Gas-liquid separator modelling and simulation with Gaussian Process models

J. Kocijan [a,b,*], B. Likar [c]

[a] *Jožef Stefan Institute, Department of System and Control, Jamova 39, Ljubljana, Slovenia*

[b] *University of Nova Gorica, Nova Gorica, Slovenia*

[c] *Sinabit, d.o.o. Vojkova 8b, Idrija, Slovenia*

Abstract

The Gaussian process (GP) model is an example of a probabilistic, nonparametric model with uncertainty predictions. It can be used for the modelling of complex nonlinear systems and also for dynamic systems identification. The output of the GP model is a normal distribution, expressed in terms of mean and variance. The modelling case study of gas-liquid separator is presented in the paper. It describes the comparison of three methods for dynamic GP model simulation in the phase of model validation. The level of computational burden associated with each approach rises with the complexity of computation necessary for approximation of uncertainty propagation.

Key words: Dynamic system models, System identification, Gaussian process models, simulation.

1 Introduction

While there are numerous methods for the identification of linear dynamic systems from measured data, the nonlinear systems identification requires more sophisticated approaches. The most common choices include artificial neural networks, fuzzy models etc. Gaussian process (GP) models present an emerging, complementary method for a nonlinear system identification. The GP model is a probabilistic, non-parametric black-box model. It differs from most of the other black-box identification approaches as it does not try to approximate the modelled system by fitting the parameters of the selected basis functions but rather searches for the relationship among measured data. GP models are closely related to approaches such as Support Vector Machines and specially Relevance Vector Machines [7]. The output of the GP model is a normal distribution, expressed in terms of mean and variance. The mean value represents the most likely output and the variance can be interpreted as the measure of its confidence. The obtained variance, which depends on the amount and quality of the available identification data, is important information, distinguishing the GP model from other methods. The GP model structure determination is facilitated as only the covariance function and the regressors of the model need to be selected. Also the number of model parameters, which need to be optimised is smaller than in other black-box identification approaches. The disadvantage of the method is the potential computational burden for optimization, which increases with the amount of data and the number of regressors. The GP model was first used for solving

* Corresponding author. tel: +386 1 477 3661, fax: +386 1 477 3994
Email addresses: jus.kocijan@ijs.si (J. Kocijan), bojan.likar@synatec.si (B. Likar).

a regression problem in the late seventies, but it gained popularity within the machine learning community in the late nineties of the twentieth century. Results of a possible implementation of the GP model for the identification of dynamic systems were presented only recently. The investigation of the model with uncertain inputs, which enables the propagation of uncertainty through the model, is given in [2] and illustrated in [5].

The paper is composed as follows. The next section will briefly describe the modelling of dynamic systems with Gaussian process models. The description of dynamic systems simulation will follow in the third section. The case study is described next. Conclusions are given at the end of paper.

2 Modelling of Dynamic Systems with Gaussian Processes

A Gaussian process is an example of the use of a flexible, probabilistic, non-parametric model with uncertainty predictions. Its use and properties for modelling are reviewed in [7].

A Gaussian process is a collection of random variables which have a joint multivariate Gaussian distribution. Assuming a relationship of the form $y = f(\mathbf{x})$ between an input $\mathbf{x}$ and output y, we have $y^1, \ldots, y^n \sim \mathcal{N}(0, \Sigma)$, where $\Sigma_{pq} = \mathrm{Cov}(y_p, y_q) = C(\mathbf{x}_p, \mathbf{x}_q)$ gives the covariance between output points corresponding to input points $\mathbf{x}_p$ and $\mathbf{x}_q$. Thus, the mean $\mu(\mathbf{x})$ (usually assumed to be zero) and the covariance function $C(\mathbf{x}_p, \mathbf{x}_q)$ fully specify the Gaussian process. Note that the covariance function $C(.,.)$ can be any function having the property of generating a positive definite covariance matrix.

A common choice is

$$C(\mathbf{x}_p, \mathbf{x}_q) \;=\; v_1 \exp\left[-\frac{1}{2}\sum_{d=1}^{D} w_d (x_p^d - x_q^d)^2\right] + \delta_{pq} v_0, \tag{1}$$

where $\boldsymbol{\Theta} = [w_1 \ldots w_D \; v_0 \; v_1]^T$ are the 'hyperparameters' of the covariance functions, v_0 is estimated noise variance, v_1 is the estimate of the vertical scale of variation, D is the input dimension and $\delta_{pq} = 1$ if $p = q$ and 0 otherwise. Other forms of covariance functions suitable for different applications can be found in [7]. For a given problem, the parameters are learned (identified) using the data at hand. After the learning, one can use the w parameters as indicators of 'how important' the corresponding input components (dimensions) are: if w_d is zero or near zero it means that the inputs in dimension d contain little information and could possibly be removed.

Consider a set of N D-dimensional input vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$ and a vector of output data $\mathbf{y} = [y^1, y^2, \ldots, y^N]^T$. Based on the data $(\mathbf{X}, \mathbf{y})$, and given a new input vector $\mathbf{x}^*$, we wish to find the predictive distribution of the corresponding output y^*. Unlike other models, there is no model parameter determination as such, within a fixed model structure. With this model, most of the effort consists in *tuning* the parameters of the covariance function. This is done by maximizing the log-likelihood of the parameters, which is computationally relatively demanding since the inverse of the data covariance matrix $(N \times N)$ has to be calculated at every iteration. Nevertheless, the number of parameters to be optimized is small ($D + 2$, see Eq. (1)), which means that optimization convergence might be faster and that the 'curse of dimensionality' so common to black-box identification methods is circumvented or at least decreased.

The described approach can be easily utilized for regression calculation. Based on training set $\mathbf{X}$ a covariance matrix $\mathbf{K}$ of size $N \times N$ is determined. As already mentioned, the aim is to find the distribution of the corresponding output y^* at some new input vector $\mathbf{x}^* = [x_1(N+1), x_2(N+1), \ldots, x_D(N+1)]^T$.

For a new test input $\mathbf{x}^*$, the predictive distribution of the corresponding output is $y^*|(\mathbf{X}, \mathbf{y}), \mathbf{x}^*$ and is Gaussian, with mean and variance

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

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

where $\mathbf{k}(\mathbf{x}^*) = [C(\mathbf{x}^1, \mathbf{x}^*), \ldots, C(\mathbf{x}^N, \mathbf{x}^*)]^T$ is the $N \times 1$ vector of covariances between the test and training cases, and $\kappa(\mathbf{x}^*) = C(\mathbf{x}^*, \mathbf{x}^*)$ is the covariance between the test input and itself.

Gaussian processes can, like neural networks, be used to model static nonlinearities and can therefore be used for modelling of dynamic systems [1–3,5] if delayed input and output signals are fed back and used as regressors. In such cases an autoregressive model is considered, such that the current output depends on previous outputs, as well as on previous control inputs.

$$\begin{aligned} \mathbf{x}(k) &= [y(k-1), y(k-2), \ldots, y(k-L), u(k-1), u(k-2), \ldots, u(k-L)]^T, \\ \hat{y}(k) &= f(\mathbf{x}(k)) + \epsilon, \end{aligned} \tag{4}$$

where k denotes the consecutive number of data sample. Let $\mathbf{x}$ denote the state vector composed of the previous outputs y and inputs u up to a given lag L, and ϵ is white noise.

As can be seen from the presented relations, the obtained model not only describes the dynamic characteristics of nonlinear system, but also provides

information about the confidence in these predictions by means of prediction variance. The Gaussian process can highlight areas of the input space where prediction quality is poor, due to the lack of data, by indicating the higher variance around the predicted mean.

3 Dynamic model simulation

When only the mean values of model predicted values are feed back the simulation was named 'naive'. However, to get more realistic picture of the dynamic model multi-step ahead prediction we have to take account of the uncertainty of future predictions which provide the 'inputs' for estimating further means and uncertainties. The partial overview of results given in [2] is given as follows.

In the case of multi-step ahead prediction we wish to make a prediction at $\mathbf{x}^*$, where input vector $\mathbf{x}^*$ contains also uncertain inputs fed back from outputs. Within a Gaussian approximation, input values can be described by normal distribution $\mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*})$, where $\boldsymbol{\mu}_{\mathbf{x}^*}$ and $\boldsymbol{\Sigma}_{\mathbf{x}^*}$ are the vector and the matrix of input mean values and variances respectively. To obtain a prediction we need to integrate the predictive distribution $p(y^*|(\mathbf{X}, \mathbf{y}), \mathbf{x}^*)$ over the input distribution, that is

$$p(y^*|(\mathbf{X}, \mathbf{y}), \boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) = \int_{-\infty}^{+\infty} p(y^*|(\mathbf{X}, \mathbf{y}), \boldsymbol{\mu}_{\mathbf{x}^*}) p(\mathbf{x}^*|\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) d\mathbf{x}^*, \qquad (5)$$

where

$$p(y^*|(\mathbf{X}, \mathbf{y}), \mathbf{x}^*) = \frac{1}{\sqrt{2\pi\sigma^2(\mathbf{x}^*)}} \exp\left[-\frac{(y^* - \mu(\mathbf{x}^*))^2}{\sigma^2(\mathbf{x}^*)}\right]. \qquad (6)$$

Since $p(y^*|(\mathbf{X}, \mathbf{y}), \mathbf{x}^*)$ is a nonlinear function of $\mathbf{x}^*$, the new predictive distribution $p(y^*|(\mathbf{X}, \mathbf{y}), \boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*})$ is not Gaussian and this integral cannot be solved

without using approximation.

Approximations can be roughly divided into numerical, for example Monte-Carlo numerical methods, and analytical approximations.

3.1 Analytical approximation with Taylor expansion

In order to achieve computational simplicity the analytical approximation which consists of computing only the first two moments, namely the mean and variance of $p(f(\mathbf{x}^*)|(\mathbf{X},\mathbf{y}),\mathbf{x}^*)$ can be used.

To distinguish between $\mu(\mathbf{x}^*)$ and $\sigma^2(\mathbf{x}^*)$, the mean and variance of the Gaussian predictive distribution in the case when there are no uncertain inputs, we denote by $m(\boldsymbol{\mu}_{\mathbf{x}^*},\boldsymbol{\Sigma}_{\mathbf{x}^*})$ the mean and by $v(\boldsymbol{\mu}_{\mathbf{x}^*},\boldsymbol{\Sigma}_{\mathbf{x}^*})$ the variance of the non-Gaussian predictive distribution $p(y^*)|(\mathbf{X},\mathbf{y}),\boldsymbol{\mu}_{\mathbf{x}^*},\boldsymbol{\Sigma}_{\mathbf{x}^*})$, corresponding to $\mathbf{x}^* \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}^*},\boldsymbol{\Sigma}_{\mathbf{x}^*})$. This can be interpreted as a Gaussian approximation, such that

$$p(y^*|(\mathbf{X},\mathbf{y}),\boldsymbol{\mu}_{\mathbf{x}^*},\boldsymbol{\Sigma}_{\mathbf{x}^*}) \approx \mathcal{N}(m,v). \tag{7}$$

The predictive mean and variance of the output corresponding to a noisy input $\mathbf{x}^*$ are obtained by solving [2]

$$m = E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)], \tag{8}$$

$$\begin{aligned} v &= E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + \mathrm{var}_{\mathbf{x}^*}[\mu(\mathbf{x}^*)] \\ &= E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)^2] - (E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)])^2, \end{aligned} \tag{9}$$

where $E_{\mathbf{x}^*}[\cdot]$ denotes the expectation for expression in brackets at the noisy input $\mathbf{x}^*$.

Instead of working with the expressions of $\mu(\mathbf{x}^*)$ and $\sigma^2(\mathbf{x}^*)$, equations (8) and (9) are solved by approximating directly $\mu(\mathbf{x}^*)$ and $\sigma^2(\mathbf{x}^*)$ by their first and second order Taylor expansions respectively around $\boldsymbol{\mu}_{\mathbf{x}^*}$. The second order expansion is required in order to get a correction term for the new variance. This is a relatively rough approximation.

Consequently, within a Gaussian approximation and a Taylor expansion $\mu(\mathbf{x}^*)$ and $\sigma^2(\mathbf{x}^*)$ around $\mathbf{x}^* = \boldsymbol{\mu}_{\mathbf{x}^*}$, the predictive distribution is again Gaussian with mean and variance [2]

$$m(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) = E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)] \approx \mathbf{k}(\mu(\mathbf{x}^*)^T \mathbf{K}^{-1}\mathbf{y}, \tag{10}$$

$$\begin{aligned} v(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) &= E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + \mathrm{var}_{\mathbf{x}^*}(\mu(\mathbf{x}^*)) \\ &\approx \sigma^2(\mu(\mathbf{x}^*)) + \frac{1}{2}\mathrm{Tr}\left\{ \left.\frac{\partial^2 \sigma^2(\mathbf{x}^*)}{\partial \mathbf{x}^* \partial \mathbf{x}^{*T}}\right|_{\mathbf{x}^*=\boldsymbol{\mu}_{\mathbf{x}^*}} \boldsymbol{\Sigma}_{\mathbf{x}^*} \right\} \\ &+ \left.\frac{\partial \mu(\mathbf{x}^*)}{\partial \mathbf{x}^*}\right|^T_{\mathbf{x}^*=\boldsymbol{\mu}_{\mathbf{x}^*}} \boldsymbol{\Sigma}_{\mathbf{x}^*} \left.\frac{\partial \mu(\mathbf{x}^*)}{\partial \mathbf{x}^*}\right|_{\mathbf{x}^*=\boldsymbol{\mu}_{\mathbf{x}^*}}. \end{aligned} \tag{11}$$

For a more detailed derivation see [2]. Eqs. (10) and (11) can be applied to calculation of multi-step ahead prediction with propagation of uncertainty.

3.2 Alternative analytical approximation

The alternative approach to approximation is that instead of approximation of entire mean and variance only the integral of (5) is approximated. The simulation with this kind of approximation is named 'exact'. The expressions for mean and variance are expressions (8) and (9). We consider the Gaussian covariance function given by (1). Since what we get at the nonlinear system output as a response to noisy input with Gaussian distribution is not really

a Gaussian distribution, it is denoted by N for notational convenience (not by $\mathcal{N}$ as Gaussian distributions), because it just denotes some function of the same parametric form. We write it as we would a Gaussian distribution for $\mathbf{x}_i$, centered on $\mathbf{x}_j$:

$$C(\mathbf{x}_i, \mathbf{x}_j) = \tau N_{\mathbf{x}_i}(\mathbf{x}_j, \mathbf{W}) \tag{12}$$

with

$$\tau = (2\pi)^{D/2}|\mathbf{W}|^{1/2} v_1 \tag{13}$$

where $\mathbf{W} = \text{diag}[w_1, \ldots, w_D]$.

We have seen that in order to predict at a noisy input, we needed to integrate the predictive distribution over the input distribution (Eq. (5)). Then, a Gaussian analytical approximation of this integral reduced the problem to computing the mean and variance of $p(f(\mathbf{x})|\mathcal{D}, \boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*})$.

The exact derivations can be found in [4]. Here we are presenting just the final results.

The new predictive mean is equivalent to that obtained for a noise-free test input, except that the covariance between the noisy input and the noise-free training input is computed using a *modified* covariance function which accounts for the uncertainty on the test input. We can write

$$m(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) = \sum_{i=1}^{N} \beta_i C_{mod_1}(\boldsymbol{\mu}_{\mathbf{x}^*}, \mathbf{x}_i) \tag{14}$$

where

$$C_{mod_1}(\boldsymbol{\mu}_{\mathbf{x}^*}, \mathbf{x}_i) = v_1 |\mathbf{I} + \mathbf{W}^{-1}\boldsymbol{\Sigma}_{\mathbf{x}^*}|^{-1/2} \exp\left[-\frac{1}{2}(\boldsymbol{\mu}_{\mathbf{x}^*} - \mathbf{x}_i)^T (\mathbf{W} + \boldsymbol{\Sigma}_{\mathbf{x}^*})^{-1} (\boldsymbol{\mu}_{\mathbf{x}^*} - \mathbf{x}_i)\right] \tag{15}$$

$\mathbf{I}$ is $D \times D$ identity matrix and β_i is i-th element of vector $\boldsymbol{\beta} = \mathbf{K}^{-1}\mathbf{y}$.

That is to say, the correlation length is 'lengthened' to account for the uncertainty on the new input and the vertical amplitude of variation (formally controlled by v_1) is accordingly diminished.

The new predictive variance can also be written using modified covariance functions

$$v(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) = v_1 + \sum_{i,j=1}^{N} (\beta_i\beta_j - K_{ij}^{-1}) C_{mod_2}(\mathbf{x}_i, \mathbf{x}_j) C_{mod_3} - m^2(\boldsymbol{\mu}_{\mathbf{x}^*}, \boldsymbol{\Sigma}_{\mathbf{x}^*}) \quad (16)$$

where $C_{mod_2}(\mathbf{x}_i, \mathbf{x}_j) = \tau N_{\mathbf{x}_i}(\mathbf{x}_j, 2\mathbf{W})$
and $C_{mod_3}(\boldsymbol{\mu}_{\mathbf{x}^*}, \mathbf{x}_b) = \tau \mathcal{N}_{\boldsymbol{\mu}_{\mathbf{x}^*}} \left(\frac{\mathbf{x}_i + \mathbf{x}_j}{2}, \frac{\mathbf{W}}{2} + \boldsymbol{\Sigma}_{\mathbf{x}^*} \right)$.

4 Modelling case study

4.1 Gas-liquid separation plant

The semi-industrial process plant used for the case study in the paper is the unit for separating the gas from liquid that forms part of a larger pilot plant. The scheme of plant is given in Fig. 1.

[Insert figure 1 about here]

The role of the separation unit is to capture flue gases under low pressure from the effluent channels by means of water flow, to cool them down and then supply them under high-enough pressure to other parts of the pilot plant.

The flue gases coming from the effluent channels are absorbed by the water flow into the water circulation pipe through injector.

The water flow is generated by the water ring pump. The speed of the pump is kept constant. The pump feeds the mixture of water and gas into the tank, where gas is separated from water. Hence the accumulated gas in tank forms a sort of 'gas cushion' with increased internal pressure. Owing to this pressure, the flue gas is blown out from tank into the neutralization unit. On the other side, the 'cushion' forces water to circulate back to the reservoir. The quantity of water in the circuit is constant.

In order to understand the basic relations among variables and to illustrate the nonlinearity of the process a mathematical model is introduced. The gas-liquid separation pressure sub-system of interest can be described by a set of two equations.

$$\begin{aligned}\frac{dp_1}{dt} &= \frac{1}{S_1(h_{T_1}-h_1)}(p_0(\alpha_0+\alpha_1 p_1+\alpha_2 p_1^2 \\ &\quad - k_1 R_1^{u_1-1}\sqrt{p_1}) + (p_0+p_1)(\Phi_w - k_2 R_2^{u_2-1}\sqrt{p_1+k_w(h_1-h_{T_2})})), \\ \frac{dh_1}{dt} &= \frac{1}{S_1}(\Phi_w - k_2 R_2^{u_2-1}\sqrt{p_1+k_w(h_1-h_{T_2})}),\end{aligned} \tag{17}$$

where u_i is the command signal of valve Vi, $i=1,2$, h_i is the level in tank Ti, $i=1,2$, p_1 is the relative air pressure in tank T1, S_i is the section area of tank Ti, p_0 is atmospheric pressure, h_{Ti} is height of tank Ti, $i=1,2$, R_i is the ratio of flows at maximum and minimum aperture of valve Vi, $i=1,2$, k_i is the flow coefficient of valve Vi, $i=1,2$, Φ_w is the known constant water flow through pump P1 $i=1,2$, $\alpha_i; i=1,2,3$ are constant parameters.

From the model presented, it can be seen that the nonlinear process is of a multivariable nature (two inputs and two outputs with dynamic interactions between the channels). In our case a level feedback control was imple-

mented. Consequently the dynamic system could be approached as a single-input single-output dynamic system with the command signal of valve V_1 as the input and the pressure in tank T1 as the output. As can be seen from Eqs. (17) pressure is nonlinearly related to level and input flow which results in different dynamic behaviour depending on the operating region.

The real-time experiments were pursued in the environment schematically shown in Fig. 2.

[Insert figure 2 about here]

User-friendly experimentation with the process plant is enabled through interface with the Matlab/Simulink environment [6]. This interface enables PLC access with the Matlab/Simulink using DDE protocol via Serial Communication Link RS232 or TCP/IPv4 over Ethernet IEEE802.3. Control algorithms for experimentation can be prepared in Matlab code or as Simulink blocks and extended with functions/blocks, which access PLC. In our case all schemes for data acquisition were put together as Simulink blocks.

4.2 Process identification

Since the process to be identified is characterised as predominantly the first order system, a model of the form (18) is identified

$$p_1(k+1) = f(p_1(k), u_1(k), h_1(k)), \tag{18}$$

which means that pressure $p_1(k)$, valve signal $u_1(k)$ and liquid level $h_1(k)$ are selected for regressors. Pressure $p_1(k)$ is fed back as distribution, and the predicted mean and variance are calculated in three different ways as described in the previous section. Attempts have been made to identify the system with

a higher order model, but the results were not better.

Based on the response and iterative cut-and-try procedure, a sampling time of 15 seconds is selected. Identification data consists of pseudo random changes of valve signal u_1 in regions with different liquid level h_1, so that as wide a region as possible was encompassed in 967 samples for each signal.

Obtained hyperparameters of the first order Gaussian process were:

$$\begin{aligned}\boldsymbol{\Theta} &= [w_1, w_2, w_3, v_0, v_1] \\ &= [20.2759, 78.0774, 0.1517, 2.9145 \cdot 10^{-5}, 0.1162],\end{aligned} \tag{19}$$

where hyperparameter w_1 corresponds to pressure signal p_1, w_2 corresponds to valve signal u_1, w_3 corresponds to level signal h_1, v_0 is estimated noise variance, and v_1 is the estimate of the vertical scale of variation.

The validation signals that are given in Fig. 3 are different from the identification signals, though of the same kind. Response of the model to validation signal and comparison with process response are given in Figs. 4, 5 and 6.

[Insert figure 3 about here]

[Insert figure 4 about here]

[Insert figure 5 about here]

[Insert figure 6 about here]

It is difficult to notice differences among responses and especially confidence bounds that correspond to standard deviation multiplied by 2 in Figs. 4, 5 and 6. Confidence bounds comparison is more pronounced in the graph of standard deviation changes without contribution of estimated white noise, which is given in Fig. 7. The standard deviation in the case of 'exact' simulation is larger, though standard deviations of other two approaches also indicate

regions where identification data was sparse.

[Insert figure 7 about here]

Fitting of the response for validation signal is evaluated with the following measures:

- average squared test error

$$ASE = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2, \tag{20}$$

- log density error

$$LD = \frac{1}{2N}\sum_{i=1}^{N}\left(\log(2\pi) + \log(\sigma_i^2) + \frac{(\hat{y}_i - y_i)^2}{\sigma_i^2}\right) \tag{21}$$

The evaluation results are given in Tab. 1. The results for 'exact' simulation

Table 1

Fitting of the response for validations signal

	ASE	LD
'naive' simulation	$3.2744{\cdot}10^{-4}$	365.78
Taylor approximation	$3.2744{\cdot}10^{-4}$	149.07
'exact' simulation	$3.2753{\cdot}10^{-4}$	-1.37

show low value of both measures. The relatively low value of average squared test error in comparison with the relatively high value of log density error for 'naive' simulation and simulation with Taylor approximation of uncertainty propagation shows that while the model mean values follow the process response well, the variance may not be large enough. Nevertheless, the standard deviation still clearly indicates the regions where identification data is sparse, as can be seen from Fig. 7, which is good enough for certain purposes, for

example control design. The calculation times for 'naive' simulation can be a magnitude of order lower than in the 'exact' simulation depending on the system order.

5 Conclusions

The modelling case study of gas-liquid separator was presented in the paper. The emphasis of the paper was on the comparison of three methods for dynamic model simulation based on Gaussian processes. All three methods are approximations. The 'naive' simulation is feeding back only the mean values of model predicted values. The Taylor approximation approach and 'exact' approach approximate the model predicted distribution with Gaussian distribution, but in different ways.

The level of computational burden rises with the complexity of computation. Different approaches give different confidence bounds, but it is an important question whether the increased precision is worth increased computational complexity. The purpose of developed model is the main issue that helps answering this question. If the absolute value of variance is an issue in the developed model than computational cost is acceptable. In the case that only the indication of confidence in the model prediction is required than 'naive' simulation will suffice.

References

[1] K. Ažman and J. Kocijan. Application of Gaussian processes for black-box modelling of biosystems. *ISA Transactions*, Vol. 46, No, 4, Pages 443-457,

2007.

[2] A. Girard, C.E. Rasmussen, J. Quinonero Candela, and R. Murray-Smith. Gaussian Process Priors With Uncertain Inputs - Application to Multiple-Step Ahead Time Series Forecasting. In: *Advances in Neural Information Processing Systems*, Vol. 15, Eds. S. Becker, S. Thrun, K. Obermayer, (MIT Press), 2003, 542-552.

[3] A. Girard and R. Murray-Smith. Gaussian Process: Prediction at a noisy input and application to iterative multiple-step ahead forecasting of time-series. In: *Switching and Learning in Feedback Systems*, Eds. R. Murray-Smith, R. Shorten, Springer, 2005.

[4] A. Girrard. *Approximate Methods for Propagation of Uncertainty with Gaussian Process Models.* PhD thesis, University of Glasgow, Glasgow, UK, 2004.

[5] J. Kocijan, A. Girard, B. Banko, R. Murray-Smith. Dynamic Systems Identification with Gaussian Processes, *Mathematical and Computer Modelling of Dynamic Systems*,Vol. 11, 411-424, 2005.

[6] B. Likar. *Gaussian process based nonlinear model predictive control.* MSc thesis, University of Ljubljana, Ljubljana, 2004 (in Slovene).

[7] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for machine learning.* The MIT Press, Cambridge, MA, 2006.

Fig. 1. The scheme of gas-liquid separation plant

Fig. 2. Experimental set-up for data acquisition and control algorithm testing

Fig. 3. Validation signal and response

Fig. 4. Simulation results for 'naive' simulation

Fig. 5. Simulation results for simulation with Taylor approximation

Fig. 6. Simulation results for 'exact' simulation

Fig. 7. Standard deviations without contribution of estimated white noise