Near-real-time online process control using grey-box models

Miriam B. Dodt
PhD Candidate, Dept. of Mechanical Engineering, KU Leuven, Sint-Katelijne-Waver, Belgium

Augustin Persoons
PhD, Dept. of Mechanical Engineering, KU Leuven, Sint-Katelijne-Waver, Belgium

Matthias G. R. Faes
Professor, Chair for Reliability Engineering, TU Dortmund University, Dortmund, Germany

David Moens
Professor, Dept. of Mechanical Engineering, KU Leuven, Sint-Katelijne-Waver, Belgium

ABSTRACT: A scheme is presented with the goal to enable near-real time reliable and robust process control for fast and complex processes. For that, a grey-box modelling approach is employed, making use of adaptive Kriging surrogate modelling. A learning function that is able to refine the Kriging model locally is proposed and its stopping criterion is discussed.

How can one provide not only reliable process monitoring but also control for very fast processes near-real time? That is the question we would like to address in this paper and provide an insight to necessary considerations. Near-real time process control can prove to be very beneficial in manufacturing as diminishing quality can be detected and counter-steered early on. This can not only improve the overall product quality and reliability of the process, but possibly also extend the overall lifetime of the product, increasing the safety and reliability and reducing maintenance or production costs (Moya et al. (2020)). Fast, cost-efficient and most importantly, reliable machine-based process control is therefore of great value in industrial applications.

Detecting a control need, the input can be used to steer the process in the desired direction. The parameter needs to be chosen according to the current state of the physical system, which can be assessed through measuring devices (Nise (2011)). Those sensors however, are inevitably subjected to noise and measurement errors, which display themselves as aleatory and epistemic uncertainties. The exact effect the input parameter has on the system in a closed-loop control system is affected by uncertainty as well (Heirung et al. (2018)). Achieving a reliable control means considering these uncertainties.

Furthermore, in order to define the optimal control strategy, predictions of the process' reaction to the input parameter need to be evaluated. This requires the use of a digital model. A (numerical) model of the physical process, also called white-box model (Pitchforth et al. (2021)), has good accuracy but is expensive to evaluate and too slow to be usable alone in a near-real time context for fast, dynamic processes. Not only are these processes fast but oftentimes also physically complex, so that one model evaluation takes a significant amount of time. The discrepancy between the evaluation and the process time is therefore not negligible.

Apart from the white-box model, there is also a different group of models, the so-called black-box models (Pitchforth et al. (2021)). They are not physics- but data-based and often build a regression model (e.g. Neural Networks (Bishop (1994)), surface response models (Box and Wilson (1951)), support vector machines (Cortes and Vapnik (1995)), polynomial chaos expansion (Mai and Sudret (2017)), Gaussian process regression (Santner et al. (2003); Rasmussen and Williams (2006))) from a limited amount of calibration samples, the design of experiments (DoEs). After calibration the black-box model can be used in place of the numerical model at very limited computational cost. However, the physical model needs to be evaluated at every sample of the DoE, which can be computationally expensive. Furthermore, it introduces an additional modelling error to the system, which presents itself as an additional epistemic uncertainty.

Subsequently, neither the black-box, nor the white-box model are able to fulfill the modelling requirements on their own. In this paper, a solution is explored using the framework of grey-box models (Pitchforth et al. (2021)), which combine both types of models in one, to make use of their respective advantages. It is of great benefit to have a process model that can continuously evaluate and update itself against the measured process state and predict future instances reliably, as this allows for robust and reliable control, even if the process deviates from the expectations.

A modelling technique, whose aim is creating such a process model, is the self-improving digital twin (Kritzinger et al. (2018); Moya et al. (2020)). The digital twin mirrors the physical process from measurements of it while considering uncertainties. It then performs analyses based on the current process state. Based on the analysis the optimal input can be chosen, leading to the best possible process results. However, the process is affected by unpredictable and uncontrollable process drift, which can be caused by e.g. wear and tear of material or accumulation of dirt, which effects the process itself but also its measurements. It is clear, that proper consideration of the process drift is of great importance but not trivial to do.

In summary, a scheme is necessary, that can account for the process drift as well as for the uncertainty of the process and measurements while maximising computational speed by minimising the number of necessary white-box evaluations. In this paper, a simple grey-box model is proposed consisting of a numerical model and an adaptive Kriging surrogate model, the physics-informed white-box model and the data-driven black box model respectively. In order to take advantage of the positive aspects of both models, they are used in combination with each other. The grey-box model tries to find the optimal compromise between the amount of calibration points, which are expensive to evaluate, and the modelling error.

This paper introduces a general and simple scheme to perform process control for dynamic processes. It is proposed to achieve the objective detailed above to find a near-optimal DoE through an adaptive Kriging scheme (Jones et al. (1998); Echard et al. (2011)). At first, the methodology of the digital twin is introduced. Herein the adaptive Kriging scheme and its principal components: the learning function and stopping criterion are introduced in more detail. An application on a benchmark problem and its results is presented afterwards. Finally, the paper closes with the conclusion.

1. METHODOLOGY

Let there be a process affected by two types of parameters. The first are state variables x, that are uncontrollably effected by process drift. The second one are input or control variables v, which are used to counter the process drift. The performance $y(x,v)$ is given as a function of both. It is further assumed, that the process drift and its development is not known beforehand but past instances can be gained from measurements. The goal at the current control step i is to find the input v_{opt} that maximises the reliability. Considering a process that is significantly faster than its numerical simulation, process control is performed over multiple processes at once. The goal is then to find a unique input v that guarantees the reliability over the next t_n steps. Algorithm 1 gives an overview over the optimisation scheme.

The already often mentioned Kriging model (Matheron (1973)) is used to emulate the performance function $g(\mathbf{X})$, which is a function based on failure criteria of the system, in a way, that $g(\mathbf{X}) < 0$ correspond to system failure. The boundary, $g(\mathbf{X}) = 0$, is called the limit state function and $g(\mathbf{X}) > 0$ indicates the safe domain. Considering the sign of the function, the input space can be divided into safe (positive performance) and failure (negative performance) domains. The Kriging model is initially calibrated using n_{ini} Latin Hypercube (or other) samples evaluated using the white-box model. An adaptive Kriging meta model (Bichon et al. (2008); Echard et al. (2011)) was chosen due to its ability to give an estimation of its own localised accuracy. Kriging models are based on Gaussian processes, thus their localised variance can be seen as an indicator of the confidence of the model regarding the local predicted value (Echard et al. (2011)).

In the second step, the effect of the process drift needs to be assessed. For that, the evolution of the state variable x for the next t_n processes is extrapolated and its confidence bounds are determined. The choice of the extrapolation scheme is problem dependent and needs to be chosen appropriately. As this is not the core focus of this work, a Gaussian process model is used arbitrarily for the sake of illustration. The confidence bounds of the extrapolation result define the interval I_x^i the state variable x lies most probably in during the next t_n processes. Within this interval, no probability distribution is assumed.

In the next step it is estimated, if adjustment of the input is necessary. This is done by calculating the estimated reliability $\hat{R}$

$$\hat{R} = 1 - \Phi\left(\frac{-\hat{\mu}(\mathbf{X})}{\hat{\sigma}(\mathbf{X})}\right) \tag{1}$$

with Φ the standard Gaussian cumulative distribution function. This quantity can be interpreted as the probability, according to the Kriging model, that the sample lies in the safe domain, as it corresponds to the probability that $\hat{g}(\mathbf{X})$ is positive. The reliability is evaluated for $\mathbf{X} = [x_{worst}, v_{i-1}]$ at v of the previous control step $i-1$ and its corresponding worst-case estimate x_{worst} based on the Kriging model. Equation 1 can then be written as

$$\hat{R}_i = \min_{x \in \hat{\mathrm{I}}_x^i} \left(\hat{R}(y(x_{worst}, v_{i-1}))\right) \tag{2}$$

Algorithm 1 Pseudocode of the process control scheme

```
1. Initialise Kriging model, $x_0$
For $i = 1 : n$
  2. Estimate $\mathrm{I}_x^i$
  3. Compute $\hat{R}_i$
    If $\hat{R}_i < R_{min}$
      $x_i = x_{i-1}$
    Else
      4. Find $v_{opt}$ for $x_{worst}$ and compute $\hat{R}_{opt}$
      If $\hat{R}_{opt} < R_{crit}$
        $x_i = x_{i-1}$
      Else
        While $\mathrm{CoV}(x_{worst}, v_{opt}) > \mathrm{CoV}_{\min}$
          Refine Kriging model
        End
        Step 3 - 4
        If $\hat{R}_{opt} < R_{crit}$
          $x_i = x_{i-1}$
        Else
          Warning: Process is expected to fail.
        End
      End
    End
End
```

If $\hat{R}_i < R_{min}$, the previous v can be used for the next processes as well. R_{min} is a operator-chosen parameter determining the minimal required reliability to not change the input parameters. Through that it is verified, if the current vector of control variables already satisfies the safety criterion. The optimisation of v is then only performed if necessary, thus avoiding unnecessary calibration of the process.

If $\hat{R}_i \geq R_{min}$, an optimisation scheme is run, to find the value of the control variable v, that maximises the estimated reliability. The reliability is computed for the worst-case state variable within I_x^i the interval of possible state variables of the next t_n processes. Figure 1 visualises this step with an

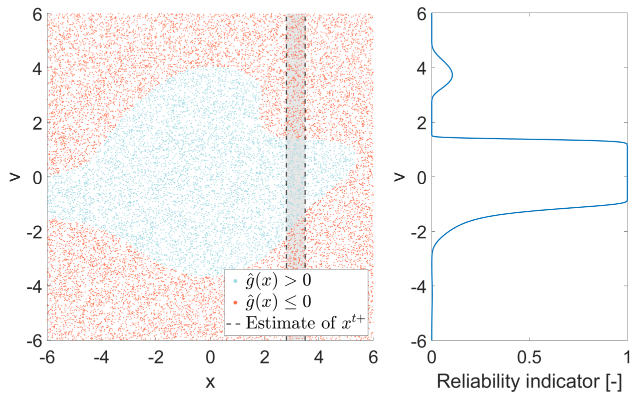

Figure 1: Reliability indicator

example. On the left hand side, the Kriging model is visible. The Latin Hypercube samples marked in red symbolise the failure domain, the blue ones show the safe domain. I_x^i for the next processes is displayed as the grey-shaded area. On the left hand side, the worst-case reliability $\hat{R}_{worst}(\hat{y})$ is displayed over the full domain of v. During the optimisation process, the maximum of this curve needs to be found. It is further visible, that the result is not necessarily unique. If the optimised reliability

$$\hat{R}_{opt} = \max_{v\in \mathrm{I}_v} \min_{x\in \hat{\mathrm{I}}_x^i} \left(\hat{R}(y(x,v))\right) \tag{3}$$

does not agree with the user-defined reliability threshold R_{crit}, it needs to be assessed, whether this is due to a badly trained Kriging meta-model or due to actual failure. In the former case, the Kriging model is refined, until the stopping criterion is satisfied. If the optimisation yielded positive results, the optimised control variable can be used as input for the next processes. Otherwise, a warning can be given and/or the process halted, until the operator restarts it.

2. THE KRIGING MODEL

The Kriging model, which uses the assumption of an underlying Gaussian process, is used as the black-box model. Kriging is a well established surrogate method, that was initially introduced for global optimisation by Jones et al. (1998) and popularised through adaptive Kriging Monte Carlo Simulation (AK-MCS) by Echard et al. (2011).

The model is used here to emulate the performance function of the process y. A Gaussian random variable, defined by its mean $\hat{\mu}$ and standard deviation $\hat{\sigma}$, is associated for each input vector $\mathbf{X} = [x, v]$. The hat symbolises that it is an estimated performance. The mean is considered as the Kriging estimated performance and the standard deviation as its indicator of precision. Further information can be found in the detailed overview provided by Moustapha et al. (2022).

The assessment of the own accuracy has allowed the development of adaptive sampling approaches (Jones et al. (1998); Echard et al. (2011)), to minimise the influence of the modeling error over the quantity of interest. The adaptive calibration scheme aims at building a near optimal design of experiment using a so called learning function identifying the samples that have the highest expected influence on the reduction of the error. The optimal sample is then evaluated using the white-box model and added to the DoE. This is of great importance for on-line monitoring and control, as the process drift may push the process into an area of the state variable, for which the calibration of the Kriging model is not sufficiently accurate. Three aspects that are key to the adaptive refinement of the Kriging model are when to refine, where to refine and when to stop. The strategy used to determine whether refinement is necessary was discussed previously. In the next sections the learning function that decides where to refine the surrogate and the stopping criterion are discussed in further detail.

2.1. *Learning function*

The true limit-state function is usually unavailable, finding a satisfying input value for the next processes is therefore solely dependent on the Kriging model of the limit state. As such, having a reliable and sufficiently accurate surrogate is of utmost importance. The refinement of the Kriging model is the crux to achieve this. Evaluating the Kriging model is computational inexpensive, however, during refinement new samples are added to the DoE. For that it is necessary to call the expensive full model. Therefore, the refinement process should be optimised to effectively add the least amount of samples necessary. To achieve that, a so-called learning function is used to weight all possible new samples and to add the sample, that gives the most information.

Usually, learning functions are used in a structural engineering context, where the global probability of failure for all possible failure modes is of interest. In the case of the optimisation scheme however, the goal is to find an input that is locally in the safe domain for all possible state parameters within the interval I_x^i. Therefore, also a local scheme is necessary, that finds a good trade-off between exploration and exploitation. As a reminder, the refinement step is performed, when no input was found guaranteeing the entire interval lies in the safe domain. The goal of this step is then to improve the calibration of the surrogate so that it represents the process better hopefully allowing to find such an input value.

This can be achieved by expanding the known safe domain, which can be done by adding new safe samples. On the one hand, when adding a sample close to the safe domain it is very likely to expand it but the expansion will be limited. On the other hand, a point far away from the known safe domain would expand it significantly but the point is far less likely to be safe.

The proposed learning function

$$L = \max\left(\Phi\left(\frac{\hat{\mu}}{\hat{\sigma}}\right)\cdot \mathbf{L}^{\mathrm{E}}\right) \qquad \forall x \in \hat{\mathrm{I}}_x^i, \forall v \tag{4}$$

is a trade-off of these two competing cases and can be seen as the expected Euclidean distance of expansion of the safe domain.

$$L_j^E = \min|(x_1,v_1)-(x_2,v_2)| \tag{5}$$
$$\text{for } x_1 \in \hat{\mathrm{I}}_x^i \text{ and } v_1 = v_2$$

is the Euclidean distance between a sample j within $[x,v]\forall x \in \mathrm{I}_x^i$, $\forall v$ and its closest neighbour in the save domain at the same v-value. Only the distance considering the state variables (horizontal axis in figure 2) is of interest, as the control value needs to be found, which has the highest reliability. Since the only goal is to expand the safe domain to the whole interval I_x^i, the state variable area considered to add a new sample is limited to this interval.

2.2. *Stopping criterion*

As previously mentioned, the Kriging model is able to assess its own estimated accuracy. This can be utilised for adaptive refinement and also for the stopping criterion. Through this criterion, the amount of added samples to the DoE is determined. If the refinement stops too early, convergence might not have been reached yet. If it stops too late however, it is computationally more expensive. The refinement should stop, when a sufficient local accuracy has been reached. This can be checked through the coefficient of variation.

A convergence study on the stopping criterion $\mathrm{CoV}_{\max}$ was performed for

$$\mathrm{CoV} = \frac{\sigma}{|\mu|} \le \mathrm{CoV}_{\max}. \tag{6}$$

The study and the results are discussed in section 3.1.

3. CASE STUDY

In the following section, the optimisation scheme and especially the proposed learning function is tested on the well-known four-branch limit state function

$$g(x,v) = \min[g_1(x,v), g_2(x,v), g_3(x,v), g_4(x,v)] \tag{7}$$

with

$$\begin{aligned} g_1(x,v) &= k_1 + \frac{(x-v)^2}{10} - \frac{x-v}{\sqrt{2}} \\ g_2(x,v) &= k_1 + \frac{(x-v)^2}{10} + \frac{x-v}{\sqrt{2}} \\ g_3(x,v) &= x+v+\frac{k_2}{\sqrt{2}} \\ g_3(x,v) &= -x-v+\frac{k_2}{\sqrt{2}}. \end{aligned}$$

with $k_1 = 3$ and $k_2 = 7$ (Echard et al. (2011); Schueremans and Van Gemert (2005); Schöbi et al. (2017)). The system considered is discrete; one process equals one time step and $x, v \in R^1 \in [-6,6]$. Within the total time of $T = 100$ processes, the control variable v can be adjusted every $t_n = 4$ processes, with a starting value of $v = 0$ for the first four processes. The evolution of the uncontrollable parameter x was estimated through Gaussian process regression with a Matern 5/2 kernel, the

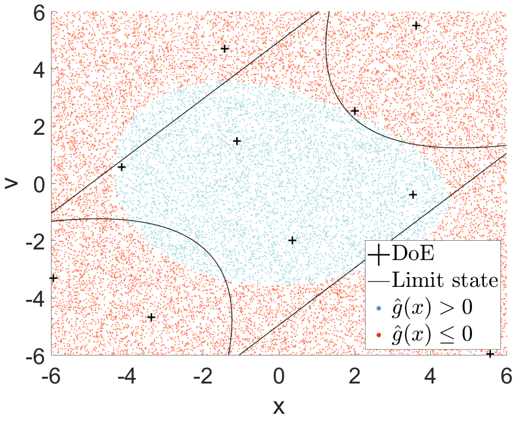

Figure 2: Initial Kriging model, $n_{ini} = 10$.

boundaries of the interval were chosen as the $1-\sigma$-Interval. The measurements of x were taken for every process and a Gaussian white noise was added onto them to mirror noisy measurements. It should be noted that this example is a toy example used as a first illustration of the approach and is not representative of the expected process drift of actual physical processes. It is furthermore recommended to use a higher-confidence interval than done in this example. R_{min} and R_{crit} were chosen with 0.1 and 0.2 respectively.

As the white-box model, the analytical performance function is used. The black-box model is a Kriging model initially trained on 10 Latin hypercube (LH) samples (figure 2). New samples are added according to the proposed learning function, if no input was found that would result in a sufficient performance, until the stopping criterion has been reached. The Kriging function was evaluated on a set of $2,000,000$ Latin Hypercube samples. Both LH designs used the same seed, which was varied during the experiment.

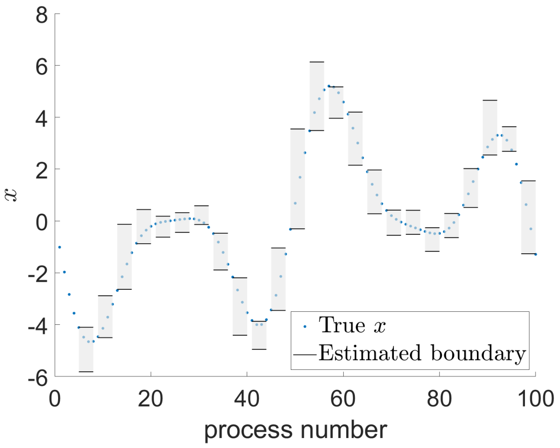

Figure 3: Extrapolation of x for $t_n = 4$, $T = 100$.

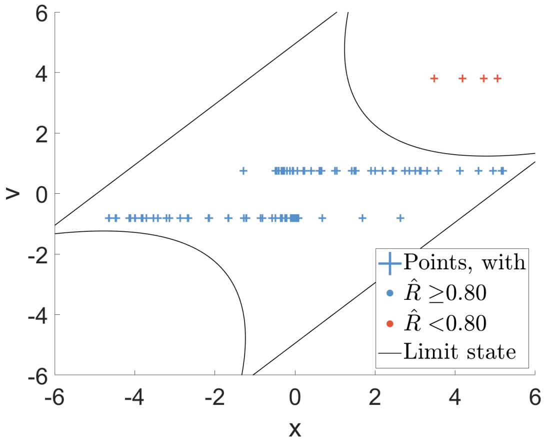

Figure 4: Results of the optimised input parameter with noise-free x compared to the true limit-state function.

3.1. Results

In order to have good control results, it is necessary to properly estimate the time evolution of x. Figure 3 shows, that this is the case, with the blue points denoting the noise-free value of x and the grey-shaded area marking the estimated interval $\hat{\mathrm{I}}_x^{t_i}$ at $t = t_i : t_i + t_n$. The control results with the proposed learning function and $\mathrm{CoV}_{\max} = 0.2$ are shown in figure 4 with regards to the true limit state function and in figure 5 with regards to the Kriging model the input was optimised for. The control scheme performs mostly satisfying, in only one control step are the results not in the safe domain, which is also correctly identified. The Kriging model was refined during three control steps at $t = 5$ (3 samples added), $t = 53$ (3 samples added) and $t = 57$ (8 samples added). The resulting model can be seen in figure 6. By adding 14 new samples to the DoE, sufficient accuracy for the control process has been reached. It can be expected, that the Kriging model does not require any further refinement even for future control steps. The learning function performed well and informative samples were added within the area of interest.

While the stopping criterion $\mathrm{CoV}_{\max} = 0.2$ was

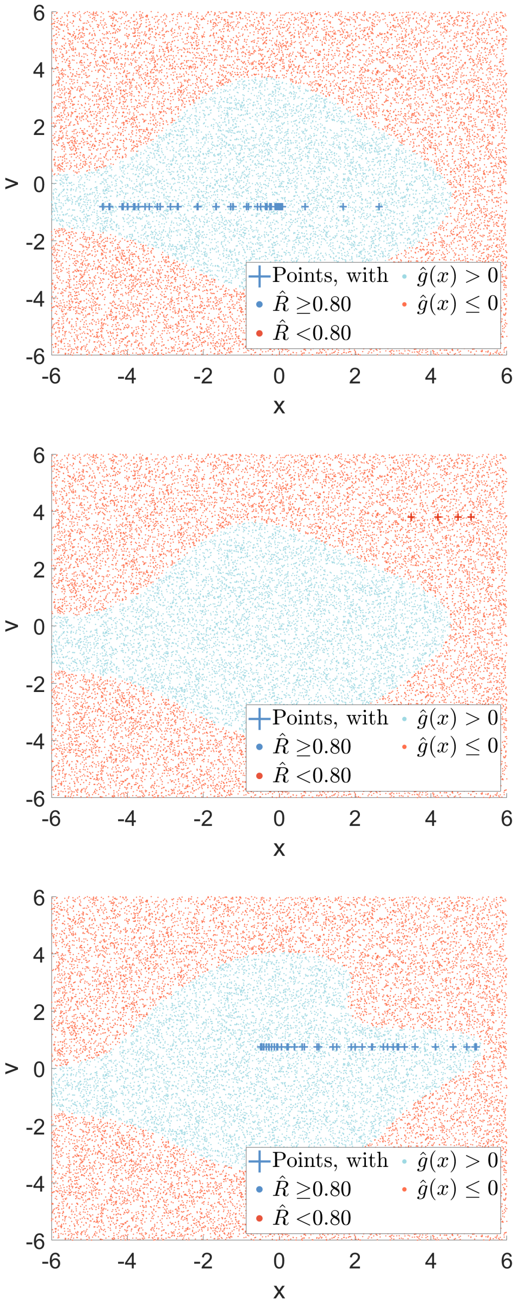

Figure 5: Optimisation results in their respective Kriging model. Top: $t_n = 5$ to 52, centre: $t_n = 53$ to 56, bottom: $t_n = 57$ to 100

chosen, different optimal values between 0.1 and 0.5 were observed for other realisations. A convergence study for the optimal CoV_{max} is difficult to perform, as the CoV at the optimisation result is of interest and this point changes after each refinement step.

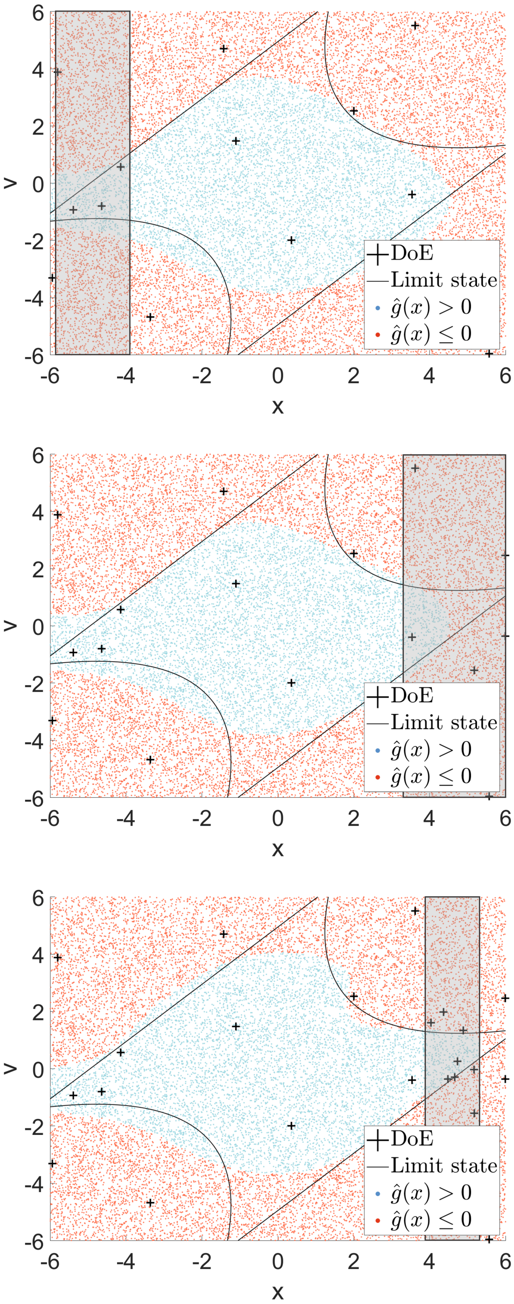

Figure 6: From top to bottom: Kriging model at $t = 5$, $t = 53$ and $t = 57$ (final Kriging model). The shaded area displays I_x^i of the respective time step.

4. CONCLUSION

In this paper a grey-box modelling scheme for on-line process control was presented. The grey-box approach is advantageous over the pure white-box or black-box approaches, as it can find a good trade-off between both models limits and possibilities. The ability to adapt to parameter deviation in unexpected areas of the problem space is of great importance to perform reliable and computational efficient process control. The proposed scheme performs satisfactory for a simple example with the additional constraint of performing process control over multiple processes at once. For the black-box model within this scheme – an adaptive Kriging surrogate – a learning function was proposed, that is well suited to perform localised refinement. While the learning function performs well and informative samples were added, its stopping criterion needs further consideration. Future research will also focus on a continuous formulation of the learning function to make it more robust in higher dimension and for larger problem spaces. All in all, the proposed scheme performs process control satisfactory for a simple example, highlighting the advantages of its simplicity and the locality of the learning function.

5. REFERENCES

Bichon, B. J., Eldred, M. S., Swiler, L. P., Mahadevan, S., and McFarland, J. M. (2008). "Efficient Global Reliability Analysis for Nonlinear Implicit Performance Functions." *AIAA Journal*, 46(10), 2459–2468.

Bishop, C. M. (1994). "Neural networks and their applications." *Review of Scientific Instruments*, 65(6), 1803–1832 Publisher: American Institute of Physics.

Box, G. E. P. and Wilson, K. B. (1951). "On the Experimental Attainment of Optimum Conditions." *Journal of the Royal Statistical Society: Series B (Methodological)*, 13.

Cortes, C. and Vapnik, V. (1995). "Support-vector networks." *Machine Learning*, 20(3), 273–297.

Echard, B., Gayton, N., and Lemaire, M. (2011). "AK-MCS: An active learning reliability method combining Kriging and Monte Carlo Simulation." *Structural Safety*, 33, 145–154.

Heirung, T. A. N., Paulson, J. A., O'Leary, J., and Mesbah, A. (2018). "Stochastic model predictive control — how does it work?." *Computers & Chemical Engineering*, 114.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). "Efficient Global Optimization of Expensive Black-Box Functions." *Journal of Global Optimization*, 13(4), 455–492.

Kritzinger, W., Karner, M., Traar, G., Henjes, J., and Sihn, W. (2018). "Digital Twin in manufacturing: A categorical literature review and classification." *IFAC-PapersOnLine*, 51(11), 1016–1022.

Mai, C. V. and Sudret, B. (2017). "Surrogate Models for Oscillatory Systems Using Sparse Polynomial Chaos Expansions and Stochastic Time Warping." *SIAM/ASA Journal on Uncertainty Quantification*, 5(1), 540–571.

Matheron, G. (1973). "The intrinsic random functions and their applications." *Advances in Applied Probability*.

Moustapha, M., Marelli, S., and Sudret, B. (2022). "Active learning for structural reliability: survey, general framework and benchmark." *Structural Safety*, 96.

Moya, B., Badías, A., Alfaro, I., Chinesta, F., and Cueto, E. (2020). "Digital twins that learn and correct themselves." *International Journal for Numerical Methods in Engineering*.

Nise, N. S. (2011). *Control systems engineering*. John Wiley & Sons, 6 edition.

Pitchforth, D., Rogers, T., Tygesen, U., and Cross, E. (2021). "Grey-box models for wave loading prediction." *Mechanical Systems and Signal Processing*, 159.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass.

Santner, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments*. Springer.

Schueremans, L. and Van Gemert, D. (2005). "Benefit of splines and neural networks in simulation based structural reliability analysis." *Structural Safety*, 27(3), 246–261.

Schöbi, R., Sudret, B., and Marelli, S. (2017). "Rare Event Estimation Using Polynomial-Chaos Kriging." *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 3(2).