

Article

Nonlinear Dynamic System Identification in the Spectral Domain Using Particle-Bernstein Polynomials

Michele Alessandrini [†], Laura Falaschetti ^{*,†}, Giorgio Biagetti , Paolo Crippa  and Claudio Turchetti 

Department of Information Engineering, Università Politecnica delle Marche, Via Brecce Bianche 12, I-60131 Ancona, Italy

* Correspondence: l.falaschetti@univpm.it; Tel.: +39-071-220-4129

† These authors contributed equally to this work.

Abstract: System identification (SI) is the discipline of inferring mathematical models from unknown dynamic systems using the input/output observations of such systems with or without prior knowledge of some of the system parameters. Many valid algorithms are available in the literature, including Volterra series expansion, Hammerstein–Wiener models, nonlinear auto-regressive moving average model with exogenous inputs (NARMAX) and its derivatives (NARX, NARMA). Different nonlinear estimators can be used for those algorithms, such as polynomials, neural networks or wavelet networks. This paper uses a different approach, named particle-Bernstein polynomials, as an estimator for SI. Moreover, unlike the mentioned algorithms, this approach does not operate in the time domain but rather in the spectral components of the signals through the use of the discrete Karhunen–Loève transform (DKLT). Some experiments are performed to validate this approach using a publicly available dataset based on ground vibration tests recorded from a real F-16 aircraft. The experiments show better results when compared with some of the traditional algorithms, especially for large, heterogeneous datasets such as the one used. In particular, the absolute error obtained with the proposed method is 63% smaller with respect to NARX and from 42% to 62% smaller with respect to various artificial neural network-based approaches.

Keywords: system identification; ground vibration; nonlinear dynamic system; particle-Bernstein polynomials; regression



Citation: Alessandrini, M.; Falaschetti, L.; Biagetti, G.; Crippa, P.; Turchetti, C. Nonlinear Dynamic System Identification in the Spectral Domain Using Particle-Bernstein Polynomials. *Electronics* **2022**, *11*, 3100. <https://doi.org/10.3390/electronics11193100>

Academic Editor: Ahmed Abu-Siada

Received: 3 September 2022

Accepted: 24 September 2022

Published: 28 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

System identification (SI) is a discipline concerned with finding the mathematical models of a dynamic system based on observations of its inputs and outputs. Due to applications of SI to a variety of fields, such as the analysis and simulation of complex systems or the control of dynamic processes, a rich body of research is available on SI [1,2].

However, the intrinsic nonlinear nature of real world phenomena makes the linear hypothesis simply an approximation of real system behavior. Therefore, nonlinear system identification (NSI) is of great interest and one of the key issues in the modeling of signals generated by artificial systems and natural phenomena. The main applicative fields of NSI are automatic control, neurosciences, communications (e.g., echo cancellation, channel equalization and amplifier nonlinearity evaluation) and signal and image processing, to just mention a few [3].

The algorithms classically used for NSI depend on the selection of a suitable model to represent the data. Some valid and widely used models are the Volterra series expansion [4–6], Hammerstein model [7–11], Wiener model [12–15] and the nonlinear auto-regressive moving average model with exogenous inputs (NARMAX) and its derivatives [16–21].

In the discrete time domain, one of the most successful approaches for nonlinear system identification is the NARMAX model (and its derivatives NARX [22,23] and NARMA [24]), in which the system is modeled in terms of a nonlinear functional expansion

of lagged inputs, outputs and prediction errors. NARMAX models have been shown to be very effective in many real-world applications [17,19,25–29], as they are powerful, efficient and unified representations of a wide variety of nonlinear systems.

The choice of the nonlinear estimator function is an important part of NARMAX models. Various estimators can be used, such as polynomials [30,31], multilayer perceptrons (MLPs) [32,33], artificial neural networks (ANNs) [34–37] and wavelet networks [38,39].

While traditional algorithms operate on time domain representations of the data, the method used in this paper takes advantage of the spectral representation of the signals through the use of the discrete Karhunen–Loève transform (DKLT), thus simplifying the complexity of the problem, especially for large temporal series. The nonlinear relationship in the spectral domain thus becomes a multivariate static function to be approximated, and an alternative algorithm based on particle-Bernstein polynomials, a variation of the classic Bernstein polynomials [40,41] extended to continuous variables, has been applied for this purpose.

To assert the effectiveness of the proposed algorithm, it has been tested on a public dataset acquired from a real system, namely the F-16 aircraft benchmark dataset, which consists of measures of the airplane’s ground vibrations, and compared to other widely used ones.

This article is organized as follows. Section 2 introduces the mathematical theory and algorithms used in the paper. Section 3 describes the dataset used for the experiments. Section 4 explains how the data have been processed prior to the actual SI phase. The experimental set-up and results are shown in Section 5, together with a discussion of the results in Section 6. Finally, some conclusions are drawn in Section 7.

2. Proposed Method

The goal of this article is the identification of nonlinear dynamic systems, given a set of inputs $u(t)$ applied to the system and the corresponding outputs $y(t)$ generated by the system itself.

In the most common case of discrete time dynamic systems, the output of the system at a given time depends on the input and output values at a certain number of previous time points in what represents the memory of the system:

$$y(t) = h(y(t-1), \dots, y(t-p), u(t-1), \dots, u(t-q)), \quad (1)$$

with $t = 1, \dots, n$.

The proposed algorithm operates in the spectral domain, thus converting the system to a function that does not depend on time, but only on a set of static parameters. To this end, the discrete Karhunen–Loève transform (DKLT) [42] is used, which has the well-known property of separating the time-dependent components from fixed parameters.

Finally, the resulting nonlinear relation is estimated by particle-Bernstein polynomials.

2.1. Spectral Representation of Nonlinear Systems

Starting with the general case of time-dependent inputs $u(t)$ and outputs $y(t)$, as used in Equation (1), it is convenient to represent them in matrix form, namely U and Y , respectively, with $U \in \mathbb{R}^{m \times N}$, $Y \in \mathbb{R}^{n \times N}$ and N being the number of realizations. It is worth noticing that the original data are not always in this suitable form, likely being composed of a few realizations of different lengths. Nonetheless, they can be turned into this form by means of windowing, or splitting the input and output files into equally sized windows whose length must be chosen so it is compatible with the properties of the data.

Then, the DKLT is applied to the resulting matrices. A possible realization of the DKLT is using singular value decomposition (SVD), which factorizes a matrix (U for example) as

$$U = V\Sigma W^T \quad (2)$$

where $\Sigma \in \mathbb{R}^{N \times N}$ is a diagonal matrix containing the singular values of U and V and W contain vectors forming the bases for the U components such that U can be represented as

$$U = VX \quad (3)$$

where V contains the time-dependent components and X the desired static components (features) of the signals. The correspondence is biunivocal such that X can be computed as

$$X = V^T U \in \mathbb{R}^{N \times N} \quad (4)$$

for the properties of SVD.

Similarly, the outputs Y can be represented in their feature space as a new matrix K such that the nonlinear system in Equation (1) can be represented in a new form as

$$k = f(x), \quad x \in \mathbb{R}^m, \quad k \in \mathbb{R}^n \quad (5)$$

with no dependency on time.

With the two transformations being biunivocal, the identification problem can be solved in the feature space, and the computed outputs can then be transformed back to the time domain.

Moreover, it is customary to apply principal component analysis (PCA) to the DKLT to reduce the dimensionality of the problem [43]. By retaining only the components of V associated with the most significant singular values, the problem can be simplified with no significant loss in signal representation. In Equation (2), assuming the singular values in Σ are sorted in descending order, this is equivalent to truncating Σ to the upper $d \times d$ part and V to the first d rows so that Equation (4) becomes

$$X_d = V_d^T U, \quad X_d \in \mathbb{R}^{d \times N} \quad (6)$$

with $d < m$. An analog reduction can be applied to the output matrix K .

In most cases, the estimation of $f(\cdot)$ in Equation (5) is performed through supervised learning, where an experimental set S of corresponding input/output realizations is given:

$$S = \{(x_j, k_j), j = 1, \dots, N\} \quad (7)$$

Thus, this reduces to a regression problem in Equation (5) through a suitable technique. Usually, a part of S is used for the actual system estimation (training), while a separate subset is used to validate the approximated function (testing).

2.2. Bernstein Polynomials

Polynomials can effectively provide a basis to represent a function to be estimated, as stated by the well-known Weierstrass approximation theorem, according to which every continuous real function $f : [0, 1] \rightarrow \mathbb{R}$ can be approximated by a polynomial with arbitrary accuracy. (The $[0, 1]$ interval can easily be generalized to an arbitrary domain.)

Bernstein polynomials (BPs) are a convenient choice for such a basis [40,41]. Given the space of polynomials of a degree m , the k^{th} BP is defined as

$$b_k^{(m)}(x) = \binom{m}{k} x^k (1-x)^{m-k}, \quad k = 0, 1, \dots, m \quad (8)$$

where $x \in [0, 1]$ and $\binom{m}{k}$ is the usual binomial coefficient.

BPs are a convenient choice for representing a function to be estimated given an input/output relationship as in Equations (5) and (7), because the associated coefficients do not need complex algorithms to be computed but depend only on the value of $f(\cdot)$ in a series of points. More precisely, say we are given the approximation sequence $B_m(x)$, expressed as

$$B_m(x) = \sum_{k=0}^m f(k/m) b_k^{(m)}(x) \tag{9}$$

Then, it can be shown that this sequence converges uniformly to $f(x)$ as $m \rightarrow \infty$ (therefore representing a constructive proof of the Weierstrass theorem). The coefficients are simply given by the function evaluated at points $\{k/m, k = 0, \dots, m\}$.

This representation can easily be extended to the case of multivariate functions. If $x = (x_1, \dots, x_d)$ and $k = (k_1, \dots, k_d)$, then Equations (8) and (9) can be redefined as

$$B_m(x_1, \dots, x_d) = \sum_{k_1=0}^m \dots \sum_{k_d=0}^m f\left(\frac{k_1}{m}, \dots, \frac{k_d}{m}\right) b_k^{(m)}(x) \tag{10}$$

and

$$b_k^{(m)}(x) = \binom{m}{k_1} \dots \binom{m}{k_d} x_1^{k_1} (1 - x_1)^{m-k_1} \dots x_d^{k_d} (1 - x_d)^{m-k_d} \tag{11}$$

In this case, computing the coefficients requires evaluating the function at points

$$x = (k_1/m, \dots, k_d/m), \quad k_1 = 0, \dots, m, \quad k_d = 0, \dots, m \tag{12}$$

which is a grid of points in \mathbb{R}^d . Provided that such a grid must be sufficiently dense to obtain a good approximation of $f(\cdot)$, this has the drawback that the computational cost increases dramatically with the dimensionality of the input space, thus becoming prohibitive even for moderately complex systems.

2.3. Particle-Bernstein Polynomials

To overcome the computational cost of BPs in the multivariate function case, recently, a new class of functions named particle-Bernstein polynomials (PBPs) has been proposed [40,42,44], which has been shown to achieve good results in regression problems while maintaining a lower computational cost with respect to classic BPs.

Meanwhile, in Equation (8), the k index must be the integer for the presence of the binomial coefficient, and the PBPs relax this constraint by replacing k with a real parameter ζ , where $\zeta \in [0, m]$. The PBP functions are then defined as

$$C_\zeta^{(m)}(x) = \alpha_\zeta^{(m)} x^\zeta (1 - x)^{m-\zeta} = \alpha_\zeta^{(m)} k_\zeta^{(m)}(x) \tag{13}$$

The $\alpha_\zeta^{(m)}$ terms are computed so that

$$\int_0^1 C_\zeta^{(m)}(x) dx = 1 \tag{14}$$

Aside from them all having the same area, PBP functions have a maximum at $x = \zeta/m$ in $[0, 1]$, a property similar to traditional BPs. Together with the fact that they can be considered mostly concentrated around their maximum for $m \gg 1$, the value of $f(\zeta/m)$ can be approximated as

$$f(\zeta/m) \cong \int_0^1 f(x) C_\zeta^{(m)}(x) dx \tag{15}$$

If a given set of arbitrary values $\{x^{(j)}, j = 1, 2, \dots, N\}$ is given together with the evaluated function $f(x^{(j)})$ at the same points, as is usual in supervised learning, the following approximation for $f(\cdot)$ in an arbitrary point holds:

$$f(\zeta/m) \cong \frac{\sum_{j=1}^N f(x^{(j)}) k_\zeta^{(m)}(x^{(j)})}{\sum_{j=1}^N k_\zeta^{(m)}(x^{(j)})} \tag{16}$$

It follows that the main advantage of PBPs is that $f(\cdot)$ can be estimated at an arbitrary point given its values in a set of N randomly chosen points, according to the most convenient criterion, and not on a fixed grid as in traditional BPs.

The advantage is even more evident when extended to multivariate functions. Given $x = (x_1, \dots, x_d)$ and $\zeta = (\zeta_1, \dots, \zeta_d)$, the $k_{\zeta}^{(m)}(x)$ term in Equation (13) can be redefined as

$$k_{\zeta}^{(m)}(x) = x_1^{\zeta_1} (1 - x_1)^{m - \zeta_1} \dots x_d^{\zeta_d} (1 - x_d)^{m - \zeta_d} \quad (17)$$

and the estimation of $f(\zeta/m)$ is given by the same expression in Equation (16), replacing the scalar x and ζ with their vector equivalents. It can be seen from Equation (17) that the estimation complexity does not increase exponentially with the dimension d of the input space as in Equation (11), thus sensibly reducing the complexity with respect to traditional BPs.

Figure 1 shows the several steps of the proposed algorithm in a concise way.

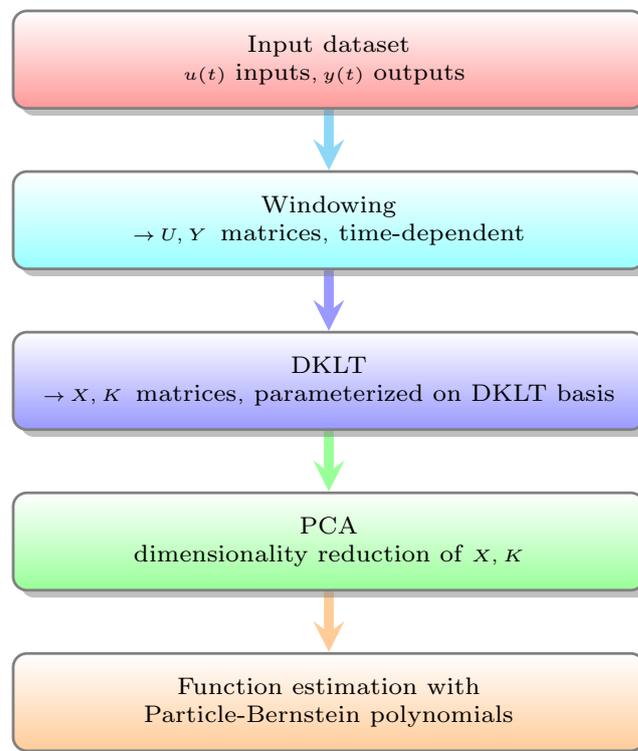


Figure 1. Schematic flowchart of the proposed algorithm.

3. Dataset

The F-16 Ground Vibration Test Benchmark Dataset [45,46] contains experimental data acquired from a full-scale F-16 aircraft on the occasion of the Siemens LMS Ground Vibration Testing Master Class, held in September 2014 at the Saffraanberg military base in Sint-Truiden, Belgium.

During the experiment, two dummy payloads were mounted at the wing tips to simulate the mass and inertia of real devices typically equipped by an F-16 in flight (Figure 2a). The aircraft structure was instrumented with accelerometers, and as the input signal, one shaker was attached underneath the right wing. The dominant source of nonlinearity in the structural dynamics was expected to originate from the mounting interfaces of the two payloads. These interfaces consisted of T-shaped connecting elements on the payload side slid through a rail attached to the wing's side (Figure 2b).



Figure 2. (a) Dummy payload mounted at the right wing tip. (b) Connection of the right-wing-to-payload mounting interface. Reproduced under CC BY-SA 4.0 from [46].

The measurements were acquired at a sampling frequency of 400 Hz. Two distinct input signals were made available: (1) the voltage measured at the output of the signal generator amplifier, acting as a reference input, and (2) the actual force provided by the shaker and measured by an impedance head at the excitation location.

Three acceleration signals were provided as output quantities. They were measured (1) at the excitation location, (2) on the right wing next to the nonlinear interface of interest and (3) on the payload next to the same interface. Measurements were performed with three different excitation types that yielded the three subsets of data described in the following sections.

3.1. Sine-Sweep Excitation with a Linear, Negative Rate

Sine-sweep excitations with a linear, negative rate of 0.05 Hz/s (sweep down) were applied. The covered input frequency range was 15–2 Hz. Seven different levels of excitation were provided as benchmark data. The lowest level at a 4.8-N input amplitude could be considered a linear dataset. Three higher excitation levels were given to function as estimation data in the nonlinear regimes of vibration, namely dataset numbers 3, 5 and 7 corresponding to 28.8, 67.0 and 95.6 N, respectively. Dataset numbers 2, 4 and 6 at 19.2, 57.6 and 86.0 N, respectively, were to be used for testing the models estimated using datasets 3, 5 and 7.

3.2. Multisine Excitation with a Full Frequency Grid

The data recorded under multisine excitations with a full frequency grid from 2 to 15 Hz were provided. At each force level, nine periods were acquired, considering a single realization of the input signal. The number of points per period was 8192. Note that transients were present in the first period of measurement. Similar to the sine-sweep case, 7 excitation levels were considered, starting from the linear data at 12.4 N RMS (dataset 1). In addition, three nonlinear estimation datasets (numbers 3, 5 and 7 at 36.8, 73.6 and 97.8 N RMS, respectively) were accompanied by their corresponding test sets (numbers 2, 4 and 6 at 24.6, 61.4 and 85.7 N RMS, respectively).

3.3. Multisine Excitation with a Random Frequency Grid

In the third set of tests, sine-sweep excitations were applied with only odd frequencies excited in the range of 1–60 Hz. Moreover, within each group of four successive excited odd lines, one frequency line was randomly rejected to act as a detection line for odd nonlinearities. In this setting, the frequency band from 1 to 60 Hz was excited. Three periods per level were recorded, considering 10 input realizations per level. The number of

points per period was 16384. Note that only the last two periods of each realization were in a steady state. The datasets were originally sampled at 200 Hz. They were upsampled to 400 Hz in the frequency domain, processing period per period and assuming the data were periodic and in a steady state.

4. Data Processing

For the system identification and its testing, the data subset with multisine excitation on a full frequency grid was used, with separate experiments for each of the three output sensors. The full data available for the mentioned input excitation were used for a total of 1290 s and at different excitation amplitudes.

To represent the input and output data in matrix form (Section 2.1) starting from a set of time series of different lengths and amplitudes, the original signals were concatenated and then split into a set of homogeneous windows. With w being the number of samples in a window and o being the number of overlapping samples between adjacent windows, the n th data window corresponded to the samples in the range

$$[n(w - o), n(w - o) + w - 1] \quad (18)$$

with $n \geq 0$.

For the input windows, a length of 8192 was used, corresponding to the period used to generate the dataset (Section 3). To overcome the problem of a limited number of realizations, the splitting of input data was performed with an overlap between windows. In this way, a moving window was used on the input data, while the matching output was taken as non-overlapping windows smaller than the input ones (Figure 3).

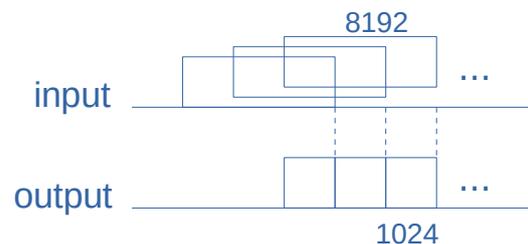


Figure 3. Windowing of input and output data.

To generate a well-balanced set of data for training and a set for testing, instead of roughly using half of the data for training and testing, as meant in the original data grouping, single inputs and outputs obtained from the different signals through windowing were randomly shuffled and then divided at an 80/20 ratio for training and testing, respectively.

The previous steps generated a pair of input/output matrices for training, named U_{train} and Y_{train} , and a similar pair for testing, named U_{test} and Y_{test} . Then, the DKLT was applied as in Section 2.1 by computing a base as in Equations (3) and (4). To avoid data leaks from testing to training, the decomposition was computed on the training matrices only. The same base was then applied to the testing material so as to obtain input/output pairs in the feature space of X_{train} , K_{train} , X_{test} and K_{test} .

Finally, before using the particle-Bernstein polynomials (Section 2.3), the input data had to be normalized in the range $[0, 1]$. The input data X_{train} and X_{test} were normalized by single components (columns) according to the following formula:

$$x_{\text{norm}} = a + (b - a) \frac{x - \min(x)}{\max(x) - \min(x)} \quad (19)$$

where x is the data component to be normalized, $a = 0$ and $b = 1$.

A prior clipping of data was performed using their mean value and standard deviation to filter out the outlier data and better distribute the resulting signal in the desired range.

Specifically, μ and σ are the mean value and standard deviation of a given component, respectively, and data were limited to the range $\mu \pm 3\sigma$.

5. Experimental Results

All the experiments on the different outputs were performed with particle-Bernstein polynomials of a degree of 40, determined experimentally as a trade-off between the resulting accuracy and numerical stability.

The input and output windows were of sizes of 8192 and 1024, respectively. For the PCA, 16 components for the input matrices and 50 for the output were used because this was experimentally determined to suffice for providing a good representation.

The absolute error of the predicted signals with respect to the true output were computed as the root mean square (RMS) of the difference of the two signals:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_{\text{predict}}[k] - y_{\text{true}}[k])^2} \quad (20)$$

The relative error was computed as the ratio between the RMSE and the RMS of the reference signal:

$$\text{RMSE}_{\text{rel}} = \text{RMSE} / \sqrt{\frac{1}{N} \sum_{k=1}^N (y_{\text{true}}[k])^2} \quad (21)$$

Figure 4 shows the true output and the error with respect to the prediction for the three output sensors, as well as a detailed comparison of the true and predicted outputs for the three output sensors.

To test the effectiveness of our approach, a comparison of the numerical results with the NARX model created with the Matlab “System Identification” tool was first conducted using the same data for training and testing as in the PBP experiment. The parameters used for the NARX model were as follows: 10 points (lags) for the input data, 20 points for the output data, a linear input regressor and a wavelet network as an output function. These parameters were chosen as a trade-off between accuracy and the resources needed by the process (time and memory).

Table 1 shows the error of the predicted output for the three output sensors compared with the NARX model.

Table 1. Comparison of errors in predicted output for testing data.

Output	PBP Rel. Error (%)	PBP Abs. Error (g)	NARX Abs. Error (g)
1	7.3	0.07	0.81
2	5.2	0.07	1.30
3	12.2	0.14	1.14

As a more exhaustive comparison, our approach was compared with deep convolutional neural networks (CNNs) as performed in [47] with the same dataset (actually a subset, as explained in the following). In [47], three types of CNNs are used: a temporal convolutional network (TCN), multilayer perceptron (MLP) and long short-term memory (LSTM). As a matter of comparison, in this case, the same data as in [47] were used, namely the multisine realizations with a random frequency grid limited to the test with a 49.0-N RMS amplitude. Moreover, only the training data were used to compute the RMSE of the identification results with no independent testing data. Another difference was the number of components needed by the PCA in our algorithm to represent the input values. In this case, 25 components were used, provided that a bigger value posed stability problems for the computations of the PBPs.

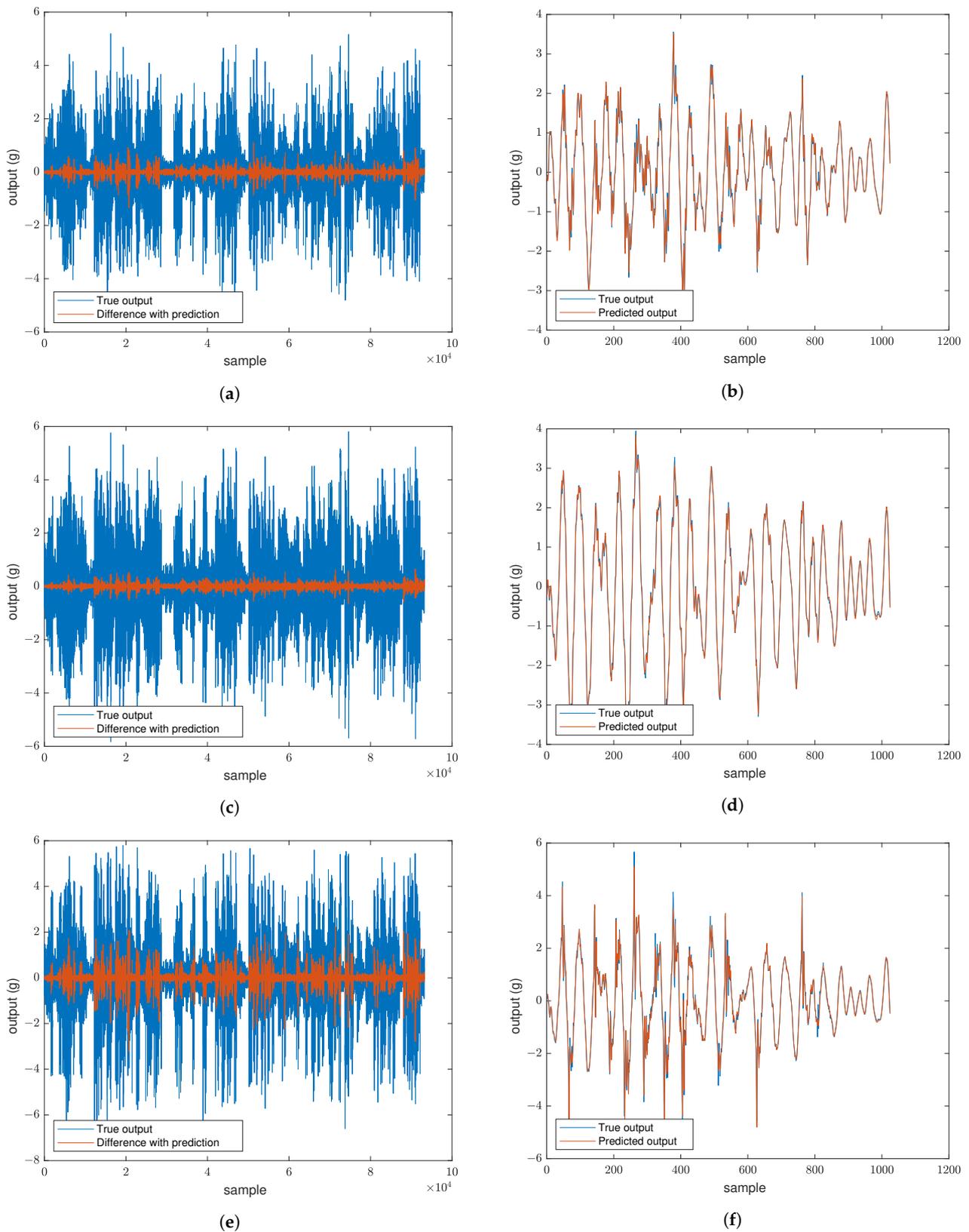


Figure 4. (a,c,e) Identification error for testing data of outputs 1, 2 and 3, respectively. (b,d,f) Comparison of true and predicted outputs for a sample window in outputs 1, 2 3, respectively.

Table 2 shows the error of the predicted data for the three output sensors for the various algorithms, with the addition of the Matlab NARX implementation with the same parameters as in the previous tests.

Table 2. Comparison of errors in predicted output for training data. * These results are provided as averages of the three outputs.

Output	PBP Abs. Error (g)	NARX Abs. Error (g)	LSTM Abs. Error (g) *	MLP Abs. Error (g) *	TCN Abs. Error (g) *
1	0.41	0.82	0.74	0.48	0.63
2	0.31	0.79	0.74	0.48	0.63
3	0.28	0.75	0.74	0.48	0.63

6. Discussion

In this work, PBPs were proposed as an alternative to the widely used system identification algorithms available in the literature, such as NARMAX and its derivatives and convolutional neural networks based on different structures.

Moreover, unlike the aforementioned algorithms, this approach does not operate in the time domain but rather in the spectral components of the signals through the use of the DKLT.

Some experiments were performed to validate this approach using a publicly available dataset based on ground vibration tests recorded from a real F-16 aircraft.

In the first experiment, we applied a PBP to the data in order to test the effectiveness of our approach, and a comparison of the numerical results with the NARX model created with Matlab's "System Identification" tool was conducted. Table 1 reports a comparison of the errors in the predicted output for the testing data between the proposed method and the NARX model. It can be seen that the PBP algorithm yielded sensibly better results.

In the second experiment, to give a more exhaustive comparison, our approach was compared with deep convolutional neural networks (CNNs), as performed in [47]. Table 2 reports a comparison of the errors in the predicted output for the training data between the various algorithms in [47] and also the Matlab NARX model. Again, the PBP algorithm yielded the best results in terms of identification error, with reductions in the absolute error ranging from 42% to 63%.

We already successfully applied this method to synthetic data, as reported in [40], and to identify two real-world nonlinear systems in the fields of speech signals and nonlinear audio amplifiers using simulated data, as shown in [42]. This is a first attempt to apply this method to real-world data, and the experimental results show that it achieved better performance, probably because the other approaches suffered from the large amount of data and their heterogeneity. Therefore, it could potentially be applied to other real datasets presenting the same complexity.

7. Conclusions

In this article, PBPs were proposed as an alternative to the widely used system identification algorithms available in the literature, such as NARMAX and its derivatives and convolutional neural networks based on different structures. The main difference with respect to the traditional algorithms is that PBPs operate on a simplified system representation in the spectral domain.

Experiments with a specific dataset showed that this approach yields better results when compared with other algorithms, proving to be effective on datasets with large amounts of data and heterogeneous compositions. In particular, the absolute error that was obtained was 63% smaller with respect to NARX and from 42% to 62% smaller with respect to various ANN-based approaches.

Author Contributions: Conceptualization, M.A., L.F., G.B., P.C. and C.T.; investigation, M.A., L.F. and C.T.; methodology, M.A., L.F., G.B., P.C. and C.T.; project administration, L.F. and C.T.; software, M.A., L.F. and G.B.; supervision, P.C. and C.T.; validation, M.A., L.F. and G.B.; visualization, M.A. and L.F.; writing—original draft, M.A., L.F. and C.T.; writing—review and editing, M.A., L.F., G.B., P.C. and C.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: This work used data publicly available from [46].

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

References

1. Yassin, I.M.; Taib, M.N.; Adnan, R. Recent advancements & methodologies in system identification: A review. *Sci. Res. J.* **2013**, *1*, 14–33.
2. Schoukens, J.; Ljung, L. Nonlinear System Identification: A User-Oriented Road Map. *IEEE Control Syst. Mag.* **2019**, *39*, 28–99. doi: 10.1109/MCS.2019.2938121. [CrossRef]
3. Turchetti, C.; Biagetti, G.; Gianfelici, F.; Crippa, P. Nonlinear System Identification: An Effective Framework Based on the Karhunen–Loève Transform. *IEEE Trans. Signal Process.* **2009**, *57*, 536–550. [CrossRef]
4. Kibangou, A.; Favier, G. Identification of fifth-order Volterra systems using iid inputs. *IET Signal Process.* **2010**, *4*, 30. [CrossRef]
5. Kim, J.; Nam, S. Bias-compensated identification of quadratic Volterra system with noisy input and output. *Electron. Lett.* **2010**, *46*, 448–450. [CrossRef]
6. Hu, Y.; Tan, L.; de Callafon, R.A. Noniterative tensor network-based algorithm for Volterra system identification. *Int. J. Robust Nonlinear Control* **2022**, *32*, 5637–5651. [CrossRef]
7. Chen, X.M.; Chen, H.F. Recursive identification for MIMO Hammerstein systems. *IEEE Trans. Autom. Control* **2010**, *56*, 895–902. [CrossRef]
8. Ren, X.; Lv, X. Identification of extended Hammerstein systems using dynamic self-optimizing neural networks. *IEEE Trans. Neural Netw.* **2011**, *22*, 1169–1179. [PubMed]
9. Zhao, W.X. Parametric identification of Hammerstein systems with consistency results using stochastic inputs. *IEEE Trans. Autom. Control* **2010**, *55*, 474–480. [CrossRef]
10. Schwedersky, B.B.; Flesch, R.C.C.; Dangui, H.A.S. Nonlinear MIMO System Identification with Echo-State Networks. *J. Control. Autom. Electr. Syst.* **2022**, *33*, 743–754. [CrossRef]
11. Chang, W.D. Identification of nonlinear discrete systems using a new Hammerstein model with Volterra neural network. *Soft Comput.* **2022**, *26*, 6765–6775. [CrossRef]
12. Chen, B.; Zhu, Y.; Hu, J.; Príncipe, J. Stochastic gradient identification of Wiener system with maximum mutual information criterion. *IET Signal Process.* **2011**, *5*, 589–597. [CrossRef]
13. Michalkiewicz, J. Modified Kolmogorov’s neural network in the identification of Hammerstein and Wiener systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 657–662. [CrossRef] [PubMed]
14. Brouri, A.; Kadi, L.; Lahdachi, K. Identification of nonlinear system composed of parallel coupling of Wiener and Hammerstein models. *Asian J. Control* **2022**, *24*, 1152–1164. [CrossRef]
15. Brouri, A. Wiener–Hammerstein nonlinear system identification using spectral analysis. *Int. J. Robust Nonlinear Control* **2022**, *32*, 6184–6204. [CrossRef]
16. Zhao, W.X.; Chen, H.F.; Zheng, W.X. Recursive identification for nonlinear ARX systems based on stochastic approximation algorithm. *IEEE Trans. Autom. Control* **2010**, *55*, 1287–1299. [CrossRef]
17. Chiras, N.; Evans, C.; Rees, D. Nonlinear gas turbine modeling using NARMAX structures. *IEEE Trans. Instrum. Meas.* **2001**, *50*, 893–898. [CrossRef]
18. Chen, L.K.; Ulsoy, A.G. Identification of a nonlinear driver model via NARMAX modeling. In Proceedings of the 2000 American Control Conference, ACC (IEEE Cat. No. 00CH36334), Chicago, IL, USA, 28–30 June 2000; Volume 4, pp. 2533–2537.
19. Amisigo, B.; Van de Giesen, N.; Rogers, C.; Andah, W.; Friesen, J. Monthly streamflow prediction in the Volta Basin of West Africa: A SISO NARMAX polynomial modelling. *Phys. Chem. Earth Parts A/B/C* **2008**, *33*, 141–150. [CrossRef]
20. Piroddi, L. Simulation error minimisation methods for NARX model identification. *Int. J. Model. Identif. Control* **2008**, *3*, 392–403. [CrossRef]
21. Chen, S.; Billings, S.A. Representation of non-linear systems: The NARMAX model. *Int. J. Control* **1989**, *49*, 1012–1032. [CrossRef]
22. Piroddi, L.; Lovera, M. NARX model identification with error filtering. *IFAC Proc. Vol.* **2008**, *41*, 2726–2731. [CrossRef]
23. Kadochnikova, A.; Zhu, Y.; Lang, Z.Q.; Kadirkamanathan, V. Integrated Identification of the Nonlinear Autoregressive Models With Exogenous Inputs (NARX) for Engineering Systems Design. *IEEE Trans. Control. Syst. Technol.* **2022**, 1–8. doi: 10.1109/TCST.2022.3171130. [CrossRef]
24. Narendra, K.S.; Parthasarathy, K. Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Netw.* **1990**, *1*, 4–27. [CrossRef]
25. Rahim, N.A.; Taib, M.N.; Adom, A.H.; Halim, M.A.A. Nonlinear System Identification for a DC Motor using NARMAX Model with Regularization Approach. In Proceedings of the International Conference on Control, Instrument and Mechatronics Engineering, CIM, Johor Bahru, Malaysia, 28–29 May 2007; Volume 7.
26. Kukreja, S.L.; Galiana, H.L.; Kearney, R.E. NARMAX representation and identification of ankle dynamics. *IEEE Trans. Biomed. Eng.* **2003**, *50*, 70–81. [CrossRef] [PubMed]

27. Boynton, R.; Balikhin, M.; Wei, H.L.; Lang, Z.Q. Applications of NARMAX in Space Weather. In *Machine Learning Techniques for Space Weather*; Elsevier: Amsterdam, The Netherlands, 2018; pp. 203–236.
28. Wei, H.L. Sparse, Interpretable and Transparent Predictive Model Identification for Healthcare Data Analysis. In *Proceedings of the Advances in Computational Intelligence*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 103–114.
29. Ayala Solares, J.R.; Wei, H.L.; Billings, S.A. A novel logistic-NARX model as a classifier for dynamic binary classification. *Neural Comput. Appl.* **2019**, *31*, 11–25. [[CrossRef](#)]
30. Chen, S.; Billings, S.A.; Luo, W. Orthogonal least squares methods and their application to non-linear system identification. *Int. J. Control* **1989**, *50*, 1873–1896. [[CrossRef](#)]
31. Billings, S.A.; Chen, S. Extended model set, global data and threshold model identification of severely non-linear systems. *Int. J. Control* **1989**, *50*, 1897–1923. [[CrossRef](#)]
32. Rahim, N.; Taib, M.; Yusof, M. Nonlinear system identification for a DC motor using NARMAX Approach. In *Proceedings of the Asian Conference on Sensors, AsiaSense 2003, Kebangsaan, Malaysia, 18 July 2003*; pp. 305–311.
33. Awadz, F.; Yassin, I.M.; Rahiman, M.H.F.; Taib, M.N.; Zabidi, A.; Hassan, H.A. System identification of essential oil extraction system using Non-Linear Autoregressive Model with Exogenous Inputs (NARX). In *Proceedings of the 2010 IEEE Control and System Graduate Research Colloquium (ICSGRC 2010), Shah Alam, Malaysia, 22 June 2010*; pp. 20–25.
34. Lü, Y.; Tang, D.; Xu, H.; Tao, S. Productivity matching and quantitative prediction of coalbed methane wells based on BP neural network. *Sci. China Technol. Sci.* **2011**, *54*, 1281–1286. [[CrossRef](#)]
35. Shafiq, M.; Butt, N.R. Utilizing higher-order neural networks in U-model based controllers for stable nonlinear plants. *Int. J. Control. Autom. Syst.* **2011**, *9*, 489–496. [[CrossRef](#)]
36. Rashid, M.T.; Frasca, M.; Ali, A.A.; Ali, R.S.; Fortuna, L.; Xibilia, M.G. Nonlinear model identification for Artemia population motion. *Nonlinear Dyn.* **2012**, *69*, 2237–2243. [[CrossRef](#)]
37. Ljung, L.; Andersson, C.; Tiels, K.; Schön, T.B. Deep Learning and System Identification. *IFAC-PapersOnLine* **2020**, *53*, 1175–1181. [[CrossRef](#)]
38. Zhao, F.; Hu, L.; Li, Z. Nonlinear system identification based on recurrent wavelet neural network. In *Proceedings of the Sixth International Symposium on Neural Networks (ISNN 2009), Wuhan, China, 26–29 May 2009*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 517–525.
39. Billings, S.A.; Wei, H.L. A new class of wavelet networks for nonlinear system identification. *IEEE Trans. Neural Netw.* **2005**, *16*, 862–874. [[CrossRef](#)]
40. Biagetti, G.; Crippa, P.; Falaschetti, L.; Turchetti, C. Machine learning regression based on particle Bernstein polynomials for nonlinear system identification. In *Proceedings of the 2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP), Tokyo, Japan, 25–28 September 2017*; pp. 1–6.
41. Farouki, R.T. The Bernstein polynomial basis: A centennial retrospective. *Comput. Aided Geom. Des.* **2012**, *29*, 379–419. [[CrossRef](#)]
42. Biagetti, G.; Crippa, P.; Falaschetti, L.; Turchetti, C. A Machine Learning Approach to the Identification of Dynamical Nonlinear Systems. In *Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO), A Coruna, Spain, 2–6 September 2019*; pp. 1–5.
43. Jolliffe, I.T. Principal component analysis. In *International Encyclopedia of Statistical Science*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 1094–1096.
44. Biagetti, G.; Crippa, P.; Falaschetti, L.; Luzzi, S.; Santarelli, R.; Turchetti, C. Classification of Alzheimer’s disease from structural magnetic resonance imaging using particle-Bernstein polynomials algorithm. In *Intelligent Decision Technologies 2019*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 49–62.
45. Noël, J.P.; Schoukens, M. F-16 aircraft benchmark based on ground vibration test data. In *Proceedings of the 2017 Workshop on Nonlinear System Identification Benchmarks, Brussels, Belgium, 24–26 April 2017*; pp. 19–23.
46. Noël, J.P.; Schoukens, M. F-16 Aircraft Benchmark Based on Ground Vibration Test Data. Available online: https://data.4tu.nl/articles/dataset/F-16_Aircraft_Benchmark_Based_on_Ground_Vibration_Test_Data/12954911 (accessed on 30 March 2022).
47. Andersson, C.; Ribeiro, A.H.; Tiels, K.; Wahlström, N.; Schön, T.B. Deep convolutional networks in system identification. In *Proceedings of the 2019 IEEE 58th Conference on Decision and Control (CDC), Nice, France, 11–13 December 2019*; pp. 3670–3676.