

Hammerstein Model Identification for Autonomous Vehicle Dynamics by Two-Stage Algorithm [†]

György Istenes ^{1,*} , Dániel Pup ¹, György Terdik ²  and József Bokor ³

¹ Vehicle Industry Research Center, Széchenyi István University, Egyetem tér 1, 9026 Győr, Hungary; pupd@ga.sze.hu

² Department of Information Technology, Faculty of Informatics, University of Debrecen, Egyetem tér 1, 4032 Debrecen, Hungary; terdik.gyorgy@inf.unideb.hu

³ Institute for Computer Science and Control (SZTAKI), Kende u. 13-17, 1111 Budapest, Hungary; bokor.jozsef@sztaki.hun-ren.hu

* Correspondence: istenes.gyorgy@ga.sze.hu

[†] Presented at the Sustainable Mobility and Transportation Symposium 2024, Győr, Hungary, 14–16 October 2024.

Abstract: In this paper, the nonlinear identification (ID) of the lateral dynamics of a road vehicle is presented. The mathematical description of lateral dynamics is crucial for developing various self-driving functions. One method of describing dynamics is system identification from measured data. During the measurements, the steering servo of a test vehicle kept in straight-line motion by a self-driving function was artificially excited. A Hammerstein–Wiener model was successfully applied for the identification of these measurements. A nonlinear estimator was used during the fitting, which needed high computing power. For the Hammerstein–Wiener model, we used the two-stage algorithm (TSA) with a bilinear estimation method, which makes it possible to apply linear regression. We compared these methods during simulations and real data.

Keywords: system identification; Hammerstein model; bilinear parametrization; vehicle yaw dynamics



Citation: Istenes, G.; Pup, D.; Terdik, G.; Bokor, J. Hammerstein Model Identification for Autonomous Vehicle Dynamics by Two-Stage Algorithm. *Eng. Proc.* **2024**, *79*, 54. <https://doi.org/10.3390/engproc2024079054>

Academic Editors: András Lajos Nagy, Boglárka Eisinger Balassa, László Lendvai and Szabolcs Kocsis-Szürke

Published: 6 November 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

One type of nonlinear identification method is block-oriented models. They consist of linear dynamics and one or more static nonlinearities. If this nonlinearity is placed before the linear dynamics, it is called a Hammerstein model. If this nonlinearity is placed after the linear dynamics, it is called a Wiener model. If there is a nonlinearity before and after the linear dynamics, it is called a Hammerstein–Wiener model [1,2]. In this paper, we deal only with the Hammerstein model, whose block diagram is shown in Figure 1.

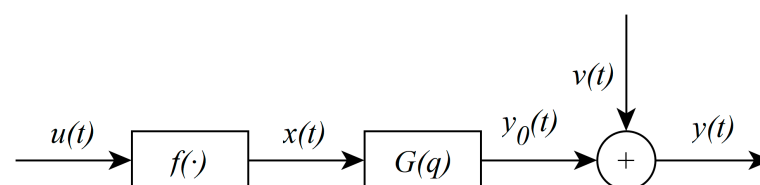


Figure 1. A discrete Hammerstein model.

There are many studies in the literature on modeling nonlinear systems using Hammerstein models and on the application of these models in various identification and control problems. One approach is the two-stage approximation approach, where the linear dynamics are described using the finite impulse response (FIR) in time series terms via the finite moving average (MA) model [3]. Support vector methods (SVM) were shown to provide further possibilities to learn the nonlinearities while parameterizing the linear

dynamics either by autoregressive (ARX), rational transfer function (ARMAX), or output error (OE) models.

It has been shown that the FIR and ARX cases (least-squares (LS) SVM and TSA) lead to the need for solving a weighted least-squares problem followed by a rank 1 approximation of an estimated matrix providing a Kronecker product of FIR parameters using the denominator (AR) parameters [4]. The solution to ARMAX and OE parametrizations requires solving quadratic programs (QPs), as shown in [5].

In this paper, a special version of the Hammerstein model is used and identified using the two-stage algorithm. The major advantage of the TSA algorithm is its ability to identify nonlinear dynamics using linear regression. Previously, this method has only worked effectively on simulated data [3]. Our paper aims to demonstrate the effectiveness of the TSA algorithm on a real dynamical system by comparing it with the results obtained using the Hammerstein model, which was implemented in MATLAB. The linear dynamics of the system are reached using an FIR parametrization and provide a simple technique that leads to a bilinearly parameterized linear regression [3]. We approximate a rational transfer function by truncating its infinite MA form by a finite one. These methods are compared with data from measurements of a Nissan Leaf test vehicle [6].

The calculations in this paper were performed using MATLAB R2021b software. For Hammerstein model identifications, MATLAB built-in functions were used [7].

2. Lateral Dynamics of an Autonomous Vehicle

The measurements were carried out using a Nissan Leaf test vehicle capable of fully autonomous driving. The data used for model identification were taken from these measurements. A high-level controller keeps the vehicle on the trajectory defined by the autonomous function, and a low-level controller determines the appropriate power steering control voltage. During the tests, this steering servo was artificially excited while the autonomous function control algorithm forced the vehicle to move in a straight line at a constant speed. The artificial excitation was a filtered pseudo random binary sequence (PRBS) signal. This is a signal with constant power within the bandwidth. The band limit is 1.59 Hz, considering the steering servo's operating limits. Measurements were taken at five different speeds. The input to the dynamics model is therefore the steering control voltage, which is proportional to the torque delivered by the steering servo, and the output is the lateral acceleration of the vehicle. More information on the test vehicle, measurement conditions, and artificial excitation is provided in [8]. We consider a discrete single-input single-output (SISO) Hammerstein model with input u and output y . The input nonlinearity f is assumed to be a static (memoryless) function; it is computed using a linear combination of known basis functions $\phi_j(t)$, and it results in $x(t) = f(u(t))$ and is provided as follows:

$$x(t) = f(u(t)) = \sum_{j=1}^m a_j \phi_j(u(t)) \quad (1)$$

In particular, $\phi_j(u) = u^j$ is set. We compute the output y passing $x(t)$ through the linear filter $B(q)/F(q)$. Finally, the output $y(t)$ is measured with an additive-colored noise $v(t-d)$ with delay d . More precisely,

$$y(t) = \frac{B(q)}{F(q)} x(t - n_k) + v(t - d), \quad (2)$$

where the transfer function $B(q)/F(q)$ is provided in terms of a backward shift operator q (i.e., $q^{-1}x(t) = x(t-1)$) as follows:

$$B(q) = \sum_{k=0}^{n_b} \beta_k q^{-k}, \quad F(q) = 1 - \sum_{k=1}^{n_\varphi} \varphi_k q^{-k}, \quad (3)$$

where $\beta_0 \neq 0$ and $\varphi_{n_\varphi} \neq 0$. In the sequel, we assume the following:

- The orders n_b and n_φ ($n_\beta < n_\varphi, n_\beta = n_\varphi - 1$ in practice), the delay n_k from the input to the output in terms of the number of samples, and the delay d of the noise v are assumed to be known a priori;
- The poles of the transfer function B/F have a modulus less than 1.

3. Approximating the Hammerstein Model

The approximation of the Hammerstein model is necessary because the linear core of the Hammerstein model implemented in MATLAB is an output error model, which approximates the transfer function with a rational function. By contrast, the TSA approximates it with a moving average model. To make the two comparable, we approximate the linear core of the Hammerstein model with a moving average model.

Now consider the series expansion of the following transfer function:

$$\frac{B(q)}{F(q)} = \sum_{j=0}^{\infty} c_j q^{-j} \tag{4}$$

This series expansion exists since the poles have a modulus less than 1. The coefficients c_j can be recursively calculated using the method of matching coefficients, i.e.,

$$B(q) = F(q) \sum_{j=0}^{\infty} c_j q^{-j}, \tag{5}$$

which provides the system of equations for finding the coefficients c_j . In practice, one can approximate the transfer function B/F using a finite sum as follows:

$$\frac{B(q)}{F(q)} \sim \sum_{j=0}^{n_c} c_j q^{-j} = C(q), \tag{6}$$

in other words, the Hammerstein model is ‘simplified’ using the moving average transfer function $C(q)$ as follows:

$$y(t) = C(q)x(t - n_k) + v(t - d) \tag{7}$$

It is well known that a rational transfer function has priority in several ways over a moving average transfer function; nevertheless, this nonlinear model (7) makes possible the application of a simple bilinear estimation of the parameters. Figure 2 shows the power spectrum of a Hammerstein model’s linear core, which was identified from measurement data and its approximations. The order of the numerator and denominator is three, and the order of approximations are 10, 30, 50, and 100.

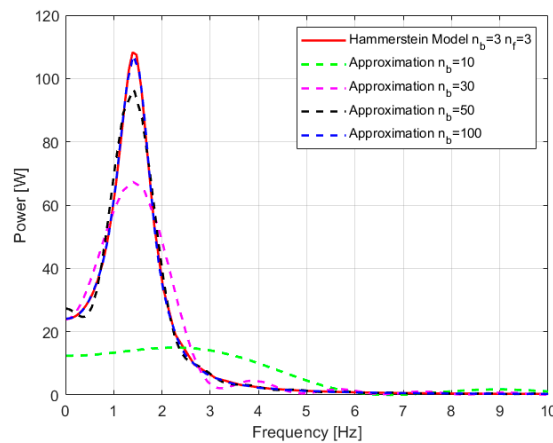


Figure 2. A discrete Hammerstein model.

4. Bilinear Parameter Estimation

Let u_{n_k} denote the shifted input, i.e., $u_{n_k}(t) = u(t - n_k)$, and let us rewrite Equation (7) in a more detailed form:

$$y(t) = \sum_{j=0}^{n_c} \sum_{k=1}^m c_j a_k \phi_k(u_{n_k}(t - j)) + v(t - d), \quad (8)$$

which is a bilinear equation in terms of the parameters $\mathbf{a} \in \mathbb{R}^m$ and $\mathbf{c} \in \mathbb{R}^{n_c}$. This equation implies that the parameters are not unique since one can multiply one term with a nonzero constant to yield the product $c_j a_k$ and divide the other by the same constant. Without a loss of generality, we shall assume that $\|\mathbf{c}\| = 1$ from now on. Equation (8) is bilinear in terms of the vectors $\mathbf{a} = [a_1, \dots, a_m]^T$ and $\mathbf{c} = [c_1, \dots, c_{n_c}]^T$; hence, Equation (8) can be written as

$$y(t) = \mathbf{c}^T \Phi(t) \mathbf{a} + v(t - d), \quad (9)$$

where the matrix $\Phi \in \mathbb{R}^{n_c \times m}$ is defined by the basis functions ϕ_j and the input u_{n_k} as

$$\Phi(t) = \begin{bmatrix} \phi_1(u_{n_k}(t - 1)) & \phi_2(u_{n_k}(t - 1)) & \cdots & \phi_m(u_{n_k}(t - 1)) \\ \phi_1(u_{n_k}(t - 2)) & \phi_2(u_{n_k}(t - 2)) & \cdots & \phi_m(u_{n_k}(t - 2)) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(u_{n_k}(t - n_c)) & \phi_2(u_{n_k}(t - n_c)) & \cdots & \phi_m(u_{n_k}(t - n_c)) \end{bmatrix} \quad (10)$$

We notice that the bilinear Equation (9) has a linear form if we replace the parameters \mathbf{a} and \mathbf{c} with their product:

$$y(t) = \mathbf{c}^T \Phi(t) \mathbf{a} + v(t - d) = \text{vec}^T \Phi(t) (\mathbf{a} \otimes \mathbf{c}) + v(t - d) \quad (11)$$

where $\text{vec}^T \Phi(t)$ is a row vector of column-wise vectorized $\Phi(t)$, and \otimes denotes the usual Kronecker product. Now let $\mathbf{y} \in \mathbb{R}^n$ be the column vector of observations $y(1), y(2), \dots, y(n)$, and the corresponding linear system of equations is as follows:

$$\mathbf{y} = \Psi_n (\mathbf{a} \otimes \mathbf{c}) + \mathbf{v} = \Psi_n \text{vec}(\mathbf{c} \mathbf{a}^T) + \mathbf{v} = \Psi_n \boldsymbol{\beta} + \mathbf{v}, \quad (12)$$

where $\boldsymbol{\beta} = \mathbf{a} \otimes \mathbf{c}$ and $\text{vec}^T \Phi(t)$ constitute the rows of the matrix $\Psi_n \in \mathbb{R}^{n \times mn_c}$, so

$$\Psi_n = \begin{bmatrix} \text{vec}^T \Phi(1) \\ \vdots \\ \text{vec}^T \Phi(n) \end{bmatrix}, \quad (13)$$

which is the subject of linear regression. The main issue here is finding the components \mathbf{a} and \mathbf{c} of the parameter $\boldsymbol{\beta}$.

5. Two-Stage Algorithm

Let us start with the weighted LS problem; we are given the parameter $\boldsymbol{\beta} \in \mathbb{R}^{mn_c}$ in the form $\boldsymbol{\beta} = \mathbf{a} \otimes \mathbf{c}$ and a weighting matrix $\mathbf{W} \in \mathbb{R}^{n \times n}$. Consider the LS problem

$$\frac{1}{2} \inf_{\mathbf{a}, \mathbf{c}} (\mathbf{y} - \Psi_n \boldsymbol{\beta})^T \mathbf{W} (\mathbf{y} - \Psi_n \boldsymbol{\beta}) \quad (14)$$

under the following constraints: $\|\mathbf{c}\|^2 = 1$, and the matrix $\mathbf{B} = \mathbf{c} \mathbf{a}^T \in \mathbb{R}^{n_c \times m}$ has a rank of 1. To find the solution of Equation (14), we compute the T-derivative ($D_{\boldsymbol{\beta}}^{\otimes} f(\boldsymbol{\beta}) = f(\boldsymbol{\beta}) \otimes D_{\boldsymbol{\beta}}$, see [9] for details). First, recall that $\Psi_n \in \mathbb{R}^{n \times mn_c}$, and we obtain $\Psi_n^T \mathbf{W} \mathbf{y} = \Psi_n^T \mathbf{W} \Psi_n \boldsymbol{\beta}$. If $\Psi_n^T \mathbf{W} \Psi_n$ has a full rank, then we have $\boldsymbol{\beta} = (\Psi_n^T \mathbf{W} \Psi_n)^{-1} \Psi_n^T \mathbf{W} \mathbf{y}$.

Stage 1. The singular decomposition of Ψ_n , which is $\Psi_n = \mathbf{U}_\Psi \mathbf{S}_\Psi \mathbf{V}'_\Psi$, $\mathbf{U}_\Psi \in \mathbb{R}^{n \times mn_c}$, $\mathbf{V}_\Psi \in \mathbb{R}^{mn_c \times mn_c}$, and $\mathbf{S}_\Psi \in \mathbb{R}^{mn_c \times mn_c}$, provides the weighting matrix $\mathbf{W} = \mathbf{U}_\Psi \mathbf{S}_\Psi^{-2} \mathbf{U}'_\Psi$, which produces the optimal solution of the nonlinear LS.

There are alternative weighting matrices for the optimal solution of the nonlinear LS problem (14), like $\mathbf{W} = \mathbf{I}$ (unweighted TSA) and $\mathbf{W} = \alpha (\Psi_n \Psi_n^T)^+$, where α is any positive scalar and the $(\Psi_n \Psi_n^T)^+$ is the Moore–Penrose pseudoinverse of the matrix $\Psi_n \Psi_n^T$. We obtain the estimate of the parameter vector β as $\hat{\beta} = (\Psi_n^T \mathbf{W} \Psi_n)^{-1} \Psi_n^T \mathbf{W} y$.

Stage 2. Decompose $\hat{\beta} = \text{vec} \hat{c} \hat{a}^T$ under the condition that the first nonzero entry of c is positive, and $\|\hat{a}\|^2 = 1$. Rewrite $\hat{\beta}$ in matrix form as follows: $\hat{\mathbf{B}} \in \mathbb{R}^{n_c \times m}$. Note that $\hat{\mathbf{B}}$ depends on \mathbf{W} , i.e., $\hat{\mathbf{B}} = \hat{\mathbf{B}}(\mathbf{W})$. Consider the singular value decomposition $\hat{\mathbf{B}} = \mathbf{U}_{n_c \times m} \mathbf{S}_B \mathbf{V}_{m \times m}^T$. Denote \mathbf{u}_1 and \mathbf{v}_1 as the left and right singular vectors of $\hat{\mathbf{B}}(\mathbf{W})$, respectively, according to the highest singular value s_1 . Compute $\hat{\mathbf{a}} = \epsilon_1 \mathbf{v}_1$ and $\hat{\mathbf{c}} = \epsilon_1 s_1 \mathbf{u}_1$, where $s_1 \epsilon_1$ is the sign of the first nonzero entry of \mathbf{u}_1 , and obtain $\hat{\mathbf{B}} = \hat{\mathbf{c}} \hat{\mathbf{a}}^T$, since $\mathbf{B} = 1$ is assumed.

6. Results and Conclusions

The measurements were taken at five different speeds. The input of the dynamics to be identified is the excitation voltage of the steering servo, and the output is the vehicle’s lateral acceleration. In the tests, the order of the Hammerstein model was 2/3. We approximated the estimated transfer function of the Hammerstein model using an MA transfer function with an order of 100. The two-stage algorithm was also applied with an order of 10. The order of polynomial nonlinearities was 3 in all cases. In Figure 3, the measured output is shown in blue, the output identified using the Hammerstein model is shown in red, the output identified using the Hammerstein model’s approximation is shown in green, and the output identified using the two-stage algorithm is shown in magenta. This figure shows only the first 5 s of the identification results for the 28.8 kph measurement, but all models were identified in 100 s measurement periods.

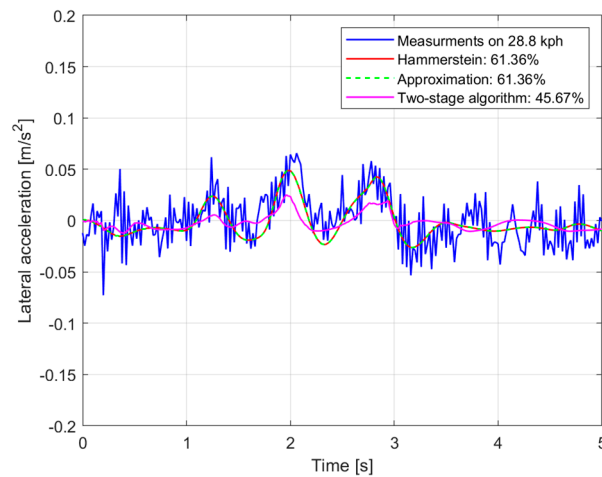


Figure 3. The fit of the identified models for the 28.8 kph measurement dataset.

More detailed results are shown in Table 1, where the $fit = 100(1 - \|y - \hat{y}\| / \|y - \bar{y}\|)$, and the BIC is the Bayesian information criterion.

From the results, it is clear that the Hammerstein model can be accurately approximated using a moving average transfer function, provided that the order of approximation is sufficiently high. However, using these measurements, the two-stage algorithm identified a model that was worse than the Hammerstein model, which was identified using the maximum likelihood (ML) method [10].

Table 1. The fit of the identified models.

Vehicle's Speed	Fit/BIC of Hammerstein Model	Fit/BIC of Hammerstein Model's Approximation	Fit/BIC of TSA Model
10.8 kph	79.46%/−28768	79.50%/−28779	24.89%/−22285
14.4 kph	75.84%/−26335	75.83%/−26332	48.25%/−22526
18.0 kph	58.46%/−22554	58.42%/−22549	40.64%/−20769
25.2 kph	58.16%/−24658	58.16%/−24659	40.67%/−22911
28.8 kph	61.36%/−24963	61.36%/−24963	45.67%/−23258

Author Contributions: Conceptualization, G.T. and J.B.; methodology, D.P., G.T. and J.B.; software, G.I. and G.T.; validation, G.I. and G.T.; formal analysis, G.I. and G.T.; investigation, G.I. and G.T.; resources, D.P.; data curation, G.I. and G.T.; writing—original draft preparation, G.T.; writing—review and editing, G.I., G.T. and J.B.; visualization, G.I.; supervision, D.P. and J.B.; project administration, D.P.; funding acquisition, D.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the European Union within the framework of the National Laboratory for Autonomous Systems (RRF-2.3.1-21-2022-00002) and was also supported by the project TKP2021-NKTA of the University of Debrecen, Hungary. Project no. TKP2021-NKTA-34 was implemented with support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, which is financed under the TKP2021-NKTA funding scheme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset is available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- Wills, A.; Schön, T.B.; Ljung, L.; Ninness, B. Identification of Hammerstein–Wiener models. *Automatica* **2013**, *49*, 70–81. [\[CrossRef\]](#)
- Giri, F.; Bai, E. *Block-Oriented Nonlinear System Identification*; Springer: London, UK, 2010; p. 419. [\[CrossRef\]](#)
- Wang, J.; Zhang, Q.; Ljung, L. Revisiting Hammerstein system identification through the two-stage algorithm for bilinear parameter estimation. *Automatica* **2009**, *45*, 2627–2633. [\[CrossRef\]](#)
- Castro-Garcia, R.; Suykens, J.A.K. Wiener System Identification using Best Linear Approximation within the LS-SVM framework. In Proceedings of the 2016 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Cartagena, Colombia, 2–4 November 2016. [\[CrossRef\]](#)
- Dhaifallah, M.A.; Westwick, D.T. Support Vector Machine Identification of Output Error Hammerstein Models. *IFAC Proc. Vol.* **2011**, *44*, 13948–13953. [\[CrossRef\]](#)
- Rödönyi, G.; Beintema, G.I.; Tóth, R.; Schoukens, M.; Pup, D.; Kisari, Á.; Vígh, Z.; Körös, P.; Soumelidis, A.; Bokor, J. Identification of the nonlinear steering dynamics of an autonomous vehicle. *IFAC-PapersOnLine* **2021**, *54*, 708–713. [\[CrossRef\]](#)
- Ljung, L. *System Identification Toolbox for Use with MATLAB*; The MathWorks Inc.: Natick, MA, USA, 2011; p. 904.
- Pup, D.; Istenes, G.; Szauter, F.; Bokor, J. Identification of lateral dynamics of an autonomous car vehicle. In Proceedings of the 12th International Conference on Mechatronics and Control Engineering—ICMCE 2024, Budapest, Hungary, 25–27 January 2024.
- Terdik, G. *Multivariate Statistical Methods: Going Beyond the Linear*; Frontiers in Probability and the Statistical Sciences; Springer International Publishing: Cham, Switzerland, 2021. [\[CrossRef\]](#)
- Ljung, L. *System Identification: Theory for the User*, 2nd ed.; Prentice-Hall: Hoboken, NJ, USA, 1999; pp. 212–213.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.