Reliable nonlinear dynamic gray-box modeling by regularized training data estimation and sensitivity analysis

Joschka Winz * **Sebastian Engell** *

* *Process Dynamics and Operations Group, Department of Biochemical and Chemical Engineering, Technische Universität Dortmund, Emil-Figge Str. 70, 44227 Dortmund, Germany (e-mail: joschka.winz@tu-dortmund.de).*

Abstract: Dynamic process models are a key requirement for advanced process control and the application of process optimization techniques. The derivation of these models is time consuming and error-prone in cases where a lack of physico-chemical understanding is present. Machine learning (ML) methods can be employed in these cases to extract models or model elements from data. To reduce the amount of necessary data and to increase the extrapolation capabilities, gray-box models can be used that combine mechanistic equations with ML models. For embedded ML-models, the selection of a suitable model structure is challenging. Therefore, we propose a methodology to approach this problem in several steps by firstly estimating what values the ML-models should predict to accurately describe the experimental data. Subsequently, the ML-submodels can be trained using any ML-toolbox. Finally, a full parameter estimation is performed using a dynamic simulation in the cost function. We investigate different algorithmic options and show promising results for a case study of the fermentation of a sporulating bacterium.

Keywords: Gray-box modeling, machine-learning, parameter-estimation, fermentation

1. INTRODUCTION

Gray-box modeling denotes the combination of models based on mechanistic knowledge, so-called white-box models, together with models based on data, so-called black-box models. The motivation of this approach lies in combining the favorable properties of the two extreme cases. This includes a physical interpretability of the results as well as the usage of the black-box submodels to extract knowledge from data in cases where the underlying phenomena are unknown or too difficult to model.

Gray-box models have been previously applied in many cases. Over the years work has been focused on the application to discrete-time models in various domains, as chemical processes (Tulleken (1993)), fermentation processes (Niu et al. (2013)) and hydrodynamic river models (Sohlberg and Sernfält (2002)). Nowadays, techniques like augmented recurrent neural networks have also been proposed for gray-box modeling in discrete time, see Halmschlager et al. (2019).

Recent work also addressed the development of continuous time gray-box models, in which the black-box part represents an embedded variable within a set of differential equations. In this case, initializing a machine learning (ML) model and estimating the parameters using the simulation of the differential equations for the evaluation of the cost function is a difficult task. This is because it is not a priori clear, what ML model structure is suitable for the problem at hand and because the adaptation of all model parameters at once is not effective without proper initialization. Therefore, the problem should be approached in a step-wise fashion. The first step entails the estimation of input-output data of the embedded black box model or models. After such an input-output/ feature/ training data set has been estimated, an analysis of the model can be performed, and the black-box models can be trained on the estimated data. In some recent work, the model was applied after this training step, while in other cases a full dynamic parameter estimation step was performed. An overview over these approaches is given in the following.

Scheffold et al. (2021) use a state estimator for the estimation of the training data. They set up a model using symbolic regression by finding suitable basis functions, that are linearly combined to give the model predictions. The model is applied in model predictive control to control a polymerization semibatch reactor without full dynamic parameter estimation.

In the work by de Prada et al. (2018), a data-estimation technique similar to the one used in this work is proposed. A piecewise constant function describing the growth of biomass in a fermentation process is estimated, which in this case leads to a quadratic optimization problem (QP). This is due to the fact, that the differential equations are linear with respect to the embedded variables. They use the ALAMO toolbox for surrogate model construction and show the successful application to the ABE fermentation

⋆ This research has been supported by the project "KI-Inkubator-Labore in der Prozessindustrie - KEEN", funded by the Bundesministerium für Wirtschaft und Klimaschutz (BMWK) under grant number 01MK20014T. This support is gratefully acknowledged.

process.
Hebing et al. (2020) consider the rate estimation problem of kinetic rates in a fermentation process. They reformulate the optimization problem for data estimation into a quadratic form where they first estimate the stoichiometric constants by an analysis of the cell internal metabolism. In contrast to de Prada et al., Hebing et al. use a piecewise linear dependency on time and at the end perform a full dynamic parameter estimation.
In this work we extend these approaches in two directions. Firstly, we consider a non-linear system as e.g. the dependency on the substrate concentration is included. This has the drawback that the problem has to be solved using non-linear programs together with a solver for differential equations. This leads to the need for a suitable initial value generation strategy.
The second extension is the analysis of the reliability of the estimated data by means of a sensitivity analysis.
The rest of this paper is structured as follows. Section 2 deals with the description of the methodology for creating dynamic gray-box models in a non-linear setting. Section 3 contains the description of the simulation model that is used as a case study for data-generation. In section 4, the results of the application of the methodology to the case study are presented. Finally, section 5 concludes the work and gives an outlook on future research.

2. MODELING METHODOLOGY

Our methodology assumes the existence of a structured finite-dimensional model that is given as a system of differential algebraic equations (DAE) as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{\Theta}}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{\varphi}) \tag{1}$$

$$\boldsymbol{0} = \boldsymbol{g}_{\boldsymbol{\Theta}}(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{\varphi}) \tag{2}$$

Here $\boldsymbol{x} \in \mathbb{R}^{n_x}$ denotes the state vector, $\boldsymbol{z} \in \mathbb{R}^{n_z}$ the vector of algebraic variables and $\boldsymbol{u} \in \mathbb{R}^{n_u}$ the vector of input variables. The right hand sides of the differential equations $\boldsymbol{f}$ and the set of algebraic equations $\boldsymbol{g}$ depend on a set of parameters $\boldsymbol{\Theta} \in \mathbb{R}^{n_\Theta}$.
The goal of the methodology besides estimating the parameters $\boldsymbol{\Theta}$ is to find a machine learning (ML) model of the embedded dependent variables $\boldsymbol{\varphi} \in \mathbb{R}^{n_\varphi}$ for which there is no known mechanistic relationship. Specifically, we do not assume that there is feature data of $\boldsymbol{\varphi}$ available.
The basis of the data-based modeling is a data set consisting of

- the values of some output (measured) variables $\{y^{exp}_{i,j,k}\} = \boldsymbol{Y}^{exp} \in \mathbb{R}^{n_y \times n_{exp} \times n_{samp}}$ and
- the values of the input variables $\{u^{exp}_{i,j,k}\} = \boldsymbol{U}^{exp} \in \mathbb{R}^{n_u \times n_{exp} \times n_{samp}}$ at each of the
- known sampling times $\{t^{exp}_{j,k}\} = \boldsymbol{T}^{exp} \in \mathbb{R}^{n_{exp} \times n_{samp}}$ as well as
- the values of the initial states $\{x^{0,exp}_{i,j}\} = \boldsymbol{X}^{0,exp} \in \mathbb{R}^{n_x \times n_{exp}}$,

which were collected by performing n_{exp} experiments. Here the vector of outputs $\boldsymbol{y}$ is related to the state vector $\boldsymbol{x}$ and to the vector of algebraic variables $\boldsymbol{z}$ by the known relation $\boldsymbol{h}$ as

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{z}). \tag{3}$$

Without loss of generality, we assume that n_{samp} data points have been collected for each of the n_y outputs in every experiment. Additionally we assume that the initial states for each experiment are known. If this data is not available, the proposed methodology can still be applied, but the subsequently discussed optimization problems have to be adjusted to also include the unknown initial states as decision variables.
A conventional approach to the problem is to intuitively guess the model structure of $\boldsymbol{\varphi}$. To this end, an estimation of the set of descriptors $\boldsymbol{d} \subset \{\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}\}$ is needed. These descriptors are the subset of the state, algebraic and input variables that serve as independent variables for the embedded dependent variables, such that $\boldsymbol{\varphi} = \boldsymbol{\varphi}_{\boldsymbol{\Theta_{ML}}}(\boldsymbol{d})$. In this notation $\boldsymbol{\Theta_{ML}}$ denotes the ML-model parameters. With $\boldsymbol{\varphi}$ defined, the full dynamic parameter estimation problem shown in (4) can be solved using suitable initial values for both the set of parameters of the DAE system $\boldsymbol{\Theta}$, and the set of parameters of the ML model $\boldsymbol{\Theta_{ML}}$.

$$\begin{aligned}
\min_{\boldsymbol{\Theta}, \boldsymbol{\Theta}_{ML}} \quad & J\left(\boldsymbol{Y}^{exp}, \hat{\boldsymbol{Y}}\right) \\
\text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{\Theta}}\left(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{\varphi}_{\boldsymbol{\Theta_{ML}}}(\boldsymbol{d})\right) \\
& \boldsymbol{0} = \boldsymbol{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \boldsymbol{\varphi}_{\boldsymbol{\Theta_{ML}}}(\boldsymbol{d})\right) \\
& x_i(t_{j,0}) = x^{0,exp}_{i,j} \qquad \forall\, i = 1, \ldots, n_x \\
& \qquad\qquad\qquad\qquad\quad\; j = 1, \ldots, n_{exp} \\
& \hat{\boldsymbol{y}} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{z})
\end{aligned} \tag{4}$$

In this formulation, the inputs $\boldsymbol{u}(t)$ are selected from the available input data $\boldsymbol{U}^{exp}$. The objective function describes the deviation from the experimental data as

$$\boldsymbol{J}\left(\boldsymbol{Y}^{exp}, \hat{\boldsymbol{Y}}\right) = \frac{1}{n_y} \sum_{i=1}^{n_y} \frac{1}{n_{exp}} \sum_{j=1}^{n_{exp}} \frac{1}{n_{samp}} \sum_{k=1}^{n_{samp}} r^2_{i,j,k}. \tag{5}$$

$r_{i,j,k}$ denotes the residuals

$$r_{i,j,k} = y^{exp}_{i,j,k} - \hat{y}_i(t^{exp}_{j,k}). \tag{6}$$

This approach comes with multiple challenges. Firstly, it is difficult to guess on the one hand the ML-model structure of $\boldsymbol{\varphi}$ and on the other hand the set of descriptors $\boldsymbol{d}$ without any knowledge of the complexity of the relationship it is used to represent. Therefore, multiple iterations of trial and error are necessary, where for each assumed model structure and each set of descriptors the optimization problem shown in (4) has to be solved.
Secondly, this parameter estimation problem is difficult to solve, since the parameters of the ML-model $\boldsymbol{\Theta_{ML}}$ have to be randomly initialized as they have no physical meaning. The set of DAEs might therefore not be stable, which is problematic for the numerical integration and the initial values might lie far from the optimal solution. This together with the fact, that in ML-models usually large parameter sets are used, leads to significant problems.
Therefore, we propose a step-wise procedure motivated by work of Hebing et al. (2020) on gray-box modelling of biotechnological processes based on dynamic metabolic flux analysis (DMFA). An overview over the steps of our methodology is given in Fig. 1.
After determining the gray-box model structure in steps 1 and 2 the modeling process is decomposed by firstly estimating training data for the adaptation of the ML models in step 3. This data defines the values of the ML-model outputs $\boldsymbol{\varphi}$ for given values of states $\boldsymbol{x}$, algebraic

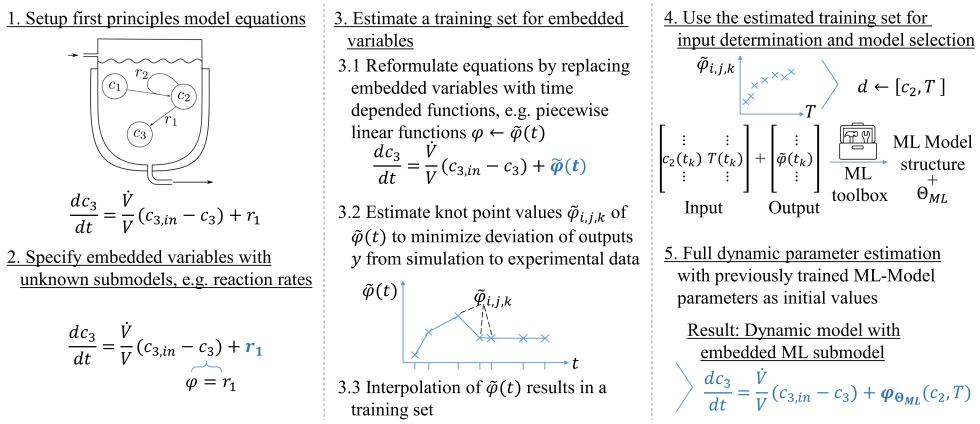

Fig. 1. Visualization of the methodology

variables $\boldsymbol{z}$ and inputs $\boldsymbol{u}$.
Using this set of feature data, the set of descriptors is obtained in step 4 by analyzing the correlation of $\boldsymbol{\varphi}$ to $\{\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}\}$ together with physical understanding. A suitable ML-model structure can be found by using any ML toolbox or automated procedures as hyperparameter optimization. By performing the training of the ML model for the estimated data, a first estimate of the ML model parameters becomes available as a result of step 4.
After the set of descriptors $\boldsymbol{d}$ and the ML-model structure of $\boldsymbol{\varphi}$ have been determined and suitable initial values for the ML-model parameters $\boldsymbol{\Theta_{ML}}$ are available, the full dynamic parameter estimation (4) can be solved relatively easily, which is denoted as step 5.
The rest of this section deals with the details that are involved in the different steps. In section 2.1, the proposed procedure to generate the feature data is described. Section 2.2 entails how regularization can be used to prevent overfitting in this context.

2.1 Time explicit reformulation

For the estimation of the training set the proposed approach follows the one applied in Hebing et al. (2020). To remove the dependency on a priori decisions regarding model structure, discriptors and ML-model parameters, the embedded variables are replaced by counter parts that depend explicitly on time only: $\boldsymbol{\varphi}_{\boldsymbol{\Theta}_{ML}}(\boldsymbol{d}) \rightarrow \tilde{\boldsymbol{\varphi}}(t)$.
A continuous piecewise linear structure is chosen for the dependency of the embedded variables $\tilde{\boldsymbol{\varphi}}(t)$ on time, as shown in (7). This provides more flexibility compared to the method of de Prada et al. (2018), who use a piecewise constant function of time. Using a continuous linear function has two advantages. Firstly, since the embedded variables are modeled as continuous functions, this is closer to the underlying relationship $\boldsymbol{\varphi}_{\boldsymbol{\Theta}_{ML}}(\boldsymbol{d})$. Secondly, if a piecewise linear function is used, interpolation in time can be applied to expand the training set significantly without creating artefacts. Other continuous functions, e.g. higher order polynomials, could also be used but are more prone to overfitting.

$$\tilde{\varphi}_i(t) = \frac{\tilde{\varphi}_{i,j,k+1} - \tilde{\varphi}_{i,j,k}}{t_{j,k+1} - t_{j,k}}(t - t_{j,k}) + \tilde{\varphi}_{i,j,k} \quad t \in [t_{j,k+1}, t_{j,k}] \tag{7}$$

The values at the "knot points" $\tilde{\varphi}_{i,j,k}$ are degrees of freedom, that are represented by the matrix $\tilde{\boldsymbol{\Phi}} = \{\tilde{\varphi}_{i,j,k}\} \in \mathbb{R}^{n_\varphi \times n_{exp} \times n_{samp}}$. They are determined by solving the training set estimation problem:

$$\begin{aligned} \min_{\tilde{\boldsymbol{\Phi}},\, \boldsymbol{\Theta}} \quad & J\left(\boldsymbol{Y}^{exp}, \hat{\boldsymbol{Y}}\right) + \lambda\, Reg\left(\tilde{\boldsymbol{\Phi}}\right) \\ \text{s.t.} \quad & \dot{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{\Theta}}\left(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \tilde{\boldsymbol{\varphi}}(t)\right) \\ & \boldsymbol{0} = \boldsymbol{g}_{\boldsymbol{\Theta}}\left(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{u}, \tilde{\boldsymbol{\varphi}}(t)\right) \\ & x_i(t_{j,0}) = x_{i,j}^{0,exp} \qquad \forall\, i = 1, \ldots, n_x \\ & \qquad\qquad\qquad\qquad\quad \forall\, j = 1, \ldots, n_{exp} \\ & \hat{\boldsymbol{y}} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{z}). \end{aligned} \tag{8}$$

In this optimization problem, the embedded variables $\boldsymbol{\varphi}$ are replaced by the corresponding time-explicit functions $\tilde{\boldsymbol{\varphi}}$.

2.2 Regularization for robustness

As the set of all values at the knot points $\tilde{\boldsymbol{\Phi}}$ contains many degrees of freedom, possibly more than the number of data points in $\boldsymbol{Y}^{exp}$, regularization is applied by adding the term $Reg(\tilde{\boldsymbol{\Phi}})$ to the cost function, weighted by the parameter λ. The regularization is applied for each experiment and for each embedded variable separately as shown in equation (9).

$$Reg\left(\tilde{\boldsymbol{\Phi}}\right) = \frac{1}{n_\varphi} \sum_{i=1}^{n_\varphi} \frac{1}{n_{exp}} \sum_{j=1}^{n_{exp}} reg_{i,j} \tag{9}$$

There exists a multitude of ways to conduct regularization for this type of problems. An overview over some frequently used formulations is shown in Table 1.

Any combination of these terms is possible, similar to elastic net regression, see Fahrmeir et al. (2007). This can lead to favorable results as shown by Scheffold et al. (2021). Due to the fact that the optimization problem (8) is high-dimensional and possibly strongly nonlinear, a methodology for finding good initial values is crucial to avoid getting trapped in local minima and long computation times. We propose to firstly solve (8) for a constant value of $\tilde{\boldsymbol{\varphi}}$ without regularization. The resulting values can be used as suitable initial values. It should be noted that for

Table 1. Overview over commonly used regularization formulations for estimating dynamic data

Kind of regularization	Equation
Differential L2	$reg_{i,j} = \frac{1}{n_{samp}-1} \sum_{k=2}^{n_{samp}} \left(\frac{\tilde{\varphi}_{i,j,k} - \tilde{\varphi}_{i,j,k-1}}{t_{j,k} - t_{j,k-1}} \right)^2$
Differential L1	$reg_{i,j} = \frac{1}{n_{samp}-1} \sum_{k=2}^{n_{samp}} \left\lvert \frac{\tilde{\varphi}_{i,j,k} - \tilde{\varphi}_{i,j,k-1}}{t_{j,k} - t_{j,k-1}} \right\rvert$
Absolute L2	$reg_{i,j} = \frac{1}{n_{samp}} \sum_{k=1}^{n_{samp}} \left(\tilde{\varphi}_{i,j,k} \right)^2$
Absolute L1	$reg_{i,j} = \frac{1}{n_{samp}} \sum_{k=1}^{n_{samp}} \left\lvert \tilde{\varphi}_{i,j,k} \right\rvert$

large sets of experimental data and complex processes, a good initialization is crucial.

The solution of the optimization problem (8) is denoted as $\tilde{\boldsymbol{\Phi}}^*$ and $\boldsymbol{\Theta}^*$. The values $\tilde{\boldsymbol{\Phi}}^*$ correspond to the simulated values of states $\hat{\boldsymbol{X}}^*$ and algebraic variables $\hat{\boldsymbol{Z}}^*$, along with the inputs $\boldsymbol{U}^{exp}$.

A larger training set can be generated by interpolating the values of $\tilde{\boldsymbol{\Phi}}^*$ in time and simulating the differential equations to obtain $\hat{\boldsymbol{X}}^*$ and $\hat{\boldsymbol{Z}}^*$.

After solving the optimization problem in (8), a sensitivity analysis is performed in order to analyze which values in $\tilde{\boldsymbol{\Phi}}$ have a strong influence on the objective function. The main idea behind this approach is to avoid the presence of decision variables that take arbitrary values as they do not change the model fit but influence the following analysis of the model. The set of sensitivities $\{\tilde{\sigma}_{i,j,k}\} = \tilde{\boldsymbol{\Sigma}} \in \mathbb{R}^{n_\varphi \times n_{exp} \times n_{samp}}$ is computed as is shown in (10) by taking the derivative of the residuals, which are defined in (6) with respect to the values at the knot points of the piecewise functions. Taking the absolute values ensures that no cancellation of positive and negative influences occurs.

$$\tilde{\sigma}_{i,j,k} = \bar{\tilde{\varphi}}_i^* \frac{1}{n_y} \sum_{i'=1}^{n_y} \frac{\bar{r}_{i'}}{n_{exp}} \sum_{j'=1}^{n_{exp}} \frac{1}{n_{samp}} \cdot \sum_{k'=1}^{n_{samp}} \left\lvert \frac{\partial r_{i',j',k'}}{\partial \tilde{\varphi}_{i,j,k}} \right\rvert \Bigg|_{\tilde{\boldsymbol{\Phi}}^*, \hat{\boldsymbol{X}}^*, \hat{\boldsymbol{Z}}^*, \boldsymbol{U}^{exp}} \tag{10}$$

Here, $\bar{\tilde{\varphi}}_i^*$, the mean value of the optimal values of the embedded variables is used for normalization, as defined in (11). $\bar{r}_{i'}$ is the mean absolute residual of output i' as shown in (12).

$$\bar{\tilde{\varphi}}_i^* = \frac{1}{n_{exp}} \sum_{j'=1}^{n_{exp}} \frac{1}{n_{samp}} \sum_{k'=1}^{n_{samp}} \tilde{\varphi}_{i,j',k'}^* \tag{11}$$

$$\bar{r}_{i'} = \frac{1}{n_{exp}} \sum_{j'=1}^{n_{exp}} \frac{1}{n_{samp}} \sum_{k'=1}^{n_{samp}} \left| r_{i,j',k'} \right| \Bigg|_{\tilde{\boldsymbol{\Phi}}^*, \hat{\boldsymbol{X}}^*, \hat{\boldsymbol{Z}}^*, \boldsymbol{U}^{exp}} \tag{12}$$

It is worth noting that the derivative is not computed for the second part of the objective function $Reg\left(\tilde{\boldsymbol{\Phi}}\right)$ as the regularization term depends on $\varphi_{i,j,k}$ regardless of its importance for describing the experimental data.

These sensitivities can be applied as a filter to generate a filtered set of values of the embedded variables $\tilde{\boldsymbol{\Phi}}^{*,filt} = \{\tilde{\varphi}_{i,j,k}^* | \tilde{\sigma}_{i,j,k} > \epsilon_i,\ i = 1 \dots n_\varphi,\ j = 1 \dots n_{exp},\ k = 1 \dots n_{samp}\}$. Analogously $\hat{\boldsymbol{X}}^{*,filt}$, $\hat{\boldsymbol{Z}}^{*,filt}$, $\boldsymbol{U}^{exp,filt}$ are obtained.

The sets $\tilde{\boldsymbol{\Phi}}^{*,filt}$, $\hat{\boldsymbol{X}}^{*,filt}$, $\hat{\boldsymbol{Z}}^{*,filt}$, $\boldsymbol{U}^{exp,filt}$ are the basis for finding a relationship for $\boldsymbol{\varphi}(\boldsymbol{d})$. Any ML-toolbox and correlation analysis tool can be used to this end.

To improve the model accuracy, the obtained parameters of the ML model $\boldsymbol{\Theta}_{ML}$ are used as initial values to solve the full dynamic parameter estimation problem (4) with the chosen ML model structure.

3. SIMULATION STUDY: FERMENTATION OF A SPORULATING MICROORGANISM

In this work we consider the case study of the fermentation of a sporulating Bacillus micro organism. Neglecting both (side-)product formation and the time delay between the beginning and the end of the sporulation as well as concentrating only one limiting substrate, the process can be described using three state variables

- X_v: Concentration of vegetative cells that undergo both growth and sporulation,
- S: Concentration of the limiting substrate that is needed for the growth reaction and
- X_s: Concentration of sporulated cells that are the product of the sporulation process.

The overall reaction system can be summarized as shown in (13)

$$S \xrightarrow[\text{growth}]{r_g} X_v \xrightarrow[\text{sporulation}]{r_s} X_s. \tag{13}$$

Here, r_g denotes the growth and r_s the sporulation reaction rate. With these reaction rates and states, a system of ordinary differential equations (ODE) arises, which is discussed in the next section.

3.1 Differential model of the case study

Introducing stoichiometry, the system of ordinary differential equations can be inferred as shown in (14)-(16).

$$\dot{X}_v = r_g - r_s \tag{14}$$

$$\dot{S} = -r_g \, Y_{X/S}^{-1} \tag{15}$$

$$\dot{X}_s = r_s, \tag{16}$$

The stoichiometry is here introduced by the yield coefficient $Y_{X/S}$. The two reaction rates r_g and r_s are described by introducing inhibition terms as shown in (17)-(18)

$$r_g = \mu_{max} \, \tilde{\mu}_T(T) \, \tilde{\mu}_S(S) \, X_v \tag{17}$$

$$r_s = k_{s,max} \, \tilde{k}_{s,T}(T) \, \tilde{k}_{s,S}(S) \, X_v. \tag{18}$$

Here, μ_{max} describes the maximum value of the reaction rate constant, which is inhibited by both the temperature T and the substrate concentration S as described by the functions $\tilde{\mu}_T(T)$ and $\tilde{\mu}_S(S)$. Similarly, the inhibition of the maximum sporulation rate $k_{s,max}$ is described with $\tilde{k}_{s,T}(T)$ and $\tilde{k}_{s,S}(S)$. Both reaction rates are assumed to depend on the vegetative cell concentration with first order kinetics as shown in (17)-(18).

The measured outputs are provided by three different measurements: the total cell concentration X_t and the concentrations of the spores and of the substrate. Therefore the function $\boldsymbol{h}$ is linear in this case and defined as follows:

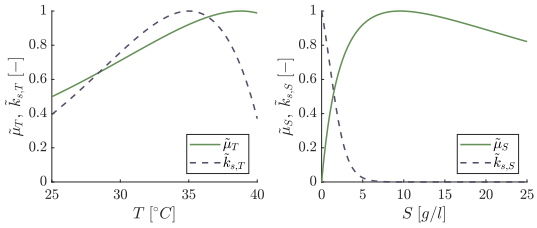

Fig. 2. Inhibition terms for the sporulation rate (dashed line) and the growth rate (full line)

$$\begin{bmatrix} X_t \\ X_s \\ S \end{bmatrix} = \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_v \\ X_s \\ S \end{bmatrix} \tag{19}$$

The following section describes the inhibition terms regarding temperature and substrate in more detail.

3.2 Temperature and substrate inhibition

The temperature inhibition of the growth reaction is modeled by the general assumption of a positive and a negative influence as shown in (20), taken from Bastin and Dochain (1990)

$$\tilde{\mu}_T(T) = a_1 \exp\left(\frac{-E_1}{RT}\right) - a_2 \exp\left(\frac{-E_2}{RT}\right). \tag{20}$$

In this equation, a_1, a_2, E_1 and E_2 are parameters while R denotes the universal gas constant.
For sporulation rate inhibition, a model from Baril et al. (2012) is used, which depends on the minimum, maximum and optimum temperatures T_{min}, T_{max} and T_{opt} as well as on a shape parameter n.

$$\tilde{k}_{s,T}(T) = \frac{(T - T_{max})(T - T_{min})^n}{(T - T_{opt})^{n-1}\left((T_{opt} - T_{min})(T - T_{opt}) - a_T\right)} \tag{21}$$

with

$$a_T = (T_{opt} - T_{max})\left((n-1)T_{opt} + T_{min} - nT\right).$$

Both functions of temperature and substrate inhibition are shown in Fig. 2.

In this figure, it can be seen that both temperature inhibition terms show a local maximum in the temperature range of 25 to 40 °C, the sporulation rate decreases substantially at both high and low temperatures. The temperature of maximum growth is higher than for maximum sporulation. The inhibition of the substrate concentration is modeled as shown in (22)-(23) with relations taken from Das and Sen (2011) and Atehortúa et al. (2007).

$$\tilde{\mu}_S(S) = \frac{S}{1 + \sum_{i=1}^{n} A_i S^i} \tag{22}$$

$$\tilde{k}_{s,S}(S) = \frac{1}{1 + e^{G_S(S - P_S)}} - \frac{1}{1 + e^{G_S(S' - P_S)}} \tag{23}$$

The resulting functions are shown in Fig. 2.

The growth rate vanishes at low substrate concentrations and saturates at around 10 g/l. With more substrate, an inhibition of the growth is observed.
For the sporulation rate, the maximum is observed when no substrate is present, as this induces stress in the cell metabolism leading to a high rate of sporulation. Even at low substrate concentrations of about 5 g/l, the sporulation rate is strongly inhibited.

4. RESULTS

The methodology presented in section 2 is applied to the simulated case study from section 3. Data was generated by performing 5 virtual experiments, $n_{exp} = 5$. In each of these experiments, the differential equations (14)-(16) were simulated from a random initial composition and temperatures that were random but constant for each experiment. For the generation of the initial values, the concentration of sporulated cells X_s was kept at zero, while both the concentrations of vegetative cells and substrate were varied in realistic ranges. 20 samples were collected for each experiment, thus: $n_{samp} = 20$. After the simulation, the output values were disturbed by simulated measurement noise computed from (24) with additive normally distributed noise as shown in (25).

$$\sigma^2_{meas}(y^{exp,no\,noise}_{i,j,k}) = (\alpha + \beta\, y^{exp,no\,noise}_{i,j,k})^2 \tag{24}$$

$$y^{exp}_{i,j,k} = y^{exp,no\,noise}_{i,j,k} + \mathcal{N}(0, \sigma^2_{meas}) \tag{25}$$

To create a dynamic gray-box model that describes the obtained experimental data accurately, the following model structure is proposed.

$$\dot{\hat{X}}_v = \varphi_1\, \hat{X}_v\, \hat{S} - \varphi_2\, \hat{X}_v \tag{26}$$

$$\dot{\hat{S}} = -\nu\, \varphi_1\, \hat{X}_v\, \hat{S} \tag{27}$$

$$\dot{\hat{X}}_s = \varphi_2\, \hat{X}_v \tag{28}$$

This is a gray-box model structure as domain knowledge is integrated into these equations:

- the presence of three states: $\hat{X}_v$, $\hat{X}_s$, $\hat{S}$
- the presence of two reactions: $\hat{r}_g = \varphi_1\, \hat{X}_v\, \hat{S}$, $\hat{r}_s = \varphi_2\, \hat{X}_v$
- a first order dependency of the growth reaction wrt. vegetative cells and substrate
- a first order dependency of the sporulation reaction wrt. vegetative cells

What is missing in this model is the kinetic relationship that is described by the embedded variables φ_1 and φ_2, i.e. how these variables depend on the states and the inputs, in this case the temperature. Additionally, the value of the stoichiometric constant ν is unknown and therefore represents the set of parameters Θ.
The following sections describe how, using the proposed methodology, knowledge about these relationships can be obtained.

4.1 Effect of the regularization on the training set

In order to estimate a training set for analysing the embedded variables φ_1 and φ_2, the optimization problem in (8) is setup in CasADi (Andersson et al. (2019)) using CVODES (Hindmarsh et al. (2005)) and solved to get $\tilde{\boldsymbol{\Phi}}^*$ and $\boldsymbol{\Theta}^*$, along with the corresponding set of inputs, states and outputs. The results of this optimization problem are shown in Fig. 3. Every set of colored lines corresponds to solving the optimization problem once, with varying values of λ. To keep the values of λ comparable, λ^2 is applied, when L2 regularization is considered.
In this figure, in the top row the predicted trajectories of the three outputs are shown. All values of λ give rise

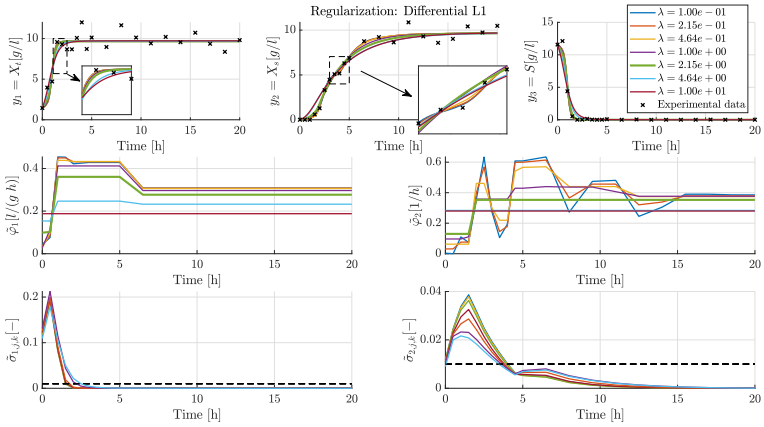

Fig. 3. Data estimation step, top row: experimental and modeled output values for different values of λ, middle row: piecewise linear functions of time with which the embedded variables $\tilde{\varphi}_1$ and $\tilde{\varphi}_2$ are described, bottom row: sensitivites of the knot points of the piecewise linear functions, the black line denotes the threshold for filtering

to reasonable trajectories for the outputs X_t and S. These two outputs are only influenced by the first embedded variable φ_1. The output trajectory varies more for lower values of λ, which is to be expected. With larger penalization of changes in the values of φ, these take lower values in general, which can be seen in the figure in the middle row on the left. With the largest tested regularization value $\lambda = 1.00e+01$, a flat response of $\tilde{\varphi}_1$ results.

For the concentration of sporulated cells X_s a strong variation in the prediction for different values of λ can be observed. Not all values of λ lead to a satisfactory result. In fact, low values as e.g. $1.00e-01$ to $4.64e-01$ lead to a significant overfitting, which can be concluded from the fact that the concentration of sporulated cells rises in multiple steps which follow the measurements corrupted by noise closely. For each of the steps the value of $\tilde{\varphi}_2$ spikes from low to high values or back at about 2, 3, 4, 8 and 13 h. This is not physically reasonable, therefore this is considered as an unwanted behavior. On the other end, a high value of λ leads to an approximately constant value of $\tilde{\varphi}_2$, which is also not desired, as this results in a trajectory of X_s, that does not follow the sharp observed increase, thus underfits the data.

Therefore, a value of λ is chosen that prevents overfitting, while not leading to underfitting as well. In this case, a value of $\lambda = 2.15e+00$ is chosen.

This decision is supported by a visualization of the mean squared error of the experimental data (MSE, calculated from $J(\boldsymbol{Y}^{exp}, \hat{\boldsymbol{Y}})$) to model complexity ($Reg(\tilde{\boldsymbol{\Phi}})$) tradeoff, shown in Fig. 4

In this figure, one can see that for the different types of regularization, different values of the tradeoff between MSE and the regularization $Reg(\tilde{\boldsymbol{\Phi}})$ result. The shown values can be interpreted as the Pareto front of the multi-objective optimization to minimize both the MSE of the experimental data and the model complexity. The closer the Pareto front gets to the utopia point the better, as this denotes the point, where both objectives are independently minimal. All methods show the expected behavior, but absolute L1 regularization shows unwanted oscillatory trajectories except for high values of λ. Note, that the values of $Reg(\tilde{\boldsymbol{\Phi}})$ cannot be directly compared between the regularization methods.

To understand the different regularization methods, the

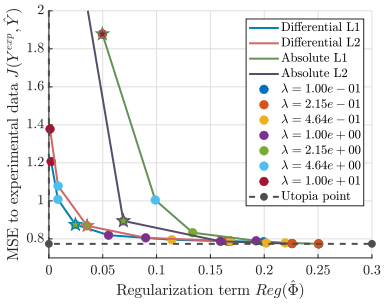

Fig. 4. Plot of model complexity vs. deviation from experimental data for different regularization weighting factors λ, the star denotes that this value of λ is chosen for further studies

chosen values of λ are applied and the resulting trajectories of the sporulated cell concentrations together with its derivatives are shown in Fig. 5.

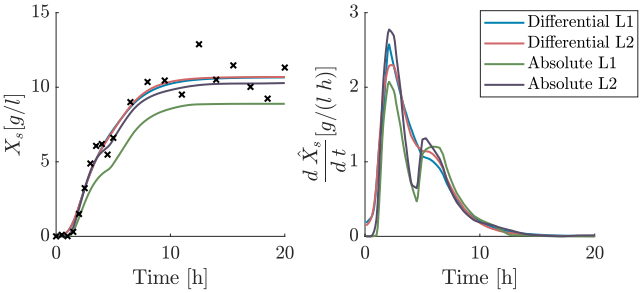

Fig. 5. Effect of choosing a regularization method, left: predicted and experimental spore concentration, right: derivative of the prediction wrt. time

Here, the previous result that the absolute L1 regularization is not working well is also visible. The strong regularization of the absolute values of $\tilde{\varphi}$ leads to no sufficient sporulation. Thus, the final spore concentration underestimates the experimental value.

The other methods seem to produce similar trajectories of X_s, though when visualizing the derivative of the output trajectory it becomes apparent, that a slight overfitting can be observed for the absolute L2 method. The differential L1 and L2 regularizations lead to physically appropriate data sets. Here, differential L1 regularization

is considered to work best due to the higher peak and slightly smaller s-shaped behavior of the derivative at 6 h, and therefore used for next steps.
From the plotted sensitivities in the bottom row of Fig. 3 it can be seen that all values of $\tilde{\varphi}_1$ after approx. 3 h and almost all values of $\tilde{\varphi}_2$ after approx. 4 h have a low impact on the prediction of the experimental data, which makes these values unreliable. These values are mostly influenced by the regularization, which is why no further significant change in these values is visible. Therefore, all values of $\tilde{\varphi}$ with a sensitivity less than 0.01 are omitted.
With the sensitivity filter applied, the final training set is finalized. An interpolation is applied to obtain 5 additional training points between the sample times, which are filtered according to interpolated sensitivities.

4.2 Feature and model selection

With the obtained training set, feature and model selection can be performed. Feature selection denotes in this case the determination of the set of descriptors $\boldsymbol{d}$. To determine the set of descriptors, the correlation plots of the resulting embedded variables to the states and the input are shown in Fig. 6.

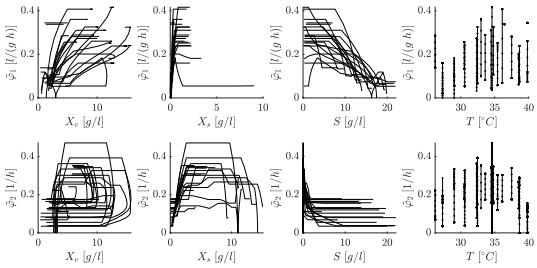

Fig. 6. Correlation plots of all states (first three columns) and the input (fourth column) to the estimated values of the embedded variables, dots are used for the visualization of the distribution

It can be seen that the two embedded variables show multiple correlations. For instance, $\tilde{\varphi}_1$, which governs the growth reaction, seems to be correlated to all states and inputs. This is due to the fact that the states themselves are not independent as they change according to the differential equations. The strongest correlations of $\tilde{\varphi}_1$ among the three state variables seems to be to the substrate concentration S.
For the embedded variable that describes the rate of sporulation, $\tilde{\varphi}_2$, the strongest correlation can also be observed for the substrate concentration, the inverse relationship is clearly visible. The fact, that the rate does not go to zero might be a result of the application of differential regularization. Changing the sporulation rate to zero comes with a regularization penalty in the cost function of (8), and might have a low influence on the change of the spore concentration, as the concentration of vegetative cells also approaches zero, which is a factor in the sporulation reaction kinetics.
For both embedded variables, a correlation to temperature can be observed. Therefore, temperature T and substrate concentration S are chosen as the set of descriptors $\boldsymbol{d}$. In the correlation plot of $\tilde{\varphi}_1$ and X_s one trajectory stands out. This might be the result of insufficient filtering. Since this phenomenon only seems to occur in this one experiment, the overall impact is assumed to be negligible. Additionally, spore and vegetative cell concentration can be disregarded anyways from the correlation plot and from the physical understanding that spores are inactive cells and that all reactions are modeled specific to the vegetative cells already. It should be noted that in cases where these physical relationships are not apparent, the correlation of the state variables among themselves can make it difficult to determine $\boldsymbol{d}$.
With the descriptors $\boldsymbol{d}$ determined, the training of an ML model $\boldsymbol{\varphi}_{\boldsymbol{\Theta}_{ML}}$ can be performed. Here, a simple artificial neural network with 1 hidden layer and 4 nodes is used with the *tanh* activation function. Training on the estimated data is done using the Levenberg-Marquardt algorithm with early stopping after 6 iterations of no further progress on a validation set. The resulting regression plot is shown in Fig. 7.

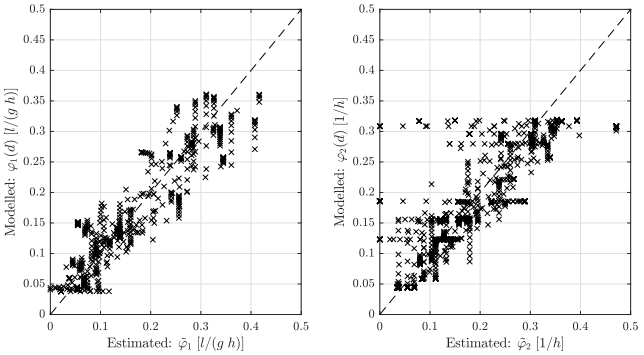

Fig. 7. Regression plot of the two embedded variables

In this figure it can be seen that the regression model gives reasonable results, even though there are some deviations between the model and the estimated data. Some horizontal and vertical clusters of points lead to the conclusion that the data estimation problem results in imperfect training data, due to strong measurement noise. As afterwards a full dynamic parameter estimation is performed, this is acceptable at this point in the procedure.

4.3 Full dynamic parameter estimation

The full dynamic parameter estimation problem, as described in (4), can be solved for the model structure, the set of descriptors and the estimated model parameters from section 4.2. The resulting model is already validated structurally. The final model predictions with conditions that were not used during training are shown in the following figure. Here, also results for a black-box model are shown, where the entire right-hand side of the ODE is described by ANN models. The parameters were trained using the full dynamic parameter estimation routine.

From Fig. 8 it can be inferred that both versions of the gray-box model show physically feasible and reasonable trajectories as opposed to the black-box model predictions. Also, even though there are some deviations in the regression shown in Fig. 7, the resulting trajectories, shown as dashed lines, approximate the experimental data quite

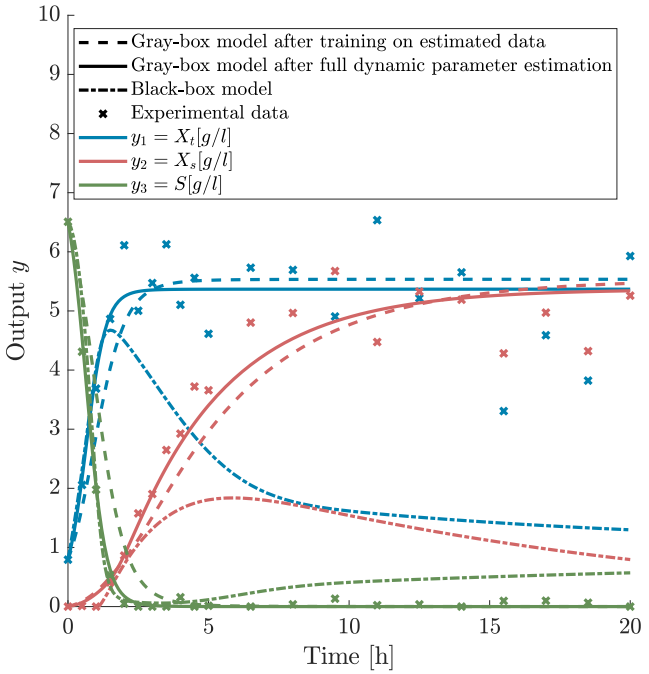

Fig. 8. Simulation results of the full dynamic models after parameter estimation for all outputs and for different models

well. This validates the data estimation step, the ML-model parameters that are obtained by this preliminary regression are already reasonably accurate and provide good initial values for the full dynamic parameter estimation problem. The results of the latter are depicted in Fig. 8 as solid lines. The final dynamic model accurately describes the system dynamics.

5. CONCLUSION AND OUTLOOK

We have proposed a comprehensive methodology for determining dynamic gray-box process models of complex dynamic processes and have shown the performance of the methodology for a simulation case of the fermentatation of a sporulating bacterium. The methodology consists of three steps. First, training data for the embedded variables, that are later described by ML-models, is estimated. This is used in the second step for model selection and training. Third, full scale dynamic parameter estimation is performed to fine-tune the parameters, using the previously obtained results for initialization.

The discussion focused on choosing an appropriate regularization method and on the use of sensitivity analysis to increase the reliability of the estimated data. This is especially important in systems where the embedded variables are multiplied with vanishing states, which commonly occurs in chemical and biochemical systems, because this causes the sensitivity and thus the reliability to decrease significantly.

Further research is needed to make the methodology scalable to large sets of experimental data. This is currently a challenge, as the set of decision variables in the estimation step increases linearly with the number of experimental data points. Distributed optimization is a promising direction to this end.

REFERENCES

Andersson, J.A.E., Gillis, J., Horn, G., Rawlings, J.B., and Diehl, M. (2019). Casadi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1), 1–36. doi: 10.1007/s12532-018-0139-4.

Atehortúa, P., Alvarez, H., and Orduz, S. (2007). Modeling of growth and sporulation of bacillus thuringiensis in an intermittent fed batch culture with total cell retention. *Bioprocess and Biosystems Engineering*, 30(6), 447–456. doi:10.1007/s00449-007-0141-0.

Baril, E., Coroller, L., Couvert, O., El Jabri, M., Leguerinel, I., Postollec, F., Boulais, C., Carlin, F., and Mafart, P. (2012). Sporulation boundaries and spore formation kinetics of bacillus spp. as a function of temperature, ph and a(w). *Food microbiology*, 32(1), 79–86. doi: 10.1016/j.fm.2012.04.011.

Bastin, G. and Dochain, D. (1990). *On-line estimation and adaptive control of bioreactors*.

Das, S. and Sen, R. (2011). Kinetic modeling of sporulation and product formation in stationary phase by bacillus coagulans rk–02 vis-à-vis other bacilli. *Bioresource Technology*, 102(20), 9659–9667.

de Prada, C., Hose, D., Gutierrez, G., and Pitarch, J.L. (2018). Developing grey-box dynamic process models. *IFAC-PapersOnLine*, 51(2), 523–528. doi: 10.1016/j.ifacol.2018.03.088.

Fahrmeir, L., Kneib, T., and Lang, S. (2007). *Regression: Modelle, Methoden und Anwendungen*. Statistik und ihre Anwendungen. Springer Berlin Heidelberg, Berlin, Heidelberg.

Halmschlager, V., Koller, M., Birkelbach, F., and Hofmann, R. (2019). Grey box modeling of a packed-bed regenerator using recurrent neural networks. *IFAC-PapersOnLine*, 52(16), 765–770. doi: 10.1016/j.ifacol.2019.12.055.

Hebing, L., Neymann, T., and Engell, S. (2020). Application of dynamic metabolic flux analysis for process modeling: Robust flux estimation with regularization, confidence bounds, and selection of elementary modes. *Biotechnology and Bioengineering*, 117(7), 2058–2073. doi:10.1002/bit.27340.

Hindmarsh, A.C., Brown, P.N., Grant, K.E., Lee, S.L., Serban, R., Shumaker, D.E., and Woodward, C.S. (2005). Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*, 31(3), 363–396.

Niu, D., Jia, M., Wang, F., and He, D. (2013). Optimization of nosiheptide fed-batch fermentation process based on hybrid model. *Industrial & Engineering Chemistry Research*, 52(9), 3373–3380. doi:10.1021/ie3022169.

Scheffold, L., Finkler, T., and Piechottka, U. (2021). Gray-box system modeling using symbolic regression and nonlinear model predictive control of a semibatch polymerization. *Computers & Chemical Engineering*, 146, 107204. doi:10.1016/j.compchemeng.2020.107204.

Sohlberg, B. and Sernfält, M. (2002). Grey box modelling for river control. *Journal of Hydroinformatics*, 4(4), 265–280. doi:10.2166/hydro.2002.0026.

Tulleken, H.J. (1993). Grey-box modelling and identification using physical knowledge and bayesian techniques. *Automatica*, 29(2), 285–308. doi:10.1016/0005-1098(93)90124-C.