Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

On the Least Squares Solutions of a System of Bilinear Equations

Er-Wei Bai and Yun Liu
Dept. of Electrical and Computer Engineering
University of Iowa, Iowa City, Iowa 52242
er-wei-bai@uiowa.edu

Abstract— **The problem of finding a least squares solution for a system of bilinear equations is investigated. Sufficient conditions to have a unique minimum are given in the cases of random inputs. Three methods, the normalized iterative method, the over-parametrization method and the numerical method are presented for solving the least squares problem along with their convergence properties. Simulation examples are provided.**

I. INTRODUCTION

Besides linear equations, one common type of models that arise in science and engineering is the system of bilinear equations. For example, we need to solve such systems in identification of a Hammerstein model which is parameterized by two sets of unknown parameters [1], [2], [4], [5]. However, unlike its linear counterpart, there is no systematic study on a system of bilinear equations and only scattered results are reported in the literature [3]. In this paper, we will study the least squares solution of a system of bilinear equations.

In a system of bilinear equations, we have two unknown vectors,

$$\mathbf{x} = (x_1, x_2, \cdots, x_m)^T \in \mathbf{R}^m$$

and

$$\mathbf{y} = (y_1, y_2, \cdots, y_n)^T \in \mathbf{R}^n.$$

The known data are a sequence of $n \times m$ dimensional matrices

$$A(i) = \begin{pmatrix} a_{11}(i) & \cdots & a_{1m}(i) \\ \vdots & \ddots & \vdots \\ a_{n1}(i) & \cdots & a_{nm}(i) \end{pmatrix}, \quad i = 1, 2, ..., N,$$

and the right-hand side vector

$$\mathbf{d} = (d_1, ..., d_N)^T \in \mathbf{R}^N.$$

where

$$d_i = \mathbf{y}^{*T} A(i)\mathbf{x}^* + v_i,$$

v_i's are unknown noises and

$$\mathbf{x}^* = (x_1^*, ..., x_m^*)^T \in \mathbf{R}^m$$

$$\mathbf{y}^* = (y_1^*, ..., y_n^*)^T \in \mathbf{R}^n$$

are the "true" values. The bilinear equations can be written in the following form:

$$\mathbf{y}^T A(i)\mathbf{x} = d_i \quad \text{for} \quad i = 1, 2, ...N. \tag{I.1}$$

The goal is to find a solution in the least squares sense

$$\begin{aligned}(\mathbf{x}, \mathbf{y}) &= \arg\min_{\mathbf{x},\mathbf{y}} J_N(\mathbf{x}, \mathbf{y}) \\ &= \arg\min_{\mathbf{x},\mathbf{y}} \frac{1}{N}\sum_{i=1}^{N}(d_i - \mathbf{y}^T A(i)\mathbf{x})^2. \end{aligned} \tag{I.2}$$

and hope the solution of (I.2) will be close to the "true" values $(\mathbf{x}^*, \mathbf{y}^*)$.

To avoid unnecessary complications, we assume that the noise v_i is zero mean i.i.d. with finite variance σ_v^2. The cost function $J_N(\mathbf{x}, \mathbf{y})$ in (I.2) can be now written as

$$J_N(\mathbf{x}, \mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{y}^T A(i)\mathbf{x} - \mathbf{y}^{*T} A(i)\mathbf{x}^* - v_i)^2.$$

Let

$$B(l,p) = \lim_{N\to\infty} B_N(l,p) \text{ with}$$

$$B_N(l,p) = \frac{1}{N}\sum_{i=1}^{N}\begin{pmatrix} a_{1l}(i) \\ \vdots \\ a_{nl}(i) \end{pmatrix}\begin{pmatrix} a_{1p}(i) & \cdots & a_{np}(i) \end{pmatrix},$$

where $l = 1, ..., m,\ p = 1, ..., m,$ and

$$C(l,p) = \lim_{N\to\infty} C_N(l,p) \text{ with}$$

$$C_N(l,p) = \frac{1}{N}\sum_{i=1}^{N}\begin{pmatrix} a_{l1}(i) \\ \vdots \\ a_{lm}(i) \end{pmatrix}\begin{pmatrix} a_{p1}(i) & \cdots & a_{pm}(i) \end{pmatrix},$$

where $l = 1, ..., n,\ p = 1, ..., n,$ and

$$\Phi_1 = \begin{pmatrix} B(1,1) & \cdots & B(1,m) \\ \vdots & \ddots & \vdots \\ B(m,1) & \cdots & B(m,m) \end{pmatrix},$$

$$\Phi_2 = \begin{pmatrix} C(1,1) & \cdots & C(1,n) \\ \vdots & \ddots & \vdots \\ C(n,1) & \cdots & C(n,n) \end{pmatrix}$$

It can be easily shown that when $N \to \infty$, $J_N \to J$ where

$$\begin{aligned} J(\mathbf{x}, \mathbf{y}) = &((x_1\mathbf{y} - x_1^*\mathbf{y}^*)^T, ..., (x_m\mathbf{y} - x_m^*\mathbf{y}^*)^T) \\ &\Phi_1\begin{pmatrix} (x_1\mathbf{y} - x_1^*\mathbf{y}^*) \\ \vdots \\ (x_m\mathbf{y} - x_m^*\mathbf{y}^*) \end{pmatrix} + \sigma_v^2 \end{aligned} \tag{I.3}$$

0-7803-9568-9/05/$20.00 ©2005 IEEE

$$= ((y_1\mathbf{x}-y_1{}^*\mathbf{x}^*)^T, ..., (y_n\mathbf{x}-y_n^*\mathbf{x}^*)^T)$$

$$\Phi_2 \begin{pmatrix} (y_1\mathbf{x}-y_1^*\mathbf{x}^*) \\ \vdots \\ (y_n\mathbf{x}-y_n^*\mathbf{x}^*) \end{pmatrix} + \sigma_v^2 \tag{I.4}$$

or equivalently,

$$J(\mathbf{x},\mathbf{y}) = \sum_{l=1}^{m}\sum_{p=1}^{m}(x_l\mathbf{y}-x_l^*\mathbf{y}^*)^T B(l,p)(x_p\mathbf{y}-x_p^*\mathbf{y}^*)+\sigma_v^2 \tag{I.5}$$

$$= \sum_{l=1}^{n}\sum_{p=1}^{n}(y_l\mathbf{x}-y_l^*\mathbf{x}^*)^T C(l,p)(y_p\mathbf{x}-y_p^*\mathbf{x}^*)+\sigma_v^2 \tag{I.6}$$

In section 2 we will discuss when the least squares solution is unique. In particular, sufficient conditions are given in the random cases. Three methods, the normalized iterative method, the over-parametrization method and the numerical method are presented to solve the least squares problem (I.2) along with their convergence in sections 3. Simulation examples are provided in section 4.

II. A SPECIAL CASE-RANDOM INPUTS

Unlike the linear case, the least squares problem (I.2) is difficult to solve if it has more than one local minimum. In this section we will discuss the conditions under which the minimum is unique for the random input case. To avoid a trivial case. it is assumed that

$$\mathbf{x}^* \neq 0,\ \mathbf{y}^* \neq 0$$

Further note if $(\mathbf{x},\mathbf{y})$ is a solution of the bilinear system (I.1), then so is $(c\mathbf{x},\mathbf{y}/c)$ for any nonzero constant c. Therefore, either $\mathbf{x}$ or $\mathbf{y}$ has to be normalized. It is also assumed that

$$||\mathbf{y}^*|| = 1$$

and the the first non-zero entry is positive.

Let $\mathbf{E}$ denote the expectation operator, the following result is derived.

Theorem 2.1: Consider the bilinear system (I.1) and the least squares problem (I.2). Assume all the components $a_{jk}(i)$ of $A(i),\ j=1,...,n,\ k=1,...,m$ are zero mean and independent random variables having variance σ_{jk}^2. Further, the variance of components in the same row or column of $A(i)$s are the same, i.e.,

$$\mathbf{E}(a_{j1}^2(i)) = \cdots = \mathbf{E}(a_{jm}^2(i)) \tag{II.7}$$

or

$$\mathbf{E}(a_{1k}^2(i)) = \cdots = \mathbf{E}(a_{nk}^2(i)), \tag{II.8}$$

for $j=1,...,n,\ k=1,...,m$. The noise v_i is zero mean i.i.d. with bounded variance σ_v^2. Then, as $N\to\infty$, the cost function $J(\mathbf{x},\mathbf{y})$ has a unique minimum achieved at $(\mathbf{x}^*,\mathbf{y}^*)$.

Proof: Under the assumptions, the matrices in the cost functions (I.5) and (I.6) become

$$B(l,p) = \begin{cases} \mathbf{0}, & l\neq p; \\ \begin{pmatrix} \sigma_{1l}^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_{nl}^2 \end{pmatrix}, & \text{l=p.} \end{cases} \tag{II.9}$$

where $l,p=1,...,m$.

$$C(l,p) = \begin{cases} \mathbf{0}, & l\neq p; \\ \begin{pmatrix} \sigma_{l1}^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_{lm}^2 \end{pmatrix}, & \text{l=p.} \end{cases} \tag{II.10}$$

where $l,p=1,...,m$.

The partial derivative of $J(\mathbf{x},\mathbf{y})$ at all local minimum points should satisfy

$$\frac{\partial J}{\partial x_k} = \sum_{p=1}^{m}\mathbf{y}^T B(k,p)(x_p\mathbf{y}-x_p^*\mathbf{y}^*) + \sum_{l=1}^{m}(x_l\mathbf{y}-x_l^*\mathbf{y}^*)^T B(l,k)\mathbf{y} = 0 \tag{II.11}$$

$$\frac{\partial J}{\partial y_j} = \sum_{p=1}^{n}\mathbf{x}^T C(j,p)(y_p\mathbf{x}-y_p^*\mathbf{x}^*) + \sum_{l=1}^{n}(y_l\mathbf{x}-y_l^*\mathbf{x}^*)^T C(l,j)\mathbf{x} = 0 \tag{II.12}$$

where $k=1,...,m,\ j=1,...,n$. Substituting the matrix $C(l,p)$ in (II.12), we have the following

$$(\sigma_{j1}^2x_1^2+\cdots+\sigma_{jm}^2x_m^2)y_j = (\sigma_{j1}^2x_1x_1^*+\cdots+\sigma_{jm}^2x_mx_m^*)y_j^*.$$

Noting that condition (II.7) $\sigma_{j1}^2=\sigma_{j2}^2=\cdots=\sigma_{jm}^2$ for $j=1,...,n$ implies

$$y_j = y_j^*\frac{\mathbf{x}^T\mathbf{x}^*}{\mathbf{x}^T\mathbf{x}} \Rightarrow \mathbf{y} = c\cdot\mathbf{y}^*$$

Substituting $B(l,p)$ into the partial derivative condition of (II.11), we get

$$(\sigma_{1k}^2y_1^2+\cdots+\sigma_{nk}^2y_n^2)x_k = (\sigma_{1k}^2y_1y_1^*+\cdots+\sigma_{nk}^2y_ny_n^*)x_k^*.$$

Since we have solved that $y_j=cy_j^*$, we have

$$\begin{aligned} &c^2(\sigma_{1k}^2y_1^{*2}+\cdots+\sigma_{nk}^2y_n^{*2})x_k \\ &= c(\sigma_{1k}^2y_1^{*2}+\cdots+\sigma_{nk}^2{y_n}^{*2})x_k^* \\ &\Rightarrow \quad x_k = \frac{1}{c}x_k^* \end{aligned}$$

Thus, $\mathbf{x}=\frac{1}{c}\cdot\mathbf{x}^*$. So the solution is unique under the normalization constraint. The similar procedure can be followed by using the condition (II.8). ■

The following corollary is a direct consequence of Theorem 2.1.

Corollary 2.1: Consider the bilinear system (I.1) and the least squares problem (I.2). Assume all the components $a_{jk}(i)$ of $A(i),\ i = 1, ..., N,\ j = 1, ..., n,\ k = 1, ..., m$ are zero mean and i.i.d. distributed with the variance σ_a^2, and the noise v_i is zero mean and i.i.d. distributed with bounded variance σ_v^2. Then as $N \to \infty$, the cost function $J(\mathbf{x}, \mathbf{y})$ has a unique minimum under the normalization constraint, achieved at $(\mathbf{x}^*, \mathbf{y}^*)$.

Though the theorem and corollary require that $A(i)$ is zero mean, the results can be easily extended to non-zero mean $A(i)$.

Let

$$\mathbf{E}A(i) = \bar{A} = \begin{pmatrix} \bar{a}_{11} & \cdots & \bar{a}_{1m} \\ \vdots & \ddots & \vdots \\ \bar{a}_{n1} & \cdots & \bar{a}_{nm} \end{pmatrix}$$

and

$$\mathbf{E}d_i = \bar{d}_i = \mathbf{y}^{*T}\mathbf{E}A(i)\mathbf{x}^* = \mathbf{y}^{*T}\bar{A}\mathbf{x}^*$$

Further, let

$$\tilde{A}(i) = A(i) - \bar{A},\ \tilde{d}_i = d_i - \bar{d}_i$$

Now, consider the least squares problem for the equation

$$\mathbf{y}^T\tilde{A}(i)\mathbf{x} = \tilde{d}_i$$

$\tilde{A}(i)$ is zero mean. Under the conditions of Theorem 2.1, the minimum is unique and achieved at $(\mathbf{x}^*, \mathbf{y}^*)$. This solves the original least squares problem. In implementation, the mean values can be calculated by the samples means

$$\bar{d}_i = \frac{1}{N}\sum_{i=1}^{N} d_i, \quad \bar{A} = \frac{1}{N}\sum_{i=1}^{N} A(i)$$

III. General cases

In this section, we consider a general case and no assumptions are made on $A(i)$ and d_i.

A. Normalized Iterative Algorithm

The iterative algorithm is initially proposed in [4] to solve the identification problem of a Hammerstein model. The method is simple, and converges fast when it converges. But the convergence can not be proved in the general case [5].

As we discussed in section 2, without loss of generality, a solution can be characterized by the normalization $||\mathbf{y}|| = 1$. A recent result shows that the normalized iterative algorithm has good convergence properties in identifying Hammerstein systems when the input is i.i.d. distributed [2]. The same algorithm can be used in solving (I.2). Given an initial estimate $\mathbf{y}_0 \neq \mathbf{0}$, the algorithm is defined as:

$$\begin{array}{l} \bar{\mathbf{x}}_k = \arg\min_{\mathbf{x}\in\mathbf{R}^m} J_N(\mathbf{x}, \mathbf{y}_{k-1}), \\ \bar{\mathbf{y}}_k = \arg\min_{\mathbf{y}\in\mathbf{R}^n} J_N(\mathbf{x}_{k-1}, \mathbf{y}), \\ \text{Let } \xi_k = \pm 1 \text{ be the sign of the first non-zero} \\ \text{entry of } \bar{\mathbf{y}}_k \text{ define} \\ \mathbf{y}_k = \xi_k\bar{\mathbf{y}}_k/||\bar{\mathbf{y}}_k||, \quad \mathbf{x}_k = \xi_k\bar{\mathbf{x}}_k \cdot ||\bar{\mathbf{y}}_k||, \\ \text{Replace } k \text{ by } k+1 \text{ and the process is repeated.} \end{array} \tag{III.13}$$

In fact, the algorithm can also be started with an estimate $\mathbf{x}_0$, define

$$\begin{array}{l} \bar{\mathbf{y}}_k = \arg\min_{\mathbf{y}\in\mathbf{R}^n} J_N(\mathbf{x}_{k-1}, \mathbf{y}), \\ \text{Let } \xi_k = \pm 1 \text{ be the sign of the first non-zero} \\ \text{entry of } \bar{\mathbf{y}}_k \text{ define } \ \mathbf{y}_k = \xi_k\bar{\mathbf{y}}_k/||\bar{\mathbf{y}}_k||, \\ \mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathbf{R}^m} J_N(\mathbf{x}, \mathbf{y})_k \\ \text{Replace } k \text{ by } k+1 \text{ and the process is repeated.} \end{array} \tag{III.14}$$

The following theorem shows that the algorithms (III.13) and (III.14) converge to a stationary point in general cases as $N \to \infty$. The proof is the same as theorem 3.2 in [2], and is omitted here.

Theorem 3.1: Consider the bilinear system (I.1). Assume the noise v_i is zero mean and i.i.d. distributed with finite variance. Consider the least squares cost function (I.3) and (I.4) with $\Phi_1 > 0, \Phi_2 > 0$. In the normalized iterative algorithms (III.13) and (III.14), let $\mathbf{z}_k = \begin{pmatrix} \mathbf{x}_k \\ \mathbf{y}_k \end{pmatrix}$ represent the kth step estimates. Suppose $\mathbf{x}_k$ and $\mathbf{y}_k \neq \mathbf{0}$ at each k. Then,

1) The sequence $\mathbf{z}_k$ generated by the normalized iterative algorithm (III.13) or (III.14) is bounded.
2) The cost function $J(\mathbf{x}_k, \mathbf{y}_k)$ is strictly convex in $\mathbf{x}_k$ ($\mathbf{y}_k$) for given $\mathbf{y}_k$ ($\mathbf{x}_k$) at each k, and the minimum of J with respect to $\mathbf{y}_k$ ($\mathbf{x}_k$) for given $\mathbf{x}_k$ ($\mathbf{y}_k$) is unique at each k.
3) Each accumulation point $\mathbf{z}_k$ of the sequence $\{\mathbf{z}_k\}$ satisfies
$$\nabla J(\mathbf{z}_k) \doteq \frac{\partial J}{\partial \mathbf{z}_k} = \mathbf{0}.$$
4) If at some k, $\mathbf{x}_k = \frac{1}{a}\cdot\mathbf{x}^*$ or $\mathbf{y}_k = a\cdot\mathbf{y}^*$ for some constant $a \neq 0$, then
$$\mathbf{y}_{k+1} = \mathbf{y}_{k+2} = ... = \mathbf{y}^*,$$
$$\mathbf{x}_{k+1} = \mathbf{x}_{k+2} = ... = \mathbf{x}^*.$$

In general, the stationary point returned by the normalized iterative algorithm is not necessarily the true solution $\mathbf{x}^*$ and $\mathbf{y}^*$. It depends on the properties of the cost function $J(\mathbf{x}, \mathbf{y})$ and the initial estimate. In the random case, however, the global optimum can be obtained

Theorem 3.2: Consider the bilinear system (I.1). Assume the noise v_i is zero mean and i.i.d. distributed with finite variance. Assume the components $a_{jk}(i)$ of $A(i), j = 1, ..., n,\ k = 1, ..., m$ are zero mean and i.i.d. distributed with variance σ_{jk}^2. Further, the variance of components in the same row or the same column of all $A(i)$s are the same, i.e.,

$$\mathbf{E}(a_{j1}^2(i)) = \cdots = \mathbf{E}(a_{jm}^2(i))$$

or

$$\mathbf{E}(a_{1k}^2(i)) = \cdots = \mathbf{E}(a_{nk}^2(i)),$$

for $j = 1, ..., n,\ k = 1, ..., m$. Assume $\Phi_1 > 0, \Phi_2 > 0$. Consider the normalized iterative algorithms (III.13) and (III.14), then as $N \to \infty$,

1. If all the variances in one column are the same, and $\mathbf{y}_0^T\mathbf{y}^* \neq 0$, the algorithm (III.13) converges to $(\mathbf{x}^*, \mathbf{y}^*)$ in one step (two least squares iterations).

2. If all the variances in one row are the same, and $\mathbf{x}_0^T\mathbf{x}^* \neq 0$, the algorithms (III.14) converge to $(\mathbf{x}^*, \mathbf{y}^*)$ in one step (two least squares iterations).

Proof:

Given the assumptions, the matrix $B(l,p),\ C(l,p)$ in the cost function (I.5) and (I.6) are given in (II.9) and (II.10).

Substitute $\mathbf{x}_0,\ C(l,p)$ into the partial derivative condition (II.12), as derived in the proof of Theorem 2.1 we get

$$||\mathbf{x}_0||^2 y_j = \mathbf{x}_0^T\mathbf{x}^* y_j^*$$

which gives

$$\bar{\mathbf{y}}_1 = \frac{\mathbf{x}_0^T\mathbf{x}^*}{||\mathbf{x}_0||^2}\cdot\mathbf{y}^* = c\cdot\mathbf{y}^*$$

and the rest follows from the last part of Theorem 3.1.

The second part can be proved by substituting $\mathbf{y}_0,\ B(l,p)$ into the partial derivative condition (II.12). It gives

$$||\mathbf{y}_0||^2 x_k = \mathbf{y}_0^T\mathbf{y}^* x_k^*$$

$$\Rightarrow \bar{\mathbf{x}}_1 = \frac{\mathbf{y}_0^T\mathbf{y}^*}{||\mathbf{y}_0||^2}\cdot\mathbf{x}^* = c\cdot\mathbf{x}^*$$

and the result follows. ■

The set $\{\mathbf{y}\in\mathbf{R}^n : \mathbf{y}^T\mathbf{y}^* = 0\}$ and $\{\mathbf{x}\in\mathbf{R}^m : \mathbf{x}^T\mathbf{x}^* = 0\}$ have zero measures in $\mathbf{R}^n$ and $\mathbf{R}^m$, respectively. Thus Theorem 3.2 tells us that the normalized iterative algorithm is globally convergent in one step with initial estimate almost everywhere under the given assumptions. Moreover, whether $\mathbf{y}_0$ is in the measure 0 set is detectable in one iteration step. Note that $\mathbf{y}_0^T\mathbf{y}^* = 0$ implies that $\bar{\mathbf{x}}_1 = 0$.

Though there is no guarantee that the stationary point returned by the algorithms (III.13) and (III.14) is the true solution $(\mathbf{x}^*, \mathbf{y}^*)$ in general, we have not seen any such an example based on a huge number of numerical simulations.

B. Over-parametrization Method

The over-parametrization method is initially proposed in [1] to identify a bilinear Hammerstein-Wiener system. It is an optimal two-stage algorithm. Consider the equation (I.1), we can write it in the following scalar form,

$$\sum_{j=1}^{m}\sum_{k=1}^{n} a_{jk}(i) y_j x_k = d_i \quad \text{for} \quad i = 1, ..., N.$$

Define

$$\begin{aligned}\boldsymbol{\theta} &= (y_1x_1, ..., y_1x_m, y_2x_1, ..., y_2x_m, ..., \\ &\quad y_nx_1, ..., y_nx_m) \\ &= (\theta_1, \theta_2, ..., \theta_{nm})\end{aligned}$$

as the over-parameterized unknown vector, and

$$\Theta_{yx} = \mathbf{y}\mathbf{x}^T = \begin{pmatrix} y_1x_1 & y_1x_2 & \cdots & y_1x_m \\ y_2x_1 & y_2x_2 & \cdots & y_2x_m \\ \vdots & \vdots & \ddots & \vdots \\ y_nx_1 & y_nx_2 & \cdots & y_nx_m \end{pmatrix}$$

which is the unknown vector $\boldsymbol{\theta}$ in a $n\times m$ matrix form. Define the data matrix

$$\Phi_N = \begin{pmatrix} a_{11}(1) & a_{12}(1) & \cdots & a_{nm}(1) \\ a_{11}(2) & a_{12}(2) & \cdots & a_{nm}(2) \\ \vdots & \vdots & \ddots & \vdots \\ a_{11}(N) & a_{12}(N) & \cdots & a_{nm}(N) \end{pmatrix}.$$

The equation (I.1) can be now written as a linear system

$$\Phi_N\boldsymbol{\theta} = \mathbf{d}. \tag{III.15}$$

We use the following two-stage algorithm to obtain an estimate of $\mathbf{y}^*$ and $\mathbf{x}^*$.

Step 1. Calculate the least-squares estimate $\hat{\boldsymbol{\theta}} = (\Phi_N^T\Phi_N)^{-1}\Phi_N^T\mathbf{d}$,
Step 2. Construct $\hat{\Theta}_{yx}$ from $\hat{\boldsymbol{\theta}}$ and let $\hat{\Theta}_{yx} = \sum_{i=1}^{\min(n,m)}\sigma_i\mu_i\nu_i^T$ be its singular value decompositions (SVD), where $\mu_i (i = 1, 2, ..., n)$ and $\nu_i (i = 1, 2, ..m)$ are n, m -dimensional orthonormal vectors, respectively,
Step 3. Let s_μ denote the sign of the first nonzero element of μ_1. Define the estimates as $\hat{\mathbf{x}} = s_\mu\sigma_1\nu_1,\ \hat{\mathbf{y}} = s_\mu\mu_1$. (III.16)

In the first stage of the above algorithm, the least-squares solution $\hat{\boldsymbol{\theta}}$ is sought to get an estimate of the over-parameterized unknown vector $\boldsymbol{\theta}$. In the second stage, the optimum $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ is given by

$$(\hat{\mathbf{x}}, \hat{\mathbf{y}}) = \arg\min_{\mathbf{x}\in\mathbf{R}^m, \mathbf{y}\in\mathbf{R}^n} ||\hat{\Theta}_{yx} - \mathbf{y}\mathbf{x}^T||_F^2.$$

The solutions of $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are provided by the SVD of $\hat{\Theta}_{yx}$, which is proved to be optimal in lemma A.1 in [1]. Because the second stage is globally optimal over all vector space, the convergence of this algorithm depends on the convergence of the over-parameterized least-squares problem in the first stage. When $\Phi_N^T\Phi_N > 0$, the disturbance on $\mathbf{d}$ is white with zero mean and finite variance, the least-squares solution $\hat{\Theta}_{yx}$ will converge to the true solution $\mathbf{y}^{*T}\mathbf{x}^*$ as $N\to\infty$. This is a standard result in linear regression. If the first stage is convergent, then $\hat{\mathbf{x}}\to\mathbf{x}^*,\ \hat{\mathbf{y}}\to\mathbf{y}^*$ in the second stage.

C. Numerical Method

The numerical method solves the nonlinear least squares problem (I.2) using a direct optimization technique. Consider the following necessary conditions for a global optimum

$$\begin{aligned}\frac{\partial J_N}{\partial x_k} &= \frac{2}{N}\sum_{i=1}^{N}[(d_i - \mathbf{y}^T A(i)\mathbf{x})\sum_{j=1}^{n} y_j a_{jk}(i)] \triangleq 0 \\ \frac{\partial J_N}{\partial y_l} &= \frac{2}{N}\sum_{i=1}^{N}[(d_i - \mathbf{y}^T A(i)\mathbf{x})\sum_{j=1}^{m} a_{lj}(i) x_j] \triangleq 0\end{aligned} \tag{III.17}$$

where $k = 1, 2, ..., m$ and $l = 1, 2, ..., n$. There are $n + m$ partial derivative conditions, and they are all in polynomial form. The goal is to solve these nonlinear equations to get all candidates of the global minimum solution $\mathbf{x}^*$ and $\mathbf{y}^*$. The global minimum solution then can be found by

substituting all possible candidates into (I.2) and checking the cost function.

There are numerical methods available for solving a group of nonlinear equations and they are provided as standard routines in some computer software packages. In our case, all equations in (III.17) are in polynomial form. A systematic procedure to solve them can be found, e.g., in elimination theory [6].

The elimination theory solves two polynomial equations $p(x,y)=0$ and $q(x,y)=0$ simultaneously by eliminating one unknown variable, say, y. For example, let the two polynomials have degrees 3 and 2 in x, respectively. They can be written in the form

$$p(x,y) = p_3(y)x^3 + p_2(y)x^2 + p_1(y)x + p_0(y)$$
$$q(x,y) = q_2(y)x^2 + q_1(y)x + q_0(y)$$

The $(3+2)\times(3+2)$ sylvester matrix is defined by $S_{pq}(x) \triangleq$

$$\begin{pmatrix} p_0(y) & 0 & q_0(y) & 0 & 0 \\ p_1(y) & p_0(y) & q_1(y) & q_0(y) & 0 \\ p_2(y) & p_1(y) & q_2(y) & q_1(y) & q_0(y) \\ p_3(y) & p_2(y) & 0 & q_2(y) & q_1(y) \\ 0 & p_3(y) & 0 & 0 & q_2(y) \end{pmatrix}$$

The resultant polynomial is then defined by

$$r(x) = Res(p(x,y), q(x,y), y) \triangleq detS_{pq}(x)$$

One standard result is that any solution (x_0, y_0) of $p(x,y) = 0$ and $q(x,y) = 0$ must satisfy $r(x) = 0$. Thus we can solve $r(x) = 0$ and have all possible candidate solutions for x_0 and check if they are the solutions for the two equations.

In (III.17), we have $m+n$ unknown variables and $m+n$ equations. We can eliminate one variable at a time using the above technique. First, we write all polynomials in descending order of the variable, then construct the sylvester matrix for each other equation and obtain the resultant polynomials of other variables. The resultant polynomials will form a group of polynomial equations with one less variable. Continuing this elimination process, we will have one polynomial equation with one variable in the end. All candidate solutions can then be found and substituted back to check the validity.

The direct optimization method is computationally expensive when the unknown vectors have high dimensions, and therefore can suffer from bad numerical stability problem. For smaller dimension problems, the method will be a good choice in finding all critical points of the cost function.

IV. Examples

In this section we provide an example to compare three methods summarized in the previous sections. Let the components of $A(i)$ be uniformly i.i.d. distributed in the interval $[2,6]$, $i = 1,2,...,N = 40$, $n = 3$, $m = 4$. The true unknown vectors are

$$\mathbf{y}^* = (0.5000, 0.7006, 0.5090)^T,$$
$$\mathbf{x}^* = (-3, 2, 5, -1)^T.$$

$d_i = \mathbf{y}^{*T} A(i)\mathbf{x}^* + v_i$ and three different noises, uniformly distributed in $[-0.025, 0.025]$, $[-0.05, 0.05]$, and $[-0.5, 0.5]$ respectively, are simulated.

Method 1: For the normalized iterative algorithm, the initial estimate was $\mathbf{y}_0 = (5,6,7)^T$. The simulation results are shown in Table I. The iterative method converges very fast. In our case, the solution converges in 4 steps for $N = 40$.

Method 2: We use the over-parametrization method to estimate the unknown vectors in the same example. After the SVD decomposition, the estimated values are shown in Table II. They are very close to the true values even for a small N=40.

Method 3: We use the numerical method to find a least squares solution of the example. The results are provided in Table III. For noise $v_i \in [-0.025, 0.025]$, 123 groups of roots are found for the polynomial equations in (III.17) by numerical methods and most are complex. We are only interested in 7 real roots that are candidates for the optimum. The cost functions $J_N(\mathbf{\Theta}, \mathbf{\Gamma})$ corresponding these 7 real roots are calculated

$$(448.69, 424.23, 447.54, 0.000197, 448.69, 446.998, 419.06).$$

So the one that gives the minimum cost 0.000197 is the solution which is listed in Table III after normalization. For noise $v_i \in [-0.05, 0.05]$, 97 roots are found, the cost functions corresponding to the 5 real roots are

$$(446.264, 11.8355, 444.946, 0.000653394, 409.987)$$

The solution is with the minimum cost 0.000653394. When the noise $v_i \in [-0.5, 0.5]$, the cost functions corresponding to the 5 real roots are

$$(427.14, 13.4343, 419.217, 482.336, 0.0748615)$$

and the solution is again with the minimum cost 0.0748615. All three estimates are close to the true values.

Therefore all the three methods work in this example. We can use one method to check the solution given by another method in practice.

References

[1] Bai, E. W. (1998), "An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems", *Automatica*, **14**, pp. 333-338.

[2] Bai, E. W. and D. Li (2004),"Convergence of the iterative Hammerstein system identification algorithm", *IEEE. Trans on Auto. Contr.*, **49**, pp. 1929-1940.

[3] Cohen, S. and C. Tomasi (1997) "Systems of Bilinear Equations", Technical Report, Dept. of Computer Science, Stanford University.

[4] Nanrendra, K. S. and P. G. Gallman (1966), "An iterative method for the identification of nonlinear systems using a Hammerstein model", *IEEE Trans. on Auto. Contr.*, **11**, pp. 546-550.

[5] Stoica, P. (1981), "On the convergence of an iterative algorithm used for Hammerstein systems", *IEEE Trans. on Auto. Contr.*, **26**, pp. 967-969.

[6] Wang, K., J. Chiasson, M. Bodson and L. M. Tolbert, "A nonlinear least-squares approach for identification of the induction motor parameters" (2004), *Proceedings of 43rd IEEE conference on Decision and Control*, pp. 3856-3861

TABLE I

NORMALIZED ITERATIVE ESTIMATE AT FIRST 5 ITERATIONS

$\mathbf{y}^{*T}$ $\mathbf{x}^{*T}$	$(0.5000, 0.7006, 0.5090)$ $(-3, 2, 5, -1)$		
$\mathbf{y}_0^T$	$(5, 6, 7)$		
v_i	[-0.025,0.025]	[-0.05,0.05]	[-0.5, 0.5]
$\mathbf{y}_1^T$ $\mathbf{x}_1^T$	$(0.4993, 0.6955, 0.5168)$ $(-0.2569, 0.1715,$ $0.4750, -0.1035)$	$(0.4974, 0.6954, 0.5186)$ $(-0.2778, 0.1978,$ $0.4751, -0.1158)$	$(0.4971, 0.7082, 0.5013)$ $(-0.2956, 0.1964,$ $0.4800, -0.0974)$
$\mathbf{y}_2^T$ $\mathbf{x}_2^T$	$(0.5002, 0.7002, 0.5094)$ $(-2.9917, 1.9910,$ $5.0019, -1.0033)$	$(0.5004, 0.6998, 0.5098)$ $(-2.9990, 2.0018,$ $5.0046, -1.0161)$	$(0.4962, 0.7142, 0.4936)$ $(-2.9878, 2.0186,$ $4.9870, -1.0184)$
$\mathbf{y}_3^T$ $\mathbf{x}_3^T$	$(0.5002, 0.7004, 0.5091)$ $(-2.9989, 1.9991,$ $5.0009, -1.0009)$	$(0.5005, 0.7004, 0.5089)$ $(-3.0012, 1.9983,$ $5.0009, -1.0001)$	$(0.4961, 0.7149, 0.4927)$ $(-2.9792, 2.0144,$ $4.9838, -1.0145)$
$\mathbf{y}_4^T$ $\mathbf{x}_4^T$	$(0.5002, 0.7004, 0.5091)$ $(-2.9992, 1.9994,$ $5.0009, -1.0008)$	$(0.5005, 0.7004, 0.5089)$ $(-3.0010, 1.9978,$ $5.0006, -0.9987$	$(0.4961, 0.7150, 0.4926)$ $(-2.9782, 2.0139,$ $4.9835, -1.0140)$
$\mathbf{y}_5^T$ $\mathbf{x}_5^T$	$(0.5002, 0.7004, 0.5091)$ $(-2.9992, 1.9994,$ $5.0009, -1.0008)$	$(0.5005, 0.7004, 0.5089)$ $(-3.0010, 1.9978,$ $5.0005, -0.9985)$	$(0.4961, 0.7150, 0.4926)$ $(-2.9781, 2.0139,$ $4.9835, -1.0140)$

TABLE II

ESTIMATES BY THE OVER-PARAMETRIZATION METHOD

v_i	$\mathbf{y}^T$	$\mathbf{x}^T$
$[-0.025, 0.025]$	$(0.4999, 0.7008, 0.5089)$	$(-2.9989, 1.9991, 5.0004, -1.0000)$
$[-0.05, \quad 0.05]$	$(0.4999, 0.7007, 0.5091)$	$(-3.0019, 1.9986, 5.0013, -0.9970)$
$[-0.5, \quad 0.5]$	$(0.4838, 0.7090, 0.5130)$	$(-3.0573, 2.0241, 5.0036, -0.9515)$

TABLE III

ESTIMATES BY THE NUMERICAL METHOD

v_i	$\mathbf{y}^T$	$\mathbf{x}^T$
$[-0.025, 0.025]$	$(0.4998, 0.7007, 0.5091)$	$(-2.9998, 2.0020, 5.0006, -1.0028)$
$[-0.05, \quad 0.05]$	$(0.4996, 0.7005, 0.5095)$	$(-2.9996, 2.0006, 4.9969, -0.9981)$
$[-0.5, \quad 0.5]$	$(0.4988, 0.7011, 0.5096)$	$(-2.9622, 1.9421, 5.0625, -1.0406)$