Modeling and parameter learning method for the Hammerstein–Wiener model with disturbance

Feng Li¹, Lianyu Chen¹, Songlin Wo¹, Shengquan Li² and Qingfeng Cao²

Abstract
In this paper, a novel modeling and parameter learning method for the Hammerstein–Wiener model with disturbance is proposed, and the Hammerstein–Wiener model is implemented to approximate complex nonlinear industrial processes. The proposed Hammerstein–Wiener model has two static nonlinear blocks represented by two independent neuro-fuzzy models that surround a dynamic linear block described by the finite impulse response model. The parameter learning method of the Hammerstein–Wiener model with disturbance can be summarized in the following three steps: First, the designed input signals are implemented to completely separate the parameter learning problem of output nonlinear block, linear block, and input nonlinear block. Meanwhile, the static output nonlinear block parameters can be learned based on input and output data of two sets of separable signals with different sizes. Second is to determine the dynamic linear block parameter using correlation analysis algorithm using one set of separable signal; thus, the process disturbance can be compensated by the calculation of correlation function. The final one is to achieve unbiased estimation of the static input nonlinear block parameters using least squares method according to the input–output data of random signal. Furthermore, with the parameter learning method, the proposed model can achieve less computation complexity and good robustness. The simulation results of two cases are provided to demonstrate the advantage of the proposed modeling and parameter learning method.

Keywords
Hammerstein–Wiener model, process disturbance, designed input signals, parameter learning

Date received: 24 December 2019; accepted: 23 February 2020

Introduction
Almost all practical industrial processes have nonlinear characteristics to some extent.¹² For decades, many methodologies have been carried out in the field of nonlinear dynamic system modeling and identification, for example, Volterra series,³ block-oriented nonlinear models,⁴⁻¹² neural networks,¹³,¹⁴ support vector machines,¹⁵,¹⁶ and Markov jump systems.¹⁷ Among these methodologies, block-oriented nonlinear models have received widespread attention due to their simple structure and excellent modeling ability.¹⁸

The Hammerstein and Wiener models represent the most familiar models of the nonlinear block-oriented models. The extension of the Hammerstein and Wiener models is the Hammerstein–Wiener model, which constituted by one dynamic linear block surrounded by two static nonlinear blocks has a great flexibility for describing practical nonlinear systems, such as electric arc furnace system,¹⁹ pH neutralization process,²⁰,²¹ continuous stirred tank reactor (CSTR),²² and fermentation bioreactor system.²³ There is an extensive body of research with regard to various learning algorithms of the Hammerstein–Wiener model, mainly including over-parameterization method,²⁴ subspace method,²³ blind method,²⁵ iterative method,²⁶,²⁷ multi-signal based method,²⁸ and maximum likelihood method.²⁹

In the work by Bai,²⁴ a two-stage parameter learning method based on least squares method and singular value decomposition technology was proposed for the Hammerstein–Wiener model: the first stage carries out parameter learning of an augmented model, and the
second stage separates the unknown parameters of the linear and nonlinear parts using singular value decomposition. Vörös developed recursive least squares (RLS) algorithms for system identification and parameter learning of the Hammerstein–Wiener model with dead-zone nonlinearity as input block and polynomial function nonlinearity as output block as in the work by Yu et al. Zhang et al. developed recursive least squares (RLS) algorithm to learn the parameters of the Hammerstein–Wiener model using the key-term separation principle, and the adaptive controller was designed. However, the common representation of the methods mentioned is that the Hammerstein–Wiener model’s learning process contains unknown parameter product of static nonlinear blocks and dynamic linear block, which increases the learning complexity and computational burden. Furthermore, The least squares methods are the dominant algorithms for system identification and parameter learning of nonlinear dynamic models owing to their simplicity in concept and convenience in implementation. Recently, the RLS methods have been applied for the parameter learning of the Hammerstein–Wiener model. It is acknowledged that RLS methods are unable to generate consistent estimation of the model parameters when the model involves process noise or measurement noise. For this reason, various modified RLS learning schemes have been developed. Based on the hierarchical identification principle and the auxiliary model identification idea, Wang and Ding presented hierarchical least squares method to learn the Hammerstein–Wiener output error model parameter. Bias compensation principle is combined with the singular value decomposition method for learning the Hammerstein–Wiener model, and an improved online two-stage identification algorithm is introduced in the work by Li et al. By extending the simplified and refined instrumental variable method, Allafi et al. studied the instrumental variable least squares parameter estimation algorithm for the fractional-order Hammerstein–Wiener model. It should be noted that the abovementioned results assumed that unknown static nonlinearities are of polynomial forms. Consequently, if the nonlinear blocks are not polynomial forms or not continuous functions, the approaches do not converge.

To circumvent the aforementioned problem, fuzzy model and neural network model have been developed for representing nonlinear blocks of the Hammerstein–Wiener model because of their strong approximation ability for nonlinear functions. Therefore, it is worth emphasizing that the parameter learning of fuzzy systems or neural networks–based Hammerstein–Wiener model has essential differences from the parameter learning of single fuzzy model or single neural network model, the difference being that the model structure constraint was imposed by the nonlinear Hammerstein–Wiener model. The parameter learning problem of the Hammerstein–Wiener model needs to take into account nonlinear transformations and linear dynamics altogether, while the parameter learning of the traditional neuro-fuzzy systems only focuses on one global nonlinear transformation. Chen and Wang reported a Hammerstein–Wiener recurrent neural network, in which systematic identification algorithm is used to identify unknown dynamic nonlinear systems. Combining the back-propagation technique with stochastic gradient algorithm, Abouda et al. researched the parameter learning method of estimating jointly unknown parameters and the intermediate variable of fuzzy Hammerstein–Wiener model. However, neural network models have strong self-learning and self-adaptive ability, but lacks transparency and cannot well express the reasoning capability of human brain, while the fuzzy system has no self-adaptive ability, and it has some boundedness in practical applications. The neuro-fuzzy model integrates the learning mechanism of neural networks with the ability of fuzzy inference technology, which can better describe the nonlinear dynamic systems than neural networks or fuzzy systems. As a result, a significant idea is to use the neuro-fuzzy model as static nonlinear blocks of the Hammerstein–Wiener model.

To deal with the issues described above, a novel modeling and parameter learning method for the Hammerstein–Wiener model with disturbance is considered, whose two static nonlinearities are represented using two independent four-layer neuro-fuzzy models, while a dynamic linear block is described using the finite impulse response model. The designed input signals are utilized to achieve the decoupling of parameter learning of output nonlinear block, linear block, and input nonlinear block. First, the output static nonlinear block parameters can be estimated using two sets of separable signals with different sizes. Moreover, correlation analysis approach is developed for determining the unknown parameters of linear block. Finally, parameter unbiased estimation of the static input nonlinear block is achieved using least squares method based on input and output random signals. The main contributions of this paper are as follows:

1. In our previous work, the input signals that consist of binary signal and random multistep
signal are used to identify the Hammerstein–Wiener model. In contrast, in this work, the first part of the special input signal is extended to include more general independent separable signals, such as binary signal, sine signal, and Gaussian signal.

2. Unlike the synchronous parameter learning methods, the designed input signals are implemented to completely separate the parameter learning problem of output nonlinear block, linear block, and input nonlinear block, which significantly simplifies the parameter learning process of the Hammerstein–Wiener model.

3. The correlation analysis algorithm is employed to learn the unknown parameters of the dynamic linear block; thus, the process disturbance can be compensated by the calculation of correlation function.

The modeling and parameter learning problem of the Hammerstein–Wiener model is briefly presented in section “The neuro-fuzzy Hammerstein–Wiener model.” The three-stage parameter learning algorithms of the neuro-fuzzy Hammerstein–Wiener model with disturbance are discussed carefully in section “Parameter learning of the neuro-fuzzy Hammerstein–Wiener model.” Section “Simulation results and analysis” gives the simulation results of complex nonlinear industrial processes. Finally, we draw some conclusions in section “Conclusion.”

The configuration of neuro-fuzzy Hammerstein–Wiener model

The Hammerstein–Wiener model

As depicted in Figure 1, we consider the following Hammerstein–Wiener model with disturbance that consists of one dynamic linear block, \( H(\cdot) \), surrounded by input static nonlinear block, \( f(\cdot) \), and output static nonlinear block, \( g(\cdot) \).

The Hammerstein–Wiener model, defined in Figure 1, is expressed by the relationships of three blocks

\[
\begin{align*}
\text{input} & : u(k) \\
\text{linear block} & : f(\cdot) \\
\text{output} & : g(\cdot)
\end{align*}
\]

where \( f(\cdot) \) is the static input nonlinear block; \( H(\cdot) \) is the dynamic linear block; \( g(\cdot) \) is the static output nonlinear block; \( c(\cdot) \) is a white noise with zero mean and unknown variance \( \sigma^2 \); \( v(k) \), \( x(k) \), and \( z(k) \) are the intermediate variables of the model; and \( u(k) \) and \( y(k) \) denote the input and output, respectively.

In the dynamic block, the current output depends on its past values. The dynamic linear block can be described as a finite impulse response, space state model, transfer function, and so on. Here, a finite impulse response model is considered, that is, \( B(z) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n_b} \), where \( z^{-1} \) represents the unit delay operator and \( n_b \) represents the model order.

In the parameter learning method of the Hammerstein–Wiener model, a crucial challenge is to discover an effective representation of static nonlinear blocks. A good modeling should not only correctly represent the static nonlinear block but also simplify the parameter learning procedure. By analyzing the superiorities of fuzzy systems and neural network models in section “Introduction,” the proposed neuro-fuzzy models are applied to the Hammerstein–Wiener model with disturbance in the next study.

The neuro-fuzzy Hammerstein–Wiener model

Four-layer neuro-fuzzy models are designed to formulate static input nonlinear block and static output nonlinear block, as shown in Figure 2.

Furthermore, the intermediate variables \( v(k) \) and \( z(k) \) are defined as

\[
\begin{align*}
v(k) &= f(u(k)) \\
x(k) &= H(z)v(k) \\
z(k) &= x(k) + c(k) \\
y(k) &= g(z(k))
\end{align*}
\]

where \( f(\cdot) \) is the static input nonlinear block; \( H(\cdot) \) is the dynamic linear block; \( g(\cdot) \) is the static output nonlinear block; \( c(\cdot) \) is a white noise with zero mean and unknown variance \( \sigma^2 \); \( v(k) \), \( x(k) \), and \( z(k) \) are the intermediate variables of the model; and \( u(k) \) and \( y(k) \) denote the input and output, respectively.

In the dynamic block, the current output depends on its past values. The dynamic linear block can be described as a finite impulse response, space state model, transfer function, and so on. Here, a finite impulse response model is considered, that is, \( B(z) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_n z^{-n_b} \), where \( z^{-1} \) represents the unit delay operator and \( n_b \) represents the model order.

In the parameter learning method of the Hammerstein–Wiener model, a crucial challenge is to discover an effective representation of static nonlinear blocks. A good modeling should not only correctly represent the static nonlinear block but also simplify the parameter learning procedure. By analyzing the superiorities of fuzzy systems and neural network models in section “Introduction,” the proposed neuro-fuzzy models are applied to the Hammerstein–Wiener model with disturbance in the next study.

The neuro-fuzzy Hammerstein–Wiener model

Four-layer neuro-fuzzy models are designed to formulate static input nonlinear block and static output nonlinear block, as shown in Figure 2.

Furthermore, the intermediate variables \( v(k) \) and \( z(k) \) are defined as

\[
\begin{align*}
v(k) &= f(u(k)) = \sum_{l=1}^{L_{input}} \phi_l^{input}(u(k)) \ w_l^{input} \\
z(k) &= g^{-1}(y(k)) = \sum_{l=1}^{L_{output}} \phi_l^{output}(y(k)) \ w_l^{output}
\end{align*}
\]

where \( \phi_l^{input}(u(k)) = (\mu_l(u(k)))(\sum_{l=1}^{L_{input}} \mu_l(u(k))) \) and \( \phi_l^{output}(y(k)) = (\mu_l(y(k)))(\sum_{l=1}^{L_{output}} \mu_l(y(k))) \), in which \( \mu_l = \exp(- (u(k) - c_l)^2) / \sigma^2_l \), \( c_l \) is the center, \( \sigma_l \) is the
width, \( w_{input}(l = 1, 2, \ldots, L_{input}) \) is the weight of the input nonlinear block, \( L_{input} \) is the total fuzzy rule, \( w_{output}(l = 1, 2, \ldots, L_{output}) \) is the weight of the output nonlinear block, and \( L_{output} \) is the fuzzy rule of the neuro-fuzzy model.

According to the given structure of the Hammerstein–Wiener model, the neuro-fuzzy Hammerstein–Wiener system consists of two static nonlinear blocks described by two independent neuro-fuzzy models and a dynamic linear block described by finite impulse response model. The elaborate structure is shown in Figure 3.

For the available observed data set and given order \( n_u \), a common method for parameter learning is to find the best fit of the Hammerstein–Wiener model by minimizing the loss function

\[
E(f(u(k)), \hat{b}_1, \ldots, \hat{b}_{n_b}, \hat{g}(\hat{z}(k))) = \frac{1}{2N} \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2 \\
\text{s.t.} \\
\hat{y}(k) = f(u(k)) \\
\hat{z}(k) = \hat{B}(z) \hat{v}(k) + e(k) \\
\hat{g}(\hat{z}(k))
\]

where \( \hat{v}(k) \) and \( \hat{z}(k) \) are the intermediate variable estimation, \( f(\cdot) \) is the input nonlinear block estimation, \( \hat{g}(\cdot) \) is the output nonlinear block estimation, \( \varepsilon \) is the given tolerance, and \( N \) is the total observed data.

**Parameter learning of the neuro-fuzzy Hammerstein–Wiener model**

In this section, a parameter learning process of neuro-fuzzy Hammerstein–Wiener model with disturbance using designed input signals is concretely discussed. The objective of the proposed learning method is to acquire the parameter estimation of three blocks.

The designed input signals are made up of separable signals and random signals, which are implemented to activate the Hammerstein–Wiener model, bringing out completely the parameter learning of three blocks (i.e. output nonlinear block, linear block, and input nonlinear block). In the parameter learning of the Hammerstein model, from the perspective of static nonlinear block, our previous research work has demonstrated the relationship of input–output signals under separable signals. Taking advantage of the idea of this work, further research of the Hammerstein–Wiener model is discussed.

**Theorem 1.** With the static input nonlinear block of the Hammerstein–Wiener model, if the model input \( u(k) \) is the separable signal, then there is a constant \( b_0 \) which makes the following equation hold

\[
R_{u_0}(\tau) = b_0 R_u(\tau), \quad \forall \tau \in Z
\]

where \( R_{u_0}(\tau) = E(v(k)u(k - \tau)) \) denotes the function constituted by input signal and corresponding output signal, \( R_u(\tau) = E(u(k)u(k - \tau)) \) denotes the function composed of input signal, and \( b_0 = E(v(k)u(k))/E(u(k)u(k)) \) denotes constant.

The certification process of Theorem 1 can be done by referring to the method by Li and Jia and hence it is omitted here.

Theorem 1 shows that \( R_{u_0}(\tau) \) constituted by model input signal and intermediate signal can be replaced by \( R_u(\tau) \) composed of model input signal. Two advantages can be obtained from Theorem 1: first, the parameter learning of static input nonlinear block and the Wiener model (i.e. combination of dynamic linear block and static output nonlinear block) are implemented. Second, the problem that the information of the intermediate variable \( \hat{v}(k) \) is not measurable can be effectively solved.

Based on the above analysis, the parameter learning process of the Hammerstein–Wiener model with process disturbance is specifically analyzed.

**Learning the static output nonlinear block parameters**

In this subsection, the result of Theorem 1 is employed in the Hammerstein–Wiener model. Two sets of separable input and output signals with different sizes are adopted to learn the parameters of the static nonlinear block. In the parameter learning process, we need to determine the center, \( c_{output} \), the width, \( \sigma_{output} \), and the weight, \( w_{output} \), of the static output nonlinear block represented by neuro-fuzzy model. The center, \( c_{output} \), and the width, \( \sigma_{output} \), can be computed using the cluster method previously proposed. To this end, the key challenge is how to estimate the weight, \( w_{output} \).
The output signals $y_1(k)$ and $y_2(k)$, corresponding to two sets of input signals with different sizes, are used. The static output nonlinear block can be expressed as

\[
  z_1(k) = g^{-1}(y_1(k)) = \sum_{i=1}^{L_{output}} \phi_i^\text{output}(y_1(k))w_i^\text{output} 
\]

\[
  z_2(k) = g^{-1}(y_2(k)) = \sum_{i=1}^{L_{output}} \phi_i^\text{output}(y_2(k))w_i^\text{output} 
\]

According to equations (2) and (3), equations (9) and (10) can be rewritten into a linear regression form as

\[
  z_1(k) = \sum_{j=1}^{n_0} b_j v_1(k-j) + e(k) 
\]

\[
  z_2(k) = \sum_{j=1}^{n_0} b_j v_2(k-j) + e(k) 
\]

Multiplying both sides of equations (11) and (12) by $u_i(k-\tau)$ and $u_2(k-\tau)$, respectively, and calculating the mathematical expectations, we get

\[
  R_{z_1 u_i}(\tau) = \sum_{j=1}^{n_0} b_j R_{v_1 u_i}(\tau-j) + R_{u_i}(\tau) 
\]

\[
  R_{z_2 u_i}(\tau) = \sum_{j=1}^{n_0} b_j R_{v_2 u_i}(\tau-j) + R_{u_i}(\tau) 
\]

$e(k)$ is independent of system input signal, $u(k)$ and $u_2(k)$; therefore, $R_{u_i}(\tau) = 0$ and $R_{u_1}(\tau)$. Furthermore

\[
  R_{z_1 u_1}(\tau) = \sum_{j=1}^{n} b_0 b_j R_{u_1}(\tau-j) 
\]

\[
  R_{z_2 u_2}(\tau) = \sum_{j=1}^{n} b_0 b_j R_{u_2}(\tau-j) = \lambda^2 \sum_{j=1}^{n_0} b_0 b_2 R_{u_1}(\tau-j) 
\]

Using Theorem 1, we obtain

\[
  R_{z_1 u_1}(\tau) = \sum_{j=1}^{n} b_0 b_j R_{u_1}(\tau-j) 
\]

\[
  R_{z_2 u_2}(\tau) = \sum_{j=1}^{n} b_0 b_j R_{u_2}(\tau-j) = \lambda^2 \sum_{j=1}^{n_0} b_0 b_2 R_{u_1}(\tau-j) 
\]

where $b_{01} = (E(v_1(k)u_1(k)))/E(u_1(k)u_1(k))$ and $b_{02} = (E(v_2(k)u_2(k)))/E(u_2(k)u_2(k))$.

Combining equation (17) with equation (18) yields

\[
  R_{z_2 u_2}(\tau) = \beta R_{z_1 u_1}(\tau) 
\]

where $\lambda = u_2/u_1$ and $\beta = \lambda^2 b_{01}/b_{02}$.

Furthermore, equation (29) can be defined as

\[
  E(z_2(k)u_2(k-\tau)) = \beta E(z_1(k)u_1(k-\tau)) 
\]

Substituting equations (9) and (10) into equation (20), we obtain

\[
  \sum_{j=1}^{L_{output}} w_j^\text{output} E(\phi_j^\text{output}(y_2(k))u_2(k-\tau)) 
\]
Equations (27) and (28) are used to calculate $R_{\phi_1 \psi_1}(\tau)$ and $R_{\phi_2 \psi_2}(\tau)$, respectively

$$R_{\phi_1 \psi_1}(\tau) = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L_{\text{input}}} \phi_{1l}(y_1(k)) u_1(k - \tau)$$  

(27)

$$R_{\phi_2 \psi_2}(\tau) = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L_{\text{input}}} \phi_{2l}(y_2(k)) u_2(k - \tau)$$  

(28)

**Learning the dynamic linear block parameters**

The input and output data of separable signals are implemented for learning the unknown parameter of the dynamic linear blocks in this subsection. According to equation (17), the dynamic linear block can be equivalently rewritten as

$$R_{z_i u_i}(\tau) = \sum_{j=1}^{n_1} b_j R_{u_i}(\tau - j)$$  

(29)

where $b_j = b_{0j} b_j$ and $b_{0j} = (E(v_1(k) u_1(k)))/(E(u_1(k) u_1(k)))$.

In the last subsection, we have estimated the static nonlinear block parameter, so correlation function, $R_{z_i u_i}(\tau)$ can be computed. Consequently, the correlation analysis algorithm can be utilized to learn linear block unknown parameters.

Assume $\tau = 1, 2, \ldots, P$, from equation (29), we get

$$R_{\theta_{\text{linear}}} = \theta_{\text{linear}} \psi$$  

(30)

where $\theta_{\text{linear}} = [b_1, b_2, \ldots, b_{n_1}], R = [R_{z_1 u_1}(1), R_{z_1 u_1}(2), \ldots, R_{z_1 u_1}(P)],$ and

$$\psi = \begin{bmatrix} R_{u_1}(0) & R_{u_1}(1) & R_{u_1}(2) & \cdots & R_{u_1}(P-1) \\ 0 & R_{u_1}(0) & R_{u_1}(1) & \cdots & R_{u_1}(P-2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & R_{u_1}(P-n_1) \end{bmatrix}$$

Define a cost function

$$E(\theta_{\text{linear}}) = ||R - \theta_{\text{linear}} \psi||^2$$  

(31)

Taking the derivative of equation (31), we obtain

$$\frac{\partial E(\theta_{\text{linear}})}{\partial \theta_{\text{linear}}} = \frac{\partial}{\partial \theta_{\text{linear}}} [(R - \theta_{\text{linear}} \psi)^T (R - \theta_{\text{linear}} \psi)]$$

$$= 2 \theta_{\text{linear}} \psi \phi T - 2 R \psi$$  

(32)

Let $\frac{\partial E(\theta_{\text{linear}})}{\partial \theta_{\text{linear}}} = 0$, we get

$$\theta_{\text{linear}} \psi \phi T = R \psi T$$  

(33)

Multiplying both sides of equation (33) by the inverse of matrix, $\phi \phi^T$, the parameter estimation is given by

$$\hat{\theta}_{\text{linear}} = R \phi T (\phi \phi^T)^{-1}$$  

(34)

The following formulas are used to calculate $R_{\phi_1 \psi_1}(\tau)$ and $R_{\phi_2 \psi_2}(\tau)$

$$R_{z_i u_i}(\tau) = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L_{\text{input}}} \phi_{il}(y_1(k)) u_1(k - \tau)$$  

(35)

$$R_{u}(\tau) = \frac{1}{N} \sum_{k=1}^{N} u(k) u(k - \tau)$$  

(36)

**Learning the static input nonlinear block parameters**

Based on two sets of separable signals, we have estimated the input nonlinear block parameter and the linear block parameter. Next, we discuss how to learn the static input nonlinear block parameters by means of the available input-output of random signals.

The learning procedure of input nonlinearity characterized by neuro-fuzzy model is to calculate the center, $c_{\text{input}}$, the width, $\sigma_{\text{input}}$, and the weight, $w_{\text{input}}$. Similar to the previous learning methods of the static output nonlinear block, the center, $c_{\text{input}}$, and the weight, $\sigma_{\text{input}}$, can be learned by the cluster method. So an obvious question is whether the weight, $w_{\text{input}}$, needs to be learned.

From equations (1) and (3), we have

$$z(k) = \sum_{j=1}^{n_1} b_j v(k - j) + e(k)$$  

(37)

Using equation (5) yields

$$z(k) = \sum_{j=1}^{n_1} \sum_{l=1}^{L_{\text{input}}} b_j \phi_{il}^\text{input} (u(k - j)) w_{i, \text{input}} + e(k)$$  

(38)

For simplicity, equation (38) is rewritten as

$$z(k) = \varphi(k) \psi_{\text{input}}^T + e(k)$$  

(39)

where $\psi_{\text{input}} = [b_1 w_{1, \text{input}}, \ldots, b_i w_{i, \text{input}}, \ldots, b_{n_1} w_{n_1, \text{input}}]^T$ and $\varphi(k) = [\phi_{1, \text{input}}(u(k-1)), \ldots, \phi_{i, \text{input}}(u(k-l)), \ldots, \phi_{n_1, \text{input}}(u(k-n_1)), \ldots, \phi_{n_1, \text{input}}(u(k-n_1))]^T$.

Defining a quadratic cost function according to equation (39)

$$J(\psi_{\text{input}}) = \sum_{k=1}^{N} ||z(k) - \varphi(k) \psi_{\text{input}}||^2$$  

(40)

Using least squares estimation method, we get

$$\hat{\psi}_{\text{input}, k} = \left[ \sum_{k=1}^{N} \varphi(k) \varphi(k)^T \right]^{-1} \left[ \sum_{k=1}^{N} \varphi(k) z(k) \right]$$  

(41)

According to equations (39) and (41), it is easy to obtain
\[ \hat{\theta}_{\text{input}}^{LS}(k) = \left( \sum_{k=1}^{N} \varphi(k)\varphi^T(k) \right)^{-1} \left\{ \sum_{k=1}^{N} \varphi(k)(\varphi)^T(k)\theta_{\text{input}} + e(k) \right\} \]

\[ \begin{align*}
&= \left[ \sum_{k=1}^{N} \varphi(k)\varphi^T(k) \right]^{-1} \left[ \sum_{k=1}^{N} \varphi(k)\varphi^T(k)\theta_{\text{input}} \right] \\
&\quad + \left[ \sum_{k=1}^{N} \varphi(k)\varphi^T(k) \right]^{-1} \left[ \sum_{k=1}^{N} \varphi(k)e(k) \right] \\
&= \theta_{\text{input}} + \left[ \sum_{k=1}^{N} \varphi(k)\varphi^T(k) \right]^{-1} \left[ \sum_{k=1}^{N} \varphi(k)e(k) \right]
\end{align*} \tag{42} \]

Note that the noise \( e(k) \) is a white noise sequence; therefore

\[ E\left[ \sum_{k=1}^{N} \varphi(k)e(k) \right] = 0 \tag{43} \]

Combing equations (42) and (43), we get

\[ E(\hat{\theta}_{\text{input}}^{LS}(k)) = E(\theta_{\text{input}}) \]

\[ + E\left[ \sum_{k=1}^{N} \varphi(k)\varphi^T(k) \right]^{-1} \left[ \sum_{k=1}^{N} \varphi(k)e(k) \right] \]

\[ = \theta_{\text{input}} \tag{44} \]

The unbiased estimation of the input nonlinear block parameter, \( \theta_{\text{input}} \), is obtained.

For simplicity, the flowchart of learning the Hammerstein–Wiener model parameters is shown in Figure 4.

Simulation results and analysis

Two numerical cases of Hammerstein–Wiener model with disturbance are offered to demonstrate the effectiveness of the proposed parameter learning algorithm.

Case 1

The nonlinear Hammerstein–Wiener model with disturbance is taken into account in which the input nonlinear block is represented using discontinuous function

\[ v(k) = \begin{cases} 2 - \cos(3u(k)) - \exp(-u(k)) & u(k) \leq 3.15 \\ 3 & u(k) > 3.15 \end{cases} \]

\[ x(k) = 0.9v(k-1) + 0.6v(k-2) + 0.3v(k-3) + 0.1v(k-4) \]

\[ z(k) = x(k) + e(k) \]

\[ y(k) = \begin{cases} 0.25\exp(z(k) - 2.5) & z(k) > 2.5 \\ 0.1z(k) & z(k) \leq 2.5 \end{cases} \tag{45} \]

Figure 4. The flowchart of learning the Hammerstein–Wiener model parameters.

where the noise \( e(k) \) is the white noise sequence.

Define the parameter estimation error of dynamic linear block at time \( k \) sampling instant as 

\[ \delta = \| \hat{\theta}_{\text{linear}}(k) - \theta_{\text{linear}} \| / \| \theta_{\text{linear}} \| \]

and the noise-to-signal ratio as 

\[ \delta_{\text{ns}} = \sqrt{(\text{var}[e(k)])(\text{var}[x(k)])} \times 100\% . \]

Two sets of binary signals with different sizes are 200 binary signals with 0 or 4 and 200 binary signals with 0 or 2, and 400 random signals between [0, 4] are employed to learn the Hammerstein–Wiener model parameters. Figure 5 shows the inputs and outputs of the Hammerstein–Wiener model.

At first, two sets of binary signals with different sizes of 200 binary signals with 0 or 4 and 200 binary signals with 0 or 2 are employed to learn static output nonlinear block parameter. Set parameters \( S_0 = 0.995, \rho = 1, \) and \( \lambda = 0.03, \) and the number of fuzzy ruler is 14. The estimation of the static output
The nonlinear block is shown in Figure 6. As can be easily seen from Figure 6, the proposed parameter learning approach can effectively represent the static output nonlinearity of the Hammerstein–Wiener model with disturbance.

Next, inputs and corresponding outputs from training samples of binary signals are employed for learning the unknown parameter of the linear block. In this study, we select 5000 binary signals with 0 or 4 for certifying the effectiveness of the correlation analysis method.

From Table 1 and Figure 7, the proposed correlation analysis algorithm can obtain more accurate parameter estimation of dynamic linear block than the RLS algorithm for uncertainty process disturbance. With the increase in noise-to-signal ratio, the proposed learning algorithm has more obvious advantages.

Based on the inputs and outputs of the random signal, the parameters of the input static nonlinear block are learned using the design parameters \( S_0 = 0.97 \), \( \rho = 1 \), and \( \lambda = 0 \), and the number of fuzzy rules is 12. Estimation of the input nonlinear block is illustrated in Figure 8. As shown in Figure 8, the proposed model and learning algorithm can effectively simulate static input nonlinearity.

Finally, to verify the availability of the proposed parameter learning method, the polynomial model–based input nonlinearity is also constructed utilizing the same observed input and output data. The polynomial model is as follows

\[
\hat{v}(k) = p_1 u(k) + p_2 u^2(k) + \cdots + p_r u^r(k) \tag{46}
\]

where \( p_1, p_2, \ldots, p_r \) are polynomial model parameters and \( r \) is the model order. The error comparisons of two different algorithms are given in Table 2, and it is considered to be the best model when the model order is 11.
From Table 2 and Figure 8, the proposed parameter learning algorithm has smaller mean square error and maximum error than the polynomial model algorithm. As a result, the proposed parameter learning algorithm can learn well the unknown parameter of the Hammerstein–Wiener model with uncertainty process disturbance.

**Case 2**

In this case, we take into account the Van de Vusse reaction kinetic process: \( A \overset{k_1}{\rightarrow} B \overset{k_2}{\rightarrow} C, A \overset{k_3}{\rightarrow} D \); the process is implemented in isothermal CSTR. In the Van de Vusse reaction process, the first and second reactions have first-order dynamic characteristics, and the third reaction has second-order dynamic characteristics. The CSTR system is represented by two nonlinear differential equations given by the following equations\(^{42}\):

\[
\begin{align*}
\frac{dC_A}{dt} &= -k_1 C_A - k_3 C_D + \frac{F}{V} (C_{Af} - C_A) \\
\frac{dC_B}{dt} &= k_1 C_A - k_2 C_B - \frac{F}{V} C_B
\end{align*}
\]

where \( F \) denotes the inlet flow rate; \( C_A \) denotes the concentration of component \( A \); \( C_B \) denotes the concentration of component \( B \); \( C_{Af} \) denotes the feed concentration of component \( A \); \( k_1, k_2, \) and \( k_3 \) denote the kinetic parameters; and \( V \) denotes the volume of the reactor. The system parameters of CSTR process are listed in Table 3.

The objective of CSTR is to regulate the concentration of component \( B, C_B \), by manipulating the inlet flow rate \( F \). In order to proceed with the proposed learning algorithm, the designed input signals are utilized to achieve system input and output in Figure 9, and the measurable data are normalized by means of \( u = (F - 34.3)/34.3 \) and \( y = (C_B - 1.12)/1.12 \). Applying the proposed parameter learning algorithm with designed parameters of static input nonlinearity and static output nonlinearity, \( S_{\text{input}}^{\text{output}} = 0.993, \rho^{\text{input}} = 1 \), and \( \lambda_{\text{input}}^{\text{output}} = 0.01; S_{\text{output}}^{\text{output}} = 0.996, \rho^{\text{output}} = 1 \), and

---

**Figure 7.** The correlation analysis algorithm (red solid line) and the recursive least squared algorithm (black dashed line): (a) \( \delta_{\text{in}} = 5.38\% \), (b) \( \delta_{\text{in}} = 12.14\% \), (c) \( \delta_{\text{in}} = 23.37\% \), and (d) \( \delta_{\text{in}} = 33.19\% \).

**Figure 8.** Comparisons of the approximation of the static input nonlinear block.

**Table 2.** Errors of various polynomial models and the proposed method.

| Algorithms     | Mean square error | Maximum error |
|----------------|-------------------|---------------|
| Polynomial model Orders |                   |               |
| 7              | 0.0116            | 0.3713        |
| 8              | 1.3 \times 10^{-4} | 0.0765        |
| 9              | 1.0 \times 10^{-4} | 0.0853        |
| 10             | 2.8 \times 10^{-4} | 0.1356        |
| Proposed method | 6.713 \times 10^{-4} | 0.0505        |

**Table 3.** Nominal operating condition and model parameters of CSTR.

| Variables | Value (unit) |
|-----------|--------------|
| \( k_1 \) | 50 \text{ h}^{-1} |
| \( k_2 \) | 100 \text{ h}^{-1} |
| \( k_3 \) | 10 \text{ L/mol h} |
| \( C_{Af} \) | 1.12 \text{ mol/L} |
| \( C_B \) | 3.0 \text{ mol/L} |
| \( F \) | 34.3 \text{ L/h} |
| \( V \) | 1 \text{ L} |
| \( C_{Af} \) | 10 \text{ mol/L} |

CSTR: continuous stirred tank reactor.
\( \lambda^{\text{output}} = 0.01; \) and the fuzzy rulers of input nonlinearity and output nonlinearity are 9 and 4, respectively.

Apply the proposed neuro-fuzzy Hammerstein–Wiener system to the design of controller; the designed control approach of the Hammerstein–Wiener model is shown in Figure 10. The design principle of the Hammerstein–Wiener model is to get rid of input and output nonlinearities and convert the nonlinear Hammerstein–Wiener system into linear system; then the linear system is controlled by the linear controller.

In this study, the nonlinear proportional–integral (PI) controller with \( K_c = 0.5 \) and \( \tau_I = 25 \) is adopted in the nonlinear control approach. For the sake of contrast, the linear PI controller, as described in equation (48) with \( K^{\text{linear}} = 2 \) and \( \tau_I^{\text{linear}} = 10 \), is adopted to control the Hammerstein–Wiener model

\[
\begin{align*}
 u(k) &= u(k-1) + K_c \left[ e(k) - e(k-1) + \frac{e(k)}{\tau_I} \right] \\
 e(k) &= y_i(k) - z(k) \quad \text{where} \quad e(k) = y_i(k) - z(k) \text{ indicates the error and } y_i(k) \text{ indicates the set point.}
\end{align*}
\]

where \( e(k) = y_i(k) - z(k) \) indicates the error and \( y_i(k) \) indicates the set point.

In Figures 11 and 12, the control performance of the linear PI controller and the nonlinear PI controller is given. According to Figures 11 and 12, it is noticeable that the traditional linear PI controller is unable to achieve good control effectiveness with regard to nonlinear Hammerstein–Wiener model. In particular, it results in a sluggish response when the value of the set point is 0.1. Nevertheless, an oscillatory response is discovered when the value of the set point is −0.5. By comparison, the designed nonlinear PI controller in this paper can effectively balance the sluggish and oscillatory response for positive and negative set points. As a result, the nonlinear PI controller can derive good tracking performance under the nominal operating condition.
Conclusion

This paper develops a modeling and parameter learning method for the Hammerstein–Wiener model with uncertainty disturbance in which two static nonlinear blocks surround a dynamic linear block. In modeling, two static nonlinearities are represented using independent neuro-fuzzy models, while a dynamic linear block is described using the finite impulse response model. The designed input signals are utilized to analyze the Hammerstein–Wiener model with uncertainty disturbance, resulting in the decoupling of parameter learning of static output nonlinearity, dynamic linear block, and static input nonlinearity. First, the output static nonlinear block parameters can be estimated using two sets of separable signals with different sizes. Moreover, correlation analysis algorithm is developed for learning the parameter of dynamic linear block based on input and output of a set of separable signal. Finally, parameter unbiased estimation of static input nonlinearity block is achieved using least squares method based on input and output of random signal. In numerical cases, two typical Hammerstein–Wiener models are employed to propose the learning algorithm: a discontinuous function and industrial CSTR. The results show that the proposed learning method yields reliable learning results with a high degree of accuracy and those which are minimally affected by disturbance.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Natural Science Foundation of Jiangsu Province under Grant No. BK20191035, Major Basic Research Project of the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant No. 19KJB120002, and the National Natural Science Foundation of China under Grant No. 61773335.

ORCID iD

Feng Li  https://orcid.org/0000-0001-9445-1627

References

1. Li H, Gao Y, Shi P, et al. Observer-based fault detection for nonlinear systems with sensor fault and limited communication capacity. *IEEE T Automat Contr* 2016; 61(9): 2745–2751.
2. Schindele D, Aschemann H and Prabel R. Nonlinear model-predictive control with hysteresis compensation of an electro-pneumatic clutch for truck applications. *Math Comp Model Dyn Syst* 2014; 20(2): 105–129.
3. Shi Y, Yu DL, Tian Y, et al. Modified Volterra model-based non-linear model predictive control of IC engines with real-time simulations. *T I Meas Control* 2015; 39(2): 208–223.
4. Li F and Jia L. Correlation analysis-based error compensation recursive least-square identification method for the Hammerstein model. *J Stat Comput Sim* 2018; 88(1): 56–74.
5. Sadeghi M and Farrokhii M. Block-oriented identification of non-linear systems with input time delay in presence of measurement noise: a Laguerre–neural network approach. *T I Meas Control* 2018; 41(1): 117–126.
6. Pouliquen M, Pigeon E and Gehan O. Identification scheme for Hammerstein output error models with bounded noise. *IEEE T Automat Contr* 2016; 61(2): 550–555.
7. Mu B, Chen HF, Wang LY, et al. Recursive identification of Hammerstein systems: convergence rate and asymptotic normality. *IEEE T Automat Contr* 2017; 62(7): 3277–3292.
8. Grzegorz M and Paweł W. Kernel-based identification of Wiener-Hammerstein system. *Automatica* 2017; 83: 275–281.
9. Beatriz dos Santos P, Elder O, Guimaraes JG, et al. A Hammerstein-Wiener model for single-electron transistors. *IEEE T Electron Dev* 2018; 66(2): 1092–1099.
10. Esmaelani L, Ghaisari J and Bagherzadeh MA. Bayesian approach to identify Hammerstein-Wiener non-linear model in presence of noise and disturbance. *IET Control Theory A* 2019; 13(3): 367–376.
11. Li J, Zheng WX, Gu J, et al. A recursive identification algorithm for Wiener nonlinear systems with linear state-space subsystem. *Circ Syst Signal Pr* 2018; 37: 2374–2393.
12. Li J, Zong T, Gu J, et al. Parameter estimation of Wiener systems based on the particle swarm iteration and gradient search principle. *Circ Syst Signal Pr* 2020, https://doi.org/10.1007/s00034-019-01329-1
13. Li J, Xu S, Hong K, et al. Neural network modeling and single-neuron proportional-integral-derivative control for hysteresis in piezoelectric actuators. *Meas Control* 2019; 52(9–10): 1362–1370.
14. Yu T, Wang H and Cao D. Synchronization of delayed neural networks with hybrid coupling via impulsive control. *T I Meas Control* 2019; 41(13): 3714–3724.
15. Wu Y and Lu Y. An intelligent machine vision system for detecting surface defects on packing boxes based on support vector machine. *Meas Control* 2019; 52(7–8): 1102–1110.
16. Liu X and Zhang K. Earth pressure balance control of shield tunneling machine based on nonlinear least squares support vector machine model predictive control. *Meas Control* 2018; 52(1–2): 3–10.
17. Zhang Y, Liu X, Jiang J, et al. Dissipativity-based resilient asynchronous control for Markov jump systems with sector-bounded nonlinearities. *T I Meas Control* 2018; 40(9): 2821–2830.
18. Giri F and Bai EW. *Block-oriented nonlinear system identification*: Berlin; Heidelberg: Springer-Verlag, 2010.
19. Yu F, Mao Z, Jia M, et al. Recursive parameter identification of Hammerstein-Wiener systems with measurement noise. *Signal Process* 2014; 105: 137–147.
20. Cai H, Li P, Su C, et al. Double-layered nonlinear model predictive control based on Hammerstein-Wiener model with disturbance rejection. *Meas Control* 2018; 51(7–8): 260–275.
21. Jeng JC and Lin YW. Data-driven nonlinear control design using virtual reference feedback tuning based on Block-oriented modeling of nonlinear systems. *Ind Eng Chem Res* 2018; 57(22): 7583–7599.
22. Li F and Jia L. Parameter estimation of Hammerstein-Wiener nonlinear system with noise using special test signals. *Neurocomputing* 2019; 344: 37–48.
23. Luo XS and Song YD. Data-driven predictive control of Hammerstein-Wiener systems based on subspace identification. *Inform Sciences* 2018; 422: 447–461.
24. Bai EW. An optimal two-stage identification algorithm for Hammerstein-Wiener nonlinear systems. *Automatica* 1998; 34(3): 333–338.
25. Bai EW. A blind approach to the Hammerstein-Wiener model identification. *Automatica* 2002; 38(6): 967–979.
26. Crama P and Schoukens J. Hammerstein-Wiener system estimator initialization. *Automatica* 2004; 40(9): 1543–1550.
27. Vörös J. Iterative identification of nonlinear dynamic systems with output backlash using three-block cascade models. *Nonlinear Dyn* 2015; 79: 2187–2195.
28. Park HC, Sung SW and Lee J. Modeling of Hammerstein-Wiener processes with special input test signals. *Ind Eng Chem Res* 2006; 45(3): 1029–1038.
29. Wills A, Schön TB, Ljung L, et al. Identification of Hammerstein-Wiener models. *Automatica* 2013; 49(1): 70–81.
30. Yu F, Mao Z and Jia M. Recursive identification for Hammerstein-Wiener systems with dead-zone input nonlinearity. *J Process Contr* 2013; 23: 1108–1115.
31. Zhang B, Hong H and Mao Z. Adaptive control of Hammerstein-Wiener nonlinear systems. *Int J Syst Sci* 2016; 47(9): 2032–2047.
32. Zhu Y. Estimation of an N-L-N Hammerstein-Wiener model. *Automatica* 2002; 38: 1607–1614.
33. Wang DQ and Ding F. Hierarchical least squares estimation algorithm for Hammerstein-Wiener systems. *IEEE Signal Proc Let* 2012; 19(12): 825–828.
34. Li Y, Mao ZZ, Wang Y, et al. Identification of Hammerstein-Wiener models based on bias compensation recursive least squares. *Acta Automatica Sinica* 2010; 36(1): 163–168.
35. Allafi W, Zajic I, Uddin K, et al. Parameter estimation of the fractional-order Hammerstein-Wiener model using simplified refined instrumental variable fractional-order continuous time. *IET Control Theory A* 2017; 11(15): 2591–2598.
36. Lang ZQ. On identification of the controlled plants described by the Hammerstein systems. *IEEE T Automat Contr* 1994; 39(3): 569–573.
37. Chen YC and Wang JS. A Hammerstein-Wiener recurrent neural network with frequency-domain eigensystem realization algorithm for unknown system identification. *J Univer Comput Sci* 2009; 15(13): 2547–2565.
38. Abouda SE, Ben Halima Abid D, Elloumi M, et al. Identification of nonlinear dynamic systems using fuzzy Hammerstein-Wiener systems. In: *Proceedings of the 19th international conference on sciences and techniques of automatic control and computer engineering*, Sousse, Tunisia, 24–26 March 2019, pp. 365–370. New York: IEEE.
39. Jia L, Yang A and Chiu MS. Research on multi-signal based neuro-fuzzy Hammerstein-Wiener model. *Acta Automatica Sinica* 2013; 39(5): 690–696.
40. Li F, Jia L and Peng D. Identification method of neuro-fuzzy-based Hammerstein model with coloured noise. *IET Control Theory A* 2017; 11(17): 3026–3037.
41. Jia L, Yang A and Chiu MS. Auxiliary model recursive least square algorithm based multi-signal identification of Hammerstein model. *J Nanjing U Sci Techno* 2014; 38(1): 34–39.
42. Hahn J and Edgar TF. A gramian based approach to nonlinearity quantification and model classification. *Ind Eng Chem Res* 2001; 40(24): 5724–5731.