---

**Article:**
Zhang, T., Barthorpe, R.J. and Worden, K. (2020) On Treed Gaussian Processes and piecewise-linear NARX modelling. Mechanical Systems and Signal Processing, 144. 106877. ISSN: 0888-3270

https://doi.org/10.1016/j.ymssp.2020.106877

---

# On Treed Gaussian Processes and Piecewise-Linear NARX Modelling

T. Zhang, R.J. Barthorpe, K. Worden[1]

*Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD*

**Abstract**

In the scope of nonlinear system identification, traditional parametric models are widely adopted as simplifying approaches to modelling the complexity of nonlinearity. However, many high order parametric models are disadvantaged due to their inherent demand for model detection and their tendency to overfit in the absence of additional validation processes. Nonparametric models, such as the Gaussian Process (GP), though being naturally exempt from model detection, can involve expensive procedures of model optimisation. This article presents a Linear Kernel Chipman-based Treed Gaussian Processes (LK-CTGP), which is essentially an assembly of simple linear parametric models using a decision tree framework, to model nonlinear systems. The piecewise-linear structure of the LK-CTGP offers a natural geometric solution to modelling nonlinear systems, where no model detection is required. The essence of simplicity from the traditional parametric model is also completely retained within each of the submodels. The effectiveness of the LK-CTGP is illustrated here via a number of case studies from simple synthetic data to experimental data, on which Nonlinear Autoregressive eXogenous (NARX) systems will built from the data for in-depth study.

*Keywords:* Time series, Autoregressive models, Decision trees, Gaussian processes

## 1. Introduction

In the context of system identification in structural dynamics, the prior choice of modelling methods always inclines towards the methods in which all the physical insights of the system can be manifested via model parameters. However, such a requirement of comprehensive interpretation is rather rare in practice. When the physical insight into the system is not accessible, black box

[1]Corresponding Author: Keith Worden (k.worden@sheffield.ac.uk)

models are often introduced to establish a relation between input and output based on purely statistical characteristics of the provided data. To achieve such an input-output mapping, over decades, myriads of model structures have been proposed, which have covered a wide range of mathematical subdivisions associated with machine learning techniques. The practice of the linear system identification has already been populated by a series of well-established theories[1, 2]. Nowadays, the general points of interest for researchers do ground towards the more challenging field of nonlinear black box system identification. However, in the statistics community, the very research topic is considered as a partial relabelling of the long existing topic of nonparametric regression modelling, from which sprouts well-known methods such as artificial neural-networks (ANNs), support vector machines (SVMs), spline models, fuzzy models etc [3]. Along with the rehabilitation of Bayesian statistics during the 1980s and the growing trend of machine learning, the Gaussian process (GP) was inevitably introduced in conducting nonparametric modelling for nonlinear systems [4]. The GP is a natural nonparametric kernel-based machine learning technique, which is capable of providing versatile regression patterns via a specification of kernel functions. Despite naturally being able to model nonlinear systems as provided by their intrinsic inferential logic, the applications of GPs are extended and made more adaptive by establishing coalescence with other machine learning techniques. Over the years, on the stem of basic GPs, myriads of derivative or collaborative methods have been developed, such as GP-neural networks, mixtures of GP experts, Treed Gaussian processes, etc. [5][6][7]. This particular paper will discuss the implementation of a *Linear kernel Chipman-based Treed Gaussian Process* (LK-CTGP) in the identification of nonlinear systems. The concrete details on the mechanism and configuration of the LK-CTGP will be expounded in the later sections. The main purpose of establishing this decision tree-based model is in general bipartite. Firstly, the ordinary GP model is deficient at countering problems with the property of heteroscedasticity (nonstationary regression); the second point is that the traditional squared-exponential (SE) kernel requires a rather expensive optimisation procedure for its hyperparameters, if the data size is too large. The LK-CTGP proposed here chooses the linear kernel in lieu of the more traditional SE kernel, and utilises the decision tree framework to break down the nonlinearity geometry-wise by recursively partitioning the input space into subregions wherein a fine linearity is preserved. Therefore, the LK-CTGP is essentially a piecewise-linear approximator incorporating a Bayesian inference framework. Speaking of modelling time series, piecewise-linear models have appeared early since the 1980s under the name of Threshold Autoregressive (TAR) model introduced by Howell Tong [8]. The original model has been extended and adapted for use under various different purposes [9][10][11].

This paper takes a pertinent look into applying the LK-CTGP to studying the

behaviour of time series data. In the domain of time series analysis (TSA), the Nonlinear Auto-Regressive Moving-Average eXogenous (NARMAX) model is a popular model-inferencing framework offering great versatility and adaptability [12]. The NARMAX model possesses a high degree of generality by taking a bipartite consideration of both the auto-regressive nature and the moving average. To achieve more parsimonious improvement, the NARX (Nonlinear Autoregressive eXogenous) model is often used, which is a special case derived from the NARMAX model, when the noise model is assumed to be white Gaussian.

The NARX model naturally inherits the character of the NAR (Nonlinear AutoRegressive) model, where the current prediction of the system output is a function of the previous inputs and outputs.

The NARX model is distinguished for its inclusion of the eXogenous term, which accommodates a separate input from outside of the system, thus such an external input is naturally independent of the development of the system output. Such a configuration can be expressed in mathematical form by the following equation,

$$y_t = f(y_{t-1}, y_{t-2}, ..., y_{t-n_y}; x_{t-1}, x_{t-2}, ..., x_{t-n_x}) + \epsilon_i \tag{1}$$

where $y_t$ is the system output at the $t^{th}$ step; $x$ is the eXogenous input; $n_y$ and $n_x$ are the lags w.r.t $y$ and $x$; $\epsilon$ represents the residual (noise), which is assumed to be white Gaussian.

There are various forms for the NARX modelling function $f$; the common choice, in a certain sense of tradition, describes the internal mechanism of the NARX system with a multivariate polynomial (MVP) function. As a member of the family of parametric modelling methods, the MVP approach to the NARX system requires a two-step procedure. The first step is to determine the specific model structure (e.g. what terms to include in the polynomial function), which is known as the model *structure detection*. The second step is to determine the corresponding expansion parameters (e.g. polynomial coefficients) based on the result of the structure detection. However, the procedure of model detection can be rather troublesome, as an inappropriate establishment of the polynomial terms can lead to either poor prediction or overfitting issues. As a result, in recent years, the nonparametric treatment of the modelling of the NARX system has gradually gained increasing attention, as it features a major advantage over the 'traditional' parametric models. Nonparametric models such as Gaussian Processes (GP) [13], require no structure detection step. Instead, since nonparametric models have a fixed group of system parameters which establish a general conformity to the data space, the user can simply specify all these parameters without

reluctantly making selections on them. For example, Gaussian Processes as a nonparametric modelling paradigm, only have three *hyperparameters* to define once, for example a squared-exponential kernel has been selected for use [14]. Despite such an advantage, the nonparametric models do suffer from higher computational cost in terms of the hyperparameter estimation, especially when handling large datasets containing thousands of data points, as the computational cost is $O(N^3)$ ($N$ is the number of data points); this cost is associated with a matrix inversion at the heart of the GP algorithm.

In this paper, the authors present an advanced parametric model for NARX systems based on the nonparametric framework provided by the Chipman-based Treed Gaussian Processes (CTGP) [15], but restricted to linear models on the leaves of the tree. This structure produces a piecewise-linear modelling capability, and is applied here to the study of three case studies, each of which carries a special purpose in demonstrating the effectiveness of the algorithm. The first case study covers a well-customised synthetic dataset to delve visually into the fitting characteristics from the LK-CTGP[2]. The second case study deals with a practical problem involving the classic Duffing oscillator data. The mathematical insight of this case is accessible, therefore the performance can yet again be compared with the exact test data comprehensively. The last case study is a rather challenging experimental one regarding the nonlinear behaviour of automotive shock absorbers. The data space is rather skewed and it is hard to make an approximate fitting from visual approximation. Besides in this case, the mathematical insight will be unavailable due to the complex data environment.

The layout of this paper is structured explicitly simple for a clear demonstration of the effectiveness of the LK-CTGP. It begins with an explanation of the mathematical theory followed by the three case studies aforementioned, where individual summaries will be included respectively. A short general summary will be given at the end of the paper.

## 2. The Linear-Kernel Chipman-based Treed Gaussian Process

### *2.1. Model Performance Assessment Criteria*

The prediction from the LK-CTGP on the NARX data can be assessed in various ways. The common basic metric is the *One Step Ahead* (OSA) predictions of the model. The OSA prediction only uses the training data as the reference for making the prediction at a given time, which is described

[2]The 'Gaussian Process' part of the name will be maintained here, despite the fact that a linear kernel is adopted in this paper, as the methodology allows a general GP formulation with minimal modification. The implications for computational cost are another matter

by,

$$y_i^* = f(y_{i-1}, ..., y_{i-n_y}; x_i, ..., x_{i-n_x+1}) \tag{2}$$

It is arguable that the OSA prediction is not such a rigorous and descriptive metric to evaluate the model prediction performance. First, the OSA prediction completely depends on the training data, thus obviously it will have problems with overfitting issues. Second, the NARX system is a self-generative or self-developing time series system, where once the initial condition is given, the subsequent process is largely deterministic. Therefore, a good metric should provide a measure on the generative performance of the model when given the same initial conditions as the training data. This leads to the *Model Predicted Output(MPO)* predictions, as shown in equation (3), where the prediction at a certain time is computed based on the predictions made up to that time,[3]

$$y_i^* = f(y_{i-1}^*, ..., y_{i-n_y}^*; x_i, ..., x_{i-n_x+1}) \tag{3}$$

Once the MPO predictions are made, the *Normalised Mean Square Error* (NMSE) of the model can be computed as,

$$NMSE(\hat{y}) = \frac{100}{N\sigma_y^2} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{4}$$

According to long-standing experience in its use, the authors consider that an NMSE less than 5.0 shows a good fit to the data, while an NMSE less than 1.0 shows a excellent fit.

### *2.2. Methodology: Gaussian Processes*

The very basic foundation of the Gaussian process algorithm is in classical Bayesian inference governed by the well known Bayes' rule,

$$posterior = \frac{likelihood \times prior}{marginal\ likelihood} \tag{5}$$

[3]In the system identification community associated with electrical or control engineering, the OSA and MPO modes of prediction are often referred to as *prediction* and *simulation*, respectively.

The relation shown above can be easily used to establish rudimentary parametric regressions by specifying a prior and likelihood distribution for the corresponding parameters. At its very simplest form, a linear regression can be achieved by setting a prior-likelihood combination for the gradient and intercept parameters in the model. The GP is a special case built on the Bayesian inference framework. In the GP, there exists no designated inference towards any function parameters, instead, the inference is associated with the function itself. The term 'function' is used to describe the immanent relations between the input and output among the data space. This relation is not necessarily accessible for mathematical expressions with closed form. In the GP, the prior will directly be incarnated in the function as a multivariate Gaussian distribution over all the data in the space.

Following the detailed discussion in [13], the prior of the GP is specified as,

$$f(x) \sim N_p(m(x), k(x, x')) \tag{6}$$

where $m(x)$ is the multivariate mean and $k(x, x')$ is a covariance function/kernel.

To further explore this specification, it is conducive to conduct a well-clarified analysis through separation of the training and testing datasets. Then the prior bears the form below,

$$\begin{bmatrix} f \\ f_* \end{bmatrix} = N(\mathbf{0}, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix})$$

where $N$ stands for the multivariate normal distribution, $X$ is the training input data (arranged in matrix form) and $X_*$ is the test input data, $K(.)$ is the covariance matrix.

The GP is a special version of Bayesian inference where the prior specifies, not only the initial preference on the output $y$ ($y = [f, f_*]$) at each input entry, but also encodes the mutual correlations among each pair of data points via the presence of the covariance function/matrix. The covariance matrix, whose entries are outputs from a covariance function pre-selected by the user, commands the form of GP likelihood as it determines the predictive function on the given training data space. From the perspective of mathematical neatness and simplicity, the GP is extremely advantageous for analytical derivation of the posterior, simply because the prior in such a matrix form allows the inference of the posterior through matrix operations without the knowledge of the expressive form of both likelihood and marginal likelihood. Again, following [13] one has,

$$
\begin{aligned}
&f_*|X, y, X_* \sim N(\bar{f}_*, cov(f_*)),\ where \\
&\bar{f}_* \triangleq \mathbb{E}[f_*|X, y, X_*] = K(X_*, X)[K(X, X) + \sigma_n^2 I]^{-1} y, \\
&cov(f_*) = K(X_*, X_*) - K(X*, X)[K(X, X) + \sigma_n^2 I]^{-1} K(X, X_*). \quad (7)
\end{aligned}
$$

By definition, the covariance matrix is constructed to reflect the statistical variation at each input entry as a result of correlations with other data points in the space. Because the influence between points is reciprocal, the covariance matrix is completely symmetrical with elements that are all scalar values. A typical covariance matrix is,

$$
COV = \begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1n} \\ k_{21} & k_{22} & \cdots & k_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ k_{n1} & k_{n2} & \cdots & k_{nn} \end{bmatrix} \quad (8)
$$

where the $k_{ij}$ is an abbreviated form of $k(x_i, x_j)$, and represents the covariance between two points $x_i$ and $x_j$.

In the covariance matrix above, it could be observed that each row or each column describes the variance interaction between one point ($i^{th}$ or $j^{th}$ if it is the row or column being picked) with other points (including itself) in the dataset.

The parametric model has a fixed pattern to reason the prediction through the *a priori* user specification of the predictive form. As a nonparametric model, the GP specifies no fixed form to accommodate the prediction. Its own basic reasoning emerges from the generating criterion of the covariance matrix, which is constructed through the covariance function as mentioned formerly. It is a common practice to define the covariance in terms of distance as to encode a belief that the influence of an observation on another decays over distance. For the GP, the standard and most commonly-used distance covariance functions are the squared-exponential (SE) covariance functions, specified by the form,

$$
k_{SE}(r) = \exp(-\frac{r^2}{2l^2}) \quad (9)
$$

where $r$ is the distance between two mutually influential data points, $l$ is the distance influence weighing parameter, or simply *length scale*. All the parameters in the covariance functions are known as the *hyperparameters*.

The SE covariance function is infinitely differentiable, so that its presentation in the form of a predictive curve will be perfectly smooth. The GP is also versatile, as the covariance function does not have to be a distance covariance. For example, the GP also allows the covariance function to be set in relation to the axial coordinates so as to plug in a fixed curve form for the regression fitting. Hence the GP could also be applied as an alternative approach to conduct Bayesian linear regression (as in the applications in this paper) etc. From a more general angle, the covariance function could be specified in some way as to conduct nonstationary regression as well (by making the kernel depend on the absolute position of points, rather than the relative positions of pairs). However, the analysis of such a GP would be accompanied by various mathematical difficulties. In order to carry out Bayesian linear regression with a GP, the linear covariance function is given as,

$$k_{linear}(x_1, x_2) = kx_1^T x_2 + b \tag{10}$$

where the superscript $^T$ denotes a matrix/vector transpose.

The nonparametric SE kernel possesses the advantage of being natively adaptive to nonlinearity, in most cases, it ensures a smooth interpolation of the data. However, the downside is that the SE kernel requires extremely expensive optimisations for its hyperparameters if a sensible prediction is desired from its output. However, using the simple linear kernel effectively transforms the GP into a classic Bayesian linear regression, in which the gradient and intercept parameters can be obtained analytically given knowledge of the noise level (encoded in another hyperparameter). If the data from the nonlinear system behave uniformly in terms of local variance, the rough value of this noise can be determined by performing a single optimisation procedure for the noise parameter on the entire dataset.

Looking back to the inference of the posterior of the GP, it could be perceived that this is just the same as any Bayesian inference; the final stage of the prediction is to select the most suitable prediction from the posterior. For the Bayesian linear regression, it is extremely simple, as the posterior predictive distribution accounts for all possible linear fittings. Thus to select the best fitting model is just to select the best fit corresponding to the highest probability from the posterior predictive distribution. In the GP, the analytical posterior predictive is conceptually the same as its counterpart in Bayesian linear regression, and it is commonly addressed with the name 'Gaussian Process Marginal Likelihood (GPML)'. The GPML describes the likelihood of prediction which accounts for all possible predictions as weighed by their corresponding probabilities. Since, given the training data and the covariance function, the GPML is a measure of the reliability of the prediction

WRT a set of pre-selected hyperparameters, the Maximum Likelihood (ML) criterion selects the predictions parametrised by the hyperparameter values which maximise the GPML as the appropriate interpretation of the data space. The ML analysis suffers from exasperated analytical and computational difficulties compared to the MAP of the Bayesian linear regression due to the complexity of the GPML function, where its function profile as related to the hyperparameters is opaque to the probe of direct differentiation analysis.

In fact, the statement of the optimisation problem is rather simple and clear, that given a function, the objective is to find its global extremity. This particular type of problem casts a long shadow in the history of mathematics; its internal concept is rather coherent to any simple problems such as finding the extremity of a parabola. However its external expression varies and is much more complicated and intricate. In the GP, by the ML criterion, the objective function is the logarithmic GPML,

$$\log p(y|X,\theta) = -\frac{1}{2}y^T K_y^{-1} y - \frac{1}{2}\log|K_y| - \frac{n}{2}\log 2\pi \tag{11}$$

where all the hyper-parameters are contained in the covariance $K_y$.

Given a new data space, the actual graphical profile of the equation above is mostly unavailable unless the data space is well organised, expressing an obvious behaviour. To search for a maximum, there are a number of difficulties to encounter. First at different selections of the covariance function, the number and the type of the hyperparameters could be radically different. Thus this issue gives rise to difficulties in constructing general analytical models for the optimisation. The next problem is that the presence of the $K_y$ also implies the whole dataset will act as a dynamic influential factor imposed on the equation, thus leading to stacked complexity in operating matrices, especially matrix inversions. The impediments in the computational and operational cost are already very demanding to the mathematical manipulations as well as computational efficiencies, but the problem of multiple local extrema is even worse for trapping the optimisation away from the global extremity.

### *2.3. Chipman-based Treed Gaussian Processes*

A *Treed Gaussian Process* (TGP), in mathematical terms, is an amalgamation of a Binary Decision Tree (BDT) and Gaussian Processes (GP) [7]. A decision tree is a logical mapping process through which the elements of a given input space will be assigned into different groups represented by leaves of the tree based on a series of criteria [16]. A binary tree basically means the criteria take the form of a simple choice of YES/NO, thus branching the underlying space into two sub groups. Through repeated application of this process, a treed structure will be established. Figure 1 shows a typical BDT.

Figure 1 is taken as a typical BDT paradigm to explain the decision tree terminologies. A BDT is essentially a bifurcating process. A parent node can bifurcate into two child sub-nodes. The root of the tree at the top is known as the root node. Those end nodes at the bottom of the tree are external nodes, also known as leaves. All the rest of nodes in the middle who do bifurcate, are called internal nodes. In such a process, the BDT may be briefly thought of as a tangible representative framework for partitioning a set of data.

Within a traditional GP setting with distance-based kernel functions, the inference is based on the argument that predictions at any point in space are latently influenced by all other data points in the training space.Thereby, a covariance matrix is constructed which contains all the weight of the influence between any pair of data points in the training space. If the covariance function is non-distance based, the GP can also simply become a parametric model. In a TGP, the GP regressions are only carried out on subsets of data corresponding to the leaves of the tree. This can result in very large computational savings e.g. if there are two leaves with half of the total dataset associated with each, the GP cost will be $O(N^3/4)$ rather than $O(N^3)$.

Gramacy introduced one type of TGP model [7] which has seen application in numerous contexts. More strictly, Gramacy's TGP (GTGP) is the first attempt to amalgamate the GP model with the decision tree. Therefore Gramacy is indubitably recognised as the founder of the TGP. Before the introduction of the GTGP, its closest predecessor is the well known Bayesian Classification and Regression Tree (BCART), introduced by Chipman *et al* [17], where the idea of a Bayesian stochastic tree was first introduced. The current paper discusses another type of TGP which will be referred to as the 'simplified TGP' or 'Chipman-based TGP'. The CTGP suggests a TGP model which completely springs from the raw ideas of Chipman's BCART model. In order to prevent any confusion between these two TGP models, one should know that the Gramacy-based TGP also borrows ideas from Chipman's theory, but only in terms of how to alter the structure of the tree. The current paper sets complete reliance on Chipman's ideas, which will be illustrated subsequently.

The CTGP samples the partitions in the training space as in a form of growing a tree. The tree is to be treated as a state in a Markov Chain, thus the holistic sampling process manifests itself as a manner of *Markov Chain Monte Carlo* (MCMC). The most crucial aim of sampling in the MCMC space is to arrive at those trees with high posterior probability which is the total product of local posteriors from all the leaves. The posterior of the tree can be expressed mathematically by,

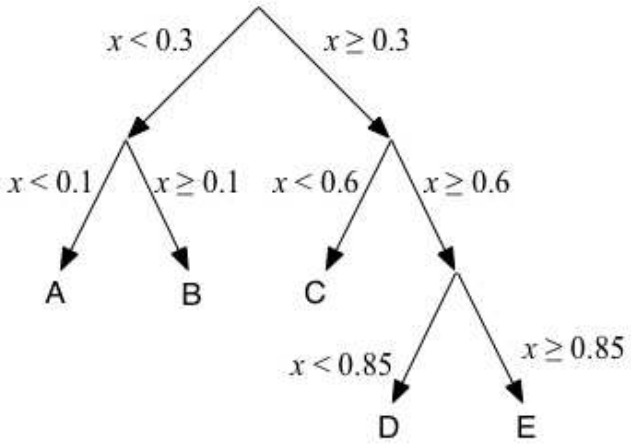


Figure 1: Typical Binary Decision Tree.

$$
\begin{aligned}
p(T|X,Y) &\propto p(Y|X,T)p(T) \\
p(Y|X,T) &= \prod_{i=1}^{b} p(Y_i|X_i)
\end{aligned}
\tag{12}
$$

where $T$ represents the tree; $X$ and $Y$ are the training inputs and outputs respectively; $b$ is the number of leaves.

To achieve suitably rapid convergence to appropriate trees, a Metropolis-Hastings (MH) algorithm with a complement of four tree-structure alteration operations is adopted.

Each step of the Markov Chain features a proposal stage and evaluation stage. The proposal stage involves the proposition of a new tree through application of one of the four operations described below. The evaluation stage involves evaluating the *Maximum a posteriori* (MAP) estimate of the newly-proposed tree, and then setting forth a comparison against the posterior from the last accepted tree. The residual that results from the subtraction between the logarithm of these two posteriors will be taken as a metric to decide on the probability that the proposal should be accepted. Through a large number of MCMC samples, the process will eventually explore the MCMC state space for high posterior probability trees, spending more time in regions of high posterior probability.

As mentioned above, the sampling of the tree structure features four operations; these operations are GROW, PRUNE, CHANGE and ROTATE. Each of these operations possesses a high similarity to the ones described in Chipman's paper regarding BCART under the exact same names, except for the ROTATE, where Chipman used the idea of SWAP. The ROTATE operation derives its originality from Gramacy.

In simple terms, the four operations have the following effect on the structure of the tree:

- GROW: add one partition by splitting one leaf node of the tree.
- PRUNE: remove one partition by joining two sibling child nodes.
- CHANGE: relocate an existent partition by changing a splitting rule in the tree.
- ROTATE: maintain the partitions but change their 'staging' sequence by rearrangement of the tree structure.

After the alteration of the tree structure comes the evaluation of the MAP of the tree. Normally this stage features an optimisation of the marginal likelihood function of each individual GP in terms of all their covariance hyperparameters, if a distance-based kernel function is chosen. To find the extrema of a given function, suggested optimisation approaches include either a gradient-based line search or a sampling-based search. In the particular application of the NARX model presented in this paper, the only parameter that requires the optimisation is the noise level, and it is treated as known. The linear function parameters can be analytically obtained through the Bayes' rule at the knowledge of the noise level.

Following the completion of the MAP evaluation, the MAP estimate is used for the Metropolis-Hastings evaluation, in which the MAP estimates of the current and last-accepted tree are compared in order to decide whether to accept the newly-proposed tree. The rule of the Metropolis-Hasting algorithm is stated as,

$$A = min(1, \frac{P(T')}{P(T^*)} \frac{Q(T^*; T')}{Q(T'; T^*)}) \tag{13}$$

where $*$ denotes the proposed tree, $'$ denotes the current tree state, $Q$ is the transition probability which depicts the probability of jumping between states in the Markov space, $P$ is the MAP of the tree. The proposed state will be accepted with probability $A$.

To compare and contrast between the two TGP approaches is not within the chief concerns of this paper. However, still for not downplaying its

importance, a brief summary is provided here. For more concrete and detailed comparison, one can refer to.... The fundamental difference of the newly developed CTGP model from th original GTGP model lies mainly on the ground of its internal modelling structure. The GTGP incorporates a full probabilistic sampling scheme, involving a four-layer hierarchic specification for its priors. It features a more sizeable pool of hyperparameters, whose introduction and specification purport to establish a Gibbs sampling based system to entitle a full probabilistic behaviour of the sampling process. However, in contrast, the CTGP is compromisingly less probabilistic, as the local posterior of each leaf is obtained deterministically via optimisation. This trade-off does not necessarily conclude that the CTGP is a simplified version of the GTGP apart from being merely structure-wise. The GTGP naturally outstrips the CTGP on the ground of sampling rate, however, as so many more hyper-parameters are involved in comparison, the search for the global optimum will take more iterations, as well as a concession in the overall posterior accuracy. Despite the well-known issue with the local optima for a deterministic optimisation process, the current CTGP uses a probabilistically guided searching scheme to ensure a local posterior optimum at each leaf. In short, the GTGP holds an edge of advantage in overall speed, but less accurate in giving the final optimal posterior. Being less robust and flexible is another major downside for the GTGP, as more hyperparameters lead to more prior assumptions.

## 3. Case Studies: Synthetic Data for Piecewise-Linear Systems

The first case study covers a pertinently-designed synthetic dataset that is used to demonstrate the effectiveness of the LK-CTGP characteristically. This case study consists of three sub-cases, which are arranged in order to study the performance of the LK-CTGP at different levels of linearity and complexity in terms of the number of time series lags. The first case study is on a synthetic bilinear NARX system with single lag, while the second one is a simple deepening of complexity by constructing a trilinear NARX system with single lag. The third and final member of the set considers a bilinear system with three lags.

The simple linear-based case study allows a better visualisation of the actual curve fitting from the LK-CTGP, that thereby, the characteristics of prediction from the algorithm can be analysed more intuitively. The reason for choosing only up to three lags, is due to the limiting problem of the curse of dimensionality (COD). The COD is a common issue associated with high-dimensional data analysis, where the escalation in the dimensionality requires an exponential increase in the number of data points to characterise the behaviour of the data set.

### *3.1. Case study 1: Bilinear NARX system*

The simple bilinear NARX system is set as a basic example to study, where only a single lag is considered along with a simple uniform random eXogenous input. The generative function of the model is,

$$\begin{aligned} y_i &= -0.7y_{i-1} + 10^{-6}x_{i-1};\ \text{if} \quad y_{i-1} < 0 \\ y_i &= 0.7y_{i-1} + 10^{-6}x_{i-1};\ \text{if} \quad y_{i-1} > 0 \end{aligned} \tag{14}$$

where $x \sim \text{Unif}(-5, 5)$.

4000 data points were generated through this process with Gaussian-distributed measurement errors added subsequently (0.001% of rms). The current choice of a distinctively small noise has made concessions toward the fact that the performance of the MPO model is heavily noise-dependent, since the MPO, in contrast with the OSA, make each prediction in the time axis based on its previous predictions, which allows the error to augment.

Apparently in this 3D data space, there is only one natural split at $y = 0$ visually in the form of a straight line. Via applying the LK-CTGP, this split has been successfully and precisely located within 500 MCMC rounds (Given 500 rounds, 20 repetitive runs unanimously give the exact same partition for this case). Figure 2 below shows the fitting of the training data space with two regions produced by LK-CTGP which exactly captured (split at $y_{n-1} = 0.003e^{-6}$) the bilinear nature of the data. The partition has been placed at the correct vicinity around $Y_{n-1} = 0$(The exact value is 0.002 for the case shown in Figure 2).

According to the fit from Figure 3, the LK-CTGP produced an excellent MPO prediction with an error less than 0.001 for the NARX system.

### *3.2. Case study 2: Trilinear system with one lag*

The second case study of the trilinear system is governed by the generative function,

$$\begin{aligned} y_i &= -0.7y_{i-1} + 10^{-6}x_{i-1};\ \text{if} \quad y_{i-1} < 0 \\ y_i &= 0.7y_{i-1} + 10^{-6}x_{i-1};\ \text{if} \quad 0 < y_{i-1} < 3 \\ y_i &= -1.5y_{i-1} + 6.6 + 10^{-6}x_{i-1};\ \text{if} \quad y_{i-1} > 3 \end{aligned} \tag{15}$$

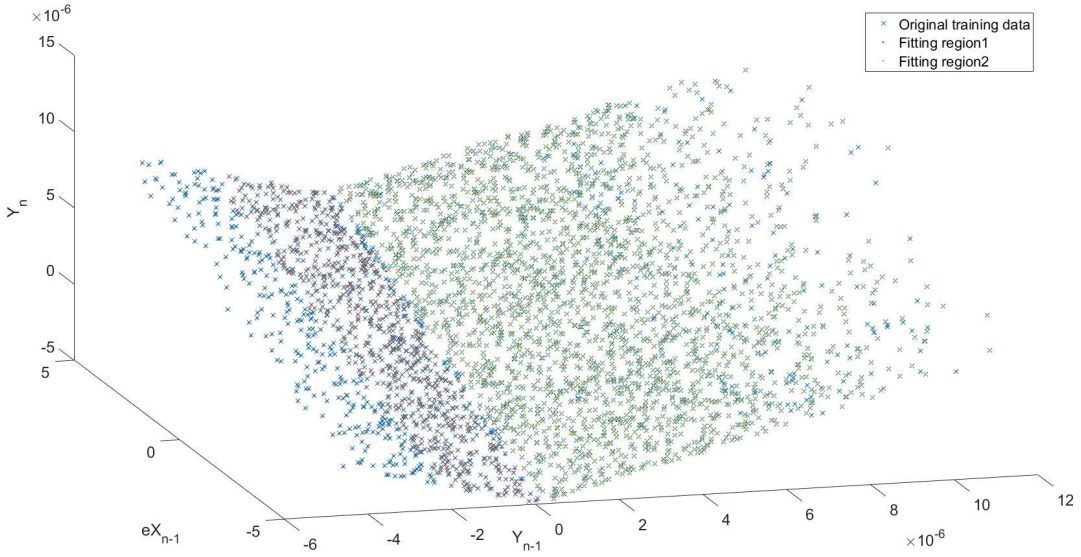


Figure 2: Model fit on the bilinear system with one lag.

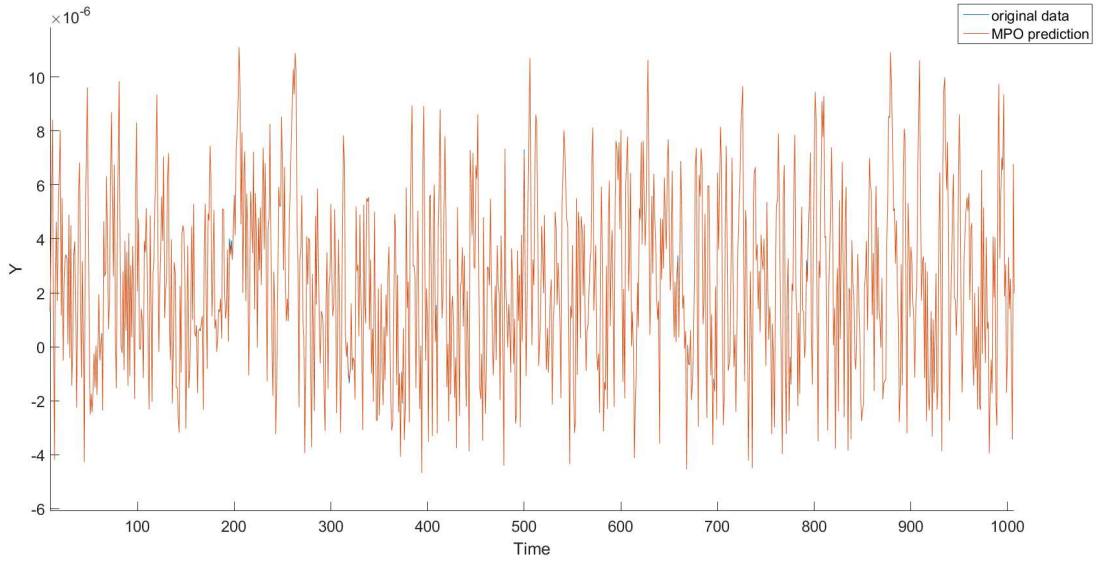


Figure 3: MPO prediction on the bilinear system with one lag.

The noise level is set as the same across all case studies. Via applying the LK-CTGP, the fitting pattern has been shown in Figure 4.

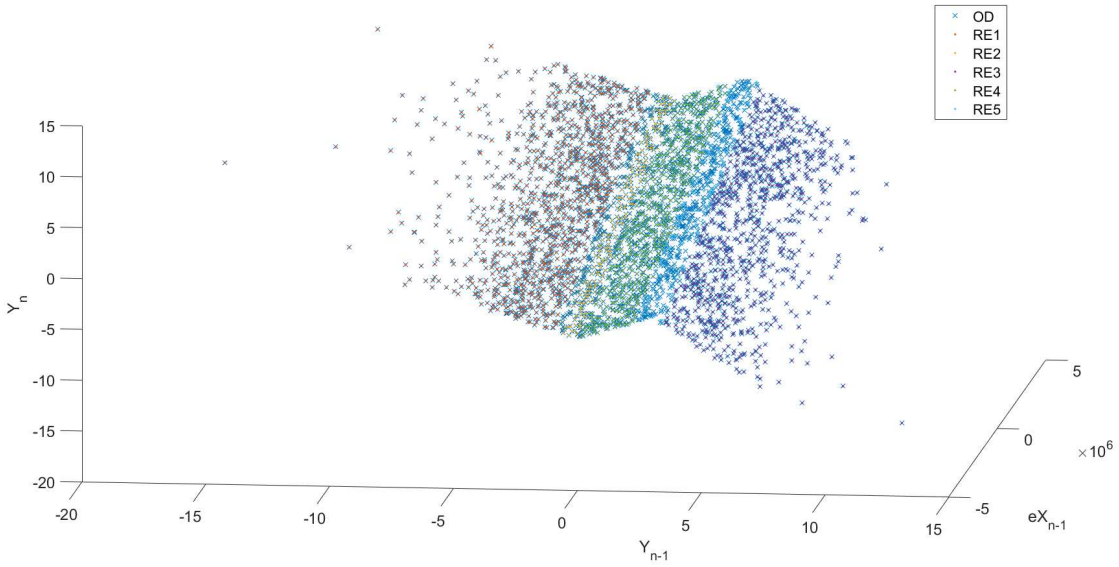


Figure 4: Fitting on the trilinear system with one lag.

20 individual runs have been carried out consecutively to explore the uncertainty in the partitioning pattern. Figure 4 does display the most typical result (13/20), and the rest 7 runs, though in resemblance, give a slight adaption by perfectly giving one partition at one or both joint sections of the planes. As a result, this indicates that for this particular trilinear case study, the CTGP is more inclined to conservatively explain the transient part using more partitions.

The fitting outcome indicates that six regions are generated to model the trilinear system, which differs from the ideal three-region scheme. The extra three regions are comparatively much smaller than the three main regions, as they are produced to interpret the transitions between two adjacent regions; two are placed around $Y_{n-1} = 0$ and one is placed around $Y_{n-1} = 3$. Theoretically, the additional interpretation with these extra regions is redundant, if the algorithm can smartly find the two exact locations to make partitions. However, during the process of MCMC search, it is rather rare to locate accurately the exact transitional locations without making inferior partitions in the first place. As having been explained before, the MCMC walk in the tree structure space is guided by four tree-structure alternating operations (Grow, Prune, Change, Rotate), but this does not assure that, the full set of optimal partitions will be eventually achieved through such a process. Especially when the tree-structural complexity has

been raised during the early stage of the MCMC process, which prevents a direct walk into the optimal state via the four operations. Accordingly, for complicated datasets, issues with over-splitting are inevitable for such type of decision tree-based models. Despite not being able to provide an exact set of ideal partitions, the LK-CTGP still performed well in breaking down the nonlinearity, resulting in a prediction with minimal errors. Figure 5 shows the MPO predictions (MSE=0.82) generated from such a fitting scheme by the algorithm.

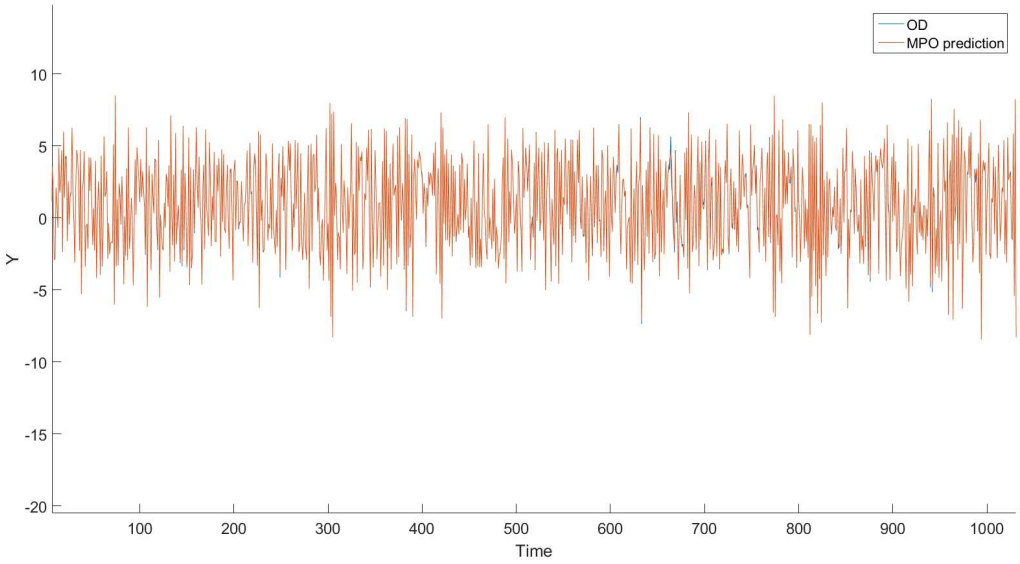


Figure 5: MPO predictions for the trilinear system with one lag.

### *3.3. Case study 3: Bilinear system with three lags*

Increasing the number of lags does raise the complexity dramatically as a result of the expanded dimensionality of the training dataset. Recalling the Curse Of Dimensionality, at higher dimensions, a dataset with fixed size will experience a general loss of resolution of its behaviour, becoming less smooth and more sparse. Apart from the effects on the data, high dimensionality will also cause troubles for the MCMC searching process for the tree, because the size of the sampling space is proportional to the dimensionality.

For the bilinear system with three lags, the generative function is given as,

$$
\begin{aligned}
y_i &= -0.7y_{i-1} + 0.7y_{i-2} - 0.2y_{i-3} + 10^{-6}x_{i-1};\ if \quad y_{i-1} < 0 \\
y_i &= 0.7y_{i-1} - 0.7y_{i-2} + 0.3y_{i-3} + 10^{-6}x_{i-1};\ if \quad y_{i-1} > 0
\end{aligned}
\tag{16}
$$

Although the fitting on the training data cannot be visualised, the mathematical result shows that the LK-CTGP still successfully generated one partition and placed it in the close vicinity of $Y = 0$. However, on average, it took significantly longer to arrive at that correct partition, compared to the bilinear case with single lag (see Table 1). Figure 6 shows the MPO predictions on the test data. The overall prediction still fits well on the original data, however, there are two intervals ([500,550] and [650,700]) where the MPO predictions failed in tracking the progress. The zoomed-in view in Figure 7 at the error section [500,550] shows that the first major departure from the test data occurs at the iteration 480 where $Y = 0$. $Y = 0$ is the exact joint section between two linear hyper-planes. This implies that the system is not well modelled close to the discontinuity as attributed to the nature of a partitioning based method. The induced error will propagate into the following predictions. However, the error is very likely to be incurred by the inaccurate placement of the partition in that area. However, along with the expansion of the dimensionality, inevitably the chance of drawing out that partition will fast dwindle. Besides, every newly-introduced lag has to carry the predictive error incurred before, thus total error accumulates faster with added lags.

Table 1 below summarises the performance of the LK-CTGP on each case study in terms of prediction error, number of splits generated and computational cost (Each MCMC round takes approximately 0.1s in real time on a four-core Xeon). Wherein, it can be seen that the increase in the number of lags does heavily weaken the prediction accuracy of the algorithm, despite the fact that the correct split has been located without incurring any over-split. The possible explanations can be largely ascribed to the problems with the curse of dimensionality. such a problem can also be observed from the drastically-increased number of MCMC rounds to locate the split. The added piecewise-linear segment naturally increased the overall prediction error, when compared with the bilinear system. But the influence is not as severe as in the case of escalated dimensionality. Therefore, it can be said the issue with over-splitting won't damage the overall performance to a significant level. To demonstrate the necessity of partitioning the training data space, the prediction NMSE of a linear fitting model has been listed for all four cases as a reference. The huge errors given by the linear models indicate its incapability at capturing or approximating the piecewise behaviour of

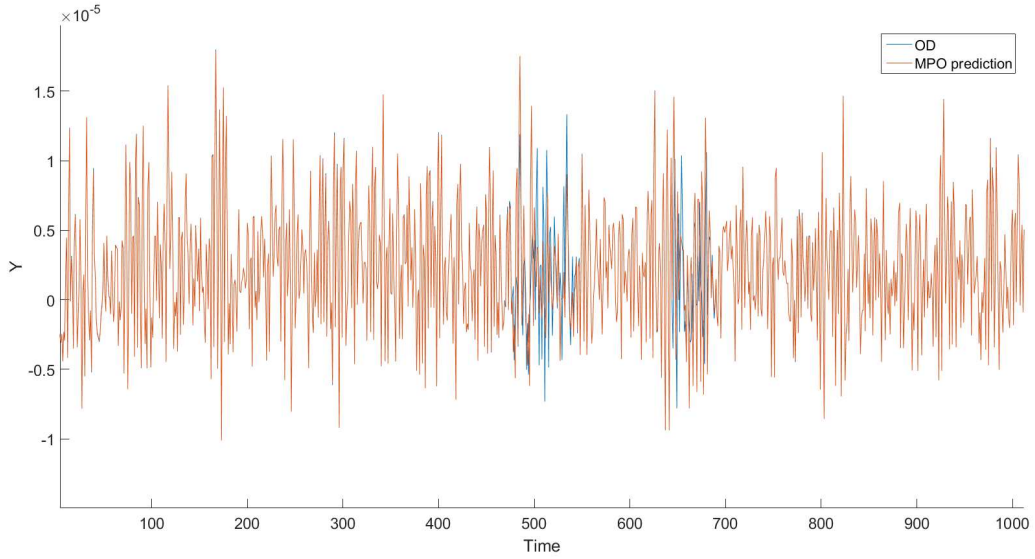


Figure 6: MPO predictions for the bilinear system with three lags (NMSE=3.1).

the time series, as well it shows the AR process is rather sensitive to the accuracy of the fitting.

| Models | NMSE | linear Ref.NMSE | Number of splits | Avg MCMC steps |
|---|---|---|---|---|
| Bilinear1lag | 0.001 | $> 1000$ | 1 | $< 200$ |
| Bilinear2lags | 1.2 | $> 1000$ | 1 | $< 700$ |
| Bilinear3lags | 3.1 | $> 1000$ | 1 | $< 2000$ |
| Trilinear | 0.3 | $> 1000$ | 5 | $< 800$ |

Table 1: Comparisons of performance for all the case studies

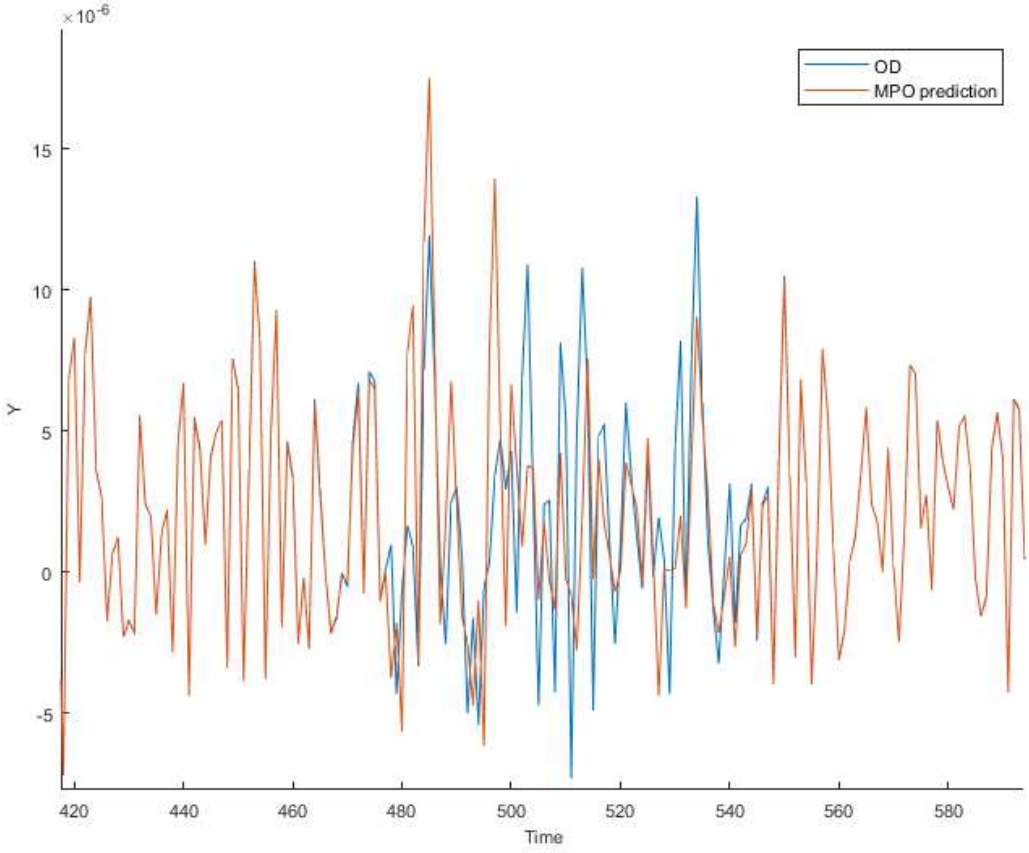


Figure 7: Zoomed MPO predictions at the area of increased errors.

## 4. Case study: Duffing Oscillator

The Duffing oscillator is a classic example for studying NARX behaviour. The physical insight is explicit as it can be formulated into mathematical equations. In this case study, a simple Duffing dynamical model will be studied. The data will be processed preliminarily through a linear regression process. Then a subsequent LK-CTGP study will be performed. Both OSA predictions and MPO predictions will be acquired for comparison.

The basic generative function for this Duffing oscillator NARX system is given as follows,

$$y_n = ay_{n-2} + by_{n-1} + cy_{n-1}^3 + dx_{n-1} + \epsilon \tag{17}$$

where $y$ is the output response variable, displacement; $x$ is the eXogenous input variable, the external force; $\epsilon$ is the noise; $a$,$b$,$c$,$d$ are the parameters.

The generative function in equation (17) shows the true relation between the input and output to produce a sequence of responses $y$ in terms of time is a 3D cubic function with input variables $y_{n-1}$, $y_{n-2}$ and $x_{n-1}$. Therefore, a simple linear regression model is supposed to be inept at comprehensively describing the behaviour of the data. Naturally the cubic behaviour can be approximated by a piecewise-linear model, when given the correct partitioning locations to reasonably accommodate each individual linear model. The LK-CTGP offers a binary process to recursively partition the input space into regions, which is considered to be efficient and effective.

Before delving into the performance of the CTGP on the data, it is worthwhile to study the generative function theoretically. From equation (17), it is clear that the nonlinearity is introduced by the cubic term associated with the first lag of $y$. from the partial differentiation with respect to each of the input variables, it can be found that only in the dimension of $y_{n-1}$, does the function have non-constant slope. Such a fact manifests that the entire input space has no gradient change in either of the dimensions of $y_{n-2}$ or $x_{n-1}$; therefore, the reasonable partitions are exclusive to taking in the dimension of $y_{n-1}$, where once the other two input variables are held fixed, the curve simplifies to a visualisable classic one-dimensional cubic function. Through performing the partial differentiation on equation (17) with respect to $y_{n-1}$ and equating the expression to zero, one can obtain the two roots,

$$r_{1,2} = \pm\sqrt{\frac{-b}{3c}} \tag{18}$$

Naturally these two roots represent the local peak and trough of the cubic function. The 3D cubic surface under the current Duffing setup can naturally be approximated as a three-piece linear system by putting partitions at these two roots if the $y$ set contains both roots. However, one should be aware that, the highly nonlinear parts at the peak and trough may require extra partitions to accommodate using piecewise linear approximators. Another thing worthy of notice here is that, the AR setup listed above in equation (17) relies on the chaotic stability of the Duffing system, which is a common paradigm in studying chaos theory [18]. The eXogenous input representing random excitation is a crucial influencing factor to such stability, which is amplitude dependent as shown by Rand [18].

The current value set-up for equation (17) is $a = -0.98$, $b = 1.97$, $c = -5000$, $d = 1e-6$ and $\epsilon = 0.0001 * \sigma_y$ (0.001% of RMS value). The eXogenous input is uniformly drawn from [-100,100]. 4000 points are generated. Under such a set-up, the $y$ values are bounded in [-0.004,0.004], while the two roots are $\pm 0.114$. The $y$ values do not contain both the peak and trough of the NARX generative function in equation (17). Since the $y$ values concentrate in the small centre interval between $\pm 0.114$, a weak nonlinearity in the system is to be expected. Figure 8 shows the original time series data of $y$.

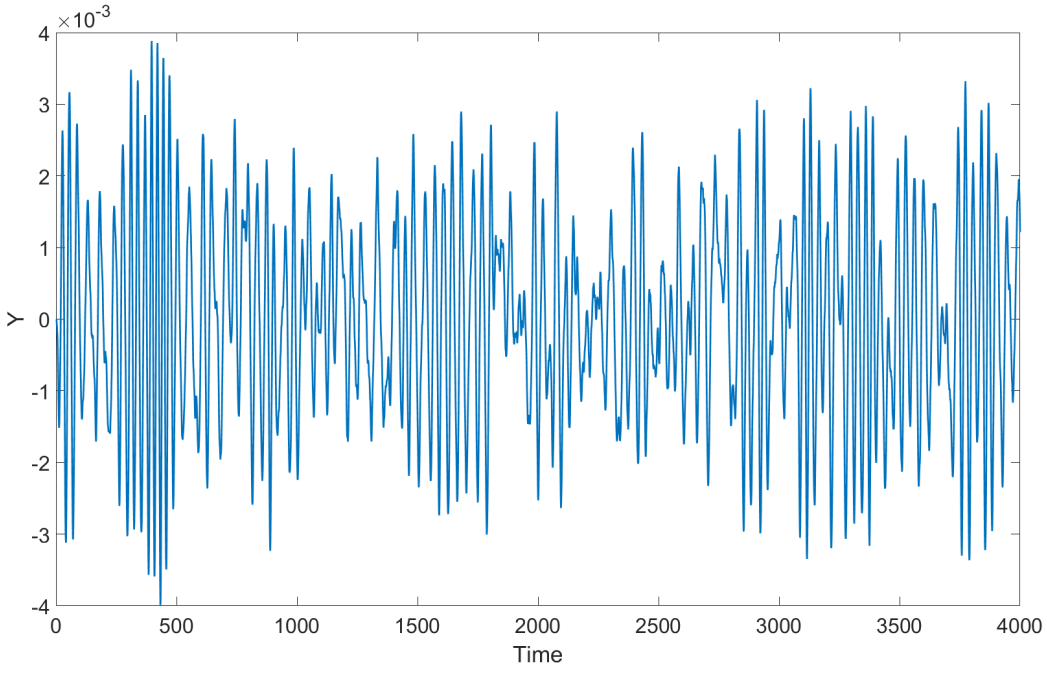


Figure 8: Original Time Series data for Duffing oscillator case study.

When applying a simple least-squares (LS) model to the data space constructed by input variables $[y_{n-1}, y_{n-2}, u_{n-1}]$ and output $y_n$, the OSA predic-

tion gives an extremely good fit to the original data by producing an NMSE value of 0.02 which is much less than 1.0. However, as has been addressed before, the OSA prediction is a rather rough metric to assess the NARX model. The MPO predictions of the first 1500 data points from the LS model are shown in Figure 9.

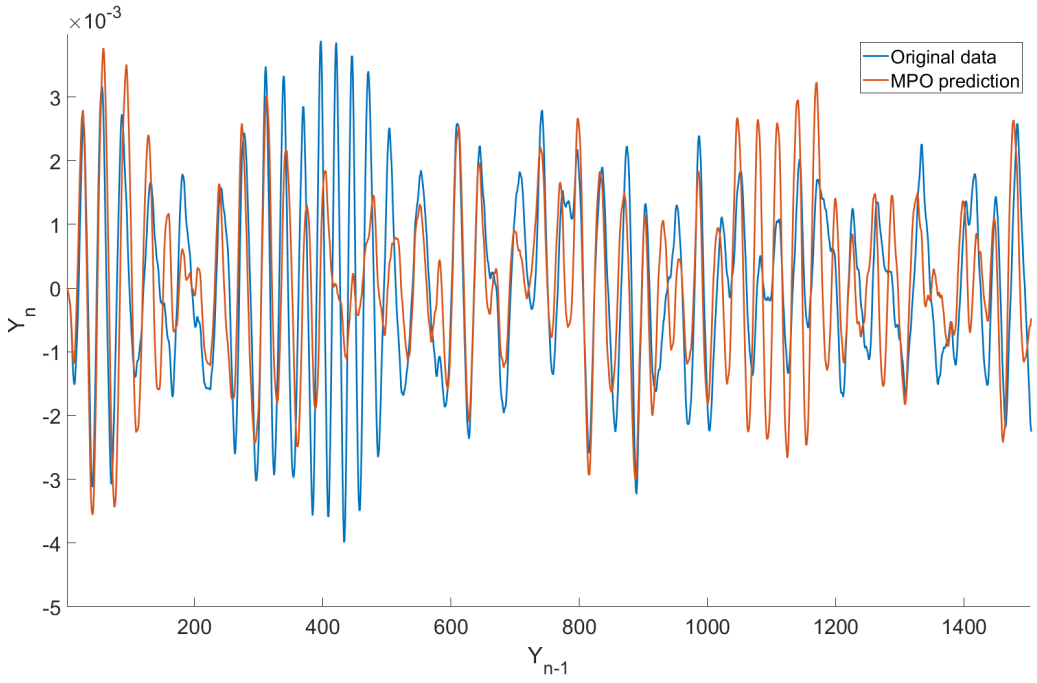


Figure 9: MPO predictions of LS model against original data.

Figure 9 shows that, even though the OSA predictions of the LS model show exceedingly good agreement to the original data at a low level of nonlinearity, the small nonlinearity in the system can be large enough to cause significant departures from the original data in the MPO predictions. Here the NMSE for the MPO predictions of the LS model is 69.5.

If the LK-CTGP is applied, the MPO predictions for the first 1500 data points (for graphical clarity) are as shown in Figure 10.

The LK-CTGP produced six leaves to accommodate the nonlinearity in the system. All the leaves are generated from partitions concentrated in the dimension of $y_{n-1}$, therefore, these partitions are reasonably placed. Figure 10 shows a good agreement between the MPO predictions of the LK-CTGP model and the original data. The NMSE here is decreased to 1.26. Another fairly interesting observation from the LK-CTGP plot is that, most of the major departures from the test data curve concentrate towards the middle section ($y = 0$), both end sections ($y = \pm 0.004$,) and in-between

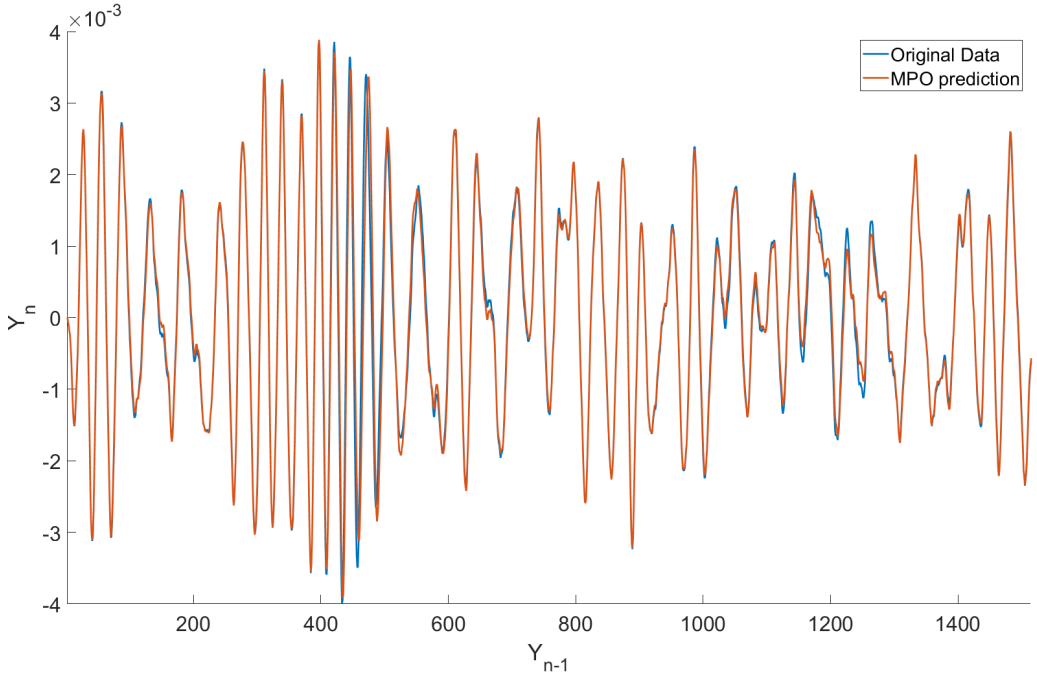


Figure 10: MPO predictions of LK-CTGP model against original data.

sections ($y = \pm 0.0015$). $y = 0, \pm 0.0015$ are approximately three of the seven partitions, which manifests that the error arises at the presence of discontinuity. Besides, it is natural for Bayesian methods to produce less accurate predictions at the boundaries of the data, where the data points correlate with less points than the central points.

## 5. Case study: Automotive Shock absorber

In the case of applying the LK-CTGP to studying the behaviour of a shock absorber, the objective of the modelling is to use the algorithm to establish a relation between the restoring force and other factors. The study of the shock absorber is a more challenging case compared to the case study of the Duffing oscillator, because the exact governing equation is not available and more factors will influence the result.

The shock absorber is a crucial part in the assembly of the automobile suspension system, whose characteristics contribute heavily to the ride comfort and handling properties of a vehicle. The conventional industrial modelling treatment to the shock absorber tends to simplify its mechanism as a basic linear spring-damper system. However, the experiments studied by Lang [19] and Hagedorn and Wallaschek [20] have brought debate against the validity of such an assumption of linearity. In their work, the shock absorber

behaviour is significantly nonlinear, and behaves quite distinctly when the shock absorber is in compression or rebound.

Lang [19], developed a rather rigorous analytical model to describe such nonlinear behaviour. His model is heavily parameterised with 87 parameters being introduced. Though it did show a good agreement with the experiment, the Lang model is far from serving the purpose of generalisation, as it only applies to a particular absorber at a certain configuration.

Automotive dampers are known to be highly frequency and amplitude dependent; this means that identification is generally complicated and even nonlinear models are limited in their capability if they have constant parameters. However, beyond the scope of accounting for the physical insight comprehensively, a straightforward approach of obtaining experimental characterisation can be applied by repeatedly taking measurements of the restoring force and velocity at different levels of excitation frequency and amplitude, the actual profile of the characteristics diagram can eventually be plotted for fixed frequencies. Despite such measurements inevitably having to discard some information, which makes the acquired data too coarse for accurate simulations, it provides an opportunity of constructing restoring force surfaces [21] which inherently carry the information of displacement and velocity. The paramount benefit from such restoring force surfaces is that they are a nonparametric representation which is independent of the *a priori* model of the structure.

In fact, the data considered in this paper come from a test carried out using random excitation, so the circumstances of the test mean that frequency effects are 'averaged out' throughout the data. Details of the test can be found in [21], but essentially a shock absorber was blocked at one end and a given velocity profile was imposed at the other end; measurements of the force were obtained at the blocked end. The object of the test was to characterise the force in the absorber as a function of displacement and velocity. It was assumed that any dynamic behaviour would be minimised in the test in the sense that the force function $F(y, \dot{y})$ would be static and there would be no inertial effects (although the absorber is blocked, there is relative movement of the two ends, and internal movement of the absorber fluid between chambers). In this paper, the process $\dot{y} \longrightarrow F$ is modelled. In the first instance, the LK-CTGP is used to determine the 'static' properties of the relationships, but then extended to see if dynamic behaviour can be identified and modelled. As will be shown, the nonlinearity in the absorber is fairly clearly of a piecewise-smooth variety, and so the LK-CTGP is expected to be aplicable.

The dataset consists of 7192 data points featuring three variable dimensions: the restoring force, displacement and velocity. In order to commence a time series analysis with the LK-CTGP under the considerations above, the

restoring force is indicated as the NARX model output, and the velocity is chosen as the exogenous input. The reason for choosing the velocity over the displacement in this case is that the measurements from the displacement sensor was subject to a higher level of noise, which can incur high levels of prediction error, as the time series analysis is rather sensitive to noise. To avoid potential over-fitting, the first 5000 data points are selected as the training dataset, while the rest goes in the bracket of testing data. The overall dataset is presented in Figures 11 (3D perspective) and 12 (2D perspective with one dimension projected out).

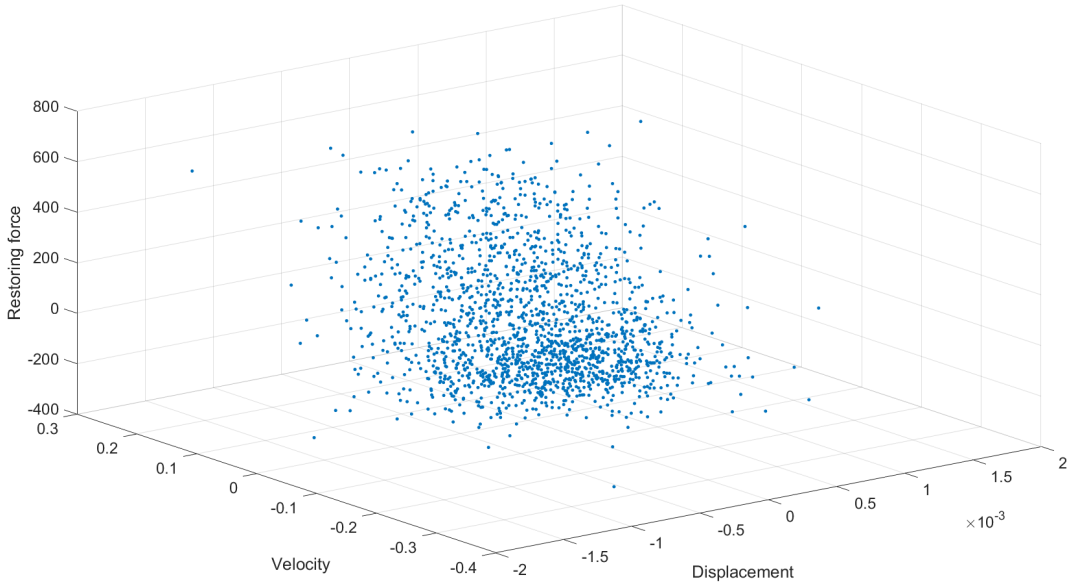


Figure 11: Original training data space of the shock absorber.

From Figure 12, along with the dimension of the displacement being projected out, the small variation in the vertical direction suggests that the restoring force does not vary much with the change of displacement. Therefore, the relation between the restoring force and the other two factors can be approximately treated as a one-sided dependence on the velocity. Applying the LK-CTGP to study the 'static' behaviour of the data was not anticipated to present a serious problem. Figure 13 depicts the regions after the process of partitioning. The figure shows that four piecewise-linear regions were generated from the LK-CTGP. By comparing Figure 13 with Figure 12, the mild curvature between [-0.05,0] in the velocity dimension is successfully modelled as a narrow linear plane indicated by the yellow middle region in Figure 13. The single partition in the displacement dimension also supports the aforementioned supposition of displacement independence.

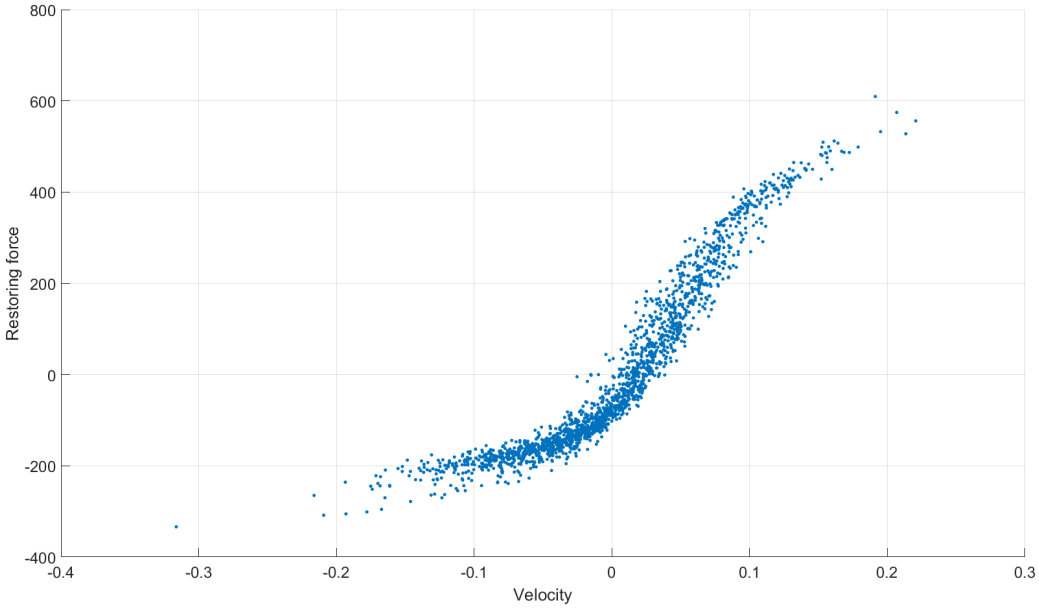


Figure 12: Side view of the data space with the displacement projected out.

Under such a static fitting scheme, the prediction on the test data can be plotted against the measured data, as in Figure 14. In general, the static model captured the behaviour of the restoring force rather well. The bottom part of the data is appropriately established by the model, while the top part occasionally has been more severely over or under estimated. The NMSE of the prediction is 1.3, which as it is from a static model, is rather good; however, it must be remembered that the test was conceived in order to simplify any dynamics.

An alternative approach to modelling, motivated by Giacomin's work [22], led to the use of a neural network approach to approximate the behaviour of the shock absorber [21]; the authors tried to model the force-velocity curve (e.g. Figure 12) using the same hyperbolic tangent transfer function used commonly in neural networks, the full dynamical model was specified by,

$$m\ddot{y} + c\dot{y} + ky + \alpha[tanh(\beta\dot{y} + \gamma) - tanh(\gamma)] = x(t) \tag{19}$$

where $m, c, k$ are the classic coefficients involved in a mass-spring-damper system; $\alpha, \beta, \gamma$ are used to compensate for the nonlinear characteristics associated with the nonlinear damping.

The model is nonlinear in the parameters and requires an iterative approach

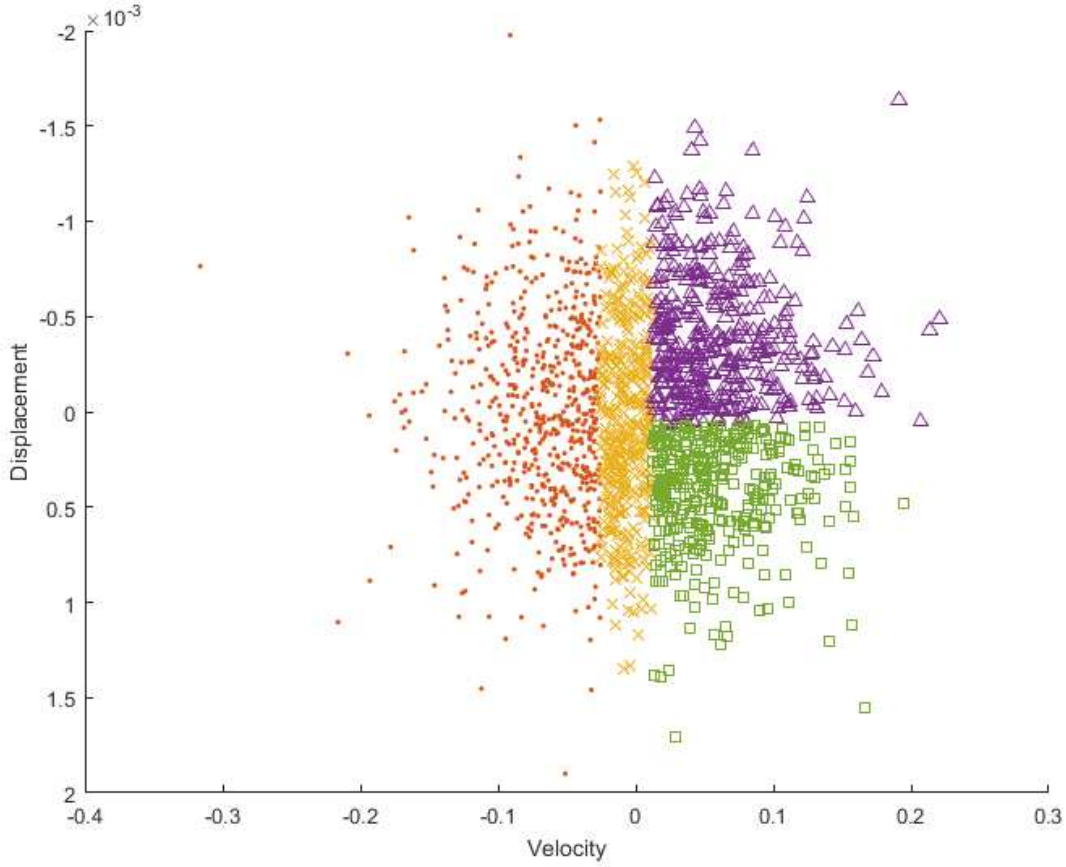


Figure 13: Layout of regions after partitioning.

to least-squares; the coefficients were obtained through a process of gradient descent (or backpropagation in neural network terminology), when given the data to learn. The study showed that a static ($m = 0$) approximating model generated a prediction with an NMSE=7, which is significantly larger than the error produced by the LK-CTGP. The paper also looked at polynomial models of the form,

$$m\ddot{y} + \sum_{i=1}^{N_p} c_i \dot{y}^i + ky = x(t) \tag{20}$$

The prediction accuracy proved very much dependent on the order of the polynomials assumed. With a linear model, the prediction NMSE was 15; when a $9^{th}$-order model was assumed, the error reduced to 0.9. However, the model did not generalise, and became unstable if prediction on an independent test set was attempted. The static study of the shock absorber using the LK-CTGP has generated a significantly improved prediction with an error very close to the best error achieved through analytical modelling at high polynomial order, with none of the stability problems as the LK-CTGP is linear with positive damping in the asymptotic regimes.

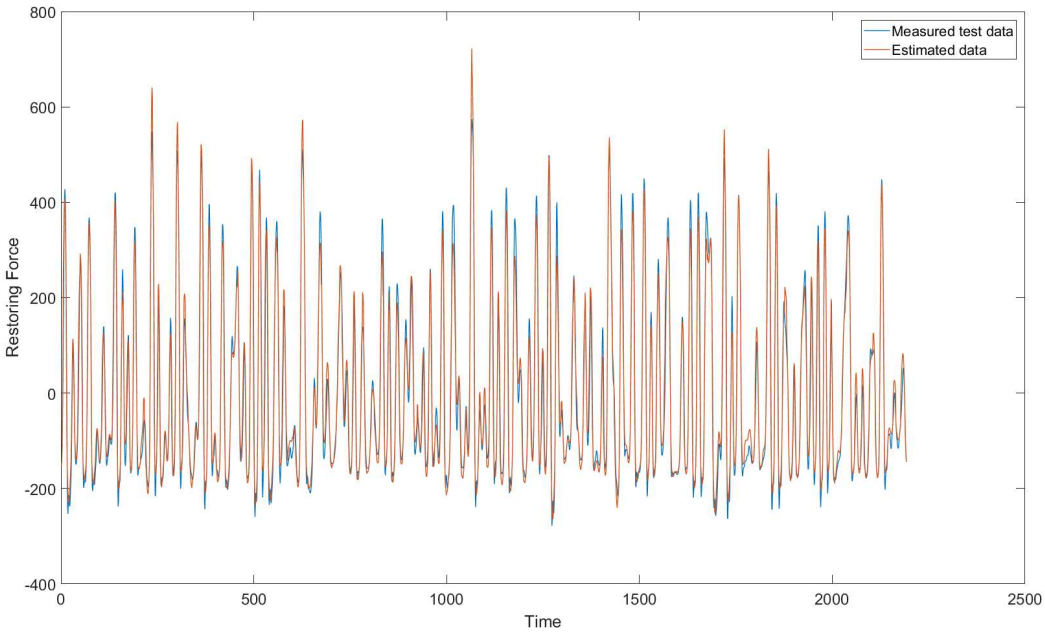


Figure 14: Prediction given by static modelling(NMSE=1.3).

However, the LK-CTGP formulated as an piecewise-ARX model is actually capable of dynamic modelling, so it offers an opportunity here to see if there are any explicable dynamic effects in the shock absorber data as measured. As before, the input is taken as the velocity and the output as force, and the LK-CTGP is applied using a single lag on the output. The advantage of starting with this minimal model is that one can physically visualise the variable space.

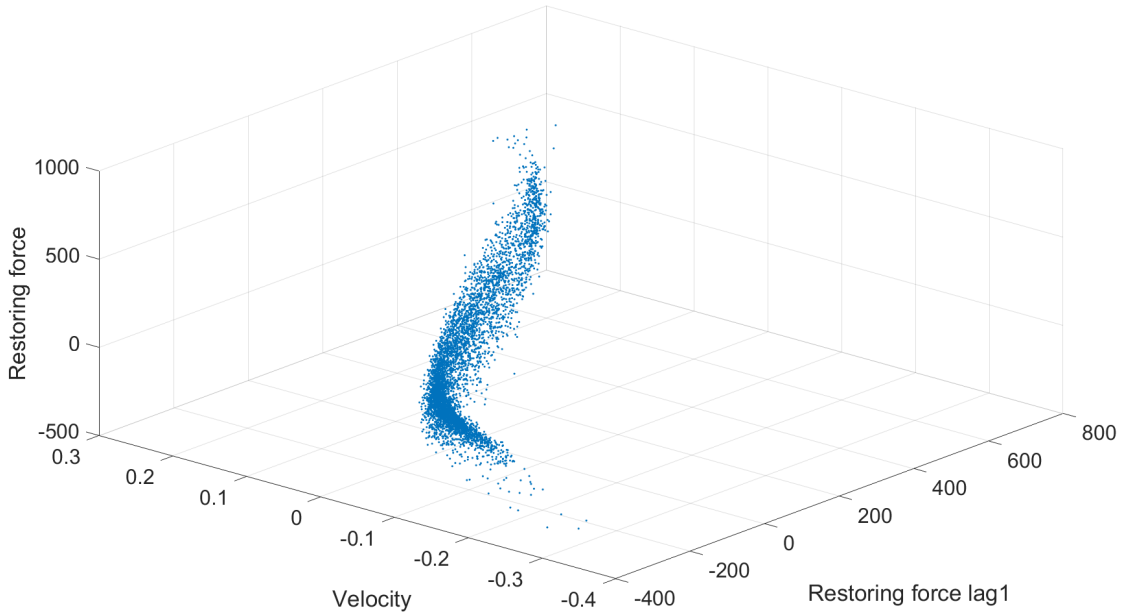


Figure 15: Training data space of shock absorber NARX system.

Figure 15 presents the training data within the variable space of the model with a single lag. Following the profile of the data surface, one can picture it as a contorted flat plane, in which the middle section stays largely flat with both ends folding towards opposite directions. It is rather difficult to visually imagine the appearance of the final fit, considering the contorted shape as well as the misalignment of the data surface with the axis. The LK-CTGP is capable of dissecting the complexity into piecewise linear simplicity. After allowing the algorithm to run for only 1000 MCMC rounds, the fit to the training data space is presented in Figure 16 (The units have been normalised).

Figure 16 demonstrates explicitly that the nonlinearity of the contorted surface can be broken down into seven segments of piecewise-linear planes. The bright red and dark blue regions at both ends do fold in opposite directions and are seemingly perpendicular to each other. The complete

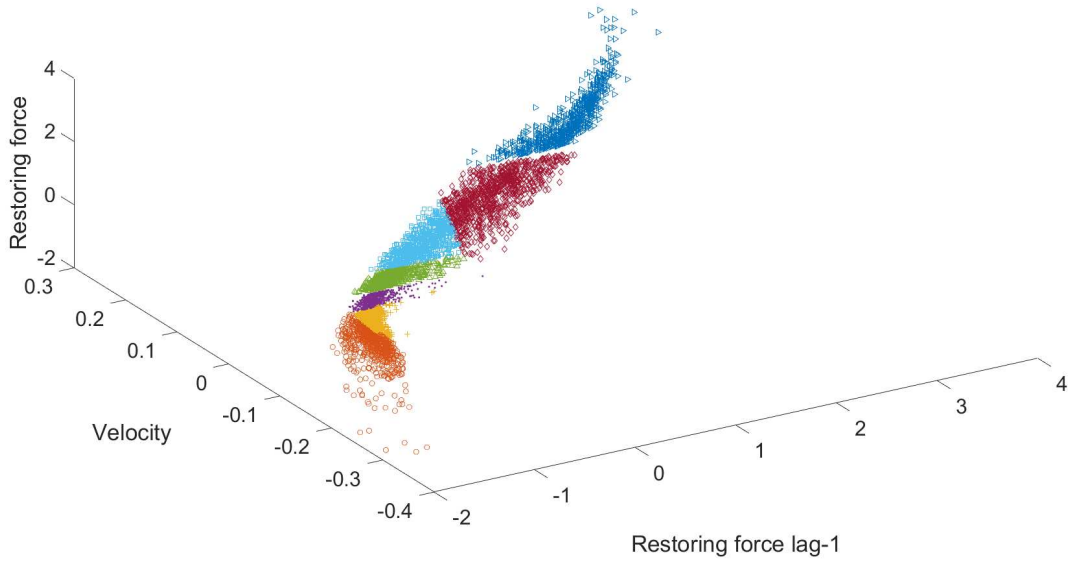


Figure 16: NARX data fitting.

gradation of the curvature from the middle dark red plane to the bright red end section is almost perfectly expressed by four transitional regions in between.

Figure 17 is a 2D perspective view of the partitioning plane, where all the horizontal and vertical partitions are neatly arranged for a global inspection.

The model fit here led to the MPO predictions on the 2000 test data shown in Figure 18. The NMSE error is approximately 23, which is a fairly large error compared to that for a static model; however, considering the error propagation in the NARX model, such a number could be considered as reasonable. The main issue is that the model is not capable of encompassing the frequency dependence of the real absorber. In the NARX model, the frequency dependence manifests as a substantial component of correlated noise which seriously interferes with the predictive capability of the model. The exercise has largely proved unsuccessful because any dynamic effects present in the data are clearly small compared to the effects of the 'correlated noise'. However, it is rather clear that the algorithm has successfully captured the general behaviour of the data by producing a set of predictions largely conforming to the original test data, despite having to some extent, underestimated the span of the amplitude.

Although the initial results were not promising, one more lag was added into

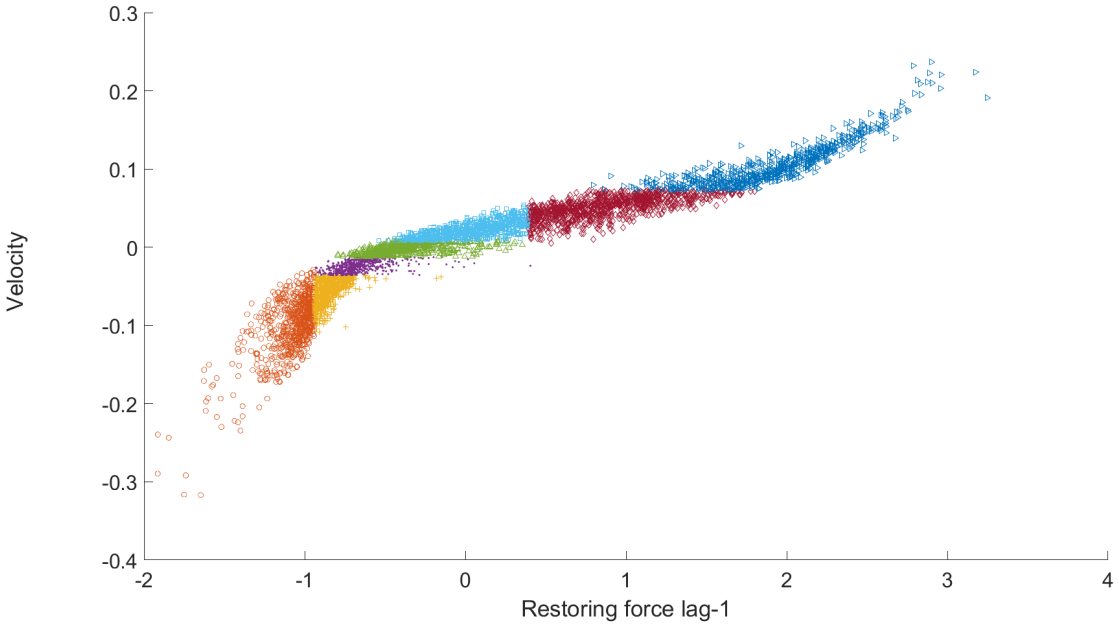


Figure 17: Layout of the partitions in the input variable plane.

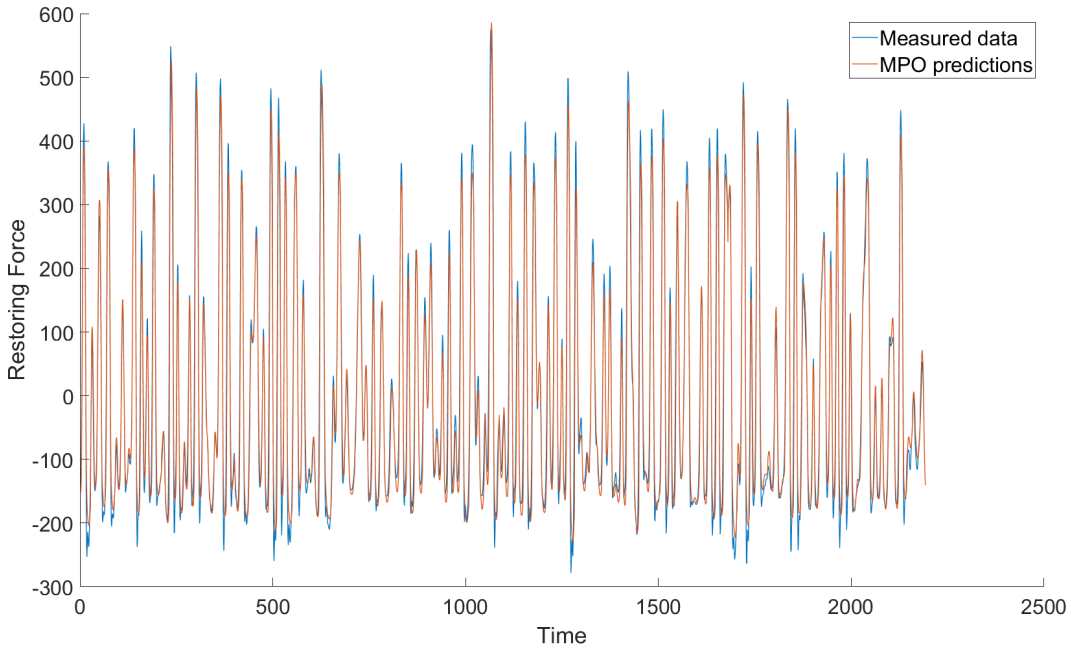


Figure 18: MPO predictions on the test set (NMSE=23).

the model; the four dimensional variable space that resulted is not accessible for an intuitive visualisation. However, the goodness of fit is reflected in the final prediction error. Figure 19 shows the MPO predictions given by the LK-CTGP on the NARX test data with two lags. When compared to the the single lag case, the additional lag seems have actually reduced the geometrical complexity of the partitioning in the training dataspace. With the same number of partitions as the single lag case, the NMSE has been reduced from 23 to 15. By comparing Figure 19 to Figure 18, it is fairly interesting to see that in the two-lag case, as quite contrary to the single lag case, the MPO predictions actually tend to overestimate the variation of the data over the time. One can also observe that, the modelling of the lower bounds of the original test data has been significantly improved by taking in this extra lag.

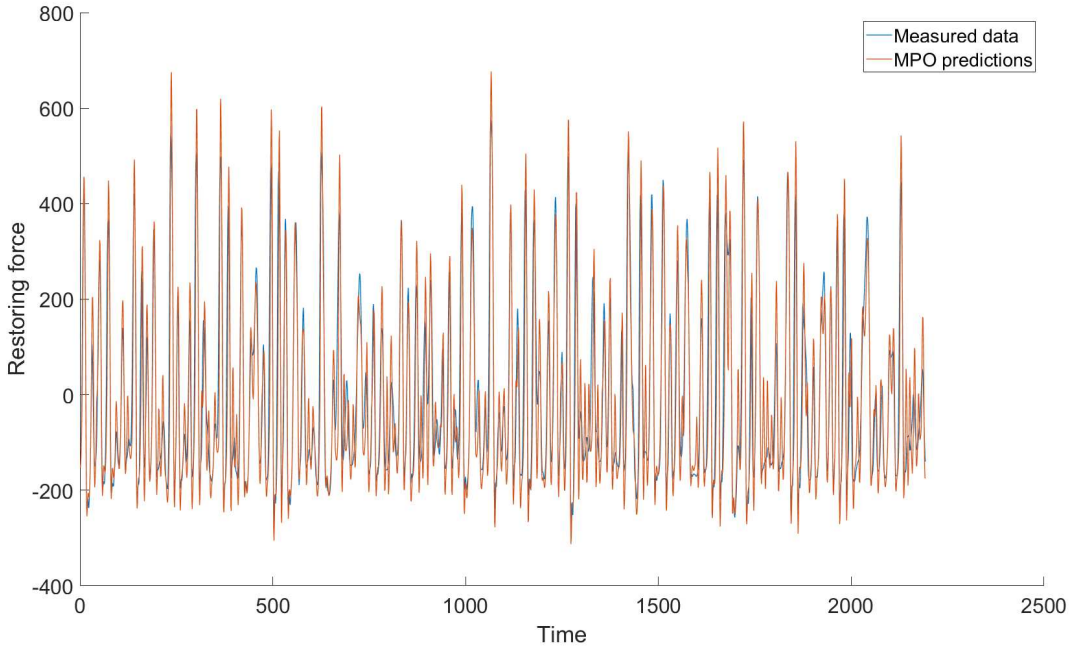


Figure 19: MPO predictions on the test set with 2 lags (NMSE=15).

Due to the limits imposed by the curse of dimensionality, further introducing lags into the system drastically exasperates the prediction. The NMSE generated are egregiously high, which can be ascribed to a complete failure of capturing the trend from the beginning of the NARX process.

## 6. Conclusions

This paper presented a preliminary study of a piecewise-linear NARX system using the LK-CTGP model. The LK-CTGP model is easy to specify its

parameters and counters nonlinearity by producing partitions of the variable space and assigning locally-linear models to each partition. Through a number of case studies, the effectiveness of the LK-CTGP in modelling NARX systems has been demonstrated. By examining the results both visually and mathematically, in all the cases, the LK-CTGP generated reasonable partitions to assure local regional linearity was achieved. The case studies on the synthetic piecewise-linear system indicated that the algorithm does partition into physically appropriate regions; however, it is noticed that the algorithm tends to over-split at partition boundaries when more linear planes are added, which shows that the algorithm is conservative at modelling sharp changes. One possible explanation for this is rather simple; the algorithm switches leaves on the basis of the prediction variable; however, it does not take account of the fact that previous values of lagged variables may be found in different partitions. Intuitively, one can see that the algorithm could compensate by grouping partition boundaries in order to smooth a transition. Apart from such a drawback, the overall fitting is considered as appropriate according to the predictive errors produced. The later case study on the Duffing oscillator shows that the MPO predictions are extremely sensitive to the presence of nonlinearity. The global linear least-square method failed at predicting the time evolution of the data, despite the fact that the OSA predictions were very good indeed. The LK-CTGP model successfully solved the problem through partitioning the data space. All the partitions were symmetrically placed in the dimension where the nonlinearity was present. The final, experimental, case study of the automotive shock absorber has shown that, the predictions produced by the LK-CTGP on the static data system excelled the predictions given by most of the previous analytical models. In the later NARX system study, the algorithm successfully allocated sensible partitions to segregate the intricately contorted surface into linear planes. However, due to the inevitable error propagation in NARX model as the result of the (effective) correlated noise, the MPO predictive error could not compete with the predictions based on the static system. This problem could potentially be circumnavigated by generalising the linear models on the leaves to ARMAX models and allowing locally-linear noise models. This will be considered for further work.

The problem with the current LK-CTGP model is that the model requires an initial estimate of the noise parameter rather than allowing it to be passively learnt and updated throughout the learning process. In this paper, this noise parameter is briefly determined by running a preliminary GP on the entire dataset, where the overall noise parameter can be optimised. However, the global noise parameter does not necessarily agree with the local noise levels after the variable space has been partitioned, especially when the data is characterised by heteroscedasticity. In all the synthetic case studies presented in the paper, all the datasets have stationary variances, therefore, the global

variance given by the nonparametric GP model should be a reasonable approximation to the local variances after partitioning. However, without the accurately-learnt variance, the confidence interval cannot be displayed for analysis in this paper. Technically, the LK-CTGP can still optimise the local noise parameter to obtain the correct variance through the same optimisation process as for an SE kernel GP. However, the computational cost would then be similar to a full Gaussian TGP. The major advantage of using the LK-CTGP is its efficiency as supported by the existence of analytical solutions to its linear structural parameters. Since no confidence interval is computed, to some extent, the linear kernel GP segment in the model can be substituted by even simple least-squares methods, in which case the computational cost is even lower.

More fundamentally, the piecewise-linear model might not be the best piecewise model for the NARX system, because too many regions are required to ensure a good linearity in each region. The general CTGP undoubtedly has more potential in studying the NARX model. Those distance-based kernels can offer smoother curve fittings for the TGP model, which seems to be promising; however, the computational cost will drastically increase.

## Acknowledgements

The author would like to acknowledge the provision of an R code of TGP by Robert Gramacy. A special acknowledgement is given to EPSRC for financially supporting the author TZ to produce this paper.

## References

[1] L. Ljung, System Identification: Theory for the User, Prentice Hall, Englewood Cliffs, 1987.

[2] T. Söderström, P. Stoica, System identification, Prentice-Hall, 1989.

[3] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Delyon, L. Ljung, J. Sjöberg, Q. Zhang, Nonlinear black-box models in system identification: Mathematical foundations, Automatica 31 (1995) 1725–1750.

[4] J. Sjöberg, Q. Zhang, A. Benveniste, B. Deylon, P. Glorennee, H. Hjalmarsson, A. Juditsky, L. Ljung, Nonlinear black-box modelling in system identification: model structures and algorithms, Automatica, 1995.

[5] J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, J. Sohl-Dickstein, Deep neural networks as gaussian processes, arXiv preprint arXiv:1711.00165.

[6] C. E. Rasmussen, Z. Ghahramani, Infinite mixtures of gaussian process experts, in: Advances in neural information processing systems, 2002, pp. 881–888.

[7] R. Gramacy, H. Lee, Bayesian treed Gaussian process models with an application to computer modeling, Journal of the American Statistical Association 103 (2008) 1119–1130.

[8] H. Tong, Threshold models in time series analysis—30 years on, Statistics and its Interface 4 (2) (2011) 107–118.

[9] M. L. Yaghin, A. Mojtahedi, M. Ettefagh, M. Aminfar, Experimental investigation of tarmax model for modeling of hydrodynamic forces on cylinder-like structures, Journal of Marine Science and Application 10 (3) (2011) 281.

[10] G. Fouskitakis, S. Fassois, Functional series tarma modelling and simulation of earthquake ground motion, Earthquake engineering & structural dynamics 31 (2) (2002) 399–420.

[11] E. H. Fırat, SETAR (self-exciting threshold autoregressive) non-linear currency modelling in EUR/USD, EUR/TRY and USD/TRY parities, 2017.

[12] S. Billings, Nonlinear System Identification: NARMAX, Methods in the Time, Frequency, and Spatio-Temporal Domains, Wiley-Blackwell, 2013.

[13] C. Rasmussen, C. Williams, Gaussian Processes for Machine Learning, The MIT Press, 2006.

[14] K. Worden, W. Becker, T. Rogers, E. Cross, On the confidence bounds of Gaussian process NARX models and their higher-order frequency response functions, 2018.

[15] T. Zhang, On treed Gaussian processes for modelling structural dynamic systems, Ph.D. thesis, University of Sheffield (2018).

[16] P. Chaudhuri, W.-D. Lo, W.-Y. Loh, C.-C. Yang, Generalized regression trees, Statistica Sinica (1995) 641–666.

[17] H. Chipman, E. George, R. McCulloch, Bayesian CART model search, Journal of the American Statistical Association (443) (1998) 935–948.

[18] R. H. Rand, Lecture notes on nonlinear vibrations, 2012.

[19] H. Lang, A study of the characteristics of automotive hydraulic dampers at high stroking frequencies, 1978.

[20] P. Hagedorn, J. Wallaschek, On equivalent harmonic and stochastic linearization for nonlinear shock-absorbers, in: Nonlinear Stochastic Dynamic Engineering Systems, Springer, 1988, pp. 23–32.

[21] K. Worden, G. Tomlinson, Nonlinearity in Structural Dynamics: Detection, Modelling and Identification, Institute of Physics Press, 2001.

[22] J. Giacomin, Neural network simulation of an automotive shock absorber, Engineering Applications of Artificial Intelligence 4 (1991) 59–64.