Tutorial example of Gaussian Process prior Modelling Applied to Twin-tank System

G.J. Gray[1], R. Murray-Smith[1,3], K. Thompson[2], D.J. Murray-Smith[2]
[1]Dept. Computing Science
[2]Dept. Electronics and Electrical Engineering
University of Glasgow
Glasgow G12 8QQ
Scotland, UK

[3]Hamilton Institute
National Univ. of Ireland, Maynooth
Co. Kildare, Ireland

Abstract

Gaussian process priors are used to model a twin-tank system as a tutorial example in the use of Gaussian process priors in the context of dynamic systems modelling. The model learned provides a good description of the real system, and the advantages of input-dependent uncertainty and propagation of uncertainty are shown. Some observations of practical issues when using Gaussian process models with dynamic system data are described.

1 Introduction

In this tutorial paper, Gaussian Process modelling is applied to laboratory scale experimental apparatus – a twin-tank system. A simulation study shows that the Gaussian Process (GP) with training data consistinh of the system response to small inputs can provide robust interpolating between these local sources of information. The Gaussian process approach to modelling is also applied to a real system (a teaching laboratory twin-tank system) using small input experimental data as the training data. The robustness of the model is validated using recorded system response to a large excursion input. The effect of derivative data is also investigated.

2 Gaussian Process Modelling

In a Bayesian framework the model is based on a prior distribution over the infinite-dimensional space of functions. As in (O'Hagan 1978), such priors can be defined as Gaussian processes. Figure 1 gives an illustration of drawing realisations from Gaussian process priors.

These models have attracted a great deal of interest recently – see for example reviews such as (Williams 1998, MacKay 1999). (Rasmussen 1996) showed empirically that Gaussian processes were extremely competitive with leading nonlinear identification methods on a range of benchmark examples. The further advantage that they provide analytic predictions of model uncertainty makes them very interesting for control applications.

Use of GPs in a control systems context is discussed in (Murray-Smith *et al.* 1999, Leith *et al.* 2000). Integration of prior information in the form of state or control linearisations is presented in (Solak *et al.* 2003). A simulation of Model Predictive Control with GPs is presented in (Kocijan *et al.* 2003).

2.1 The covariance function

The choice of covariance function is very important (MacKay 1999). We are interested in modelling a continuous dynamic system, and we assume a constant mean, that the process is stationary, and that the covariance function depends only on the distance between the inputs x. The covariance function $C(x^p, x^q)$ expresses the covariance between y^p and y^q. A common choice for the function $C(x^p, x^q)$ is

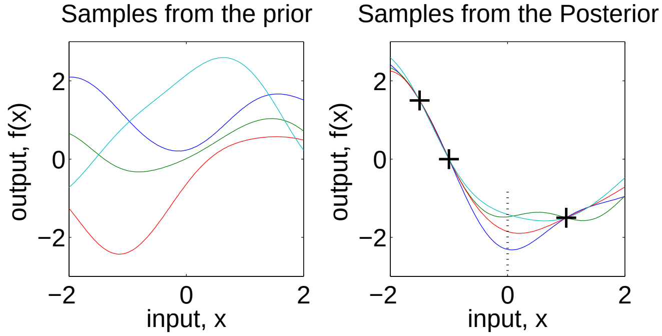

Figure 1: Sampling from a Gaussian Process. Sample realisations are drawn from the Gaussian process prior (left hand figure), and then sample functions are drawn from prior conditioned on three training points (right hand figure).

$$C\left(x^p, x^q\right) = v_1 \exp\left[-\frac{1}{2}\sum_{d=1}^{D}\frac{\left(x_d^p - x_d^q\right)^2}{w_d^2}\right] + v_0\delta_{ij}, \tag{1}$$

where D is the input dimension and x represents the data. v_0, v_1, and w_d are hyperparameters of the covariance function. These are identified from the training data using a maximum likelihood optimisation.

2.2 Prediction with Gaussian Processes

Gaussian Process modelling predicts the behaviour of a dynamic system by predicting the distribution of the next data point based on the system input and the predicted distribution of the current point. The predictive distribution is obtained by conditioning on the training data to obtain $p\left(f(x^*) \mid x^*, \mathcal{D}\right)$.

Since the variables are Gaussian, the predicted values are also Gaussian and have a mean and variance,

$$\mu(x^*) = \mathbf{k}(x^*)^T K^{-1}\mathbf{y} \tag{2}$$

$$\sigma^2(x^*) = k(x^*) - \mathbf{k}(x^*)^T K^{-1}\mathbf{k}(x^*) \tag{3}$$

where $k(x^*) = \left[C(x^1, x^*), ..., C(x^N, x^*)\right]^T$ is the $N \times 1$ vector of covariances between the new point and the training targets and $k(x^*) = C(x^*, x^*)$.

2.3 Gaussian Process configuration

As with other nonlinear modelling techniques, there are some key design decisions to make in setting up the GP. The training data is stored and used for prediction of each test point when performing inference with the Gaussian process model. This means that reducing the amount of training data reduces the computation required and therefore the computer execution time of the prediction. It is also necessary to choose the dynamic inputs that will be used for the GP prediction. This includes previously predicted values of the output as well as the input. For example, for a system with first-order like behaviour and a single input, the GP might use two variables in the training data set

$$y_k = f(y_{k-1}, u). \tag{4}$$

The sampling interval is also of course very important. As well as increasing the quantity of data and therefore computation time, too small an interval can also lead to numerical difficulties when inverting the covariance matrix. This problem is related to the choice of covariance function and its suitability for NARMAX type models which used delayed values of outputs and inputs. The functions currently used do not take into account the extra covariance among the model responses associated with inputs composed of previous outputs.

2.4 The iterative k-step ahead prediction of time series

The Gaussian Process as described above could be used to predict the mean and variance one step ahead. However, after the first step of a simulation involving auto-regressive components in the input the inputs for subsequent predictions will be uncertain, as they are based on previous outputs. We will know the mean and variance of the inputs from the distribution of the previous states. We should now therefore assume that the test input x^* has a Gaussian distribution, $x^* \sim \mathcal{N}(m_x, \Sigma_x)$. In (Girard *et al.* 2003, Quinonero-Candela *et al.* 2003), we suggest an analytical Gaussian approximation to solve this integral, computing only the mean and variance of $p(y^*|m_x, \Sigma_x)$. This allows us to propagate the uncertainty over the k-step ahead prediction horizon.

3 Simulation Study: The Twin Tank System

The Gaussian Process modelling technique was applied to the modelling of a coupled water tank system(Murray-Smith 1995). The system consists of two coupled water tanks with a connecting pipe between them, an input to the first tank, and an output pipe from the second tank. The system is nonlinear since even at the simplest approximation, the flow through the orifice varies as the square root of the water depth. The system is shown in Figure 2.

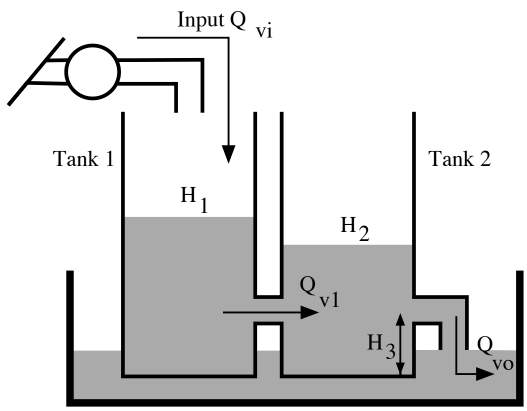

Figure 2: Twin tank system

3.1 Experimental Design

For nonlinear modelling, it is desirable to be able to predict the response to large inputs across most or all of the operating range of the dynamic system. Safety or operational reasons often make it difficult to collect data for larger inputs. It is useful therefore to be able to use small perturbation data collected at various operating points to generate a nonlinear model valid close to the equilibrium manifold.

This situation was created in the laboratory and in simulation using a coupled water tank system. The system was put into equilibrium at various operating points throughout its operating range and small step inputs were applied to the system. The GP model predicted from these data was used to predict the system response to much larger inputs. This experiment was done both with simulated and experimental data. The simulated data was generated using a simple nonlinear model.

The training data should include some points in steady state so that the GP can learn about this condition, and the sampling interval should be chosen so that system dynamics are accurately represented.

The experiments were conducted with sample intervals of 100 seconds and 200 seconds. The data was actually sampled at 10 second intervals - so the effect of filtering the data while sampling was also investigated.

The GP process was used to identify H_2. The inputs used for the training data were H_1 and H_2^{k-1} (H_2 delayed by one sampling interval). The effect of using a third input H_2^{k-2} (H_2 delayed by two sampling intervals) was also investigated.

3.2 Simulation study

A nonlinear model of the twin tank system (Murray-Smith 1995) was used to generate small perturbation training data as well as large perturbation validation data in order to evaluate the GP modelling method.

The fluid flow in and out of the tanks can be described by,

$$A_1 \frac{dH_1}{dt} = Q_{vi} - Q_{v1} \tag{5}$$

$$A_2 \frac{dH_1}{dt} = Q_{v1} - Q_{vo} \tag{6}$$

where Q_{vi} is the flow into tank one, Q_{v1} is the flow from tank one to tank two, and Q_{vo} is the output flow from tank two. A_1 and A_2 are the cross-sectional areas of tanks one and two respectively. H_1 is the depth of the water in tank one and H_2 is the depth in tank two.

It is known that for this system, the flow between the tanks and particularly in the output pipe is nonlinear and the main modelling error. If the connecting pipe and the output pipe are assumed to be orifices, Bernoulli's theorem applies and the flow can be calculated thus,

$$Q_{v1} = c_{d1} a_1 \sqrt{2g\,(H_1 - H_2)} \tag{7}$$

$$Q_{vo} = c_{d2} a_2 \sqrt{2g\,(H_2 - H_3)} \tag{8}$$

where c_{d1} and c_{d2} are empirical discharge coefficients for the connecting pipe and the output pipe respectively, a_1 is the area of the orifice between tanks one and two and a_2 is the area of the orifice out of tank two. While this approximation may be valid for the pipe connecting tanks one and two (it is 5.5mm long), it is known to be inaccurate for the outlet pipe from tank two (10.6mm long).

This mathematical model (equations (5), (6), (7), (8)) was used to generate some simulated data for the system. The experiment consisted of five simulations starting from different initial conditions. Each simulation consisted of a train of pulses of alternating sign and different amplitudes. Gaussian noise was added to the data generated to simulate measurement noise. The mean noise value was 1% of the mean signal value. The simulation data used for training is shown in Figure 3. The three variables plotted are the inputs H_1 and H_2^{k-1} (H_2 delayed by one sampling interval) and the output H_2.

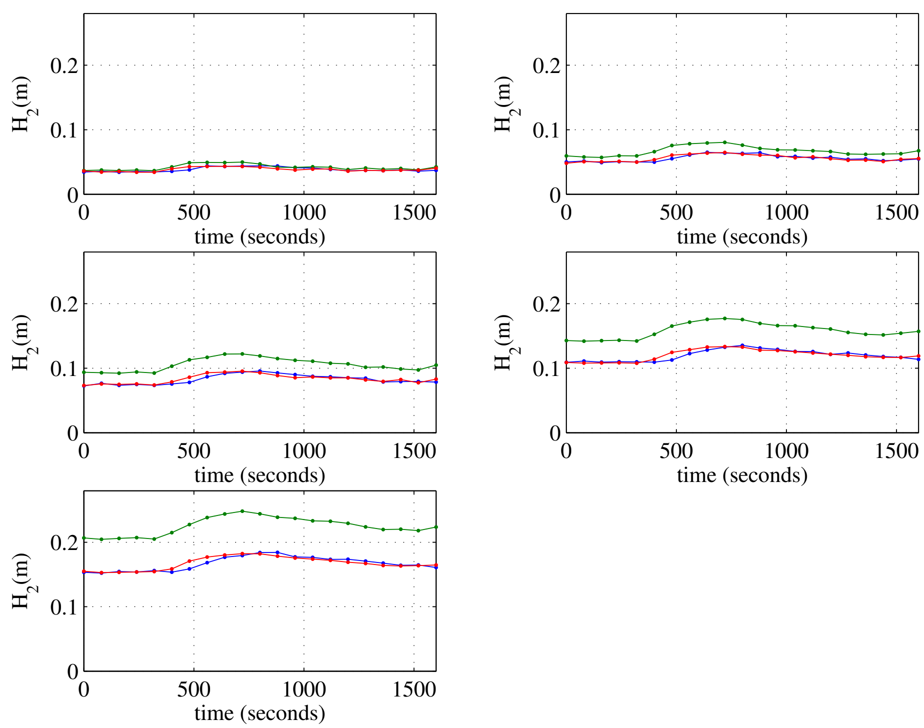

Figure 3: Simulated GP training data

A second set of validation data was simulated. This consisted of a sequence of larger pulses exciting the system over its entire dynamic range. The time response is shown in Figure 4.

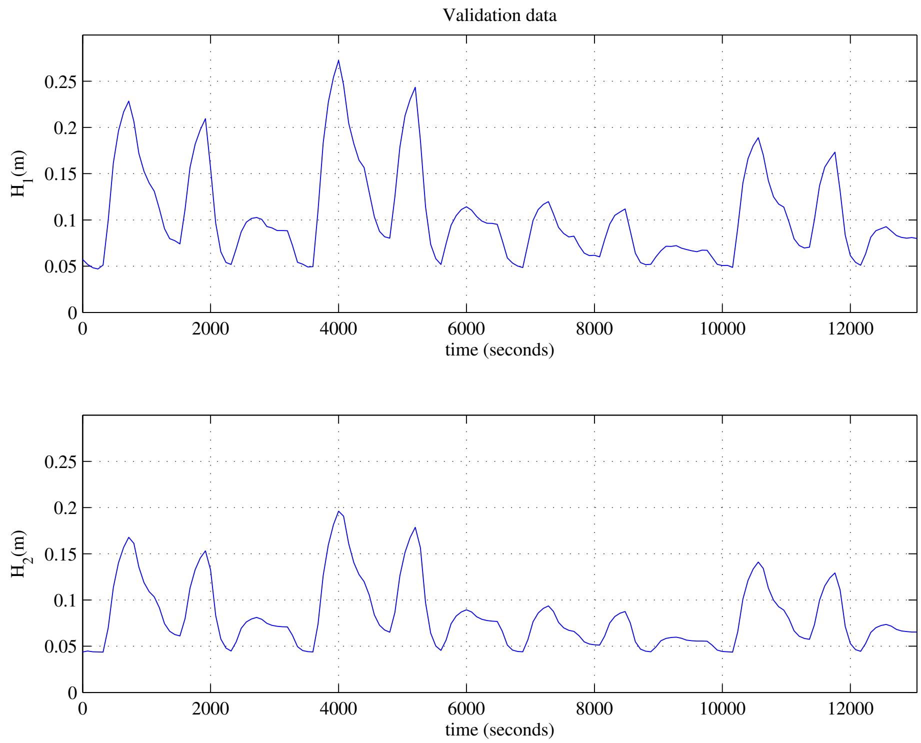

Figure 4: Simulated validation data

3.3 Training a GP on the simulation data

This simulated data was used to train a GP and the GP model was then validated by simulating one large input and comparing the GP predicted output with the simulated output data. The results are shown in Figure 5, which is a simulation (not a one-step-ahead) using the GP model.

The error bounds in Figure 5 show the predicted standard deviation of the GP model output predictions ($\pm 2\sigma$), based on the k-step-ahead propagation of uncertainty used in (Girard *et al.* 2003). This prediction of uncertainty is a very useful output from the GP simulation. In this case it shows that confidence in the predicted initial response to a large input is less than the the confidence in the predicted steady-state value. This is not unexpected as the training data contained much smaller test inputs than those applied in the validation.

4 Identification from Experimental Data

As the results on the simulated twin-tank model were encouraging we then applied the Gaussian Process model to data obtained from a laboratory coupled water tank system, as illustrated in Figure 2.

4.1 Training Data

As a nonparametric model the training data remains part of the final Gaussian Process model. Prediction with the GP requires inversion of a matrix of this training data at least once, so computation time increases with quantity of training data ($O(N^3)$ for N training points). If the sampling interval of the training data is too small relative to the dynamics of the system or if there is repetition in the training data (for example, similar responses to similar inputs at similar system states – a common feature in level control data), the rank of the matrix Q can be reduced and this leads to numerical problems when inverting Q. Ideally, the training data should be generated by exciting the dynamic system in different ways and recording data at a sampling interval slightly below that required to reveal the system dynamics (half of the fastest time constant of the system).

As with other empirical modelling techniques, it is necessary to define which inputs should be used. Inputs in this context also include delayed outputs. So for example, when modelling a first order dynamic system (Equation 9), it would be necessary to have two inputs - the system input u and the system state x delayed one time interval.

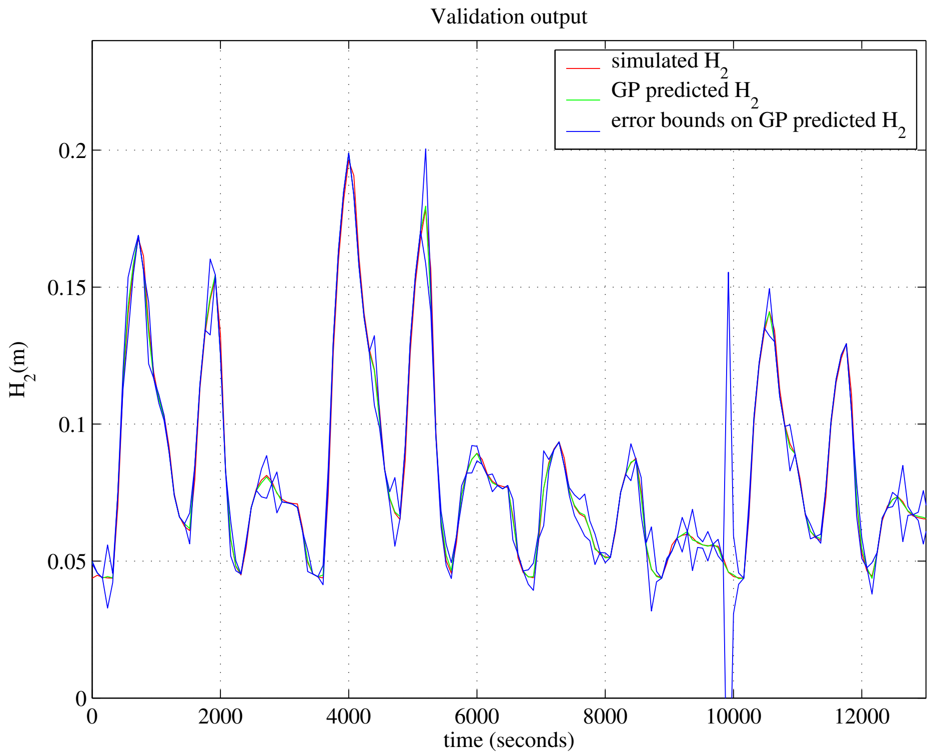

Figure 5: Time response showing simulated data and GP model output

$$\dot{x} = ax + bu \tag{9}$$

For the twin tank system, the training data was derived from several small input step response experiments at different operating points (Figure 6) as with the simulation study described above. The steps were increasing and decreasing. The system was allowed to settle to a steady state before applying the test input in each case. The water depth of each tank (H_1 and H_2) was recorded. The data was then combined to form the GP training data set.

4.2 GP model of the experimental data

The parameters of the covariance function are optimised using standard optimisation algorithms (in this case a conjugate gradient approach), to maximise the likelihood of the model. We can now use the model to predict outputs, and see where the model is most uncertain. The GP predicts the mean and the variance at any point in its input space, conditioned on the training data. A plot of the predicted variance can be useful to visualise the training data set. Figure 7 shows the predicted variance for this training data set. The training data points are also plotted on this graph. As would be expected, predicted variance is less where there is a higher concentration of training data. The plot shows that GP predictions will be more accurate when H_1 is close to 0.15 and H_2 is close to 0.1. This graph should be read with Figure 6.

4.2.1 Running a simulation

The GP model output is predicted using Equations (2) and (3). These equations predicted the expected value and the variance of the next data point given the current system state. This prediction then becomes the basis for predicting the next data point and so on. Unless the noise level on the training data is high, it is often necessary to add a small (e.g. 1^{-5} 'jitter' term to the diagonal entries of matrix Q to prevent numerical problems with the matrix inversion. This jitter term affects the absolute value of the predicted variance, but the state-dependent variance levels indicate areas of the system space where the model prediction is more dependable.

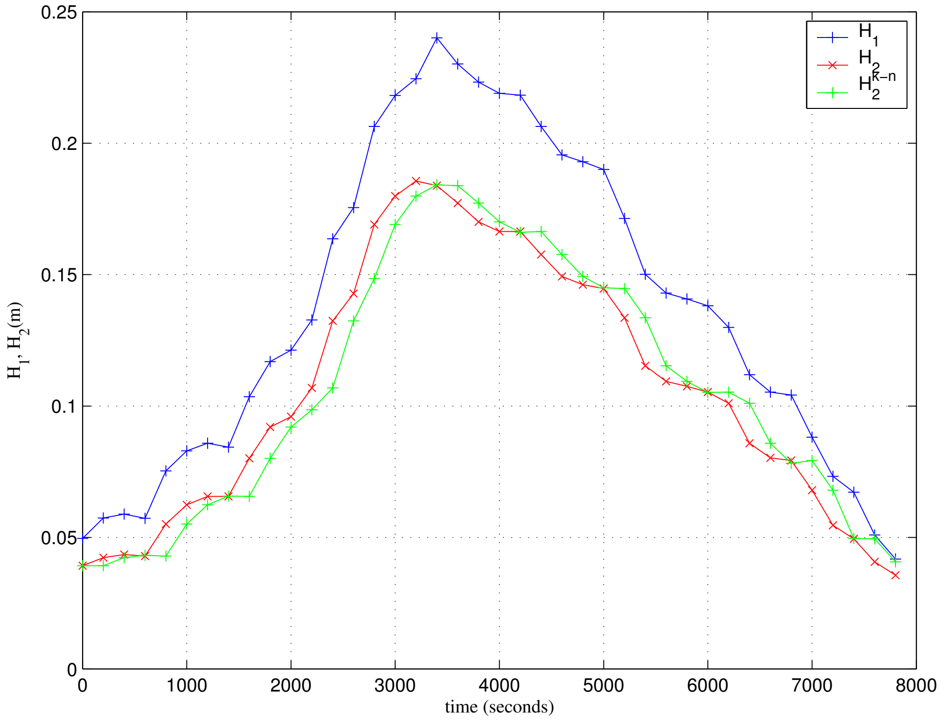

Figure 6: Experimental GP training data

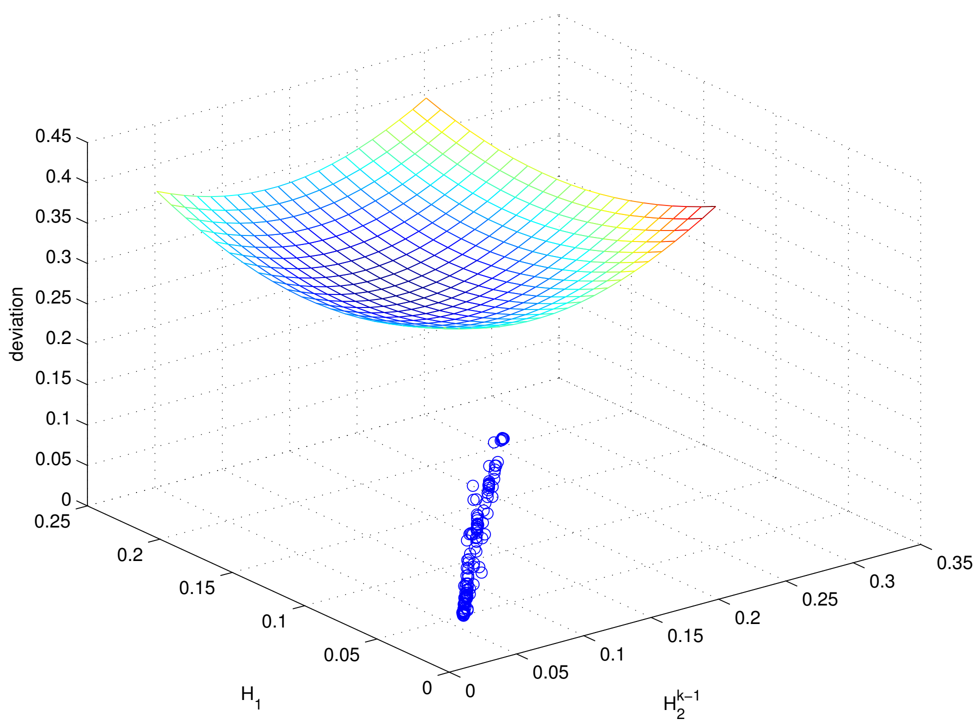

Figure 7: Variance of prediction, with experimental GP training data

4.2.2 Validation data

An experiment with larger inputs was performed using the same apparatus. A sequence of positive and negative large step inputs was applied. The GP was used to predict this response using the training data collected from the small perturbation experiments.

This represents an important test of the GP model. The training data consisted only of responses to small inputs. This test data consists of much larger inputs. This system is nonlinear and any linear representation would not be able to accurately predict system response to large inputs given only a model based on small inputs. This however is a common situation in dynamic system modelling. Often dynamic models are expected to accurately predict system behaviour in situations for which no training data exists.

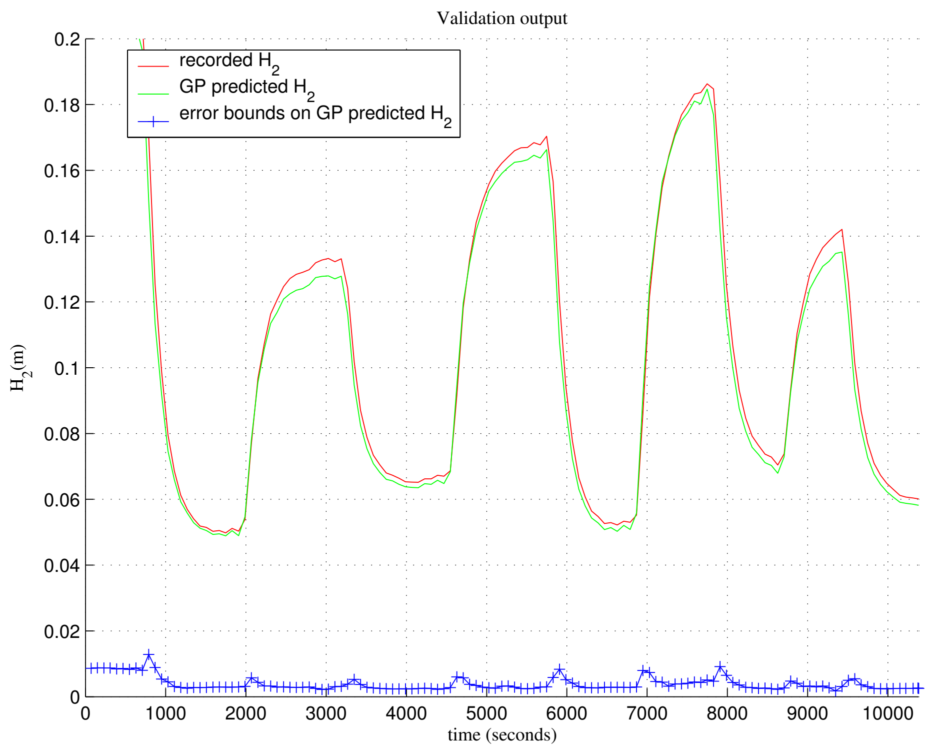

Figure 8: Twin tank system pulsed input response - recorded data

This data set consisted of three consecutive pulses and included both positive and negative steps. The result of the GP prediction is included in Figure 8. Figure 9 shows the same validation test but this time the plot indicates the error between predicted response and actual response superimposed on the GP predicted standard deviation. The GP predicted standard deviation correlates well with the error in the validation.

It can be seen from Figure 8 that the GP predicts system response accurately despite the fact that the training data was composed of much smaller steps than the validation data. This is a useful result since for many systems, experiments involving large scale inputs are often difficult or impossible to do due to operational or safety reasons.

In this case, k-step ahead prediction was used to model the system. That means that the GP was given only the initial conditions of the system and the prediction of each subsequent point was based on the predicted mean and variance of the previous point.

It was determined that although a larger number of points allows the GP to predict with lower variance, it is important not to over-sample the system. That is, prior knowledge of the system dynamics should be used to determine the system bandwidth and the training data should be configured accordingly.

4.3 Using derivative information

It is possible to use derivative information in Gaussian Process modelling. This has the advantage of reducing the number of points required in the training data set and therefore reducing the computational complexity of the model without reducing the fidelity of the model.

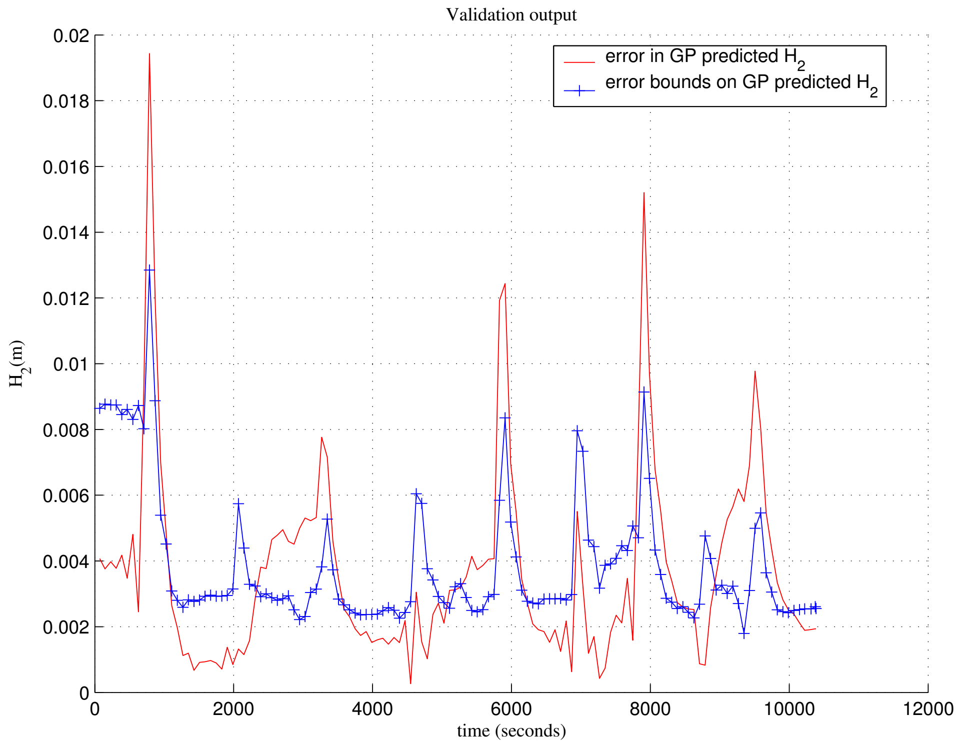

Figure 9: Twin tank system pulsed input response - GP predicted response and standard deviation

This was applied to the same system and same data used in the above example. This system has two inputs and one output, so it was necessary to calculate partial derivatives of the output with respect to each input at several points in the simulation. The variance of the estimated derivatives was also calculated, as this information can be used in the GP to improve the accuracy of both the predicted mean and variance.

4.3.1 Calculation of the partial derivatives

The same data set was used (Figure 6). this consists of small increasing and decreasing steps. The partial derivatives ($\frac{\partial H_2}{\partial H_1}$ and $\frac{\partial H_2}{\partial H_2^{k-1}}$) were calculated for each small step by representing each step response as a first order system and using a least squares parameter estimation algorithm to estimate the derivatives as shown. The first order model of system is $H_2 = mH_1$, the least squares estimation equations are

$$m = \frac{\sum_{i=1}^{n} Y_i X_i}{\sum_{j=1}^{n} X_j^2}. \tag{10}$$

To calculate the variance of m, σ^2,

$$\sigma^2 = \frac{\sum \left(Y_i - \hat{Y}_i\right)^2}{n-1}, n > 1. \tag{11}$$

All five steps of the training data were each represented by their derivatives. This reduced the number of training points in the GP from 17 to 5. The output is shown in Figure 10.

This compares favourably with the results for the 17 point dataset (Figure 8). For comparison, the GP was run with the four points representing the second rising step and no derivative information (Figure 11).

Figure 12 shows the error between the predicted response and the actual response for the validation data for both optimisations (with the derivative data and with the reduced subset). It is clear that the prediction with the derivative information is more accurate.

It can be seen that the derivative information in this case gives a more efficient GP model without reducing accuracy. In particular, it can be seen from Figure 12 that the prediction of the trasient behaviour is significantly better with the derivative information. This is because this operating region (values of H_1 and H_2 is not represented be the four data-points of the second rising step, whereas the derivative data included rising and follow steps.

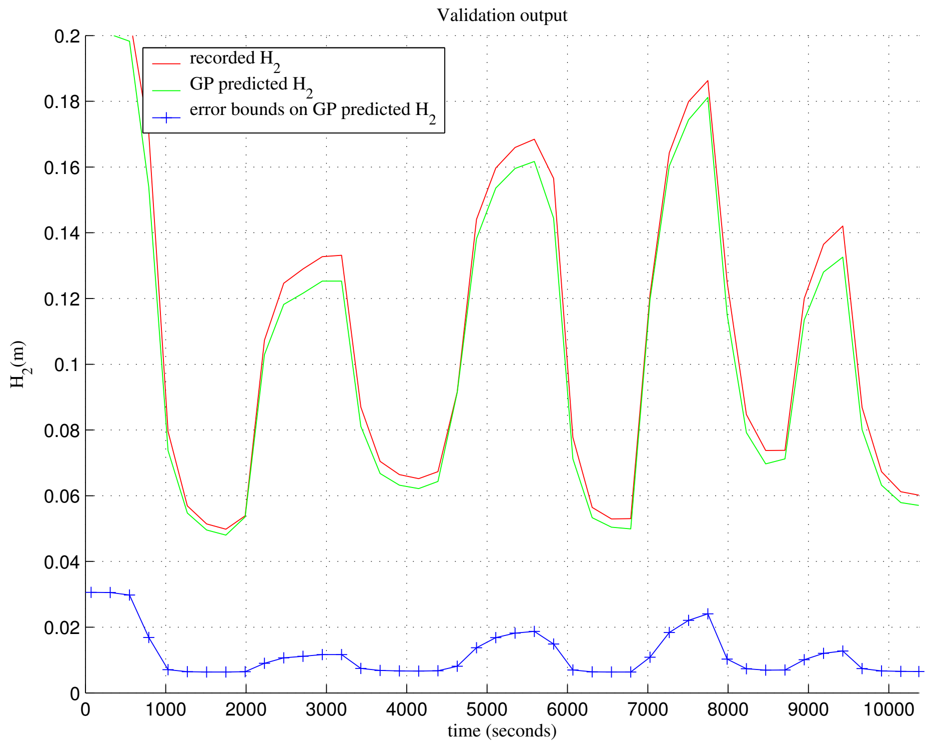

Figure 10: Twin tank system pulsed input response - GP predicted response and standard deviation using derivative information

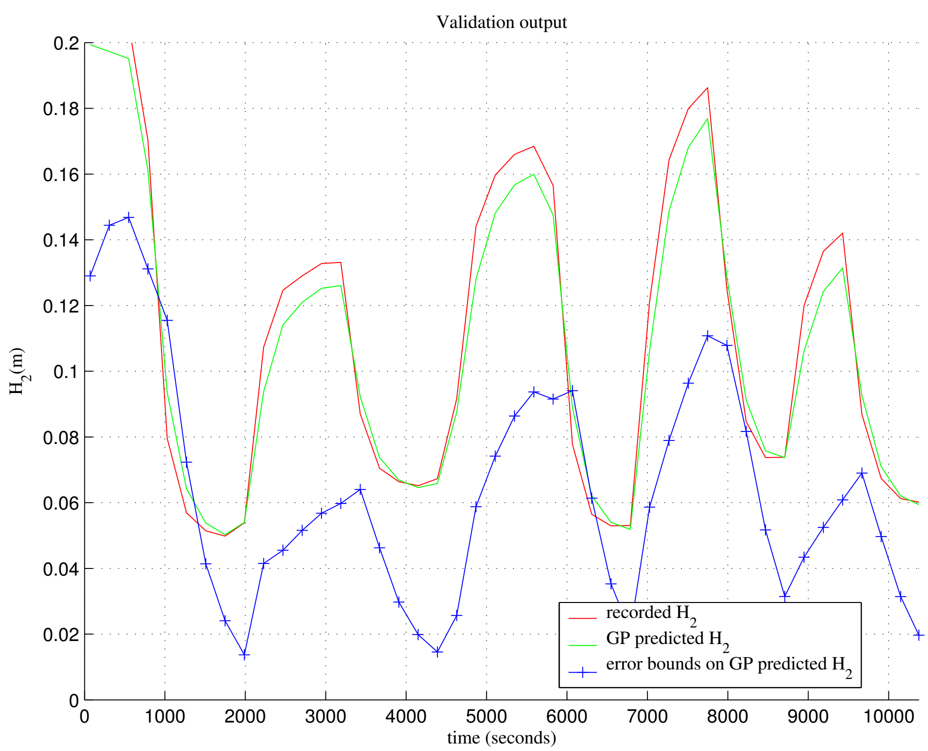

Figure 11: Twin tank system pulsed input response - GP predicted response and standard deviation using reduced input data set (4 points)

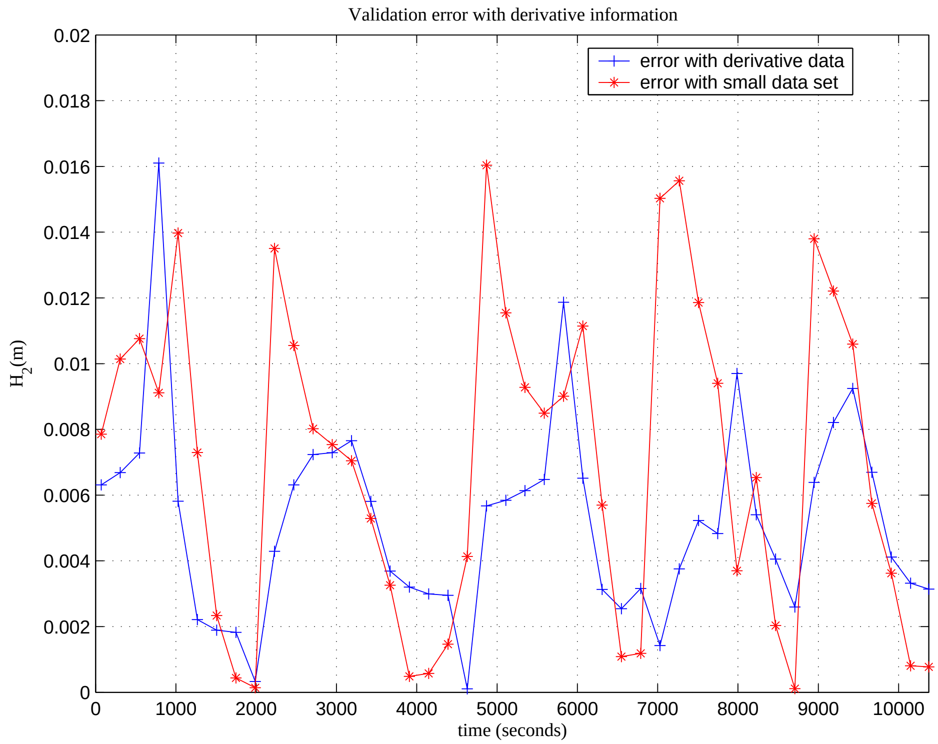

Figure 12: Comparison of optimisation with derivative information and with reduced data set

5 Practical observations

As part of creating this worked example, the issue of sampling data points from dynamic system responses was a significant component of the effort required to tune the model.

- Computational issues: Because the computational load grows approximately as $O(N^3)$ the number of training data used has a significant effect on the feasibility of the approach.
- On- & off-equilibrium data: The bulk of the data sampled from many practical systems will be close to equilibrium, while transient data will be relatively rare, and will sparsely populate off-equilibrium regions of the state-space. The use of derivative observations from locally identified linear models, on-equilibrium, and normal data off-equilibrium is an important way of coping with this aspect, and worked well in our experiments.
- Sampling rate: For dynamic systems we are typically sampling a continuous trajectory into discrete points at discrete moments in time. The variance of GP predictions depends on the number of observations close to the test point, so it appears that the choice of sampling rate can prove to have arbitrary effects on the prediction variance, and can cause numerical problems in the matrix inversions. This is due to our use of overly simple covariance functions which do not take into account the covariance among observations in time. Time-dependent disturbances have been considered in (Murray-Smith and Girard 2001), but further consideration of the effects of incomplete state information, and the effects of NARMAX-type representations, which include other outputs in the inputs, will require some work.
- *Steady state issues:* When a controlled system is in steady state, the training data and the output data are constant. This can lead to a rapid deterioration in the condition number of the covariance matrix, and increases the computational cost of the model.

6 Conclusions

This paper has shown how a Gaussian Process can be used to model a dynamic system. Experimental data was collected from a coupled water tank system. Small test inputs at several operating points were used. The GP model was constructed from a carefully selected subset of the recorded data. This model was then validated by comparing its predicted output with recorded output for a different set of experimental data collected during a large excursion experiment. The GP predicted model output accurately and also provided a useful prediction of expected variance for that output.

Another GP model was generated from the same set of recorded training data but this time using a smaller number of points by incorporating calculated derivative information. The predicted model response was good and generated with a reduced computational load. The case study highlighted the need for more attention to be paid to appropriate choice of covariance functions suitable for dynamic systems.

Acknowledgements

The authors gratefully acknowledge the support from EPSRC grant *Modern statistical approaches to off-equilibrium modelling for nonlinear system control* GR/M76379/01, support from EPSRC grant GR/R15863/01, and Science Foundation Ireland grant 00/PI.1/C067, and the *Multi-Agent Control* Research Training Network by EC TMR grant HPRN-CT-1999-00107. Carl Rasmussen and Agathe Girard have both contributed to this work with software and suggestions.

References

Girard, A., C. E. Rasmussen, J. Quinonero-Candela and R. Murray-Smith (2003). Gaussian process priors with uncertain inputs – application to multiple-step ahead time series forecasting. In: *Advances in Neural Information processing Systems 15* (S. Becker, S. Thrun and K. Obermayer, Eds.).

Kocijan, J., R. Murray-Smith, C.E. Rasmussen and B. Likar (2003). Predictive control with Gaussian process models. In: *IEEE Eurocon 2003: The International Conference on Computer as a Tool*.

Leith, D. J., R. Murray-Smith and W. E. Leithead (2000). Nonlinear structure identification: A Gaussian Process prior/Velocity-based approach. In: *Control 2000, Cambridge*.

MacKay, D. J. C. (1999). Introduction to Gaussian Processes. NIPS'97 Tutorial notes.

Murray-Smith, D.J. (1995). *Continuous System Simulation*. Chapman and Hall. London.

Murray-Smith, R. and A. Girard (2001). Gaussian Process priors with ARMA noise models. In: *Irish Signals and Systems Conference, Maynooth*. pp. 147–152.

Murray-Smith, R., T. A. Johansen and R. Shorten (1999). On transient dynamics, off-equilibrium behaviour and identification in blended multiple model structures. In: *European Control Conference, Karlsruhe, 1999*. pp. BA–14.

O'Hagan, A. (1978). On curve fitting and optimal design for regression (with discussion). *Journal of the Royal Statistical Society B* **40**, 1–42.

Quinonero-Candela, J., A. Girard, J. Larsen and C. E. Rasmussen (2003). Propagation of uncertainty in Bayesian kernels models – application to multiple-step ahead forecasting. ICASSP 2003.

Rasmussen, C. E. (1996). Evaluation of Gaussian Processes and other Methods for Non-Linear Regression. PhD thesis. Graduate department of Computer Science, University of Toronto.

Solak, E., R. Murray-Smith, W. E. Leithead, D. J. Leith and C. E. Rasmussen (2003). Derivative observations in Gaussian process models of dynamic systems. In: *Advances in Neural Information processing Systems 15* (S. Becker, S. Thrun and K. Obermayer, Eds.).

Williams, C. K. I. (1998). Prediction with Gaussian processes: From linear regression to linear prediction and beyond. In: *Learning and Inference in Graphical Models* (M. I. Jordan, Ed.). pp. 599–621. Kluwer.