true and model outputs on the product manifold of the manifold of the symmetric positive definite matrices and two Euclidean spaces. We also formulate the minimization problem on the quotient manifold under a specified group action by the orthogonal group, in order to reduce the search dimension of the former problem. We propose Riemannian conjugate gradient (CG) methods for both problems. It is shown that the only difference between the methods is vector transports in the algorithms. The vector transport for the latter problem is the projection onto the horizontal space which is the subspace of a tangent space of the product manifold. The projection is obtained by using the skew-symmetric solution to a linear matrix equation. The vector transport for the former problem is the best among the other methods such as the product manifold. We demonstrate that the CG method for the latter problem is considerably better than randomly choosing a point on the product manifold. We prove that there exists a unique skew-symmetric solution to the equation under a mild assumption. Furthermore, we provide two proofs that a Riemannian metric on the quotient manifold can be endowed by using the metric on the product manifold.

We suggest that we choose initial points in the proposed methods by using the result of the modified ordinary Multivariable Output-Error State-space (MOESP) method. The initial point is considerably better than randomly choosing a point on the product manifold. We demonstrate that the CG method for the latter problem is the best among the other methods such as the ordinary MOESP method, N4SID method, the proposed modified MOESP method, the steepest descent method for both problems, and the CG method for the former problem.

Index Terms—Riemannian optimization, symmetry, system identification

I. INTRODUCTION

Many important systems such as electrical network systems [1], multi-agent network systems [2], [3], and temperature dynamics in buildings [4], [5] can be expressed as

\[
\begin{align*}
\dot{x}(t) &= Fx(t) + Gu(t), \\
\dot{y}(t) &= Cx(t),
\end{align*}
\]

where \(x(t) \in \mathbb{R}^n\), \(u(t) \in \mathbb{R}^m\), and \(y(t) \in \mathbb{R}^p\) are the state, input, and output of this system, respectively, and \(F \in Sym(n)\), \(G \in \mathbb{R}^{n \times m}\), and \(C \in \mathbb{R}^{p \times n}\) are constant matrices to be identified. Because the matrix \(F\) is symmetric, we call system (1) a linear continuous-time system with symmetry. Although system (1) includes many important systems, system identification methods of system (1) have not been developed so far. More specifically, although there are many identification methods such as prediction error methods [6]–[9] and subspace identification methods [10]–[16] for discrete-time systems, and ones for continuous-time systems [17]–[22], it is difficult to identify the matrix \(F\) to be symmetric from \(N+1\) input-output data pairs \(\{(u(0), y(0)), (u(h), y(h)), \ldots, (u(Nh), y(Nh))\}\) with the sampling interval \(h\), where \(y(0), y(h), \ldots, y(Nh)\) are noisy data observed from the true system which is different from system (1). This is because no one has developed a system identification method of the corresponding discrete-time system

\[
\begin{align*}
\dot{x}_{k+1} &= A\hat{x}_k + B\hat{u}_k, \\
\hat{y}_k &= C\hat{x}_k,
\end{align*}
\]

where \(\dot{x}_{k} := \hat{x}(kh)\), \(\hat{u}_{k} := \hat{u}(kh)\), \(\hat{y}_{k} := \hat{y}(kh)\), and

\[
\begin{align*}
A := \exp(Fh) &\in Sym_+(n), \\
B := \left(\int_0^h \exp(Ft)dt\right)G.
\end{align*}
\]

That is, the existing methods in [6]–[16] for identifying the triple \((A, B, C)\) do not provide a symmetric positive definite matrix \(A\).

For this reason, we develop a method for identifying

\[
\Theta := (A, B, C) \in \Theta := Sym_+(n) \times \mathbb{R}^{n \times m} \times \mathbb{R}^{p \times n}.
\]

That is, we identify the matrix \(A\) to be symmetric positive definite. If this is achieved, we can also obtain the matrices \(F\) and \(G\) by

\[
F = \log A/h, \\
G = \left(\int_0^h \exp(Ft)dt\right)^{-1} B,
\]

respectively. Here, the matrix \(F\) is symmetric and is uniquely determined, because the map \(\exp : Sym(n) \to Sym_+(n)\) is bijective [23]. To this end, we consider the following problem.
Problem 1: Suppose that input-output data \{\((u_0, y_0), (u_1, y_1), \ldots, (u_N, y_N)\)\} and the state dimension \(n\) are given. Then, solve
\[
\begin{align*}
\text{minimize} & \quad f(\Theta) := ||e(\Theta)||^2_2 \\
\text{subject to} & \quad \Theta \in M.
\end{align*}
\]

Here,
\[
e(\Theta) := (y_1 - \hat{y}_1(\Theta), y_2 - \hat{y}_2(\Theta), \ldots, y_N - \hat{y}_N(\Theta)),
\]
and \(\hat{y}_k(\Theta)\) is \(\hat{y}_k\) obtained by equating \(\hat{u}_k\) of (2) to \(u_k\). That is, we develop a novel prediction error method on the manifold \(M\). Note that \(\hat{y}_k(\Theta)\) is different from \(y_k\), because \(y_k\) is obtained by observing the output of the true system. That is, system (2) is not the true system but is a mathematical model of that. Furthermore, note that it is inadequate to consider a vector parameter \(\theta\) obtained by regarding all the elements of the system matrices as
\[
\theta := \begin{pmatrix} \text{vec}(A) \\ \text{vec}(B) \\ \text{vec}(C) \end{pmatrix},
\]
although [6], [2] used such vector parameter \(\theta\), where \(\text{vec}\) denotes the usual vec-operator. This is because this parameterization does not reflect the symmetric positive definiteness of the matrix \(A\). That is, it is difficult to develop an algorithm for solving an equivalent problem with Problem 1 using the vector parameter \(\theta\) instead of the triple \(\Theta\) of the matrix parameters. This means that we cannot apply the results of [6], [9] for solving Problem 1.

We also consider another problem, because it is possible to reduce the dimension of the problem of minimizing \(||e(\Theta)||^2_2\) under the assumption that the initial state \(\hat{x}_0\) equals zero. This is because \(\Theta\) and
\[
U \circ \Theta := (U^T A U, U^T B, C U)
\]
realize input-output equivalent systems for any \(U \in O(n)\), and \(U \circ \Theta \in M\), where \(\circ\) denotes a group action of \(O(n)\) on \(M\). Thus, they attain the same value of the prediction error on the manifold \(M\), i.e., \(||e(\Theta)||^2_2 = ||e(U \circ \Theta)||^2_2\). This leads to the idea that we equate \(\Theta\) with \(U \circ \Theta\) to reduce the dimension of the problem of minimizing \(||e(\Theta)||^2_2\). To this end, we endow \(M\) with an equivalence relation \(\sim\), where \(\Theta_1 \sim \Theta_2\) if and only if there exists \(U \in O(n)\) such that \(\Theta_2 = U \circ \Theta_1\). Defining the equivalence class \([\Theta]\) by \([\Theta] := \{\Theta_1 \in M | \Theta_1 \sim \Theta\}\), we can equate \(\Theta\) with any \(\Theta_1\) that is equivalent to \(\Theta\), and thus Problem 1 is transformed into the following problem.

Problem 2: Suppose that input-output data \{\((u_0, y_0), (u_1, y_1), \ldots, (u_N, y_N)\)\} and the state dimension \(n\) are given. Then, solve
\[
\begin{align*}
\text{minimize} & \quad g([\Theta]) := ||e([\Theta])||^2_2 \\
\text{subject to} & \quad [\Theta] \in M/O(n) := \{[\Theta]|\Theta \in M\}.
\end{align*}
\]

That is, we also develop a prediction error method on the quotient manifold \(M/O(n)\). Note that this development is different from that in [8], which considered a group action of the general linear group \(GL(n)\) on a manifold instead of \(O(n)\). It is not adequate to use the action in [8] for our problem, because the action does not preserve the symmetric positive definiteness of the matrix \(A\) in general. For this reason, we consider the group action of \(O(n)\) on the manifold \(M\).

The contributions of this paper are summarized as follows.
1) We propose Riemannian conjugate gradient (CG) methods for solving Problems 1 and 2. The method for Problem 1 is developed by introducing the Riemannian metric on \(M\) as
\[
\langle (\xi_1, \eta_1, \zeta_1), (\xi_2, \eta_2, \zeta_2) \rangle_\Theta := \text{tr}(A^{-1} \xi_1 A^{-1} \xi_2) + \text{tr}(\eta_1^T \eta_2) + \text{tr}(\zeta_1^T \zeta_2)
\]
for \((\xi_1, \eta_1, \zeta_1), (\xi_2, \eta_2, \zeta_2) \in T_\Theta M\). To develop the CG method for Problem 2, we provide two proofs that a Riemannian metric on \(M/O(n)\) can be endowed by
\[
\langle \tilde{\xi}, \tilde{\zeta} \rangle_{[\Theta]} := \langle \tilde{\xi}_\Theta, \tilde{\zeta}_\Theta \rangle_\Theta,
\]
where \(\xi, \zeta \in T_{[\Theta]}(M/O(n)), \Theta \in \pi^{-1}([\Theta]), \) and \(\tilde{\xi}_\Theta\) and \(\tilde{\zeta}_\Theta\) are the horizontal lifts of \(\xi\) and \(\zeta\) at \(\Theta\), respectively, and \(\pi : M \to M/O(n)\) denotes the canonical projection. In the first proof, we use a property derived from a specific expression of the horizontal space in \(T_\Theta M\) with respect to metric (7). In the second proof, we do not use the property. It is also shown that the only difference between the algorithms for Problems 1 and 2 is vector transports in the algorithms, and the vector transport for Problem 2 is the projection onto the horizontal space. The projection is obtained by using the skew-symmetric solution to a linear matrix equation. We prove that there exists a unique skew-symmetric solution to the equation under a mild assumption.
2) We propose a technique for choosing initial points of proposed algorithms for solving Problems 1 and 2 based on the ordinary Multivariable Output-Error State-space (MOESP) method [15]. More precisely, because the ordinary MOESP method cannot provide \(A \in \text{Sym}_n(n)\) in general, we modify the method in such a way that the matrix \(A\) belongs to \(\text{Sym}_n(n)\). Numerical experiments demonstrate that the proposed modified ordinary MOESP method provides a considerably better initial point in the algorithms for solving Problems 1 and 2 than randomly choosing one from the manifold \(M\).
3) We demonstrate that the CG method for Problem 2 is the best among other methods. The other methods are the ordinary MOESP method, subspace state space system identification (N4SID) method [13], the proposed modified MOESP method, the steepest descent (SD) method for Problems 1 and 2, and CG method for Problem 1. It is illustrated that our proposed optimization methods improve the result of the proposed modified ordinary MOESP method.

The remainder of this paper is organized as follows. In Section III, we discuss Riemannian geometries of Problems 1 and 2. In Section IV, we propose optimization algorithms for solving Problems 1 and 2, and a technique for choosing an initial point in the algorithms. In Section V, we numerically
compare the ordinary MOESP method, N4SID, and our proposed methods. Finally, conclusions are presented in Section VI.

Notation: The sets of real and complex numbers are denoted by \( \mathbb{R} \) and \( \mathbb{C} \), respectively. The symbol \( \sqrt{\cdot} \) denotes the imaginary unit. The symbols \( \text{Sym}(n) \) and \( \text{Skew}(n) \) denote the sets of symmetric matrices and skew-symmetric matrices in \( \mathbb{R}^{n \times n} \), respectively. The manifold of symmetric positive definite matrices in \( \mathbb{R}^{n \times n} \) is denoted by \( \text{Sym}_+ (n) \). The symbol \( O(n) \) denotes the orthogonal group in \( \mathbb{R}^{n \times n} \). The tangent space at \( p \) on a manifold \( M \) is denoted by \( T_pM \). The identity matrix of size \( n \) is denoted by \( I_n \).

Given a vector \( v \in \mathbb{R}^n \), \( \|v\|_2 \) denotes the Euclidean norm, i.e.,
\[
\|v\|_2 := \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}.
\]

Given a matrix \( A \in \mathbb{R}^{n \times m} \), \( \|A\|_F \) denotes the Frobenius norm, i.e.,
\[
\|A\|_F := \sqrt{\text{tr}(AT A)},
\]
where the superscript \( T \) denotes the transposition, \( \lambda(A) \) denotes the set of eigenvalues of \( A \), \( \text{tr}(A) \) denotes the trace of \( A \), i.e., the sum of the diagonal elements of \( A \), and \( \text{sk}(A) \) denotes the skew-symmetric part of \( A \), i.e., \( \text{sk}(A) = \frac{A-A^T}{2} \).

Given a matrix \( A \in \mathbb{R}^{n \times m} \), \( A^\dagger \) denotes the pseudo-inverse of \( A \). For any matrices \( A \in \mathbb{R}^{n \times m} \) and \( B \in \mathbb{R}^{p \times q} \), \( A \otimes B \) denotes the Kronecker product of \( A \) and \( B \), i.e.,
\[
A \otimes B := \begin{pmatrix}
a_{11}B & a_{12}B & \cdots & a_{1m}B \\
a_{21}B & a_{22}B & \cdots & a_{2m}B \\
\vdots & \vdots & \ddots & \vdots \\
a_{n1}B & a_{n2}B & \cdots & a_{nm}B
\end{pmatrix} \in \mathbb{R}^{np \times mq}.
\]

Given a smooth real-valued function \( f \) on a finite dimensional Euclidean space \( E \), the Fréchet derivative \( Df(x) : E \to \mathbb{R} \) of \( f \) at \( x \in E \) is defined as a linear operator such that
\[
\lim_{\xi \to 0} \frac{\|f(x + \xi) - f(x) - Df(x)[\xi]\|_2}{\|\xi\|_2} = 0,
\]
where \( Df(x)[\xi] \) is called the directional derivative of \( f \) at \( x \) in the direction \( \xi \).

II. RIEMANNIAN GEOMETRIES OF PROBLEMS 1 AND 2

A. Riemannian geometry of Problem 1

In first-order optimization algorithms such as the steepest descent and conjugate gradient methods on the manifold \( M \), we need the Riemannian gradient of the objective function \( f \). To this end, we have to introduce a Riemannian metric into \( M \). In this paper, we define Riemannian metric \( \langle \cdot, \cdot \rangle \) on \( M \).

Let \( \bar{f} \) denote the extension of the objective function \( f \) to the ambient Euclidean space \( \mathbb{R}^{n \times n} \times \mathbb{R}^{n \times m} \times \mathbb{R}^{p \times n} \). Then, the Euclidean gradient of \( f \) at \( \Theta \) is given by
\[
\nabla \bar{f}(\Theta) = (G_A, G_B, G_C),
\]
where
\[
G_A := -2 \sum_{k=0}^{N} \sum_{i=0}^{k-1} A^{k-i-1}C^T(y_k - \hat{y}_k(\Theta)) \tilde{x}_i^T, \\
G_B := -2 \sum_{k=1}^{N} \sum_{i=0}^{k-1} A^{k-i-1}C^T(y_k - \hat{y}_k(\Theta)) u_i^T, \\
G_C := -2 \sum_{k=1}^{N} (y_k - \hat{y}_k(\Theta)) \tilde{x}_k^T.
\]

In \([8]\), we can find a detailed derivation for a more complicated system. Because we introduced Riemannian metric \( \langle \cdot, \cdot \rangle \), we obtain the Riemannian gradient
\[
\nabla f(\Theta) = (\text{Asym}(G_A)A, G_B, G_C).
\]

For a detailed explanation, see \([24]\).

B. Riemannian geometry of Problem 2

In Section IV, we need a vector transport on the manifold \( M/O(n) \) to develop a Riemannian conjugate gradient method. To this end, we use the orthogonal projection \( P_{\Theta}^h \) onto the horizontal space \( \mathcal{H}_\Theta \) that is the orthogonal complement of the vertical space \( \mathcal{V}_\Theta := T_\Theta \pi^{-1}(\{\Theta\}) \) in \( T_\Theta M \) with respect to metric \( \langle \cdot, \cdot \rangle \). As mentioned in Section I, the map \( \pi : M \to M/O(n) \) denotes the canonical projection, i.e., \( \pi(\Theta) = [\Theta] \) for any \( \Theta \in M \).

In order to derive \( P_{\Theta}^h \), we need to explicitly describe the vertical space \( \mathcal{V}_\Theta \) and the horizontal space \( \mathcal{H}_\Theta \). First, we specify \( \mathcal{V}_\Theta \). Consider any curve \( \Theta(t) \) on \( O(n) \) with \( \Theta(0) = \Theta \) that is expressed as
\[
\Theta(t) = (U^T(t)AU(t), U^T(t)B, CU(t)),
\]
where \( U(t) \) denotes a curve on \( O(n) \) with \( U(0) = I_n \). Differentiating the both sides with respect to \( t \), we obtain that
\[
\dot{\Theta}(0) = (U^T(0)A + A\dot{U}(0), \dot{U}^T(0)B, C\dot{U}(0)),
\]
where \( \dot{U}(0) \in T_{I_n}O(n) \cong \text{Skew}(n) \). Thus, we have that
\[
\mathcal{V}_\Theta = \{(-U'A + AU', -U'B, CU')|U' \in \text{Skew}(n)\}.
\]

Next, we characterize the horizontal space \( \mathcal{H}_\Theta \). Let \( (A', B', C') \in \mathcal{H}_\Theta \). That is,
\[
((-U'A + AU', -U'B, CU'))_\Theta = 0
\]
for all \( U' \in \text{Skew}(n) \). This means that
\[
\text{tr}(U'(2A'A^{-1} + BB'^T + C'^TC')) = 0.
\]
Because \( U' \in \text{Skew}(n) \) is arbitrary, we conclude that
\[
2A'A^{-1} + BB'^T + C'^TC' \in \text{Sym}(n).
\]
That is,
\[
\text{sk}(2A'A^{-1} + BB'^T + C'^TC') = 0.
\]
Thus,
\[
\mathcal{H}_\Theta \subset \{(A', B', C') | \text{sk}(2A'A^{-1} + BB'^T + C'^TC') = 0\}.
\]
Conversely, if \((A', B', C') \in \{(A', B', C') : \operatorname{sk}(2A'A^{-1} + BB'T + CT'C') = 0\}\), we obtain that \((A', B', C') \in \mathcal{H}_\Theta\), because (14) holds. Hence, we have that
\[
\mathcal{H}_\Theta = \{(A', B', C') : \operatorname{sk}(2A'A^{-1} + BB'T + CT'C') = 0\}.
\] (15)

We are in a position to describe the orthogonal projection \(P^h_{\mathcal{H}}\) onto the horizontal space \(\mathcal{H}_\Theta\).

**Theorem 1:** The orthogonal projection \(P^h_{\mathcal{H}}\) onto \(\mathcal{H}_\Theta\) is given by
\[
P^h_{\mathcal{H}}(\eta) = \eta + (XA - AX, XB, -CX),
\] where \(\eta = (a, b, c) \in T_0M\), and \(X\) is the skew-symmetric solution to the linear matrix equation
\[
L_1(X) + 2L_0(X) + \beta = 0,
\] where the linear matrix mappings \(L_0, L_1 : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n}\) are defined by
\[
L_0(X) := AXA^{-1} + A^{-1}XA - 2X,
L_1(X) := (BB'T + CT'C)X + (BB'T + CT'C),
\] respectively, and \(\beta := 2\operatorname{sk}(2A'A^{-1} + bb'H'T + cc'T').\)

**Proof:** Because \(T_0M = \mathcal{V}_\Theta \oplus \mathcal{H}_\Theta\), \(\eta\) can be uniquely decomposed into
\[
\eta = \eta^s + \eta^h, \quad \eta^s \in \mathcal{V}_\Theta, \quad \eta^h \in \mathcal{H}_\Theta.
\]
Since \(\eta^s \in \mathcal{V}_\Theta\), there exists \(X \in \text{Skew}(n)\) such that
\[
\eta^s = (XA + AX, XB, -CX).
\]
Thus, \(\eta^h\) can be described as
\[
\eta^h = (a + XA - AX, b + XB, c - CX).
\]
Because \(\eta^h \in \mathcal{H}_\Theta\), we obtain that
\[
\operatorname{sk}(2(a + XA - AX)A^{-1} + B(b + XB)T + CT(c - CX)) = 0.
\]
It follows from this equation that (17) holds, because \(a^T = a\) and \(X^T = -X\).

We can guarantee that there exists a unique solution \(X \in \text{Skew}(n)\) to (17) under the assumption
\[
\dim(\ker(\lambda I_n - A) \cap \ker B^T \cap \ker C) \leq 1 \quad \text{for all } \lambda \in \mathbb{R}.
\] (18)
Assumption (18) holds if the matrix \(A\) has only simple eigenvalues, because then we have \(\dim(\ker(\lambda I_n - A)) \leq 1\) for all \(\lambda \in \mathbb{R}\). Furthermore, if \((A, C)\) is observable, i.e.,
\[
\operatorname{rank}(\lambda I_n - A - C) = n \Leftrightarrow \ker(\lambda I_n - A) \cap \ker C = \{0\}
\]
for all \(\lambda \in \mathbb{C}\), then (18) holds. Analogously, the controllability of \((A, B)\), i.e.,
\[
\operatorname{rank}(\lambda I_n - A - B) = \operatorname{rank}(\lambda I_n - A - B^T) = n,
\]
also implies (18).

**Theorem 2:** Assume (18), and let \(\mathcal{L} := \mathcal{L}_1 + 2\mathcal{L}_0\). Then, \(\ker \mathcal{L} = \ker \mathcal{L}_1 \cap \ker \mathcal{L}_0 \subset \ker \mathcal{L}_0 \subset \text{Sym}(n)\). In particular, \(\mathcal{L} : \text{Skew}(n) \rightarrow \text{Skew}(n)\) is an automorphism. That is, for any \(Y \in \text{Skew}(n)\), there exists a unique \(X \in \text{Skew}(n)\) with \(\mathcal{L}(X) = Y\).

**Proof:** Using the Kronecker product and vec-operator, the operators \(\mathcal{L}_0\) and \(\mathcal{L}_1\) have the matrix representations
\[
K_0 = A^{-1} \otimes A + A \otimes A^{-1} - 2I_n, \quad K_1 = I_n \otimes (BB'T + CT'C) + (BB'T + CT'C) \otimes I_n.
\]
Both are symmetric, and \(K_1\) is positive semidefinite. Thus, \(\mathcal{L}_1 \geq 0\). Note also that both summands of \(K_1\) and thus of \(\mathcal{L}_1\) are positive semidefinite, whence
\[
\mathcal{L}_1(X) = 0 \Rightarrow (BB'T + CT'C)X = 0.
\] (19)
If \(\lambda_j, \lambda_k \in \lambda(A)\) with corresponding orthonormal eigenvectors \(v_j, v_k\), then
\[
\mathcal{L}_0(v_jv_k^T) = \mu_{jk}v_jv_k^T,
\]
where \(\mu_{jk} := \frac{(\lambda_j - \lambda_k)^2}{\lambda_j \lambda_k}\). From the \(n\) orthonormal eigenvectors \(v_j, j = 1, 2, \ldots, n\), of the matrix \(A\), we thus obtain \(n^2\) orthonormal eigenvectors \(v_jv_k^T, j, k = 1, 2, \ldots, n\), of the linear matrix mapping. Because \(\mu_{jk} \geq 0\) for all \(j, k\), it follows that \(\mathcal{L}_0 \geq 0\). Together with \(\mathcal{L}_1 \geq 0\), this implies that
\[
\ker \mathcal{L} = \ker \mathcal{L}_1 \cap \ker \mathcal{L}_0
\] (20)
as shown in [26]. Moreover, the kernel of \(\mathcal{L}_0\) is spanned by the matrices \(v_jv_k^T + v_kv_j^T\) and \(v_jv_k^T - v_kv_j^T\) with \(\lambda_j = \lambda_k, j, k = 1, 2, \ldots, n\). That is,
\[
\ker \mathcal{L}_0 \cap \text{Skew}(n) = \operatorname{span}\{v_jv_k^T - v_kv_j^T | \lambda_j = \lambda_k\}.
\]
The matrix \(A\) can be expressed as
\[
A = V \operatorname{diag}(\lambda_{n_1}I_{n_1}, \lambda_{n_2}I_{n_2}, \ldots, \lambda_{n_n}I_{n_n})V^T,
\]
where \(n_1 + n_2 + \cdots + n_l = n\), \(\lambda_1 = \cdots = \lambda_{n_1} < \lambda_{n_1+1} = \cdots = \lambda_{n_1+n_2} < \cdots < \lambda_{n_1+n_2+\cdots+n_{l-1}+1} = \cdots = \lambda_n\), and after suitable ordering and partitioning,
\[
V = (V_1 \cdots V_l) = (v_1 \cdots v_n)
\]
is orthogonal with \(\text{Im} V_j = \ker(\lambda_{n_j}I_{n_j} - A)\). We thus obtain that
\[
\ker \mathcal{L}_0 \cap \text{Skew}(n) = \{V \operatorname{diag}(S_1, S_2, \ldots, S_l)V^T | S_j \in \text{Skew}(n_j)\}.
\]
To see this, note that the right hand side is the linear subspace of \(\text{Skew}(n)\), spanned by
\[
v_jv_k^T - v_kv_j^T = (e_je_k^T - e_ke_j^T)V^T,
\]
where \(\lambda_j = \lambda_k\) and \(e_j\) is the \(j\)-th unit vector in \(\mathbb{R}^n\). Thus, it follows from (19) and (20) that \(U \in \ker \mathcal{L} \cap \text{Skew}(n)\) implies \(U = V \operatorname{diag}(S_1, S_2, \ldots, S_l)V^T\) with \((BB'T + CT'C)U = 0\). In particular, we have that
\[
\begin{align*}
0 &= B'TUV_j = B'TV_jS_j \\
0 &= CV_j = CV_jS_j
\end{align*}
\]
for \(j = 1, 2, \ldots, l\), and thus
\[
\ker(\lambda_{n_j}I_{n_j} - A) \cap \ker B^T \cap \ker C \supset \text{Im} (V_jS_j).
\]
Therefore,
\[
\dim(\ker(\lambda_{n_j}I_{n_j} - A) \cap \ker B^T \cap \ker C) \geq \text{rank} S_j.
\] (21)
Because each $S_j \in \text{Skew}(n_j)$ necessarily has even rank, assumption (18) and (21) yield that rank $S_j = 0$ for $j = 1, 2, \ldots, l$, whence $U = 0$. This implies that

$$\text{Ker} L \cap \text{Skew}(n) = \{0\},$$

or equivalently $\text{Ker} L \subset \text{Sym}(n)$. Eq. (22) implies that $L : \text{Skew}(n) \to \text{Skew}(n)$ is an automorphism. □

In the following, we show that a Riemannian metric on $M/O(n)$ can be endowed by (8). To this end, we have to prove that for any $\Theta_1, \Theta_2 \in \pi^{-1}(\Theta)$, we obtain that

$$\langle \xi_{\Theta_1}, \xi_{\Theta_2} \rangle_{\Theta_1} = \langle \xi_{\Theta_2}, \xi_{\Theta_2} \rangle_{\Theta_2}. \quad (23)$$

Note that $\bar{\xi}_\Theta$ and $\tilde{\xi}_\Theta$ are the horizontal lifts of $\xi$ and $\xi$ at $\Theta$, respectively. That is, $\bar{\xi}_\Theta$ and $\tilde{\xi}_\Theta$ are the unique elements of $H_\Theta$ satisfying $D\pi(\Theta)[\xi_\Theta] = \xi$ and $D\pi(\Theta)[\xi_\Theta] = \xi$, respectively. To prove this, we first note that (15) leads us to the latter lemma.

**Lemma 1:** For any $U \in O(n)$,

$$(A', B', C') \in H_\Theta \Leftrightarrow (U^T A' U, U^T B', C' U) \in H_{U \Theta}. \quad (24)$$

From Lemma 1, we obtain the following theorem.

**Theorem 3:** Let $\xi_\Theta := (A', B', C')$ be the horizontal lift of $\xi \in T_{\Theta}(M/O(n))$ at $\Theta \in M$. Then, for any $U \in O(n)$, $U \circ \xi_\Theta := (U^T A' U, U^T B', C' U)$ is the horizontal lift of $\xi$ at $U \circ \Theta \in M$; i.e.,

$$\xi_{U \Theta} = U \circ \xi_\Theta. \quad (25)$$

**Proof:** Let $h : M/O(n) \to \mathbb{R}$ be any smooth function, and let $\hat{h} := h \circ \pi : M \to \mathbb{R}$. Because $\pi(U \circ \Theta) = \pi(\Theta)$ for any $\Theta \in M$, we have that

$$\hat{h}(U \circ \Theta) = \hat{h}(\Theta). \quad (26)$$

Differentiating both sides of (25) in the direction $\tilde{\xi}_\Theta$, we obtain that

$$D\hat{h}(U \circ \Theta)[D(U \circ \Theta)[\xi_\Theta]] = D\hat{h}(\Theta)[\xi_\Theta]. \quad (27)$$

Because $\tilde{\xi}_\Theta$ is the horizontal lift of $\xi$ at $\Theta$, we get

$$D\hat{h}(\Theta)[\xi_\Theta] = D\hat{h}(\pi(\Theta))[D\pi(\Theta)[\xi_\Theta]] = D\hat{h}(\pi(\Theta))[\xi].$$

Furthermore, we have that

$$D(U \circ \Theta)[\xi_\Theta] = U \circ \xi_\Theta.$$

Thus, (26) implies that

$$D\hat{h}(U \circ \Theta)[U \circ \xi_\Theta] = D\hat{h}(\pi(U \circ \Theta))[\xi]. \quad (27)$$

Because

$$D\hat{h}(U \circ \Theta)[U \circ \xi_\Theta] = D\hat{h}(\pi(U \circ \Theta))[D\pi(U \circ \Theta)[U \circ \xi_\Theta]],$$

and $h$ is any smooth function, (27) yields that

$$D\pi(U \circ \Theta)[U \circ \xi_\Theta] = \xi. \quad (28)$$

Because Lemma 1 implies $U \circ \xi_\Theta \in H_{U \Theta}$, (28) yields (25). □

We provide another proof of Theorem 3 in Appendix A. It follows from (7) that for any $U \in O(n)$

$$\langle (U^T A_1 U, U^T B_1', C' U), (U^T A_2 U, U^T B_2', C' U) \rangle_{U \Theta} = \langle (A_1', B_1', C_1'), (A_2', B_2', C_2') \rangle_{\Theta}. \quad (29)$$

From Theorem 3 and (29), for any $\Theta_1, \Theta_2 \in \pi^{-1}(\Theta)$, we obtain (23).

In numerical computation, we can use the horizontal lift $\text{grad}_{\Theta_1} g_{\Theta_1}$ of the gradient $\text{grad} g_{\Theta_1}(\Theta)$ at $\Theta_1$. The horizontal lift $\text{grad}_{\Theta_1} g_{\Theta_1}$ belongs to the horizontal space $H_{\Theta_1}$, and we obtain that

$$\text{grad}_{\Theta_1} g_{\Theta_1} = \text{grad} f(\Theta), \quad (30)$$

as shown in (27).

### III. Optimization Algorithms for Solving Problems 1 and 2

This section proposes optimization algorithms for solving Problems 1 and 2, and provides a technique for choosing initial points in the algorithms.

**A. Optimization algorithms**

Algorithm 1 describes a Riemannian conjugate gradient (CG) method for solving Problem 1. Here, the exponential map $\text{Exp}$ on the manifold $M$ is given by

$$\text{Exp}_\Theta(A', B', C') = (A^{1/2} \exp(A^{1/2} A^{-1}/2) A^{1/2}, B + B', C + C'),$$

and the parallel transport, i.e., a vector transport, $P$ on $M$ is given by

$$P_{\Theta_1, \Theta_2}(A', B', C') = ((A_2 A_1^{-1})^{1/2} A' ((A_2 A_1^{-1})^{1/2})^T, B', C'), \quad (31)$$

where $\Theta_i = (A_i, B_i, C_i) \in M$ $(i = 1, 2)$, as shown in (28). We here note that $G_A$ in (10) can be rewritten as

$$G_A = -2 \sum_{i=0}^{N-1} \sum_{k=i+1}^{N} A^{k-i-1} C^T(y_k - \hat{y}_k(\Theta)) \hat{x}_i^T. \quad (32)$$

Thus, we can recursively calculate $G_A$ as

$$G_A(i + 1) = G_A(i) - 2\gamma(i) \hat{x}_i^T N_{(i+1)};$$

where

$$G_A(0) = 0, \quad \gamma(i) = C^T(y_{N-i} - \hat{y}_{N-i}(\Theta)) + A \gamma(i-1), \quad \gamma(0) = C^T(y_N - \hat{y}_N(\Theta)).$$

In fact, $G_A(N) = G_A$. Similarly, $G_B$ in (11) can be calculated as

$$G_B(i + 1) = G_B(i) - 2\gamma(i) u_{N-(i+1)}^T;$$

where

$$G_B(0) = 0.$$
Algorithm 1 Optimization algorithm for solving Problem 1.
1: Set input-output data \( \{(u_0, y_0), (u_1, y_1), \ldots, (u_N, y_N)\} \), the state dimension \( n \), and an initial point \( \Theta_0 := ((A)_0, (B)_0, (C)_0) \in M \).
2: Set \( \eta_0 = -\nabla f(\Theta_0) \) using \([13]\).
3: for \( k = 0, 1, 2, \ldots \) do
4: Compute a step size \( t_k > 0 \), and set
\[
\Theta_{k+1} = \text{Exp}_{\Theta_k}(t_k \eta_k).
\]
5: Set
\[
\beta_{k+1} = \frac{\|g_{k+1}\|^2}{\langle g_{k+1}, P_{\Theta_k, \Theta_{k+1}}(\eta_k) \rangle_{k+1} - \langle g_k, \eta_k \rangle_{k}},
\]
where \( g_k := \nabla f(\Theta_k) \), and \( \| \cdot \|_k \) and \( \langle \cdot, \cdot \rangle_k \) denote the norm and the inner product in the tangent space \( T_{\Theta_k} M \), respectively.
6: Set
\[
\eta_{k+1} = -g_{k+1} + \beta_{k+1} P_{\Theta_k, \Theta_{k+1}}(\eta_k).
\]
7: end for.

To describe the modified ordinary MOESP method, we first define the input and output block Hankel matrices as
\[
U_{0|k-1} := \begin{pmatrix}
    u_0 & u_1 & \cdots & u_{j-1} \\
    u_1 & u_2 & \cdots & u_j \\
    \vdots & \vdots & \ddots & \vdots \\
    u_{k-1} & u_k & \cdots & u_{k+j-2}
\end{pmatrix} \in \mathbb{R}^{km \times J},
\]
\[
Y_{0|k-1} := \begin{pmatrix}
    y_0 & y_1 & \cdots & y_{j-1} \\
    y_1 & y_2 & \cdots & y_j \\
    \vdots & \vdots & \ddots & \vdots \\
    y_{k-1} & y_k & \cdots & y_{k+j-2}
\end{pmatrix} \in \mathbb{R}^{kp \times J},
\]
respectively. Here, \( J \) satisfies \( N > J + k - 2 \), and the entries of \( U_{0|k-1} \) and \( Y_{0|k-1} \) are constructed from a part of input-output data \( \{(u_0, y_0), (u_1, y_1), \ldots, (u_N, y_N)\} \).

Using the block Hankel matrices \( U_{0|k-1} \) and \( Y_{0|k-1} \), the modified ordinary MOESP method is described as follows:
1) Calculate the LQ factorization
\[
\begin{pmatrix}
    U_{0|k-1} \\
    Y_{0|k-1}
\end{pmatrix} = \begin{pmatrix}
    L_{11} & 0 \\
    L_{21} & L_{22}
\end{pmatrix} \begin{pmatrix}
    Q_1^T \\
    Q_2^T
\end{pmatrix},
\]
where \( L_{11} \in \mathbb{R}^{km \times km} \), \( L_{21} \in \mathbb{R}^{kp \times km} \), \( L_{22} \in \mathbb{R}^{kp \times kp} \), \( Q_1 \in \mathbb{R}^{J \times km} \), and \( Q_2 \in \mathbb{R}^{J \times kp} \). Here, \( L_{11} \) and \( L_{22} \) are lower triangular matrices, and \( Q_1 \) and \( Q_2 \) satisfy \( Q_1^T Q_1 = I_{km} \), \( Q_2^T Q_2 = I_{kp} \), and \( Q_1^T Q_2 = 0 \).
2) Compute the SVD of the matrix \( L_{22} \) as follows:
\[
L_{22} = U_1 U_2 \begin{pmatrix}
    \Sigma_1 & 0 \\
    0 & \Sigma_2
\end{pmatrix} \begin{pmatrix}
    V_1^T \\
    V_2^T
\end{pmatrix},
\]
where \( U_1 \in \mathbb{R}^{kp \times n} \) and \( \Sigma_1 \in \mathbb{R}^{n \times n} \).
3) Define the extended observability matrix as
\[
\mathcal{O}_k := U_1 \Sigma_1^{1/2}.
\]

Determine the matrix \( C \) by
\[
C = \mathcal{O}_k(1 : p, 1 : n),
\]
where \( \mathcal{O}_k(a_1 : a_2, b_1 : b_2) \) is the matrix constructed from the row vectors between \( a_1 \)-th and \( a_2 \)-th row vectors of \( \mathcal{O}_k \) and the column vectors between \( b_1 \)-th and \( b_2 \)-th column vectors of \( \mathcal{O}_k \). The matrix \( A \) is determined by solving the linear equation
\[
\mathcal{O}_k^1 A = \mathcal{O}_k^2,
\]
where
\[
\mathcal{O}_k^1 := \mathcal{O}_k(1 : p(k-1), 1 : n),
\]
\[
\mathcal{O}_k^2 := \mathcal{O}_k(p+1 : kp, 1 : n).
\]

4) Let
\[
U_2^T L_{21} L_{11}^{-1} = \begin{pmatrix}
    M_1 & M_2 & \cdots & M_k
\end{pmatrix},
\]
where \( M_i \in \mathbb{R}^{(kp-n) \times p} \) and \( M_i \in \mathbb{R}^{(kp-n) \times m} \) \( (i = 1, 2, \ldots, k) \), and let
\[
\tilde{L}_i = (L_i L_{i+1} \cdots L_k) \in \mathbb{R}^{(kp-n) \times (k+1-i)p}.
\]

\[\]
The matrix $B$ is determined by solving the linear equation

$$
\begin{pmatrix}
\tilde{L}_2 \mathcal{O}_{k-1} \\
\tilde{L}_3 \mathcal{O}_{k-2} \\
\vdots \\
\tilde{L}_k \mathcal{O}_1
\end{pmatrix}
= 
\begin{pmatrix}
\mathcal{M}_1 \\
\mathcal{M}_2 \\
\vdots \\
\mathcal{M}_{k-1}
\end{pmatrix},
$$

(38)

where

$$
\mathcal{O}_i := \mathcal{O}_k((i-1)p + 1 : ip, 1 : n)
$$

for $i = 1, 2, \ldots, k - 1$.

Approximate solutions of (37) and (38) are obtained by solving

$$
\min_A h(A),
$$

(39)

$$
\min_B \left\| \begin{pmatrix}
\tilde{L}_2 \mathcal{O}_{k-1} \\
\tilde{L}_3 \mathcal{O}_{k-2} \\
\vdots \\
\tilde{L}_k \mathcal{O}_1
\end{pmatrix}
- 
\begin{pmatrix}
\mathcal{M}_1 \\
\mathcal{M}_2 \\
\vdots \\
\mathcal{M}_{k-1}
\end{pmatrix}
\right\|_F,
$$

(40)

respectively, where

$$
h(A) := \|\mathcal{O}_k^1 A - \mathcal{O}_k^2\|^2_F.
$$

The solution to (40) can be expressed as

$$
B = \begin{pmatrix}
\tilde{L}_2 \mathcal{O}_{k-1} \\
\tilde{L}_3 \mathcal{O}_{k-2} \\
\vdots \\
\tilde{L}_k \mathcal{O}_1
\end{pmatrix}^T
\begin{pmatrix}
\mathcal{M}_1 \\
\mathcal{M}_2 \\
\vdots \\
\mathcal{M}_{k-1}
\end{pmatrix}.
$$

(41)

Similarly, if $A \in \mathbb{R}^{n \times n}$, then it follows from (39) that

$$
A = (\mathcal{O}_k^1)^T \mathcal{O}_k^2.
$$

(42)

However, we cannot adopt (42) as an initial point $(A)_0$ in Algorithm 1, because the matrix $A$ given by (42) is not contained in $\text{Sym}_+(n)$ in general.

For this reason, we consider the following problem.

**Problem 3:**

$$
\begin{align*}
\text{minimize} & \quad h(A) \\
\text{subject to} & \quad A \in \text{Sym}_+(n).
\end{align*}
$$

To solve Problem 3, we introduce the Riemannian metric of $\text{Sym}_+(n)$ as

$$
\langle \xi_1, \xi_2 \rangle_A := \text{tr}(A^{-1} \xi_1 A^{-1} \xi_2)
$$

(43)

for $\xi_1, \xi_2 \in T_A \text{Sym}_+(n)$. Then, we obtain the Riemannian gradient of $h$ as

$$
\text{grad } h(A) = 2 \text{Sym}((\mathcal{O}_k^1)^T \mathcal{O}_k^1 A - (\mathcal{O}_k^1)^T \mathcal{O}_k^2) A
$$

Thus, grad $h(A) = 0$ if and only if

$$
WA + AW + Q = 0,
$$

(44)

where

$$
W := -(\mathcal{O}_k^1)^T \mathcal{O}_k^1,
Q := (\mathcal{O}_k^1)^T \mathcal{O}_k^2 + (\mathcal{O}_k^2)^T \mathcal{O}_k^1.
$$

Thus, from Theorem 4.1 and Proposition 4.2 in [30], we obtain the following theorem.

**Theorem 4:** Suppose that $\text{rank}(\mathcal{O}_k^1) = n$. Then, the unique solution $A$ to (44) is given by

$$
A = \int_0^\infty \exp(W \tau)Q \exp(W \tau) d\tau.
$$

(45)

In particular, if $Q \in \text{Sym}_+(n)$, we have $A \in \text{Sym}_+(n)$.

Hence, we have the following corollary.

**Corollary 1:** Suppose that $\text{rank}(\mathcal{O}_k^1) = n$ and $Q \in \text{Sym}_+(n)$. Then, the unique solution $A$ to Problem 3 is given by (45).

Unfortunately, if $Q \not\in \text{Sym}_+(n)$, Problem 3 may not have a solution. However, if $\text{rank}(\mathcal{O}_k^1) = n$, we have the symmetric matrix $A$ given by (45) that is the unique solution to the following relaxed problem.

**Problem 4:**

$$
\begin{align*}
\text{minimize} & \quad h(A) \\
\text{subject to} & \quad A \in \text{Sym}(n).
\end{align*}
$$

Thus, we can construct the matrix $A \in \text{Sym}_+(n)$ using the unique solution to Problem 4, as shown in Algorithm 2.

**Algorithm 2 Construction method of the matrix $A \in \text{Sym}_+(n)$.**

1: Suppose that $\text{rank}(\mathcal{O}_k^1) = n$ and $Q \not\in \text{Sym}_+(n)$. Set $\epsilon > 0$.
2: Solve Lyapunov equation (44).
3: Let $A = \sum_{i=1}^n \lambda_i v_i v_i^T$ be the eigenvalue decomposition of the solution $A \in \text{Sym}(n)$.
4: $A \leftarrow \sum_{i=1}^n \max(\epsilon, \lambda_i) v_i v_i^T$.

Note that Algorithm 2 means that if the solution $A$ to (44) has non-positive eigenvalues, we replace the eigenvalues by $\epsilon > 0$. If $A^* \in \text{Sym}_+(n)$ is the matrix constructed by using Algorithm 2, then we have that

$$
\|A^* - A\|_F = \sqrt{\sum_{i=1}^n \max(\epsilon, \lambda_i) - \lambda_i}^2.
$$

In summary, we can obtain an initial point $(A_0, B_0, C_0)$ in Algorithm 1 using (36), (41), and the solution to Lyapunov equation (44) (and Algorithm 2 if necessary).

**IV. NUMERICAL EXPERIMENTS**

In this section, we demonstrate the effectiveness of the proposed method. To this end, we assume that the true system is given by

$$
\begin{align*}
\dot{x}(t) &= F^* x(t) + G^* u(t), \\
y(t) &= C^* x(t) + v(t),
\end{align*}
$$

(46)
that the true system is system (1) with measurement noise numerically identical. Thus, we showed three graphs of the
Here, we draw this figure from 10-th iteration. As mentioned A.

The initial state $\sigma$ is a zero mean i.i.d. Gaussian process corresponding to measurement noise with variance $\sigma^2$. That is, we assume that the true system is system (1) with measurement noise $v(t)$. The initial state $x(0)$ is a zero mean Gaussian random variable with variance 1, and the inputs $u_1(t)$ and $u_2(t)$ are zero mean Gaussian random variables with variance 100. Furthermore, we set the sampling interval as $h = 0.1$, and the parameter $c$ in Algorithm 2 as $\epsilon = 0.01$.

\[ A. \quad \sigma^2 = 5 \]

Fig. [ illustrates the results obtained by the SD and CG methods for Problems 1 and 2 with $N = 1000$ and $\sigma^2 = 5$. Here, we draw this figure from 10-th iteration. As mentioned in Section IV the SD method for Problems 1 and 2 are numerically identical. Thus, we showed three graphs of the SD and CG methods for Problem 1, and the CG method for Problem 2. Among the three methods, the CG method for Problem 2 is the most efficient.

We next evaluated the results with respect to the eigenvalues of estimated matrix $F$, and the relative $H^2$ and $H^\infty$ norms between the difference of transfer functions of the true continuous-time system and continuous-time systems obtained by the ordinary MOESP method [15], N4SID method [13], proposed modified ordinary MOESP method, and proposed optimization methods. To this end, we define $G$ and $\hat{G}$ as the transfer functions from the input $u$ to the output $y$ of the true and obtained systems, respectively. That is,

\[ \hat{G}(s) := C(sI_n - F^*)^{-1}G^*, \quad s \in \mathbb{C}, \]
\[ G(s) := C(sI_n - F)^{-1}G, \]

where $F$, $G$, and $C$ are estimated matrices of $F^*$, $G^*$, and $C^*$, respectively. Here, we estimate the matrices $F$ and $G$ using (5) and (6), respectively. Using $G$ and $\hat{G}$, we compared the two indices

\[ g_2 := \frac{||G - \hat{G}||_{H^2}}{||G||_{H^2}} \quad \text{and} \quad g_\infty := \frac{||G - \hat{G}||_{H^\infty}}{||G||_{H^\infty}}. \]

1) Ordinary MOESP method: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_\infty$ using the ordinary MOESP method [13]. First, we estimated the matrices $A$, $B$, and $C$ in (2) using (42), (41), and (35), respectively. Using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

\[ F_{MO} = \begin{pmatrix} F_{MO_{1,1}} & F_{MO_{1,2}} \\ F_{MO_{2,1}} & F_{MO_{2,2}} \end{pmatrix}, \]
\[ F_{MO_{1,1}} = \begin{pmatrix} -2.3708 + 0.1311\sqrt{-1} & 2.3841 - 2.1604\sqrt{-1} \\ -1.2404 - 0.5712\sqrt{-1} & 0.3992 + 4.9109\sqrt{-1} \end{pmatrix}, \]
\[ F_{MO_{1,2}} = \begin{pmatrix} 2.2735 + 0.6424\sqrt{-1} & -8.0686 - 10.5852\sqrt{-1} \\ 3.3881 - 0.4618\sqrt{-1} & -9.3184 + 7.6088\sqrt{-1} \end{pmatrix}, \]
\[ F_{MO_{2,1}} = \begin{pmatrix} -2.7298 + 2.6608\sqrt{-1} & -7.2405 - 2.5091\sqrt{-1} \\ 9.9442 - 11.5096\sqrt{-1} & 12.0855 + 10.9299\sqrt{-1} \end{pmatrix}, \]
\[ F_{MO_{2,2}} = \begin{pmatrix} -1.2114 + 13.0370\sqrt{-1} & 9.8170 - 12.2938\sqrt{-1} \\ -9.3208 - 3.7312\sqrt{-1} & 13.5371 + 9.3870\sqrt{-1} \end{pmatrix}, \]
\[ G_{MO} = \begin{pmatrix} 0.2870 - 0.3583\sqrt{-1} & 0.3947 - 0.3820\sqrt{-1} \\ 0.6338 + 0.4031\sqrt{-1} & 0.1517 + 0.4296\sqrt{-1} \end{pmatrix}, \]
\[ C_{MO} = \begin{pmatrix} -0.6754 + 0.3588\sqrt{-1} - 0.0011 - 0.0142 \end{pmatrix}, \]

respectively. Thus, we obtained

\[ \lambda(F_{MO_{1}}) = \{- 0.6305 + 31.4159\sqrt{-1}, \quad \lambda(F_{MO_{2}}) = \{- 0.9156 \pm 17.8223\sqrt{-1}, -1.8315\}, \]
\[ g_2 := 1.5062, \quad g_\infty := 2.5790. \]

2) N4SID method: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_\infty$ using N4SID [13]. First, we estimated the matrices $A$, $B$, and $C$ in (2) using the MATLAB command n4sid. Using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

\[ F_{N4} = \begin{pmatrix} -6.6957 & 9.4528 & -0.0672 & -7.3267 \\ 1.9620 & -6.8972 & 5.2249 & 8.7292 \\ -0.4200 & -3.0843 & 5.0029 & -28.4599 \\ -1.1456 & -0.9297 & 16.8523 & -2.9744 \end{pmatrix}, \]
\[ G_{N4} = \begin{pmatrix} 0.0048 - 0.0013 \\ -0.0015 - 0.0007 \\ -0.0000 - 0.0009 \\ -0.0002 0.0000 \end{pmatrix}, \]
\[ C_{N4} = \begin{pmatrix} 178.0494 & -87.2066 & 13.7368 & 3.9060 \end{pmatrix}, \]

respectively. Thus, we obtained

\[ \lambda(F_{N4}) = \{1.2203 \pm 22.0744\sqrt{-1}, -2.2049, -11.8001\}. \]

Because the matrix $F_{N4}$ is unstable, $g_2$ and $g_\infty$ are not defined.
3) Modified ordinary MOESP method: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_{\infty}$ using the modified ordinary MOESP method explained in Section III-B. First, we estimated the matrices $A$, $B$, and $C$ in (2) using Algorithm 2, (4), and (5), respectively. Using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

$$
F_{MO2} = \begin{pmatrix}
-9.3473 & 15.6483 & 6.9429 & -1.6782 \\
15.6483 & -39.3803 & 2.9600 & -0.7155 \\
6.9429 & 2.9600 & -44.7384 & -0.3174 \\
-1.6782 & -0.7155 & -0.3174 & -45.9750
\end{pmatrix},
$$

$$
G_{MO2} = \begin{pmatrix}
-1.3693 & 0.4057 \\
1.8487 & 0.6781 \\
0.5272 & -0.4779 \\
-0.1605 & -0.1601
\end{pmatrix},
$$

$$
C_{MO2} = (-0.6754 \ 0.3588 \ -0.0011 \ 0.0142),
$$

respectively. Thus, we obtained

$$
\lambda(F_{MO2}) = \{-1.2859, -46.0517, -46.0517, -46.0517\},
$$

and

$$
g_2 = 0.4580, \quad g_{\infty} = 0.6294.
$$

4) SD method for Problems 1 and 2: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_{\infty}$ using the SD method. That is, we estimated the matrices $A$, $B$, and $C$ in (2) using the algorithm obtained by replacing steps 5 and 6 in Algorithm 1 with $\eta_{k+1} = -\text{grad}(\theta_{k+1})$. Next, using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

$$
F_{SD} = \begin{pmatrix}
-11.2106 & 14.8349 & 6.5852 & -1.5916 \\
14.8349 & -39.6883 & 2.8168 & -0.6812 \\
6.5852 & 2.8168 & -44.8038 & -0.3016 \\
-1.5916 & -0.6812 & -0.3016 & -45.9788
\end{pmatrix},
$$

$$
G_{SD} = \begin{pmatrix}
-1.3591 & 0.4171 \\
0.8718 & 0.7777 \\
0.3384 & -0.4571 \\
-0.1366 & -0.1631
\end{pmatrix},
$$

$$
C_{SD} = (-0.6756 \ 0.3576 \ -0.0015 \ 0.0144),
$$

respectively. Thus, we obtained

$$
\lambda(F_{SD}) = \{-3.5678, -46.0099, -46.0517, -46.0521\},
$$

and

$$
g_2 = 0.2588, \quad g_{\infty} = 0.2130.
$$

5) CG method for Problem 1: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_{\infty}$ using the CG method for Problem 1. That is, we estimated the matrices $A$, $B$, and $C$ in (2) using Algorithm 1. Next, using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

$$
F_{CG_1} = \begin{pmatrix}
-4.2603 & -0.6398 & 3.3293 & -0.3647 \\
-0.6398 & -6.7043 & 9.8801 & -3.4325 \\
3.3293 & 9.8801 & -43.4391 & -0.7245 \\
-0.3647 & -3.4325 & -0.7245 & -45.9343
\end{pmatrix},
$$

$$
G_{CG_1} = \begin{pmatrix}
0.2132 & 1.1265 \\
1.5350 & 1.0716 \\
4.0694 & 1.2941 \\
0.9015 & -1.6191
\end{pmatrix},
$$

$$
C_{CG_1} = (-0.6875 \ 0.2881 \ 0.2331 \ -0.1280),
$$

respectively. Thus, we obtained

$$
\lambda(F_{CG_1}) = \{-3.7308, -4.1767, -46.0517, -46.3788\},
$$

and

$$
g_2 = 0.2190, \quad g_{\infty} = 0.2135.
$$

6) CG method for Problem 2: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_{\infty}$ using the CG method for Problem 2. That is, we estimated the matrices $A$, $B$, and $C$ in (2) using the algorithm obtained by replacing the parallel transport $P_{\theta_k, \theta_{k+1}}$ with the orthogonal projection $P_{\theta_k}$ onto the horizontal space $H_{\theta_{k+1}}$. Next, using (5) and (6), we constructed the matrices $F$, $G$, and $C$ in (1) as

$$
F_{CG_2} = \begin{pmatrix}
-3.1144 & 2.1809 & 4.2045 & -0.7137 \\
2.1809 & -11.1890 & 9.3867 & -3.5112 \\
4.2045 & 9.3867 & -44.6347 & 0.5504 \\
-0.7137 & -3.5112 & 0.5504 & -47.3603
\end{pmatrix},
$$

$$
G_{CG_2} = \begin{pmatrix}
-3.0928 & -0.1305 \\
-7.7175 & -1.7608 \\
27.0768 & 5.3806 \\
-22.5453 & -3.9871
\end{pmatrix},
$$

$$
C_{CG_2} = (-0.6419 \ 0.3059 \ 0.0410 \ -0.0189),
$$

respectively. Thus, we obtained

$$
\lambda(F_{CG_2}) = \{-1.3831, -9.9077, -46.0510, -48.9566\},
$$

and

$$
g_2 = 0.1786, \quad g_{\infty} = 0.1126.
$$

B. $\sigma^2 = 10$

Fig. 2 illustrates the results obtained by the SD and CG methods for Problems 1 and 2 with $N = 1000$ and $\sigma^2 = 10$. Similarly to the case where $\sigma^2 = 5$, the CG method for Problem 2 is the most efficient among the three methods.

We next evaluated the results with respect to the eigenvalues of estimated matrix $F$, $g_2$ and $g_{\infty}$. 


respectively. Thus, we obtained
\[ \lambda(F_{N4}) = \{0.4995, -0.4599 + 31.4159\sqrt{-1}\} - 1.7703 \pm 2.1110\sqrt{-1}\].
Because the matrix \( F_{N4} \) is unstable, \( g_2 \) and \( g_\infty \) are not defined.

3) Modified ordinary MOESP method: We evaluated the eigenvalues of the estimated matrix \( F \), \( g_2 \), and \( g_\infty \) using the modified ordinary MOESP method explained in Section III-B. First, we estimated the matrices \( A \), \( B \), and \( C \) in (2) using Algorithm 2, (41), and (56), respectively. Using (5) and (6), we constructed the matrices \( F \), \( G \), and \( C \) in (1) as
\[
F_{N4} = \begin{pmatrix} -4.3641 + 0.0192\sqrt{-1} & 7.6328 + 0.0609\sqrt{-1} \\ -0.2241 - 0.0019\sqrt{-1} & -3.3070 - 0.0600\sqrt{-1} \\ -0.4381 - 0.1859\sqrt{-1} & 1.8039 - 0.5881\sqrt{-1} \\ -0.3969 + 0.0608\sqrt{-1} & 1.7931 + 0.1922\sqrt{-1} \end{pmatrix},
\]
\[
F_{N4_2} = \begin{pmatrix} -1.2948 - 2.4419\sqrt{-1} & 5.4369 + 2.4716\sqrt{-1} \\ -3.0791 + 0.2397\sqrt{-1} & -9.4391 - 0.2427\sqrt{-1} \\ 0.7130 + 23.5969\sqrt{-1} & 3.7682 - 23.8840\sqrt{-1} \\ 1.2195 - 7.7119\sqrt{-1} & 3.4581 + 7.8057\sqrt{-1} \end{pmatrix},
\]
\[
G_{N4} = \begin{pmatrix} 0.0022 - 0.0002\sqrt{-1} & -0.0003 - 0.0001\sqrt{-1} \\ -0.0004 + 0.0000\sqrt{-1} & -0.0000 + 0.0000\sqrt{-1} \\ 0.0001 + 0.0015\sqrt{-1} & 0.0000 + 0.0014\sqrt{-1} \\ 0.0002 - 0.0005\sqrt{-1} & 0.0001 - 0.0004\sqrt{-1} \end{pmatrix},
\]
\[
C_{N4} = \begin{pmatrix} 278.5705 & -148.7674 & -25.8993 & -74.6933 \end{pmatrix},
\]
respectively. Thus, we obtained
\[ \lambda(F_{N4}) = \{0.4995, -0.4599 + 31.4159\sqrt{-1}\} - 1.7703 \pm 2.1110\sqrt{-1}\].

2) \( N4\text{SID} \) method: We evaluated the eigenvalues of the estimated matrix \( F \), \( g_2 \), and \( g_\infty \) using \( N4\text{SID} \) [13]. First, we estimated the matrices \( A \), \( B \), and \( C \) in (2) using the MATLAB command \textit{n4sid}. Using (5) and (6), we constructed the matrices \( F \), \( G \), and \( C \) in (1) as
\[
F_{MO1} = \begin{pmatrix} -0.7990 & -14.6742 & -5.2901 & 5.3362 \\ 13.7728 & 0.1413 & -6.1225 & -4.3852 \\ 4.2195 & 4.8604 & -0.1182 & 23.0974 \\ -1.0320 & 4.0842 & -16.9264 & -3.2957 \end{pmatrix},
\]
\[
G_{MO1} = \begin{pmatrix} 0.8066 & 0.1778 \\ 1.1520 & -0.5703 \\ -0.3373 & -0.7997 \\ -1.2931 & 0.2338 \end{pmatrix},
\]
\[
C_{MO1} = \begin{pmatrix} -0.4916 & 0.1069 & -0.4906 & 0.1031 \end{pmatrix},
\]
respectively. Thus, we obtained
\[ \lambda(F_{MO1}) = \{-0.9921 \pm 20.9310\sqrt{-1}, -1.0436 \pm 15.0585\sqrt{-1}\}, \]
and
\[ g_2 = 2.5722, \quad g_\infty = 3.3564. \]

4) SD method for Problems 1 and 2: We evaluated the eigenvalues of the estimated matrix \( F \), \( g_2 \), and \( g_\infty \) using the SD method. That is, we estimated the matrices \( A \), \( B \), and \( C \) in (2) using the algorithm obtained by replacing steps 5 and...
respectively. Thus, we obtained

$$\lambda(F_{SD}) = \{-14.5548, -46.0189, -46.0517, -46.0520\},$$

and

$$g_2 = 0.5733, \quad g_\infty = 0.3757.$$

5) **CG method for Problem 1**: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_\infty$ using the CG method for Problem 1. That is, we estimated the matrices $A$, $B$, and $C$ in (\ref{eq:problem_1}) using Algorithm 1. Next, using (\ref{eq:problem_1}) and (\ref{eq:problem_2}), we constructed the matrices $F$, $G$, and $C$ in (\ref{eq:problem_2}) as

$$F_{CG1} = \begin{pmatrix}
-28.9063 & -17.8376 & 4.0392 & 4.5628 \\
-17.8376 & -27.0745 & -3.7803 & -5.0407 \\
4.0392 & -3.7803 & -44.6777 & 0.7836 \\
4.5628 & -5.0407 & 0.7836 & -44.6376
\end{pmatrix},$$

$$G_{CG1} = \begin{pmatrix}
-0.5190 & 1.0993 \\
2.7839 & -1.7395 \\
-1.7051 & -2.3228 \\
-0.3645 & 1.8687
\end{pmatrix},$$

$$C_{CG1} = \begin{pmatrix}
-0.3999 & 0.0595 & -0.3790 & 0.0880
\end{pmatrix},$$

respectively. Thus, we obtained

$$\lambda(F_{CG1}) = \{-7.9924, -45.1972, -46.0514, -46.0552\},$$

and

$$g_2 = 0.3373, \quad g_\infty = 0.3579.$$

6) **CG method for Problem 2**: We evaluated the eigenvalues of the estimated matrix $F$, $g_2$, and $g_\infty$ using the CG method for Problem 2. That is, we estimated the matrices $A$, $B$, and $C$ in (\ref{eq:problem_2}) using the algorithm obtained by replacing the parallel transport $P_{\theta_k, \theta_{k+1}}^b$ in Steps 5 and 6 in Algorithm 1 with the orthogonal projection $P_{\theta_k, \theta_{k+1}}^b$ onto the horizontal space $H_{\theta_k, \theta_{k+1}}$. Next, using (\ref{eq:problem_2}) and (\ref{eq:problem_2}), we constructed the matrices $F$, $G$, and $C$ in (\ref{eq:problem_2}) as

$$F_{CG2} = \begin{pmatrix}
-26.0776 & -20.4786 & 5.0622 & 5.0877 \\
-20.4786 & -23.8908 & -3.8960 & -6.0702 \\
5.0622 & -3.8960 & -43.3360 & 0.3589 \\
5.0877 & -6.0702 & 0.3589 & -44.1505
\end{pmatrix},$$

$$G_{CG2} = \begin{pmatrix}
-5.8646 & 1.6955 \\
1.7710 & -5.2829 \\
-11.2820 & -4.2790 \\
4.2135 & 3.2455
\end{pmatrix},$$

$$C_{CG2} = \begin{pmatrix}
-0.0586 & 0.0073 & -0.0867 & 0.0283
\end{pmatrix},$$

respectively. Thus, we obtained

$$\lambda(F_{CG2}) = \{-2.0091, -43.3345, -46.0479, -46.0635\},$$

and

$$g_2 = 0.3579, \quad g_\infty = 0.2035.$$

C. **Discussion**

From the viewpoint of $g_2$, $g_\infty$, and the largest eigenvalue of the matrix $F$, we conclude that the CG method for Problem 2 is the best among other methods. The following explains the reasons in detail.

In both cases where $\sigma^2 = 5$ and $\sigma^2 = 10$, we could find that the ordinary MOESP and N4SID methods are worse than others from the viewpoint of $g_2$ and $g_\infty$. In particular, the N4SID method provided unstable systems in both cases, although the true system is asymptotically stable. Moreover, the numerical experiments showed that the ordinary MOESP and N4SID methods may produce complex-valued matrices $F$ and $G$. This is because the methods do not provide a symmetric matrix $A$. In contrast to the methods, together with (\ref{eq:problem_2}) the proposed modified ordinary MOESP method, i.e., Algorithm 2, produces a real-valued symmetric matrix $F$. Moreover, Figs. 3 and 4 illustrate that the proposed method provides a considerably better initial point $((A)_{0}, (B)_{0}, (C)_{0})$ in Algorithm 1 than randomly choosing $((A)_{0}, (B)_{0}, (C)_{0}) \in M$. All the proposed optimization methods improved the result of the proposed modified ordinary MOESP method. When $\sigma^2 = 5$, the CG method for Problem 2 was better than the SD method and CG method for Problem 1 from the viewpoint of $g_2$ and $g_\infty$. When $\sigma^2 = 10$, the CG method for Problem 1 was slightly better than the CG method for Problem 2 from the viewpoint of $g_2$, although the CG method for Problem 2 was better than the CG method for Problem 1 from the viewpoint of $g_\infty$.

The transient state $\hat{x}(t)$ in (\ref{eq:problem_2}) is dominated by the largest eigenvalue of the matrix $F$ under $\hat{u}(t) = 0$. That is, if the largest eigenvalues of the matrices $F$ and $F^*$ are closer, then the trajectories of $\hat{x}(t)$ of (\ref{eq:problem_2}) and $\hat{x}(t)$ of (\ref{eq:problem_2}) are more similar under $\hat{u}(t) = 0$ and $u(t) = 0$. For $\sigma^2 = 5$ and $\sigma^2 = 10$, the only CG method for Problem 2 provided the matrix $F$ with the largest eigenvalue that is close to that of $F^*$.

V. **Conclusion**

We have developed an identification method of linear continuous-time systems with symmetry by formulating minimization problems on a product manifold and its quotient manifold by the orthogonal group, where the product manifold is composed of the manifold of the symmetric positive definite matrices and two Euclidean spaces. We have proposed Riemannian CG methods for both problems, and choosing initial points in the proposed methods by using the result of the modified ordinary MOESP method. According to the numerical experiments, the initial point was considerably better than randomly choosing one on the product manifold. We have demonstrated that the CG method for the problem on the quotient manifold is the best among other methods.
In this appendix, we provide another proof of Theorem 3.

Lemma 2: For any $g \in G$ and $x \in \mathcal{M}$,
\[ \mathcal{V}_{\phi_g}(x) = D\phi_g(\mathcal{V}_x). \]  

Proof: Let $\xi \in \mathcal{V}_{\phi_g}(x) = T_{\phi_g(x)}\pi^{-1}(\{\phi_g(x)\})$. Then, there exists a curve $\gamma$ such that $\gamma(0) = \phi_g(x)$ and $\gamma(0) = \xi$. Because $G$ acts on $\mathcal{M}$, $\gamma_0(t) := \phi_g^{-1}(\gamma(t))$ is on $\pi^{-1}(\{x\})$. We have that
\[ \gamma_0(0) = \phi_g^{-1}(\phi_g(x)) = x, \]
and
\[ \dot{\gamma}_0(t) = D\phi_g^{-1}(\gamma(t))\dot{\gamma}(t). \]
Hence, it follows from (47) that
\[ \dot{\gamma}_0(0) = D\phi_g^{-1}(\phi_g(x))[\xi] = (D\phi_g(x))^{-1}[\xi] \in T_{x}\pi^{-1}(\{x\}) = \mathcal{V}_x, \]
and thus $\xi \in D\phi_g(\mathcal{V}_x)$. That is,
\[ \mathcal{V}_{\phi_g}(x) \subset D\phi_g(\mathcal{V}_x). \]
Considering the dimension of both sides, we obtain (48). □

Lemma 2 implies the following theorem.

Theorem 5: Suppose that the group action $\phi_g$ is an isometry; i.e., for any $g \in G$ and any $\xi_1, \xi_2 \in T_x\mathcal{M}$,
\[ \langle D\phi_g(\xi_1), D\phi_g(\xi_2) \rangle_{\phi_g(x)} = \langle \xi_1, \xi_2 \rangle_x. \]  

Then,
\[ \mathcal{H}_{\phi_g}(x) = D\phi_g(\mathcal{H}_x). \]  

Proof: Taking the orthogonal complement of both sides of (48), we have that
\[ \mathcal{H}_{\phi_g}(x) = (D\phi_g(\mathcal{V}_x))^\perp. \]  

Because (49) holds, we obtain that
\[ \langle D\phi_g(\xi_1), D\phi_g(\xi_2) \rangle_{\phi_g(x)} = \langle \xi_1, \xi_2 \rangle_x = 0 \]
for any $\xi_1 \in \mathcal{V}_x$ and $\xi_2 \in \mathcal{H}_x$. This means that $D\phi_g(x)[\xi_2] \in (D\phi_g(V_x))^\perp$, which yields

$$D\phi_g(\mathcal{H}_x) \subseteq (D\phi_g(V_x))^\perp.$$  

Considering the dimension of both sides, we have that

$$D\phi_g(\mathcal{H}_x) = (D\phi_g(V_x))^\perp. \quad (52)$$

It follows from (51) and (52) that (50) holds. \hfill \Box

Theorem 5 is a corollary of Theorem 5. This is because the group action $\phi_U(\Theta) := U \circ \Theta$ is an isometry, as shown in (29). Thus, from Theorem 5 we obtain that

$$\mathcal{H}_{\phi_U(\Theta)} = D\phi_U(\mathcal{H}_{\Theta}).$$

This means that (24) holds.

REFERENCES

[1] J. C. Willems, “Realization of systems with internal passivity and symmetry constraints,” *Journal of the Franklin Institute*, vol. 301, no. 6, pp. 605–621, 1976.

[2] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[3] R. Olfati-Saber and R. M. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[4] T. Hatanaka, X. Zhang, W. Shi, M. Zhu, and N. Li, “Physics-integrated hierarchical/distributed HVAC optimization for multiple buildings with robustness against time delays,” in *IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 6573–6579.

[5] C. Lidström and A. Rantzer, “Optimal $H_\infty$ state feedback for systems with symmetric and Hurwitz state matrix,” in *American Control Conference (ACC)*, IEEE, 2016, pp. 3366–3371.

[6] L. Ljung, *System Identification*. Prentice-Hall, 1999.

[7] T. Mckelvey, A. Helmersson, and T. Ribarits, “Data driven local coordinates for multivariable linear systems and their application to system identification,” *Automatica*, vol. 40, no. 9, pp. 1629–1635, 2004.

[8] H. Sato and K. Sato, “Riemannian Optimal System Identification Algorithm for Linear MIMO Systems,” *IEEE Control Systems Letters*, vol. 49, no. 2, pp. 376–381, 2017.

[9] A. Wills and B. Ninness, “On gradient-based search for multivariable system estimates,” *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 298–306, 2008.

[10] T. Katayama, *Subspace Methods for System Identification*. Springer Science & Business Media, 2006.

[11] W. E. Larimore, “Canonical variate analysis in identification, filtering, and adaptive control,” in *Decision and Control, 1990. Proceedings of the 29th IEEE Conference on*. IEEE, 1990, pp. 596–604.

[12] S. I. Qin, “An overview of subspace identification,” *Computers & Chemical Engineering*, vol. 30, no. 10, pp. 1502–1513, 2006.

[13] P. Van Overschee and B. De Moor, “N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems,” *Automatica*, vol. 30, no. 1, pp. 75–93, 1994.

[14] ——, “A unifying theorem for three subspace system identification algorithms,” *Automatica*, vol. 31, no. 12, pp. 1853–1864, 1995.

[15] M. Verhaegen and P. Dewilde, “Subspace model identification part 1. The output-error state-space model identification class of algorithms,” *International Journal of Control*, vol. 56, no. 5, pp. 1187–1210, 1992.

[16] M. Verhaegen and A. Hansson, “N2SID: Nuclear norm subspace identification of innovation models,” *Automatica*, vol. 72, pp. 57–63, 2016.

[17] M. C. Campi, T. Sugie, and F. Sakai, “An iterative identification method for linear continuous-time systems,” *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1661–1669, 2008.

[18] H. Garnier, M. Gilson, T. Bastogne, and M. Mensler, “The contsid toolbox: A software support for data-based continuous-time modelling,” in *Identification of Continuous-Time Models from Sampled Data*. Springer, 2008, pp. 249–290.

[19] M. Johansson, R. and Verhaegen and C. T. Chou, “Stochastic theory of continuous-time state-space identification,” *IEEE Transactions on Signal Processing*, vol. 47, no. 1, pp. 41–51, 1999.