Identification of Hammerstein-Wiener Systems in Closed-loop

Hajime Ase* and Tohru Katayama†
* Mathematical Technology Institute, Chofu, Tokyo 182-0014, Japan
† Professor Emeritus Kyoto University, Kyoto 606-8501, Japan
E-mail: tohru_katayama(at)nifty.com

Abstract

This paper presents a method of identifying closed-loop Hammerstein-Wiener systems, where a linear plant in the forward loop is stable or semi-stable. In the framework of the direct approach [17], an output error (OE) model is derived with the help of basis function expansions of nonlinearities. First an approximate linear state space model of the forward loop is identified using a subspace identification method. Then, initialized by the identified linear model, the mean square of the output error is optimized by using a gradient-based search. Some numerical results are included to show the applicability of the proposed method.

1 Introduction

A number of papers have been published for the identification of Hammerstein and Hammerstein-Wiener (NLN) systems [2, 27, 8, 26, 20, 25, 1], and several methods have been developed for the closed-loop identification of Hammerstein systems. In fact, a subspace identification method is developed for MIMO closed-loop Hammerstein systems [15]. Also, the instrumental variables (IV) method [16, 12] and an iterative prediction error method (PEM) [13] are applied to the closed-loop identification of Hammerstein systems. To the best of our knowledge, however, less attention has been paid to the identification of NLN systems in closed-loop unstable plants. Thus, we limit our attention to the identification of Hammerstein-Wiener systems with stable or semi-stable plants in the forward loop.

This paper is organized as follows. The problem is stated in Section 2. In Section 3, the procedure of identifying Hammerstein-Wiener systems in closed-loop is outlined. In Section 4, by introducing the basis function expansions of nonlinearities, an OE model for the forward loop is derived, and the optimization of the MSE of output is discussed. It is shown that the optimization problem is reduced to a separable least-squares, which is solved using a gradient method. The overall algorithm is summarized in Section 5. Section 6 gives some numerical results to show the applicability of the technique, and Section 7 concludes the paper.

2 Problem statement

Consider a scalar discrete-time NLN system in closed-loop in Fig. 1, where \((r_1(t), r_2(t))\) are the exogenous inputs, \(u(t)\) is the control input, \(y(t)\) is the output, and \(\nu(t)\) is the output noise, and where \(G(z)\) and \(K(z)\) are linear subsystems, called the plant and the controller, respectively, and \(f(\cdot)\) and \(h(\cdot)\) are nonlinearities.

![Fig. 1: A closed-loop NLN system.](image)

The following assumptions are made for the NLN system in closed-loop.

Assumption 1

(i) The exogenous inputs \((r_1(t), r_2(t))\) are zero-mean stationary Gaussian processes with finite covariances, where \(r_2(t)\) is persistently exciting (PE).

(ii) The output noise \(\nu(t)\) is a zero-mean stationary white Gaussian process uncorrelated with the exogenous inputs \((r_1(t), r_2(t))\).
The linear system $G(z)$ is strictly proper, and is stable, or semi-stable.

(iv) The controller $K(z)$ is proper, but the knowledge of $K(z)$ is not needed in identification.

The processes $u(t), v(t), w(t), y(t)$ are 2nd-order stationary; this implies that the closed-loop system is “stable.”

(v) The nonlinearity functions $f(\cdot)$ and $h(\cdot)$ are measurable functions with $f(0) = 0$ and $h(0) = 0$, satisfying $E[f^2(v)] < \infty$, $E[h^2(w)] < \infty$, where $E[\cdot]$ denotes the mathematical expectation.

Let $G(z) = (A, B, C)$ be a minimal realization of the plant. Then, the NLN system in the forward loop is expressed as:

\[
x(t+1) = Ax(t) + Bh(u(t))
\]

\[
w(t) = Cx(t)
\]

\[
y(t) = f(w(t)) + \nu(t)
\]

where $x(t) \in \mathbb{R}^n$ is the state vector, and the feedback control $u(t)$ is given by

\[
u(t) = r_2(t) + K(r_1(t) - y(t))
\]

The problem is to develop a direct method of identifying the NLN system in the closed loop of Fig. 1. More precisely, we identify a state space model of $G(z)$ and models of nonlinearities $f(\cdot)$ and $h(\cdot)$ based on a set of input-output data $\{u(t), y(t) \mid t = 1, \ldots, N\}$ by optimizing the OE model, where the exogenous inputs $(r_1(t), r_2(t))$ are not used in the direct identification method [17].

Note that we cannot identify the three blocks independently in the NLN structure of Fig. 1 from $u(t)$ and $y(t)$, since for any constants $a, b, c$ with $abc = 1$, the input-output characteristics of two realizations $(h, G, f)$ and $(ah, bG, cf)$ are the same. Thus, we should normalize the gain of each block to get a final representation.

3 Outline of identification procedure

For the iterative PEM identification, we employ the OE model of Fig. 2, of which derivation is provided in Section 4. The output error $\varepsilon(t)$ is the difference between the observed output $y(t)$ and the model output $\hat{y}^0(t)$, so that $\varepsilon(t) = y(t) - \hat{y}^0(t)$. Note that to get bounded model outputs $\hat{y}^0(t)$ in the OE model, the matrix $A$ must be stable or semi-stable.

The problem is to minimize the sample MSE $V_N := \frac{1}{N} \sum_{t=1}^{N} \varepsilon^2(t)$, which is non-convex with respect to model parameters. Thus, we need good initial estimates to get good identification results.

The identification procedure is divided into the following two steps.

**Initialization.** We apply the ORT method [14] or PO-MOESP method [22] to the input-output data $\{u(t), y(t)\}$ in order to obtain an approximate state space model $\hat{G}_d(z) = (A^{(0)}, B^{(0)}, C^{(0)})$ of the plant in the forward loop [1].

Note that the estimates of semi-stable poles by the subspace methods often lie outside the unit circle, generating unbounded $\hat{y}^{(0)}(t)$. To avoid this, we use stabilized subspace methods by incorporating a stabilization technique [5]; see Subsection 6.3.

**Optimization.** Then, initialized by the approximate linear model $(A^{(0)}, B^{(0)}, C^{(0)})$, we optimize the sample MSE $V_N$ to find the estimates of the system parameters and of the coefficients of basis function expansion of nonlinearities. The problem is reduced to a separable least-squares [9, 4], which is solved by using a DDLCE-based gradient search [18, 22, 24].

4 Identification of NLN system in closed-loop

Given a linear approximation of the forward loop in Step 1, we consider the problem of identifying nonlinearities and of improving the estimates of the linear system.

**Assumption 2** The nonlinearities are approximated by a basis function expansion, for which we can use any basis functions. In this paper, the Legendre polynomial expansion of finite order is employed [11].

4.1 Basis function expansion of nonlinearities

Suppose that the nonlinearities $h(\cdot)$ and $f(\cdot)$ are respectively approximated as

\[
v(t) = h(u(t)) \simeq \sum_{l=1}^{L} \alpha_l \zeta_l(u(t))
\]

\[
y^0(t) = f(w(t)) \simeq \sum_{l=1}^{L} \beta_l \zeta_l(w(t))
\]

where $\zeta_l(\cdot), l = 1, \ldots, L$ are basis functions.

Let $\zeta(u(t)) := [\zeta_1(u(t)) \cdots \zeta_L(u(t))]^T \in \mathbb{R}^L$, and $\alpha := [\alpha_1 \cdots \alpha_L]^T \in \mathbb{R}^L$. Let the overparametrized matrix be defined as [2]

\[
\tilde{B} := B \alpha^T = \begin{bmatrix} b_1 \alpha_1 & \cdots & b_1 \alpha_L \\ \vdots & \ddots & \vdots \\ b_n \alpha_1 & \cdots & b_n \alpha_L \end{bmatrix} \in \mathbb{R}^{n \times L}
\]
Moreover, define
\[ \theta := \begin{bmatrix} \text{vec}(A) \\ \text{vec}(B) \\ \text{vec}(C) \end{bmatrix} \in \mathbb{R}^{n_{\theta}}, \quad n_{\theta} := n^2 + (L + 1)n \]
where the operator vec(·) stacks all the column vectors of a matrix on top of each other.

Since, from (3), the estimate of \( v(t) \) is expressed as \( \hat{v}(t) = A\hat{x}(t, \theta) + \hat{B}\zeta(u(t)) \), we see from (1a) and (1b) that
\[
\hat{x}(t+1, \theta) = Ax(t, \theta) + \hat{B}\zeta(u(t)) \\
\hat{w}(t, \theta) = C\hat{x}(t, \theta)
\]
where \( \hat{x}(t, \theta) \) and \( \hat{w}(t, \theta) \) indicate their dependence on the parameter vector \( \theta \) and on the basis functions expansions of nonlinearities. It also follows from (4) that the noise-free output is approximated as
\[
\hat{y}^W(t) = \zeta^T(\hat{w}(t, \theta))\beta
\]
where \( \beta := [\beta_1 \cdots \beta_L]^T \in \mathbb{R}^{L} \).

#### 4.2 Optimization of MSE

For simplicity, define \( r(t) := \begin{bmatrix} r_1(t) \\ r_2(t) \end{bmatrix} \) and \( U_{t-1} := \begin{bmatrix} u(t-1) \cdots u(t-2) \cdots u(1) \end{bmatrix} \in \mathbb{R}^{N-1} \); also \( R_{t-1}, Y_{t-1} \) and \( N_{t-1} \) are defined similarly. We see from (1a) and (1b) that \( w(t) \) is a function of the past inputs \( U_{t-1} \). Hence, from (1c), the output is described by
\[
y(t) = \varphi_t(U_{t-1}) + \nu(t)
\]
where \( \varphi_t : \mathbb{R}^{N-1} \rightarrow \mathbb{R} \) is a nonlinear mapping. Similarly, \( \hat{w}(t, \theta) \) is a function of \( U_{t-1} \); we see from (6) that
\[
\hat{y}^W(t) = \hat{\varphi}_t(U_{t-1}, \theta, \beta)
\]
where \( \hat{\varphi}_t : \mathbb{R}^{N-1} \rightarrow \mathbb{R} \) is a nonlinear mapping for fixed \((\theta, \beta)\). Thus the output error is expressed as
\[
\varepsilon(t) = [\varphi_t(U_{t-1}) - \hat{\varphi}_t(U_{t-1}, \theta, \beta)] + \nu(t)
\]
(7)

For simplicity, we write \( \varphi_t := \varphi_t(U_{t-1}) \) and \( \hat{\varphi}_t := \hat{\varphi}_t(U_{t-1}, \theta, \beta) \). Note from (2) that \( U_{t-1} \) is a function of past \( R_{t-1} \) and \( Y_{t-1} \), where the latter is disturbed by the past output noise \( N_{t-1} \). Since \( \nu(t) \) is uncorrelated with \( U_{t-1} \), we have \( E\{[\varphi_t - \hat{\varphi}_t]\nu(t)\} = 0 \) in (7). Thus,
\[
E[\varepsilon^2(t)] = E\{(\varphi_t - \hat{\varphi}_t)^2\} + \sigma^2_v
\]
Suppose that \( \varepsilon(t) \) is ergodic. Then, for a sufficiently large \( N \), the sample MSE is expressed as
\[
\frac{1}{N} \sum_{t=1}^{N} \varepsilon^2(t) \approx \frac{1}{N} \sum_{t=1}^{N} (\varphi_t - \hat{\varphi}_t)^2 + \sigma^2_v
\]
This implies that minimizing the sample MSE is nearly equivalent to minimizing \( \frac{1}{N} \sum_{t=1}^{N} (\varphi_t - \hat{\varphi}_t)^2 \) [21].

If there exist a set of model parameters \((\hat{\theta}, \hat{\beta})\) such that \( E\{(\varphi_t - \hat{\varphi}_t)^2\} = 0 \) holds, we say that the model set includes the “true” model [17]. In this case, the minimum MSE will be close to \( \sigma^2_v \); otherwise it is larger than \( \sigma^2_v \). Thus, we see that \( \sigma^2_v \) is a lower bound of the minimum MSE.

#### 4.3 Separable least-squares

Define the augmented \( N \)-dimensional vectors
\[
\mathcal{Y} := \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \quad \hat{\mathcal{Y}}^0 := \begin{bmatrix} \hat{y}^0(1) \\ \vdots \\ \hat{y}^0(N) \end{bmatrix}, \quad \mathcal{E} := \begin{bmatrix} \varepsilon(1) \\ \vdots \\ \varepsilon(N) \end{bmatrix}
\]
and the \( N \times L \) regression matrix
\[
\Phi(\theta) := \begin{bmatrix} \zeta_1(\hat{w}(1, \theta)) & \cdots & \zeta_L(\hat{w}(1, \theta)) \\ \vdots & \ddots & \vdots \\ \zeta_1(\hat{w}(N, \theta)) & \cdots & \zeta_L(\hat{w}(N, \theta)) \end{bmatrix} \in \mathbb{R}^{N \times L}
\]
Then, it follows that the output error vector is expressed as
\[
\mathcal{E}(\theta, \beta) = \mathcal{Y} - \Phi(\theta)\beta
\]
Thus, the sample MSE \( \frac{1}{N} \sum_{t=1}^{N} \varepsilon^2(t) = \frac{1}{N} \| \mathcal{E}(\theta, \beta) \|^2 \) becomes
\[
V_N(\theta, \beta) = \frac{1}{N} \| \mathcal{Y} - \Phi(\theta)\beta \|^2
\]
(8)
We minimize (8) with respect to \((\theta, \beta)\). Since \( \mathcal{E}(\theta, \beta) \) is linear with respect to \( \theta \), the optimization problem is called a separable least-squares [9]. Since \( \beta = \Phi^T(\theta)\mathcal{Y} \) minimizes (8) for a fixed \( \theta \), the output error vector becomes \( \mathcal{E}(\theta) = (I_N - \Phi(\theta)\Phi^T(\theta))\mathcal{Y} \) where \( -1 \) denotes the pseudo-inverse [10]. Thus, (8) reduces to
\[
V_N(\theta) = \frac{1}{N} \| (I_N - \Phi(\theta)\Phi^T(\theta))\mathcal{Y} \|^2
\]
(9)
which is called the variable projection functional. We minimize (9) with respect to \( \theta \), while the parameter \( \beta \) is recovered from \( \hat{\beta} = \Phi^T(\theta)\mathcal{Y} \).

#### 4.4 Gradient method

We can show [9] that the gradient of the output error vector with respect to \( \theta \) is given by
\[
\frac{\partial \mathcal{E}(\theta)}{\partial \theta} = -[\mathbb{D}(\theta)]^{-1}\mathcal{Y} \in \mathbb{R}^{N \times n_{\theta}}
\]
(10)
where \( \mathbb{D}(\theta) \) is a three dimensional tensor. Since \( \Phi(\theta) = \zeta(\hat{w}(t, \theta)) \) by definition, we have
\[
[\mathbb{D}(\theta)]_{l_1 l_2 l_3} = \frac{\partial\zeta_l(\hat{w}(t, \theta))}{\partial \theta_j} = \zeta_l(\hat{w}(t, \theta)) \frac{\partial \hat{w}(t, \theta)}{\partial \theta_j}
\]
(11)
where \( l = 1, \ldots, N; l = 1, \ldots, L; j = 1, \ldots, n_{\theta} \).

Differentiating (5a) and (5b) with respect to \( \theta_j \), we see that the partial derivatives \( \frac{\partial \hat{w}(t, \theta)}{\partial \theta_j} \) satisfy the following sensitivity equations
\[
\frac{\partial \hat{x}(t+1, \theta)}{\partial \theta_j} = A \frac{\partial \hat{x}(t, \theta)}{\partial \theta_j} + \frac{\partial A}{\partial \theta_j} \hat{x}(t, \theta) + \frac{\partial B}{\partial \theta_j} \zeta(u(t))
\]
(12a)
\[
\frac{\partial \hat{w}(t, \theta)}{\partial \theta_j} = C \frac{\partial \hat{x}(t, \theta)}{\partial \theta_j} + \frac{\partial C}{\partial \theta_j} \hat{x}(t, \theta)
\]
(12b)
We compute (5) and (12) for \( t = 1, \cdots, N \). Then, by using (10) and (11), we have the Jacobian

\[
J(\theta) := \frac{\partial \mathcal{E}(\theta)}{\partial \theta} = \begin{bmatrix}
\frac{\partial \mathcal{E}_1(\theta)}{\partial \theta_1} & \cdots & \frac{\partial \mathcal{E}_t(\theta)}{\partial \theta_t} \\
\vdots & \ddots & \vdots \\
\frac{\partial \mathcal{E}_N(\theta)}{\partial \theta_1} & \cdots & \frac{\partial \mathcal{E}_N(\theta)}{\partial \theta_N} 
\end{bmatrix}
\]  

(13)

Let \( \theta^{(r)} \) be the \( r \)th estimate of \( \theta \), i.e.,

\[
\theta^{(r)} := \begin{bmatrix}
\text{vec}(A^{(r)}) \\
\text{vec}(B^{(r)}) \\
\text{vec}(C^{(r)})
\end{bmatrix} \in \mathbb{R}^{n_\theta}
\]

(14)

To use the DDLC parametrization [18], we define

\[
Q^{(r)} = \begin{bmatrix}
(A^{(r)})^T \otimes I_n - I_n \otimes A^{(r)} \\
(B^{(r)})^T \otimes I_n - I_n \otimes C^{(r)}
\end{bmatrix} \in \mathbb{R}^{n_\theta \times n^2}
\]

(15)

where \( \otimes \) denotes the Kronecker product [10]. In the DDLC parametrization, the parameter \( \theta^{(r)} \) should be updated in the directions of the orthogonal complement of \( \text{Im}(Q^{(r)}) \) at each iteration [18, 24, 22]. To find such directions, we compute the SVD of (15), i.e.

\[
Q^{(r)} = U_Q S_Q V_Q^T, \quad U_Q \in \mathbb{R}^{n \times n}
\]

(16)

Since \( P := U_Q(:, n+1 : n_\theta) \in \mathbb{R}^{n \times (n_\theta - n)} \) spans (Im\(Q^{(r)})^\perp \), we define the modified Jacobian

\[
R^{(r)} := J(\theta^{(r)}) P \in \mathbb{R}^{n \times (n_\theta - n^2)}
\]

(17)

Then, the iteration algorithm is given by

\[
d^{(r)} = -[(R^{(r)})^T R^{(r)} + \lambda I]^{-1} (R^{(r)})^T \mathcal{E}(\theta^{(r)})
\]

(18a)

\[
\theta^{(r+1)} = \theta^{(r)} + \sigma P d^{(r)}
\]

(18b)

where \( \lambda > 0 \) is a regularization parameter, and a step size \( \sigma > 0 \) is determined by a simple line search.

**Remark 1** Note that the above gradient method can also be applied to Wiener and Hammerstein systems in closed-loop with minor modifications. \( \square \)

### 5 Algorithm of identifying NLN systems in closed-loop

The algorithm for identifying Hammerstein-Wiener systems in closed-loop is summarized below.

**Identification algorithm**

**Step 1.** Identify a linear forward loop model \( \bar{G}_d(z) = (\bar{A}, \bar{B}, \bar{C}) \) of the NLN system by the ORT or the POMOESP method from given input-output data. Then, define the initial values

\[
\theta^{(0)} := \begin{bmatrix}
\text{vec}(\bar{A}) \\
\text{vec}(\bar{B}) \\
\text{vec}(\bar{C})
\end{bmatrix} \in \mathbb{R}^{n_\theta}
\]

(19)

where \( \bar{B} = \bar{B}(\alpha^{(0)})^T \). Compute \( \beta^{(0)} = \Phi^1(\theta^{(0)}) \mathcal{Y} \), and set \( r = 0 \).

**Remark 2** Suppose that the output nonlinearity \( f(\cdot) \) is linear, e.g., \( f(w) = w \). Then the forward loop of Fig. 1 becomes a Hammerstein system. Thus, for given \( (\bar{A}, \bar{B}, \bar{C}) \), the output error is a linear function with respect to \( \alpha \). Hence, we can easily compute the least-squares estimate \( \hat{\alpha} \), which is used as the initial \( \alpha^{(0)} \) in Step 1 to define \( \bar{B} \). Then, we form the initial regression matrix \( \Phi(\theta^{(0)}) \) by using inputs \( u(t), t = 1, \cdots, N \) to compute \( \beta^{(0)} \).

**Step 2.** Compute (5) and (12) for \( t = 1, \cdots, N \) with initial conditions

\[
\bar{x}(1, \theta^{(r)}) = 0, \quad \frac{\partial \bar{x}(1, \theta^{(r)})}{\partial \theta_j} = 0, \quad j = 1, \cdots, n_\theta
\]

Then, compute the Jacobian \( J(\theta^{(r)}) \) by (13).

**Step 3.** Compute the SVD of (16) to get \( R^{(r)} \) of (17). Then, update the current vector \( \theta^{(r)} \) by using (18a) and (18b).

**Step 4.** If the convergence conditions are satisfied, retrieve the system parameters \( (\bar{A}, \bar{B}, \bar{C}) \) from \( \theta^{(r+1)} \) and compute \( \bar{V}_N(\theta^{(r+1)}) \) of (9). Then, go to Step 5. Otherwise, set \( r := r + 1 \) and go to Step 2.

**Step 5.** The coefficients for \( f(\cdot) \) are given by \( \beta = \Phi^1(\theta^{(r+1)}) \mathcal{Y} \). From the SVD, the rank-one decomposition of \( \bar{B} \) is given by \( \bar{B} \simeq U_1 \tilde{\alpha}_1 V_1^T \), where \( U_1 \in \mathbb{R}^n \) and \( V_1 \in \mathbb{R}^k \) are respectively the left and right singular vectors corresponding to the largest singular value \( \tilde{\alpha}_1 \) of \( \bar{B} \) [2]. Put \( \gamma := V_1^T(1) \). Then, we get the normalized coefficients \( \alpha \) for \( h(\cdot) \) with \( \alpha_1 = 1 \) by \( \alpha := V_1/\gamma \), and define \( B := U_1 \tilde{\alpha}_1 \). Thus, the estimates of \( (\bar{A}, \bar{B}, \bar{C}) \) and \( \alpha, \beta \) are obtained. \( \square \)

### 6 Simulation results

Some simulation results are included to show the applicability of the present procedure.

<table>
<thead>
<tr>
<th>Table 1: Summary of Monte Carlo simulations.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( G(z) )</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>S1</td>
</tr>
<tr>
<td>S2</td>
</tr>
<tr>
<td>S3</td>
</tr>
<tr>
<td>S4</td>
</tr>
<tr>
<td>S5</td>
</tr>
<tr>
<td>S6</td>
</tr>
<tr>
<td>S7</td>
</tr>
</tbody>
</table>

Table 1 shows the summary of simulation results for NLN systems (S1~S5) and Wiener systems (S6~S7). For S1~S5, \( G(z) \) is stable, \( K(z) \) is a PI controller and \( N = 10,000 \); for S6~S7, \( G(z) \) is semi-stable, \( K(z) \) is a proportional plus 1st-order lag controller and \( N = 5,000 \). The second column denotes the order.
of the linear system $G(z)$, and Poly, Sat and DZ in the 3rd- and 4th-column denote polynomial, saturation and deadzone, respectively. We also employ the Legendre polynomial expansion with $L = 3$ for Poly; otherwise $L = 13$. The signal-to-noise ratio is defined as $S/N := 20 \log_{10}(\sigma_{w}/\sigma_{v})$ (dB), and the RMSE is the root of the minimum MSE.

We see that for S1, S3, S6 with polynomial nonlinearities, the performance of identification is quite good; the RMSE is very close to the lower bound $\sigma_{w}$. For other cases, the performance is somewhat degraded. Three cases of S3, S5 and S7 are shown below in more detail.

### 6.1 5th-order linear system with polynomial nonlinearities (S3)

Consider the case where the plant is given by

$$G(z) = \frac{3(0.0275z + 0.0551)}{z^5 - 2.3443z^4 + 3.081z^3 - 2.5274z^2 + 1.2415z - 0.3686}$$

and the PI controller is

$$K(z) = 0.3 \left[1 + \frac{z}{60(z-1)} \right]$$

The nonlinearities are 3rd-order polynomials

$$h(u) = 1.5u - (2/9)u^3$$

$$f(v) = (4/9)v^3$$

<table>
<thead>
<tr>
<th>$\sigma_{r_1}$</th>
<th>$\sigma_{r_2}$</th>
<th>$\sigma_{\nu}$</th>
<th>$\sigma_{w}$</th>
<th>$\sigma_{y_0}$</th>
<th>$\sigma_{y}$</th>
<th>$S/N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.4</td>
<td>0.4889</td>
<td>0.6127</td>
<td>0.3604</td>
<td>0.02</td>
<td>25.1</td>
</tr>
</tbody>
</table>

We assume that the inputs $(r_1(t), r_2(t))$ are mutually uncorrelated white noises with $N(0, \sigma_{r_1}^2)$ and $N(0, \sigma_{r_2}^2)$, respectively, and $\nu(t)$ is also a white noise with $N(0, \sigma_{\nu}^2)$. The simulation conditions are displayed in Table 2, where the standard deviations $\sigma_{u}, \sigma_{w}, \sigma_{y_0}$ are sample values. The number of block rows of data matrices in subspace identification methods is fixed as $k = 15$.

Table 3: RMSE performance for 50 simulation runs.

<table>
<thead>
<tr>
<th>Method</th>
<th>Initial</th>
<th>Optimized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct ORT</td>
<td>Mean 0.07220</td>
<td>0.02012</td>
</tr>
<tr>
<td></td>
<td>Std 0.01654</td>
<td>0.00921</td>
</tr>
<tr>
<td>Direct PO-MOESP</td>
<td>Mean 0.05929</td>
<td>0.02060</td>
</tr>
<tr>
<td></td>
<td>Std 0.02312</td>
<td>0.00954</td>
</tr>
</tbody>
</table>

* Std:= standard deviation.

Table 3 shows simulation results initialized by the direct ORT and PO-MOESP methods, where the initial performance is computed by using $(\theta^{(0)}, \beta^{(0)})$; see Remark 2 in Section 5. Though there are some differences in the initial results by two subspace identification methods, the optimized results are nearly the same.

Also, we see that the performance is quite close to the lower bound. This observation is clearly confirmed by comparing the initial and optimized estimates of poles of Fig. 3 and of Bode magnitudes of Fig. 4. Fig. 5 shows the plots of estimated nonlinearities in the interval $(-1.5, 1.5)$. Though only the results initialized by the direct ORT method are shown, the results by other initializations are nearly the same as can be inferred from Table 3.

### 6.2 5th-order linear system with deadzone-saturation nonlinearities (S5)

We consider the 5th-order system

$$G(z) = \frac{5.5(0.0275z + 0.0551)}{z^5 - 2.3443z^4 + 3.081z^3 - 2.5274z^2 + 1.2415z - 0.3686}$$

Fig. 3: Initial estimates of poles of $G(z)$ by the direct ORT method (left) and optimized estimates (right), where $+$ (black) shows the true poles.

Fig. 4: Initial estimates of Bode magnitude of $G(z)$ by the direct ORT method (left) and optimized ones (right), where the black line indicates the true one.

Fig. 5: Optimized estimates of nonlinearities $h(\cdot)$ (left) and $f(\cdot)$ (right) initialized by the direct ORT method.
where the PI controller is given by

$$K(z) = 0.5 \left[ 1 + \frac{z}{60(z-1)} \right]$$

Also, the input nonlinearity is a deadzone and the output nonlinearity \(f(\cdot)\) is a saturation, i.e.

\[
h(u) = \begin{cases} 
1.25u + 0.5, & u \geq 0.5 \\
0, & -0.5 \leq u \leq 0.5 \\
1.25u - 0.5, & u \leq -0.5 
\end{cases}
\]

\[
f(w) = \frac{w}{\sqrt{0.1 + 0.9w^2}}, \quad -\infty < w < \infty
\]

We employ the Legendre polynomial expansion with \(L = 13\); the true nonlinearities are not in the model set. It is also assumed here that \(\nu(t)\) is a colored noise generated by \(\nu(t) = H(z)\xi(t)\), where \(\xi(t)\) is a white noise with \(N(0, \sigma^2_\xi)\), and the noise filter is given by

\[
H(z) = \frac{z^3 - 1.56z^2 + 1.045z - 0.3338}{z^3 - 2.35z^2 + 2.09z - 0.6675}
\]

We see that though the initial estimates of the Bode magnitude are quite different from the true magnitude, there is no clear difference in the optimized Bode magnitude.

Fig. 6: Initial estimates of the Bode magnitude of \(G(z)\) by the direct ORT (left) and the optimized magnitude (right), where the black line is the true magnitude.

Fig. 7: Optimized estimates of nonlinearities \(h(\cdot)\) (left) and \(f(\cdot)\) (right) initialized by the direct ORT method.

6.3 3rd-order semi-stable linear system (S7)
Consider the 3rd-order semi-stable linear system with a pole at \(z = 1\), i.e.

\[
G(z) = \frac{0.17(-0.3z + 1)}{(z - 1)(z^2 - 1.6z + 0.8)}
\]

To consider a Wiener system, let \(h(u)\) be linear, i.e. \(h(u) = u\), and the output nonlinearity be the saturation given by (20). It is also assumed that the controller is given by

\[
K(z) = \frac{0.1z + 0.3}{z + 0.9}
\]

which stabilizes responses in the closed-loop system.

Table 7 displays the RMSE performance in the Wiener system in closed-loop. Table 7 displays the performance of identification by two initialization methods. The optimized performance is quite similar for these two initializations; but it is quite larger than the lower bound \(\sigma_v = 0.041\). This is partly because the input \(u(t)\) is correlated with the colored noise \(\nu(t)\), and partly because the both nonlinearities are not in the model set.

We see that though the initial estimates of the Bode magnitude are quite different from the true magnitude, there is no clear difference in the optimized Bode magnitude in Fig. 6. Also, as shown in Fig. 7, the quality of the estimates of the deadzone is quite good, but we see some degradation in the estimates of the saturation nonlinearity.

Table 4: Simulation condition for 5th-order linear system with colored output noise and deadzone/saturation nonlinearities (S5).

<table>
<thead>
<tr>
<th>(\sigma_{r_1})</th>
<th>(\sigma_{r_2})</th>
<th>(\sigma_u)</th>
<th>(\sigma_w)</th>
<th>(\sigma_{y_0})</th>
<th>(\sigma_v)</th>
<th>(S/N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.45</td>
<td>0.5949</td>
<td>0.5950</td>
<td>0.7102</td>
<td>0.041</td>
<td>24.8</td>
</tr>
</tbody>
</table>

Table 5: RMSE performance for 50 simulation runs.

<table>
<thead>
<tr>
<th></th>
<th>Initial</th>
<th>Optimized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct approach</td>
<td>Mean</td>
<td>0.18984</td>
</tr>
<tr>
<td>ORT</td>
<td>Std</td>
<td>0.00765</td>
</tr>
<tr>
<td>Direct approach</td>
<td>Mean</td>
<td>0.19050</td>
</tr>
<tr>
<td>PO-MOESP</td>
<td>Std</td>
<td>0.00770</td>
</tr>
</tbody>
</table>

Table 6: Simulation condition for 3th-order linear system with a saturation nonlinearity (S7).

<table>
<thead>
<tr>
<th>(\sigma_{r_1})</th>
<th>(\sigma_{r_2})</th>
<th>(\sigma_u)</th>
<th>(\sigma_w)</th>
<th>(\sigma_{y_0})</th>
<th>(\sigma_v)</th>
<th>(S/N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.4</td>
<td>0.4826</td>
<td>0.5719</td>
<td>0.7502</td>
<td>0.040</td>
<td>25.5</td>
</tr>
</tbody>
</table>
Table 7: RMSE performance for 50 simulation runs.

<table>
<thead>
<tr>
<th></th>
<th>Initial</th>
<th>Optimized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct approach</td>
<td>Mean 0.097868</td>
<td>0.045048</td>
</tr>
<tr>
<td></td>
<td>Std 0.078190</td>
<td>0.001784</td>
</tr>
<tr>
<td>ORT</td>
<td>Mean 0.133558</td>
<td>0.045480</td>
</tr>
<tr>
<td></td>
<td>Std 0.027579</td>
<td>0.003812</td>
</tr>
</tbody>
</table>

disk $|z| \leq 0.999$; otherwise, no corrections are made. Though there are some differences in the two initializations, we see that the optimized RMSE performance is nearly the same and quite good; the RMSE has increased about 12% from the lower bound.

Fig. 8: Initial estimates of poles by the direct ORT method (left) and the optimized poles (right).

Figs. 8, 9 and 10 respectively depict the initial estimates of poles, Bode magnitude and nonlinearity by the ORT method, and optimized estimates. There is one very bad initial estimate of the nonlinearity as shown in Fig. 10 (left), but we see that the optimized estimates of the nonlinearity as well as poles and Bode magnitude are quite good.

Fig. 9: Initial estimates of Bode magnitude (left) of the linear system by the direct ORT method and the optimized estimates (right).

Fig. 10: Initial estimates of the saturation nonlinearity by the direct ORT (left) and the optimized estimates (right).

Table 8: Computation time per iteration.

<table>
<thead>
<tr>
<th>$n$</th>
<th>Poly ($L = 3$)</th>
<th>DZ/Sat ($L = 13$)</th>
<th>$N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1/S2</td>
<td>2</td>
<td>7.8 sec</td>
<td>10,000</td>
</tr>
<tr>
<td></td>
<td>(L = 3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S3/S5</td>
<td>5</td>
<td>14.7 sec</td>
<td>10,000</td>
</tr>
<tr>
<td></td>
<td>(L = 13)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S6/S7</td>
<td>3</td>
<td>2.4 sec</td>
<td>5,000</td>
</tr>
</tbody>
</table>

7 Conclusions

We have developed a closed-loop identification method of Hammerstein-Wiener systems, where it is assumed that the linear plant in the forward loop is stable or semi-stable. The idea of identification is quite simple: first we use the direct approach based on the ORT or PO-MOESP subspace method to obtain the initial linear state-space models of the forward loop. Then, initialized by the identified linear models, the optimization of the OE model is performed by using a version of the DDLC-based gradient search. Several simulation studies are carried out to show the feasibility of the proposed method; we see that the performance is quite good for all initializations.

Table 8 displays the approximate computation time for each iteration of the gradient based search by a desktop (Intel Core i7 3.40GHz, 4GB). Note that for S6/S7, the forward loop is a Wiener system with the output nonlinearity only. The computation time depends on the numbers of data $N$, the dimension of the system $n$ and the order $L$ of basis function expansions.

We see that the computational load to solve the optimization problems is rather heavy. For example, since usually 20 iterations are needed for convergence, it takes about $50 \times 20 = 1000$ sec, or 17 min for each run of S2. Thus, the computational efficiency should be improved by using a more advanced numerical procedure.

Also, the method of the present paper cannot be applied, if the linear plant in the forward loop is unstable. To develop a method of identifying Hammerstein-Wiener systems with an unstable linear plant, we need a more sophisticated model that simulates the closed loop systems using the information of the controller and exogenous inputs. This is a future topic of our study, for which gradient expressions for a closed-loop system may be applied [6].
References


