

An Improved Subspace Identification Method with the Minimum Variance and Input Design

Xiangyu Mao and Jianping He

Abstract—Subspace identification has received considerable attention due to its noniterative way of estimating the system model by linear algebraic steps, which identifies the parameter matrices of the system model by projecting them into a subspace related to input and output. However, these steps which do not have an explicit function make the statistical analysis of identification results much more complex, especially the analysis of the variance. Then, the complexity of the variance analysis in the subspace method leads to the difficulty of obtaining the optimal input signal to achieve stable identification results with the minimum variance. To tackle these problems, we propose an improved subspace method which has an explicit identification function with respect to the input and output to facilitate the variance analysis, and then present an input design algorithm to minimize the variance of the identification results. The proposed method identifies the Markov parameter matrix of an LTI system and obtains the identification function by constructing a block Hankel matrix related to the input and output. We design the input signal to minimize the variance of the identification results of the matrices of the system model by splitting the non-convex variance analysis problem into two sub-problems. Both of the sub-problems are relaxed and solved with the optimal solution guaranteed. We prove the convergence of our algorithm. A simulation demonstrates the effectiveness of our algorithm.

I. INTRODUCTION

System identification refers to determining a mathematical model to describe system behavior according to the input and output, which is a branch of modern control theory. The mainstream methods for identifying a linear time-invariant (LTI) system fall into two classes: the prediction error method (PEM) and the subspace method [1].

The PEM aims to minimize a cost function related to the prediction error [2]. The PEM identifies the state-space model from all the observed data, which has excellent statistical properties but often leads to a non-convex, multi-dimensional, nonlinear optimization problem [3]. Therefore, it is necessary to simplify the objective function or constraints of the PEM. Using the low-rank nature of the block Hankel matrix for decomposition [4], fitting the input signal by a polynomial [5] or simplifying the original problem by L1 regularization and LASSO [6] are efficient ways, but the simplified or relaxed PEM may no longer guarantee the optimal prediction. Moreover, when the sample size is relatively small, the prediction by the PEM tends to have a large variance [7].

Another mainstream identification method is the subspace method. The problem considered in the subspace method is to identify a similarity transformation of the system by projecting

the system parameter matrices into a subspace related to input and output [8]. Similar to the PEM, the subspace method can also lead to a non-convex optimization problem, which can be solved by using nuclear norm as the relaxation of the rank of the matrix [9], or combining the Sylvester equation with the traditional subspace method [10]. The advantage of the subspace method compared with the PEM is that the identification result is derived according to linear algebraic steps without complicated iteration [11] or parameterizing the system model [11], which means the subspace method is easier to implement. During the process of subspace identification, the state of the system is estimated simultaneously [12], and the observer can continuously update the identification results. Thus, the subspace method has been widely applied in the identification of large-scale network systems [13]–[15]. However, the linear algebraic steps in the subspace method do not provide a cost function like the PEM, making its statistical analysis much more difficult. Instrumental variable is a feasible tool to simplify the subspace method and its error analysis [16], but the use of instrumental variables reduces the accuracy [17].

In the two mainstream methods, the input signal plays a vital role, which affects the output and the identification result. Recently, plenty of research works investigate the input design based on the PEM [18]–[20]. In [21] and [22], the input design problem of minimizing the maximum error of identification is discussed under a finite-impulse response (FIR) model based on the PEM. In the subspace method, the lack of a cost function complicates the input design problem. Thus, the excitation signal in the subspace method is mainly the Gaussian white noise input signal [9], [16], which may result in unstable identification results of system matrices with a large variance due to the noise. Thus, optimal input design for the subspace method remains a practical issue while significant improvement has been made in the subspace method [23].

Compared with the PEM, the subspace method has more reliable noniterative numerical solutions [24], which leads to its rapid adoption in industry. Considering the advantages and the wide application of the subspace method, it is highly practical to design the input signal in the method. The difficulty lies in the statistical analysis, including identification variance analysis. In the literature, input design for minimizing the variance in system identification mainly considers maximizing the Fisher information matrix of the system [25]. However, this method is only applicable to the PEM. It is challenging to ensure the minimization of variance for system identification only in subspace. If the parameter matrix to be identified is an

The Dept. of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, China. E-mail address: {maoxy20, jphe}@sjtu.edu.cn.

implicit solution to an optimization problem in the subspace method, the input design problem turns to be more difficult [17]. Besides, the existing subspace identification methods require strong assumptions, including system stability or zero initial states to obtain concise expressions [26], which narrows the scope of the application of the subspace method.

The concern of this paper is to propose a subspace identification method and design the input to minimize the variance of the identification results. To solve the problems mentioned in the previous paragraph, we need to propose an improved subspace method with concise expressions to ease the statistical analysis of variance. Next, we need to obtain the expression of identification variance according to the proposed subspace method and tackle the non-convexity of the variance minimization problem. Finally, we design the input signal based on the variance analysis to minimize the variance. The main contributions in this paper are summarized as follows.

- To the best of our knowledge, we are the first to propose a subspace method with an input design algorithm to minimize identification variance. We give the upper bound of identification error under the proposed algorithm, which makes the identification results more reliable.
- An improved subspace method with an explicit expression is proposed to tackle the difficulty of variance analysis of subspace identification. The method avoids the use of instrumental variables or the requirement of system stability or zero initial states.
- We solve the variance analysis problem by splitting and transforming it into two quadratic programming sub-problems, which is the difficulty of the input design algorithm. Both of the sub-problems are solved by relaxation with the optimal solution guaranteed. Then, we prove the convergence of our input design algorithm. Compared with the common white noise input, the accuracy and the stability of the proposed algorithm have a considerable improvement, which is verified by simulation.

The remainder of this paper is organized as follows. Section II gives the notations and describes the problem of interest. Section III proposes a subspace identification method. Section IV analyzes the identification variance of the proposed method and presents an input design algorithm. Simulation results are shown in Section V, followed by conclusions and future directions in Section VI.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Basic Model and Notations

The model investigated in this paper is the state-space model of a discrete-time LTI system defined by

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + v(k) \\ y(k) = Cx(k) + w(k) \end{cases} \quad (1)$$

where $x(k) \in \mathbb{R}^m$ is the state variable, $y(k) \in \mathbb{R}^n$ is the output signal, $u(k) \in \mathbb{R}^p$ is the input signal, $v(k) \in \mathbb{R}^m$, $w(k) \in \mathbb{R}^n$ are the process noise and output noise, $A \in \mathbb{R}^{m \times m}$, $B \in \mathbb{R}^{m \times p}$, $C \in \mathbb{R}^{n \times m}$ are system parameter matrices.

The following notations are used throughout this paper. The lowercase letters x, y, u represent vectors and the uppercase letters X, Y, U represent the matrices constructed by the vectors. Let $\text{rank}(A)$ be the rank of matrix A , $\text{Tr}(A)$ be the trace of matrix A , A^L be the generalized left inverse matrix of a full column rank matrix A , respectively.

We define $X(k:k+h)$ as a transpose vector sequence from $x(k)$ to $x(k+h-1)$ and $Y^c(k:k+s)$ as the matrix formed by the row arrangement from $y(k)$ to $y(k+s-1)$.

$$\begin{aligned} X(k:k+h) &= [x^T(k), x^T(k+1), \dots, x^T(k+h-1)]^T, \\ Y^c(k:k+s) &= [y(k), y(k+1), \dots, y(k+s-1)]. \end{aligned}$$

A block Hankel matrix formed by vector $x(k)$ is defined as

$$\mathcal{H}_{h,s}[x(k)] = \begin{bmatrix} x(k) & x(k+1) & \dots & x(k+s-1) \\ x(k+1) & x(k+2) & \dots & x(k+s) \\ \vdots & \vdots & \ddots & \vdots \\ x(k+h-1) & x(k+h) & \dots & x(k+h+s-2) \end{bmatrix},$$

where h and s determine the dimension of the block Hankel matrix. Then, the following matrix composed of the block Hankel matrix $\mathcal{H}_{h,s}[y(k)]$ and $\mathcal{H}_{h+t,s}[u(k)]$ is denoted by

$$\mathcal{L}[y(k), u(k)] = \begin{bmatrix} \mathcal{H}_{h,s}[y(k)] \\ \mathcal{H}_{h+t,s}[u(k)] \end{bmatrix}.$$

For an integer $h \geq m$, the extended observability matrix O_c and the extended controllability matrix O_b are

$$\begin{aligned} O_c(h) &= [C^T, (CA)^T, \dots, (CA^{h-2})^T, (CA^{h-1})^T]^T, \\ O_b(h) &= [A^{h-1}B, A^{h-2}B, \dots, AB, B]. \end{aligned}$$

A system transformation matrix T is defined as

$$T(h) = \begin{bmatrix} 0 & & & & & \\ CB & 0 & & & & \\ CAB & CB & 0 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ CA^{h-2}B & CA^{h-3}B & \dots & CB & 0 & \end{bmatrix}.$$

The Markov parameter matrix G is defined as

$$G(h) = [CA^{h-1}B, CA^{h-2}B, \dots, CAB, CB].$$

Finally, the infinite-norm of a vector (\cdot) is denoted by $\|(\cdot)\|_\infty$, and the Frobenius norm is denoted by $\|(\cdot)\|_F$.

B. Problem Formulation

As shown in Figure 1, the problem to be considered in this paper is the subspace identification and input design of the LTI system (1), which is formulated as follows.

- Subspace identification: A subspace method is proposed to identify the Markov parameter matrix G , i.e., derive a function f of \hat{G} related to the sequence of input u and observed output y , with the proof that \hat{G} converges to the true G^* when the system has noise v and w .

$$\hat{G} = f(u, y).$$

- Input design: We obtain the optimal input u to minimize the variance of the identification, which is proved (in

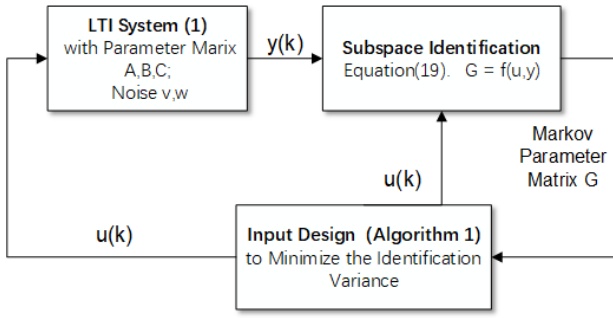


Fig. 1. Illustration of the problem of interest.

Section IV) to be equivalent to minimizing the maximum deviation of identification results as follows.

$$\begin{aligned} \min_{u(k_1:k_2)} \max_{v_i, w_i} \|G_1 - G_2\|_F \\ \text{s.t. } G_i = g(u^*, y^*, v_i, w_i) \quad i = 1, 2. \end{aligned} \quad (2)$$

where k_1, k_2 are the start and end time of input design. $g(u^*, y^*, v_i, w_i)$ is defined in section IV, which refers to the identification result of G when the noise is v_i, w_i and the true value of the input and output are u^* and y^* .

Remark 1. *If the bound of the noise v, w is known, all the possible identification results of G can be obtained, which distributes related to system input, output and noise. The aim of input design is to minimize the "area" of the distribution.*

The Markov parameter matrix G plays an essential role in system identification. The system matrices A, B, C can be obtained from G using the Ho-Kalman Algorithm, which forms a Hankel matrix H_G by G and solves A, B, C via the Singular Value Decomposition (SVD) of H_G [27]. Moreover, G is invariant in the system similarity transformation. Since the subspace method only identifies a similar transformation of the system [8], the exact A, B and C cannot be estimated when the state variables are unknown except G . Thus, we choose the matrix G as the target of identification in this paper. In addition, since there is no consensus on the criterion for evaluating the identification results, the criterion in this paper is defined as the Frobenius norm of the error of G .

The following assumptions are made throughout the paper.

Assumption 1. *The system model in (1) is minimal, and the system order is known.*

Assumption 2. *The noise v, w are zero-mean white noise which are independent of system.*

Assumption 3. *The input, output, and noise are bounded. The signal-to-noise ratio (SNR) of the system is large enough to ignore the influence of the quadratic term of the system noise v^2, w^2 on the system compared with the variables x, y or u .*

Assumptions 1 and 2 are basic guarantees for the feasibility of identification. Considering that the effect of noise on the system is generally minor, Assumption 3 is reasonable. These assumptions does not require the stability or a zero initial state, which makes them more general.

III. SUBSPACE METHOD FOR STATE-SPACE MODEL IDENTIFICATION

In this section, we first transform the system (1) and prove the equivalence of the transformation. Then, a method for estimating the Markov parameter matrix G is proposed.

A. Equivalent Transformation of the State-space Model

The state-space model of an LTI system (1) includes process noise v and output noise w . Note that the output noise w does not affect the dynamic characteristics of the system. However, the influence of the process noise v may be persistent, and the Markov parameters are difficult to identify due to the process noise [28]. An effective method to simplify the identification and the analysis of the influence of noise is transforming the process noise v into input noise which is directly related to u . Thus, we analyze (1) and construct a new model as follows.

$$\begin{cases} x(k+1) = Ax(k) + B(u(k) - e(k)) \\ y(k) = Cx(k) + w(k) \end{cases} \quad (3)$$

where $e(k) \in \mathbb{R}^p, w(k) \in \mathbb{R}^m$ represent the input noise and the output noise, respectively. $\|e\|_\infty = \|w\|_\infty = \delta$, which means e and w share the same bound. Considering the relationship between (1) and (3), we obtain the following theorem.

Theorem 1. *Suppose Assumptions 1,2,3 are true, then systems (1) and (3) are equivalent.*

Proof. In order to transform v which affects x into e which affects u , we examine the relationship between the state variable x and the input u first.

By Assumption 1, the system (1) is fully controllable when it is noise-free, i.e., if $v = w = 0, \forall x(0) = x_0$, there exist a smallest positive integer m_{x_0} and a sequence $u(0), u(1), \dots, u(m_{x_0} - 1)$, such that

$$x(m_{x_0}) = 0. \quad (4)$$

From (4), one infers that the change of x can be transformed into the change of sequence u in the LTI system. To prove that this sequence is feasible, we investigate the bound of m_{x_0} .

Denote $x(0)$ as x_0 , and expand x recursively from $x(h)$ to $x(0)$ by (1). Then, $x(h)$ is given by

$$x(h) = A^h x_0 + O_b(h)U(0:h). \quad (5)$$

When $h = m$, by Assumption 1,

$$\text{rank}(O_b(m)) = m. \quad (6)$$

Substitute $x(h) = 0$ and $h = m$ in (5). It follows that

$$O_b(m)U(0:m) = -A^m x_0. \quad (7)$$

From (6), the matrix $O_b(m)$ has full row rank. Hence, $[O_b(m), -A^m x_0]$ has full row rank, i.e.,

$$\text{rank}([O_b(m), -A^m x_0]) = \text{rank}(O_b(m)), \quad (8)$$

which means that the equation in (7) with $U(0:m)$ as a variable always has a solution. Then, m is an integer satisfying (4). Considering that m_{x_0} is the smallest integer, we have

$$m_{x_0} \leq m. \quad (9)$$

This implies that m_{x_0} has a supremum m .

Hence, we conclude that the effect of noise on x is equivalent to a change of u for a finite time m_{x_0} with the supremum m , i.e., $\forall k$, if $x(0) = v(k)$,

$$\begin{aligned} &\exists u_{v(k)}(0), u_{v(k)}(1), \dots, u_{v(k)}(m_{x_0} - 1), \\ &\text{s.t. } x(m_{x_0}) = 0. \end{aligned}$$

Define

$$e(k) = \sum_{j=k-m_{x_0}+1}^k u_{v(j)}(k-j). \quad (10)$$

Then, e in (3) is equivalent to v in (1).

Next, we prove that $e(k)$ is bounded to transform the system (1) to achieve the same bound of e and w . From (10),

$$\|e\|_\infty \leq m_{x_0} \|u_{v(j)}\|_\infty \leq m \|u_{v(j)}\|_\infty = m \frac{\|A^m v\|_\infty}{\|O_b(m)\|_\infty}.$$

By Assumption 3, v is bounded. Note that A, O_b, m are constants or constant matrices, then $e(k)$ is bounded. Define $\delta = \|w\|_\infty, \|A^m v\|_\infty / \|O_b(m)\|_\infty = c$ and

$$\frac{x'(k)}{x(k)} = \frac{e'(k)}{e(k)} = \frac{u'(k)}{u(k)} = \frac{\delta}{c}, \quad C' = \frac{c}{\delta} C.$$

Then, $e(k)'$ and $w(k)'$ share the same bound δ . \square

By Theorem 1, we relate the noise directly to the input and output by constructing an equivalent system model (3), i.e., $\forall v(k)$ in (1), there exists a sequence $e(k)$ in (3), such that the two systems can be transformed into each other through linear transformation.

B. Subspace Identification of the Markov Parameter Matrix

In this subsection, a subspace method is proposed by constructing a block Hankel matrix related to the input and output and eliminating x without using instrumental variables.

Theorem 2. Assume that the system (3) is noise-free, then the Markov parameter matrix G is given by

$$G(h) = Y^c(d : d + s) \mathcal{L}^{-1}[y(k), u(k)] \begin{bmatrix} 0 \\ I_r \end{bmatrix}, \quad (11)$$

where $d = k + h + t$, $s = h \cdot n + (h + t) \cdot p$ and $r = h \cdot n \cdot p$ are constants that determine the dimension of the matrices.

Proof. First, we derive the relationships between x and y, u .

Expand $y(k)$ recursively until $y(k + h - 1)$ by (3). Then,

$$Y(k : k + h) = O_c(h)x(k) + T(h)U(k : k + h). \quad (12)$$

Based on Assumption 1, when $h \geq m$, $\text{rank}(O_c(h)) = m$. Hence, $\exists O_c^L(h)$, s.t.

$$O_c^L(h)O_c(h) = I. \quad (13)$$

From (12) and (13),

$$x(k) = O_c^L(h)Y(k : k + h) - O_c^L(h)T(h)U(k : k + h), \quad (14)$$

which means $x(k)$ can be linearly represented by $Y(k : k + h)$ and $U(k : k + h)$.

Expand $x(k)$ recursively until $x(k + h - 1)$ by (3),

$$x(k + h) = A^h x(k) + O_b(h)U(k : k + h). \quad (15)$$

From (14) and (15), $x(k + h)$ can be linearly represented by $Y(k : k + h)$ and $U(k : k + h)$, i.e., $\exists F_1, F_2$, s.t.

$$x(k + h) = F_1 Y(k : k + h) + F_2 U(k : k + h). \quad (16)$$

Eliminate x by (16) in (3) and it follows that

$$\begin{aligned} y(k + h + t) = \\ CA^t[F_1, F_2] \begin{bmatrix} Y(k : k + h) \\ U(k : k + h) \end{bmatrix} + G(h)U(k + h : k + h + t). \end{aligned}$$

Denote $R = CA^t[F_1 \ F_2]$. Then,

$$y(d) = [R, G(h)] \begin{bmatrix} Y(k : k + h) \\ U(k : d) \end{bmatrix}. \quad (17)$$

Considering the time from d to $d + s - 1$, we obtain that

$$Y^c(d : d + s) = [R, G(h)] \mathcal{L}[y(k), u(k)]. \quad (18)$$

Note that the measure of the set of singular matrices in the real matrix space is zero. Then, for the given y and u , $\mathcal{L}[y(k), u(k)]$ is always invertible. Hence, Theorem 2 is proved by (18). \square

When the system (3) is not noise-free, and the amount of data is $N \cdot s$, we define the identification function f as

• Subspace identification function:

$$\hat{G}(h) = f(u, y) = \frac{1}{N} \sum_{i=0}^N Y^c(d : d + s) \mathcal{L}^{-1}[y(k), u(k)] \begin{bmatrix} 0 \\ I_r \end{bmatrix}, \quad (19)$$

where $k = s \cdot i$. The proof that \hat{G} converges to the true G^* is illustrated in Theorem 4, Section IV.

Theorem 2 and (19) provide a method for identifying matrix G by an expression only related to y and u , which simplify the process of error analysis. After obtaining G , the famous Ho-Kalman Algorithm [27] can be used to solve the system parameter matrix A, B, C . In the Ho-Kalman Algorithm, a Hankel matrix H_G is formed by G , then matrices A, B, C are derived via the SVD of H_G .

IV. INPUT DESIGN FOR MINIMIZING IDENTIFICATION VARIANCE

In this section, a method for obtaining the maximum deviation between the identification results is proposed, which is proved to be equivalent to minimizing the identification variance. Then, we design the system input to minimize the deviation and prove the convergence of the proposed method.

From the equivalent transformed system (3), substitute v by e , then, the input design problem (2) is equivalent to

$$\begin{aligned} &\min_{u(k_1 : k_2)} \max_{e_i, w_i} \|G_1 - G_2\|_F \\ &\text{s.t. } G_i = g(u^*, y^*, e_i, w_i) \quad i = 1, 2. \end{aligned} \quad (20)$$

From (3), it follows that $y^* = y - w, u^* = u - e$. Then, by (19), g is defined as

$$g(u^*, y^*, e_i, w_i) = f(u - e_i, y - w_i). \quad (21)$$

A. Maximum Identification Deviation

The problem (20) is a min-max problem, and the inner problem of maximum deviation needs to be solved first. In this subsection, we obtain the solution to the inner problem. It is worth noting that the noise is considered as a known variable when determining the maximum identification deviation.

Similar to $Y^c(d : d+s)$, we denote $W_i^c(d : d+s)$ as the matrix formed by the vector from $w_i(d)$ to $w_i(d+s-1)$. Define

$$[R_i \ G_i(h)] = [Y^c(d : d+s) - W_i^c(d : d+s)] \cdot \mathcal{L}^{-1}[y(k) - w_i(k), u(k) - e_i(k)]. \quad (22)$$

From (11) and (22), the g in (20) is defined as

$$g_i(u, y, e_i, w_i) = [R_i, G_i(h)] \begin{bmatrix} 0 \\ I_r \end{bmatrix} \quad i = 1, 2. \quad (23)$$

Then, the maximum identification deviation $J(u)$ is given by

$$J(u) = \max_{e_1, w_1, e_2, w_2} \| ([R_1 \ G_1(h)] - [R_2 \ G_2(h)]) \begin{bmatrix} 0 \\ I_r \end{bmatrix} \|_F. \quad (24)$$

Denote the elements of the i -th row and j -th column of the square matrices $\mathcal{L}^{-1}[y(k), u(k)]$ as $\alpha(i, j)$, $\mathcal{L}^{-1}[y(k) - w_i(k), u(k) - e_i(k)]$ as $\beta_i(i, j)$ and $\mathcal{L}[w(k), e(k)]$ as $p(i, j)$, respectively. Then,

$$\frac{\partial \beta(i, j)}{\partial p(l, t)} = (\alpha(t, j) - p(t, j))(\alpha(i, l) - p(i, l)). \quad (25)$$

Note that in the Taylor expansion of β_i with respect to p , the $p(t, j)$ and $p(i, l)$ in the partial derivative (25) corresponds to a quadratic term of p , which can be ignored by Assumption 3. Then, β_i is a proportional function of p . Denote β_i as ϵp . It follows that $\beta_i w$ can be ignored when the ϵ is relatively small (which is ensured in the input design Algorithm in Section IV). Hence, the expansion of J is reduced to

$$J^2(u) = \max_{e_1, w_1, e_2, w_2} \sum_{j=r+1}^s \left(\sum_{i=r+1}^s (\beta_1(i, j) - \beta_2(i, j)) y(d+i-1) + \sum_{i=r+1}^s (w_1(d+i-1) - w_2(d+i-1)) \alpha(i, j) \right)^2. \quad (26)$$

By Assumption 2 and (10), $e(k), w(k)$ are zero-mean white noise which are independent of the system. Denote β as $(\beta_1 - \beta_2)$, w as $(w_1 - w_2)$ and e as $(e_1 - e_2)$ in this subsection. Expanding the expression of J in (26), ignoring the quadratic term by Assumption 3, it holds that solving J is equivalent to solving two sub-problems J_1 and J_2 ,

$$J_1 = \max_w \sum_{j=r+1}^s \left(\sum_{i=r+1}^s w(d+i-1) \alpha(v, j) \right)^2, \quad (27)$$

$$J_2 = \max_{e, w} \sum_{j=r+1}^s \left(\sum_{i=r+1}^s \beta(i, j) y(d+i-1) \right)^2, \quad (28)$$

where $\|e\|_\infty \leq 2\delta$, $\|w\|_\infty \leq 2\delta$. From (27), it follows that

$$J_1 = \max_W W^T (d : d+s) H W (d : d+s), \quad (29)$$

where $H(i, j) = \sum_{k=r+1}^s \alpha(k, i) \alpha(k, j)$.

The maximization problem (29) is a quadratic programming problem with a positive semi-definite Hessian matrix H , which makes it an NP-Hard, non-convex problem [29]. Considering that J_1 is a quadratic function with a positive first coefficient for each variable w , the optimal J_1 is obtained only when all w have reached the boundary value δ or $-\delta$. When s is relatively small, (29) can be solved by enumeration. However, for the general case, it is necessary to perform a relaxation.

Denote Q_W as $W(d : d+s) W^T (d : d+s)$ and \mathcal{S} as the set of positive semi-definite matrices with the same dimension as Q_W , respectively. Then, (27) is equivalent to

$$J_1 = \max_{Q_W \in \mathcal{C}} \text{Tr}(Q_W H), \quad (30)$$

where $\mathcal{C} : \{Q : Q \leq 4\delta^2 I, Q \in \mathcal{S}, \text{rank}(Q) = 1\}$.

Considering the problem (30), define a relaxed convex feasible set $\mathcal{C}_{relax} : \{Q : Q \leq 4\delta^2, Q \in \mathcal{S}\}$ and the relaxed problem

$$Q^* = \arg \max_{Q \in \mathcal{C}_{relax}} \text{Tr}(QH). \quad (31)$$

Then, the relaxed problem (31) can be efficiently solved by semidefinite programming [25].

Theorem 3. *Denote q as the eigenvector corresponding to the largest eigenvalue of the solution Q^* of (31). Then, q is the optimal solution of (27).*

The proof of Theorem 3 can be directly obtained by referring to Section 4 in [25], since the transformed sub-problem (31) is a low dimensional case in [25]. Hence, the solution of (27) is obtained from Theorem 3.

For the solution of J_2 , we denote the objective function of J_2 by f_{j_2} , then,

$$\frac{\partial f_{j_2}}{\partial p(l, t)} = 2 \sum_{j=r+1}^s \sum_{i=r+1}^s \beta(i, j) y(d+i-1) \alpha(t, j) \alpha(i, l). \quad (32)$$

Hence, one infers that the second-order partial derivative of f_{j_2} can be regarded as a constant, i.e., the Hessian matrix H_2 of the multivariate function f_{j_2} with respect to p is a constant matrix. According to (25) and (32), the Taylor expansion of f_{j_2} related to p does not have a first-order term. Then, J_2 can be transformed into the following quadratic problem

$$J_2 = \max_P P^T H_2 P, \quad (33)$$

where P denotes the sequence composed of $p(i, j)$ and H_2 is an Hessian matrix which can be solved from (32). Then, the solution of J_2 is provided by Theorem 3. By solving (30) and (33), we obtain the optimal solution of (26), which is the maximum identification deviation.

To illustrate that minimizing the identification deviation is equivalent to minimizing the variance, we first define the variance of N times of identification G_i $i = 1, 2, \dots, N$ as

$$\mu = \sum_{i=1}^N \|G_i - \frac{1}{N} \sum_{j=1}^N G_j\|_F^2. \quad (34)$$

Similar to (26), μ is also a quadratic form of the noise with coefficients y and u . Considering that the noise is independent, and that β is a linear function of the noise from (25), then, the change of y and u dose not affect the distribution of identification results. Hence, minimizing the variance μ is equivalent to minimizing the maximum deviation J .

B. Input Design for Minimizing Identification Variance

This subsection describes the method of input design, which is shown in Algorithm 1. We take the subspace method proposed in Section III to identify the system, then combine the sequential quadratic programming (SQP) with our method for solving the maximum deviation to design u .

In the initialization phase, time is sampled discretely into multiple intervals with s time points in each interval. We set the start time k and the number of groups of data N .

Then, we construct a system model by (3). The maximum of the quadratic term of the noise which can be ignored is set as ϵ . According to the system security requirements, the bound of the infinite-norm of $y(k)$ and $u(k)$ are determined as y_m and u_m . We denote the maximum noise as δ by Assumption 2. At last, we initialize a sequence $Y(0 : s)$ and $U(0 : s)$ without special design to preliminarily identify the system.

Then, we used the subspace method to identify \hat{A}_i , \hat{B}_i and \hat{C}_i , which are required in (26) for obtaining the maximum identification deviation. We estimate $\hat{G}(h)$ from (19) by the latest data $Y((i-1)s : is)$ and $U((i-1)s : is)$, then compute \hat{A}_i , \hat{B}_i , \hat{C}_i by the Ho-Kalman Algorithm.

Next, we obtain the feasible set \mathcal{U} of input u_d . The designed u_d is feasible, if and only if both of u_d and the predicted value of y are within the safety threshold when ensuring Assumption 3. We keep the real y safe by leave a certain margin of y_m . In addition, $U(0 : s)$ in the initialization phase also needs to be in the feasible set. The predicted y is defined as follows.

$$\begin{aligned} \hat{Y}(is : (i+1)s) &= \hat{O}_c \hat{A}^h O_c(h)^L Y((i-1)s : is) \\ &+ \hat{O}_c \hat{A}^h O_c(h)^L \hat{G}_i(h) U((i-1)s : is) \\ &+ \hat{O}_c \hat{O}_b U((i-1)s : is) \\ &+ \hat{G}_i(h) U(is : (i+1)s). \end{aligned} \quad (35)$$

To ensure Assumption 3, considering that only the variable β is related to e or w in (26), from (25), the feasible set \mathcal{U} is

$$\mathcal{U} : \{u_d : \|\hat{y}\|_\infty \leq y_m, \|u\|_\infty \leq u_m, \delta \|\alpha^2\|_\infty \leq \epsilon\}. \quad (36)$$

Besides, considering that the influence of u on an LTI system control may have time delay, the Algorithm 1 borrows the idea of Model Predictive Control (MPC) to ensure safety when solving the feasible set and designing u . The $U(is : (i+1)s + m)$ is designed instead of $U(is : (i+1)s)$ to ensure that the output value of the system y does not exceed the bound in the interval of $[is, (i+1)s + m]$. Only the first s -step $U(is : (i+1)s)$ is taken in the real system to avoid the influence of time delay.

Finally, we take SQP to solve the optimal u_d in the feasible set \mathcal{U} because the expression of the inner problem (20) is implicit. SQP is a representative non-convex optimization

algorithm. The simulation in Section V shows SQP achieves satisfying results. The value of maximum deviation at each point of u_d is provided by (26).

Repeating the process of system identification and input design, the identification result is continuously updated while designing input to minimize the variance.

C. Convergence Analysis

In this subsection, we analyze the convergence of the Algorithm 1.

Theorem 4. *Considering the estimation \hat{G} under Algorithm 1, it follows that*

$$\lim_{N \rightarrow \infty} \left\| \hat{G}(h) - G^*(h) \right\|_F^2 = 0.$$

Furthermore, if the noise obeys Gaussian distribution, then with probability at least $1 - 2\exp(-t^2/2)$,

$$\left\| \hat{G}(h) - G^*(h) \right\|_F^2 \leq \frac{c_1}{N} (c_2 + t),$$

where c_1, c_2 are constants determined in initialization.

Proof. Since

$$G_k(h) = Y^c(d : d + s) \mathcal{L}^{-1}[y(k), u(k)] \begin{bmatrix} 0 \\ I_r \end{bmatrix}, \quad \forall k. \quad (37)$$

Therefore,

$$G^*(h) = \frac{1}{N} \sum_{k=1}^N G_k^*(h), \quad \hat{G}(h) = \frac{1}{N} \sum_{k=1}^N G_k(h). \quad (38)$$

Denote $\Delta G = \left\| \hat{G}(h) - G^*(h) \right\|_F^2 = \sum_{i=1}^s \Delta G_i$. Then,

$$\Delta G_i = \frac{1}{N} \sum_{i=1}^N (w_i \alpha_i + w_i \beta_i + y_i^* \beta_i), \quad (39)$$

where $w_i, \alpha_i, \beta_i, y_i$ represent the i -th column or i -th row of the matrix. By Assumptions 2 and 3, $w_i, \alpha_i, \beta_i, y_i$ are independent of each other and bounded, then $\lim_{N \rightarrow \infty} \Delta G = 0$.

When the noise obeys Gaussian distribution, by Assumption 3 and (25), β obeys Gaussian distribution. According to the fact that α and y are sequences with definite upper bounds and the Frobenius norm is Lipschitz continuous, (39) is applicable to the case of Lemma A.1 in [26]. Hence,

$$\Delta G = \sum_{i=1}^s \Delta G_i \leq \frac{s}{N} (\epsilon \delta + \epsilon^2 \delta^2 + \epsilon^2 \delta y_m) (2\sqrt{s} + t), \quad (40)$$

with probability at least $1 - 2\exp(-t^2/2)$. Considering that $\epsilon, \delta, s, N, y_m$ are all constants, the proof is complete. \square

Theorem 4 proves that the estimation error of our algorithm converges at a speed of $1/\sqrt{N}$ and converges to 0 in the infinite time domain. Actually, the designed input obtains a better result with a lower amount of data compared with the white noise input as shown in the next section.

Algorithm 1 Input Design Algorithm

Initialize: Construct the system model (3). Determine $\epsilon, \delta, h, t, N, y_m$ and u_m . Initialize a sequence of y and u .

Repeat: For $i = 1$ to $i = N$

- 1) Get the latest output data y and input data u .
- 2) Identify the system by the subspace method.
 - 2-1) Compute $\hat{G}(h)$ by (19).
 - 2-2) Obtain $\hat{A}_i, \hat{B}_i, \hat{C}_i$ by the Ho-Kalman Algorithm.
- 3) Design the input $U(is : (i+1)s + m)$ denoted by u_d
 - 3-1) Find the feasible set \mathcal{U} of u_d in (36) by $\hat{A}_i, \hat{B}_i, \hat{C}_i$.
 - 3-2) Use SQP to search for u_d of (20) in \mathcal{U} .
The maximum deviation at each u_d is provided by (26).
- 4) Take $U(is : (i+1)s)$ as input in the actual system.

V. NUMERICAL SIMULATION

This section uses a numerical simulation to compare the performance of our input design method with a white noise input and an PEM-based input design method. The simulation result verifies the effectiveness of Algorithm 1 in this paper.

We randomly generate an SISO system model of order 4 and convert it into a controllable canonical form. The conversion is reasonable because the subspace method only identifies a similar transformation of the system, and the conversion does not change the matrix G .

The controllable canonical form of the simulation model is

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1.233 & -2.174 & -1.421 & -1.208 \end{bmatrix},$$

$$B = [0 \ 0 \ 0 \ 1]^T, \quad C = [0.819 \ 0.169 \ -0.281 \ 0.266].$$

The constraints of the SISO system are $\delta = 0.05, y_m = 100, u_m = 10$. Tolerance on the constraint violation in SQP is set to default as $1e - 6$. The output constraint is soft with a certain safety margin to improve calculation efficiency, which means the system is allowed to violate the constraint in a short time. The noise is generated as a white noise sequence. We take the Frobenius norm of the Markov parameter matrix G as the index to evaluate the system identification result, which is consistent with the optimization goal of (20).

In Figure 2, we consider the scenario of identifying a running system. We use $x(0) = [0; 0.5; 0.3; 1]$ to generate an initial sequence of length 44 and apply the input design method in this paper for system identification, compared with the white noise input and the input which maximizes the Fisher information matrix based on the PEM [25]. We record the average of the error when the number of batches of data increases. It is observed that as the number of data increases, the identification errors of the three identification methods decrease. The error of the proposed method and the PEM-based method are always within the upper bound defined by Theorem 4 ($t = 5$), and much smaller than white noise input. In 80 batches of data, the error of the proposed method and the PEM-based method are 28.3% and 40.3% of the white

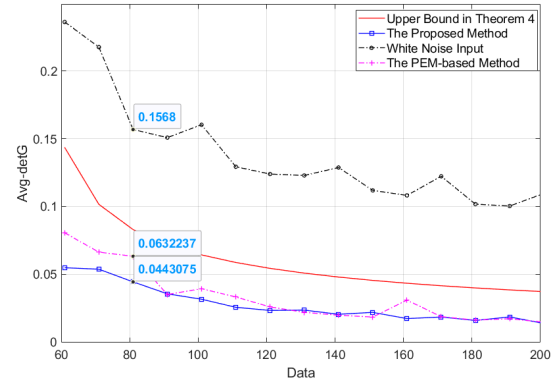


Fig. 2. The average of the error of the identification result ($\|\hat{G} - G^*\|_F$), obtained from 100 Monte Carlo runs with random noise.

noise input, respectively. Note that the performance of the PEM-based method is similar to the proposed method in large time domain. However, the error of the PEM-based method tends to be large when the amount of data is relatively small. It implies that the algorithm in this paper has advantages in fast identification in a short time. This is because that the proposed method makes full use of the subspace expression in Section III and the safety range of y , as shown in Figure 4. In addition, the identification error of the PEM-based method fluctuates when the data set is not large, and the reliability of the PEM-based result is not as high as the proposed method.

Figure 3 provides intuitive evidence that our method can reduce the maximum deviation. We consider the distribution of the results of multiple identifications of systems with the same parameter matrix and different noises. We made simulations with the three types of input in the case of 70 and 250 batches of data. The identification results of 70 batches taking random input or the PEM based input have large variance and many outliers. The proposed input design method needs only 70 batches to achieve better identification result than the 250 batches of data that traditionally uses random input.

The output signal and its prediction during the identification process are given by Figure 4. It shows that the model in this paper accurately predicts the value of y , and that the output signal fluctuates within a safe range. Since the safety constraint is soft, the output signal violates the constraint in a short time. Strict restrictions can be achieved by setting a smaller constraint tolerance. These simulation results demonstrate the effectiveness of the proposed input design algorithm.

VI. CONCLUSION

In this paper, we propose a subspace method with input design to enable the observer to identify the system more accurately by minimizing the variance of identification results of the system matrices. Our research provides a feasible way for tackle the difficulty of variance analysis and input design for subspace identification. We investigate the state-space model of the LTI system and transform the process noise into input noise. Then, we propose an improved subspace identification method to identify the Markov parameter matrix

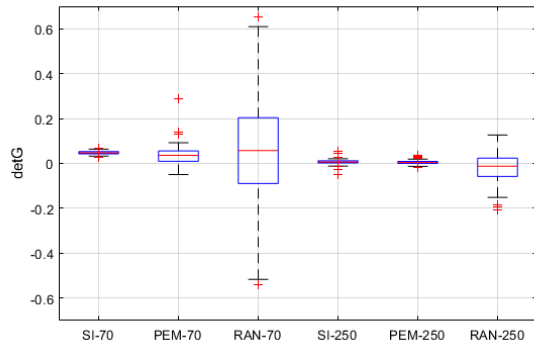


Fig. 3. Box-plot of the distribution of the error of the identification result, which compares the distribution of the subspace input design algorithm proposed in this paper (SI), the input design method based on the PEM (PEM) and the random white noise input (RAN) under 70 and 250 sets of data.

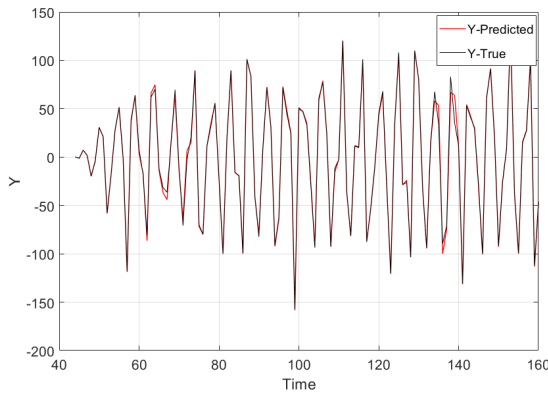


Fig. 4. The output signal (the black line) and its prediction (the red line) during the identification process.

of the system. We derive an explicit identification function and avoid the use of instrumental variables in the proposed method. We investigate the variance of the identification results of the Markov parameter matrix, transform the variance analysis problem into two sub-problems and solve the sub-problems by relaxation. Finally, an input design algorithm is presented to achieve more accurate and stable identification results via minimizing the identification variance in (34). A simulation has been provided to illustrate the effectiveness of the proposed algorithm. Future directions include i) finding a fast convergence method that can replace SQP in designing input signal while ensuring optimality ii) extending the application of input design to a locally observable network system.

REFERENCES

- [1] Z. Zhang, R. Du, and R. V. Cowlagi, "Randomized sampling-based trajectory optimization for uavs to satisfy linear temporal logic specifications," *Aerospace Science and Technology*, vol. 96, no. 1–4, p. 105591, 2020.
- [2] K. Astrom, "Maximum likelihood and prediction error methods," *IFAC Proceedings Volumes*, vol. 12, no. 8, Supplement 1, pp. 551–574, 1979.
- [3] N. Everitt, M. Galrinho, and H. Hjalmarsson, "Open-loop asymptotically efficient model reduction with the steiglitz-mcbride method," *Automatica*, vol. 89, pp. 221–234, 2018.
- [4] C. Yu, L. Ljung, and M. Verhaegen, "Identification of structured state-space models," *Automatica*, vol. 90, pp. 54–61, 2018.
- [5] F. Z. Chaoui, F. G. *, Y. Rochdi, M. Haloua, and A. Naitali, "System identification based on hammerstein model," *International Journal of Control*, vol. 78, no. 6, pp. 430–442, 2005.
- [6] W. Zhao, G. Yin, and E.-W. Bai, "Sparse system identification for stochastic systems with general observation sequences," *Automatica*, vol. 121, p. 109162, 2020.
- [7] S. K. Perepu and A. K. Tangirala, "Identification of equation error models from small samples using compressed sensing techniques," *IFAC-PapersOnLine*, vol. 48, no. 8, pp. 795–800, 2015.
- [8] A. Micchi and G. Pannocchia, "Comparison of input signals in subspace identification of multivariable ill-conditioned systems," *Journal of Process Control*, vol. 18, no. 6, pp. 582–593, 2008.
- [9] C. Yu, L. Ljung, A. Wills, and M. Verhaegen, "Constrained subspace method for the identification of structured state-space models (cosmos)," *IEEE Transactions on Automatic Control*, vol. 65, no. 10, pp. 4201–4214, 2020.
- [10] M. Inoue, "Subspace identification with moment matching," *Automatica*, vol. 99, pp. 22–32, 2019.
- [11] M. Verhaegen and V. Verdult, *Filtering and System Identification: A Least Squares Approach*, 1st ed. USA: Cambridge University Press, 2007.
- [12] K. Peterzell, W. Scherrer, and M. Deistler, "Statistical analysis of novel subspace identification methods," *Signal Processing*, vol. 52, no. 2, pp. 161–177, 1996.
- [13] C. Yu and M. Verhaegen, "Subspace identification of distributed clusters of homogeneous systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 463–468, 2017.
- [14] A. Haber and M. Verhaegen, "Subspace identification of large-scale interconnected systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 10, pp. 2754–2759, 2014.
- [15] J. S. Grover, C. Liu, and K. Sycara, "Parameter identification for multi-robot systems using optimization based controllers (extended version)," 2020.
- [16] C. Yu, J. Chen, and M. Verhaegen, "Subspace identification of individual systems in a large-scale heterogeneous network," *Automatica*, vol. 109, p. 108517, 2019.
- [17] M. Verhaegen and A. Hansson, "N2sid: Nuclear norm subspace identification of innovation models," *Automatica*, vol. 72, pp. 57–63, 2016.
- [18] B. Wahlberg, H. Hjalmarsson, and M. Annergren, "On optimal input design in system identification for control," in *IEEE CDC*, 2010, pp. 5548–5553.
- [19] M. Annergren, C. A. Larsson, H. Hjalmarsson, X. Bombois, and B. Wahlberg, "Application-oriented input design in system identification: Optimal input design for control [applications of control]," *IEEE Control Systems Magazine*, vol. 37, no. 2, pp. 31–56, 2017.
- [20] M. Gevers, L. Mišković, D. Bonvin, and A. Karimi, "Identification of multi-input systems: variance analysis and input design issues," *Automatica*, vol. 42, no. 4, pp. 559–572, 2006.
- [21] K. Lindqvist and H. Hjalmarsson, "Identification for control: adaptive input design using convex optimization," in *IEEE CDC*, 2001, pp. 4326–4331.
- [22] M. Casini, A. Garulli, and A. Vicino, "Input design in worst-case system identification using binary sensors," *IEEE Transactions on Automatic Control*, vol. 56, no. 5, pp. 1186–1191, 2011.
- [23] S. J. Qin, "An overview of subspace identification," *Computers & Chemical Engineering*, vol. 30, no. 10, pp. 1502–1513, 2006.
- [24] P. Misra and M. Nikolaou, "Input design for model order determination in subspace identification," *AIChE Journal*, vol. 49, no. 8, pp. 2124–2132, 2003.
- [25] I. R. Manchester, "Input design for system identification via convex relaxation," in *IEEE CDC*, 2010, pp. 2041–2046.
- [26] S. Oymak and N. Ozay, "Non-asymptotic identification of lti systems from a single trajectory," in *IEEE ACC*, 2019, pp. 5655–5661.
- [27] B. L. HO and R. E. Kalman, "Editorial: Effective construction of linear state-variable models from input/output functions," *at - Automatisierungstechnik*, vol. 14, no. 1–12, pp. 545–548, 1966.
- [28] Y. Zheng and N. Li, "Non-asymptotic identification of linear dynamical systems using multiple trajectories," *IEEE Control Systems Letters*, vol. 5, no. 5, pp. 1693–1698, 2021.
- [29] N. I. M. Gould and P. L. Toint, *Numerical Methods for Large-Scale Non-Convex Quadratic Programming*. Boston, MA: Springer US, 2002, pp. 149–179.