

Article

Robust Model Selection: Flatness-Based Optimal Experimental Design for a Biocatalytic Reaction

Moritz Schulze ^{1,2} and René Schenkendorf ^{1,2,*} 

¹ Institute of Energy and Process Systems Engineering, Technische Universität Braunschweig, Franz-Liszt-Straße 35, 38106 Braunschweig, Germany; mo.schulze@tu-braunschweig.de

² Center of Pharmaceutical Engineering, Technische Universität Braunschweig, Franz-Liszt-Straße 35a, 38106 Braunschweig, Germany

* Correspondence: r.schenkendorf@tu-braunschweig.de

Received: 29 November 2019; Accepted: 27 January 2020; Published: 5 February 2020



Abstract: Considering the competitive and strongly regulated pharmaceutical industry, mathematical modeling and process systems engineering might be useful tools for implementing quality by design (QbD) and quality by control (QbC) strategies for low-cost but high-quality drugs. However, a crucial task in modeling (bio)pharmaceutical manufacturing processes is the reliable identification of model candidates from a set of various model hypotheses. To identify the best experimental design suitable for a reliable model selection and system identification is challenging for nonlinear (bio)pharmaceutical process models in general. This paper is the first to exploit differential flatness for model selection problems under uncertainty, and thus translates the model selection problem to advanced concepts of systems theory and controllability aspects, respectively. Here, the optimal controls for improved model selection trajectories are expressed analytically with low computational costs. We further demonstrate the impact of parameter uncertainties on the differential flatness-based method and provide an effective robustification strategy with the point estimate method for uncertainty quantification. In a simulation study, we consider a biocatalytic reaction step simulating the carbonylation of aldehydes, where we successfully derive optimal controls for improved model selection trajectories under uncertainty.

Keywords: model selection; model-based design of experiments; differential flatness; parameter uncertainty; point estimate method; nonlinear programming

1. Introduction

Quality by design (QbD) and quality by control (QbC) are critical aspects in pharmaceutical manufacturing, aiming for low-cost but high-quality drugs [1]. For research and development costs in the competitive and strongly regulated pharmaceutical industry have been rising over recent decades leading to shrinking profit margins [2], QbD and QbC have attracted much notice. Mathematical models and process systems engineering might be useful tools for implementing effective QbD/QbC strategies, which is particularly true for the rapidly growing biopharmaceutical market [3–5]. A crucial task in modeling biopharmaceutical manufacturing processes is the reliable identification of model parameters and proper model candidates from a set of various model hypotheses. Model-based design of experiment (MBDoE) approaches, where the challenge of precise parameter identification and reliable model selection is formulated as an optimization problem, have proven beneficial over the last few decades [6–9]. Please note that the literature distinguishes between statistical design of experiment and MBDoE approaches. Methods of the former as response surface methods are popular and commonly used because of their simplicity, but, for the same reason, are not able to sufficiently handle complex dynamic systems [10]. On the other hand, the MBDoE approach frequently aims for

optimal control actions, where commonly used numerical methods give rise to nonlinear programming problems that need substantial amounts of computational time and efficient solvers [11]. Moreover, robustification against model parameter uncertainties further complicates the problem, as statistical quantities must be considered when solving the underlying optimization problem. In addition to the actual MBDoE optimization framework and data acquisition, recent studies have shown that a well-posed parameter identification and model selection problem is equally important [12–14]. In this context, the consideration of input residuals in parameter estimation beyond the classical approach of output residuals has drawn attention in the literature [13–16]. For an effective solution to the robust model selection problem, we aim at implementing an inverse modeling technique by using the differential flatness concept [17]. In systems theory, differential flatness implies that analytical expressions of the system's states and controls exist that are functions of so-called flat outputs and their derivatives. Hence, by implementing a differential flatness strategy to solve an MBDoE problem, we can avoid solving differential and sensitivity equations numerically as required by standard control vector parameterization strategies for model selection. Please also note that while optimizing the flat output trajectory, the number of continuous optimization variables is reduced to the number of flat outputs, i.e., an additional reduction of computational costs as compared to standard nonlinear programming techniques with control and state variable discretization [18]. Methods based on differential flatness have been heavily used in research and industry—primarily in the design of open-loop controllers and the planning of state trajectories in (electro)mechanical systems [19]. In the case of the MBDoE technique, strategies exploiting differential flatness have barely been considered [15,16,20,21], and—to our knowledge—have not been applied to model selection problems under uncertainty before. Thus, this study addresses two open research challenges: (i) to apply the flatness concept to multi-model problems, and (ii) to integrate parameter uncertainties into the flatness-based optimization problem for robust model selection. In detail, we aim at discarding inappropriate model candidates from a set of competing model hypotheses. To this end, we show that the considered model candidates satisfy the differential flatness condition, and we solve an optimization problem under uncertainty that makes use of the flatness property. For the sake of illustration, we consider a biocatalytic process from a carbonylation reaction system that forms an essential precursor in the synthesis of natural products and pharmaceuticals [22]. Please note that carbon-carbon bond-forming reactions are the backbone of numerous high-value molecules in industrial organic synthesis. However, biocatalytic production of the related bulk and fine chemicals and active pharmaceutical ingredients (APIs) is insufficiently explored today. Therefore, the application of enzymes in organic synthesis is currently an important research topic to exploit the significant potential of improving manufacturing processes in the chemical and pharmaceutical industry under the agenda of green chemistry [23,24]. An essential problem with all mathematical models and model-based design concepts, including MBDoE, is the uncertainty in model parameters. To ensure reliable inferences, parametric uncertainty should be taken into account in the MBDoE approach [6,14,25,26]. In general, probability-based concepts have attracted considerable attention in robust optimization. In the literature, between classical and Bayesian settings for handling model predictive errors as a potential consequence of uncertain model parameters can be distinguished. In the former, the strategies have in common that the model output differences in the objective function are weighted by an error covariance matrix of the model prediction to avoid experimental regions where model predictions are poor [7,27–29]. One drawback of these strategies is that the prediction error covariance matrix is calculated from local sensitivities, where designs of experiments for local sensitivities might provide misleading information compared to designs based on global sensitivities [30]. On the other hand, Bayesian model selection with consideration of parametric uncertainties requires the determination of computationally expensive integrals, in particular, if the recommended bias-free numerical evaluation path is chosen [31,32]. More broadly speaking, uncertainty propagation and quantification still pose considerable challenges in computational efficiency and numerical accuracy. In this context, we propose to diminish this computational burden by using an efficient and effective rule to determine statistical moments

circumventing a local approximation, namely the point estimate method (PEM). The PEM has proven beneficial in various engineering problems [33], including robust optimization problems in complex (bio)chemical process design [3,34].

The paper is organized as follows. In Section 2, we cover the basics of the inverse modeling and differential flatness concept. In Section 3, we introduce the flatness-based MBDoE technique with the PEM for its robustification under parameter uncertainties. In Section 4, we apply the proposed MBDoE concept to a biocatalytic reaction network, and we discuss the results in Section 5. Conclusions can be found in Section 6.

2. Inverse Modeling

In process systems engineering, dynamic process models are often mathematically described with ordinary differential equations (ODEs). Their state-space formulation is

$$\frac{dx}{dt} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}), \quad (1)$$

where t is the time, $\mathbf{x}(t) \in \mathbb{R}^n$ is the system states, $\mathbf{u}(t) \in \mathbb{R}^m$ is the system inputs, and $\boldsymbol{\theta} \in \mathbb{R}^p$ is the parameter vector. The nonlinear function $\mathbf{f}: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ represents the vector field of the dynamic system. The output function is defined as

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}), \quad (2)$$

with the system output vector $\mathbf{y}(t) \in \mathbb{R}^q$ and $\mathbf{h}: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}^q$ as the (non-)linear output function. Please note that in many cases, the system outputs are a (linear) function of the system states only; that is, $\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t))$. Equations (1) and (2) form the process model, \mathcal{M} . Please note that in the following, time and parameter dependencies are specified only when the context requires it.

As opposed to solving the ODE system (Equation (1)) in a forward manner to conventionally exploit the outputs \mathbf{y} given the inputs \mathbf{u} , inverse modeling aims at reconstructing the inputs \mathbf{u} for given outputs \mathbf{y} [35–37]. Please note that the process of inverse simulation is not to be confused with inverse problems, where in the latter the goal is to estimate unknown model parameters $\hat{\boldsymbol{\theta}}$ [35]. The different principles of forward simulation, inverse modeling, and the inverse problem of parameter identification are visualized in Figure 1.

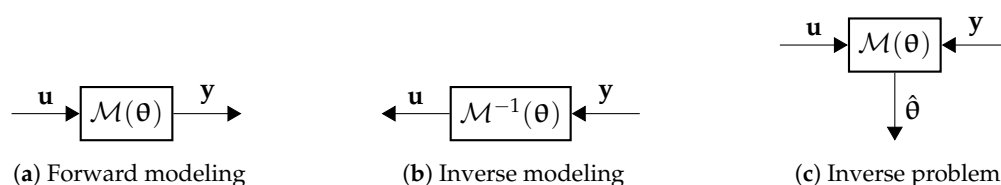


Figure 1. Illustration of the different principles of a standard process simulation configuration (a), the inverse modeling setting to reconstruct system inputs (b), and the inverse problem for parameter identification (c).

In addition to numerical inversion approaches as model inversion- or inverse simulation-based techniques [38], algebraic methods have been proposed for inverse modeling [39], among which strategies based on differential flatness are important representatives.

2.1. Differential Flatness

In systems theory, within the context of algebraic model inversion techniques and aiming at feedforward control problems, differential flatness was introduced by Fliess et al. [17] in 1992.

A (non-)linear process model (Equations (1) and (2)) is called differentially flat or shortly flat if there exists an output vector

$$\mathbf{y}^{\text{flat}} = \mathbf{h}^{\text{flat}}(\mathbf{x}, \mathbf{u}, \dot{\mathbf{u}}, \dots, \mathbf{u}^{(s)}, \boldsymbol{\theta}), \quad (3)$$

with a finite value $s \in \mathbb{N}$ and the smooth mapping function $\mathbf{h}^{\text{flat}} : \mathbb{R}^n \times (\mathbb{R}^m)^{s+1} \times \mathbb{R}^p \rightarrow \mathbb{R}^q$, referred to as a flat output which fulfills the following conditions:

1. The states and the controls can be described as a function of the flat output and its derivatives:

$$\mathbf{x} = \boldsymbol{\Psi}_x(\mathbf{y}^{\text{flat}}, \dot{\mathbf{y}}^{\text{flat}}, \dots, \mathbf{y}^{\text{flat}(r)}, \boldsymbol{\theta}), \quad (4)$$

$$\mathbf{u} = \boldsymbol{\Psi}_u(\mathbf{y}^{\text{flat}}, \dot{\mathbf{y}}^{\text{flat}}, \dots, \mathbf{y}^{\text{flat}(r+1)}, \boldsymbol{\theta}), \quad (5)$$

with the smooth mapping functions $\boldsymbol{\Psi}_x : (\mathbb{R}^m)^{r+1} \times \mathbb{R}^p \rightarrow \mathbb{R}^n$ and $\boldsymbol{\Psi}_u : (\mathbb{R}^m)^{r+2} \times \mathbb{R}^p \rightarrow \mathbb{R}^m$.

2. The dimensions of the control and the flat output vector are equal:

$$\dim \mathbf{y}^{\text{flat}} = \dim \mathbf{u}. \quad (6)$$

Here, $r \in \mathbb{N}$ specifies the number of occurring derivatives. Generally speaking, r is not known beforehand, apart from single-input single-output (SISO) systems, where $r = n - 1$ [40]. There is an infinite number of flat output candidates, and often, they are a function of the states only [40]—similar to the fact that the real output function of a dynamic system is, in many cases, exclusively a function of the states.

Flat outputs might have no direct physical meaning, or they might be identical to measurable quantities; then also referred to as flat inputs [41]. Moreover, between local and global flatness must be distinguished, where for the local form, the differential flatness is valid only for a restricted domain bound by occurring singularities [42]. For certain kinds of singularities, a globally flat system can be designed following a superposition concept. To this end, individual local model inversion domains are created from different local flat outputs where the union of the local domains provides a global framework for the flatness-based model inversion [42]. Alternatively, a transformation approach exists to bypass points of a singularity [43]. In addition to differential flatness of ODE systems, flatness-based methods are also being used for discrete-time systems and partial differential equations (PDEs); see [44–46], and references within. Application scenarios based on differential flatness cover predominantly (electro)mechanical problems, but also process technologies like heat exchangers [47], crystallizers [48] or (bio)reactors [45,49], lithium-ion batteries [50], and more.

2.2. Flat Output Identification

To implement a flatness-based concept, two key challenges exist: First, determining if a system is differentially flat, and second, constructing a flat output alternatively. There is no general method for either of those challenges. However, exceptions exist, and include systems that are linearizable by static feedback control, which are flat by definition. Considering a linear system, controllability conditions differential flatness and vice versa [43]. Methods of determining flat outputs and the mapping function \mathbf{h}^{flat} , respectively, are currently being researched, and the interested reader is referred to [19,51,52] and the references therein. Moreover, the construction of so-called flat inputs (i.e., reconstructed inputs based on the given output function (Equation (2))) was shown for SISO systems and a limited set of multi-input and multi-output (MIMO) cases [53].

Alternatively, in systems theory, valuable information about system characterization based on the interaction of inputs, system states and outputs can also be extracted from graph theory [39,54,55]. Typically, observability or controllability measures are derived for (non)linear state-space models using directed graphs (digraphs) [56–58]. A digraph $D = [V, E]$ with $n + m + q$ vertices $v_i \in V$ and edges $e \in E$ can be constructed from adjacency matrices A_u , A_x , and A_y of the process model for the inputs,

system states, and outputs, respectively. The a_{ij} -th element of A_x or A_u is set to 1 if derivatives $\frac{\partial f_i(\mathbf{x}, \mathbf{u})}{\partial x_j}$ or $\frac{\partial f_i(\mathbf{x}, \mathbf{u})}{\partial u_j}$ exist, respectively, and to 0 if this is not the case. If these derivatives exist (i.e., $\partial f_i / \partial x_j \neq 0$ or $\partial f_i / \partial u_j \neq 0$), then there is an edge from vertex v_j to vertex v_i . Analogously, the adjacency matrix for the outputs \mathbf{y} and the output function $h(\mathbf{x}, \mathbf{u})$ can be defined. Regarding the flatness property, in the paper by Csercsik et al. [55], an algorithm for flatness analysis of MIMO systems is introduced that uses an *explicit expressibility graph*. Please note that from a given digraph, the explicit expressibility graph is formed with reversed edge directions, with the outputs replaced by the flat outputs, and with the self-loop system states dependencies omitted. The following theorem then gives a necessary condition for differential flatness based on the adjacency matrices A_u , A_x , and A_y [55].

Theorem 1. *For a prospective set of flat output-input pairs, m pairwise disjoint paths must exist in the explicit expressibility graph, the union of which covers each vertex of the graph.*

In summary, existing construction methods are often restrictive and require cumbersome calculations. In practice, for the identification of flat outputs (Equation (3)) a sequential strategy including an expert guessing for a flat output candidate followed by a subsequent validation step in terms of Equations (4)–(6) has proven favorable. This trial-and-error approach is facilitated by two facts. First, numerous technical systems are indeed flat systems. Second, frequently, comparable to Lyapunov functions in control theory, informative flat outputs have physical signification [40]. Once the flat output configuration is identified, the flat output functions y^{flat} must be parameterized using for instance spline functions.

2.3. Basis Splines

The flat outputs y^{flat} have to be parameterized while limited by the fact that they have to satisfy solutions of the dynamic system (Equations (1) and (2)). For flat outputs, basis functions from a large set of possibilities, ranging from simple polynomials to more complex functions, can be chosen. Basis splines, i.e., piece-wise polynomial functions also known as B-splines, offer great freedom of action, are flexible and well-studied, and libraries for their usage are implemented in many different programming languages [59]. The classical application of splines is the approximation and interpolation of data points.

A flat output based on B-splines is defined by

$$y^{\text{flat}}(t) = \sum_{i=1}^l \theta_i^* N_{i,k}(t), \quad (7)$$

where $N_{i,k}$ are B-splines of order k on l collocation points, and θ_i^* is a coefficient vector, whose elements are referred to as meta-parameters. Please note that the meta-parameters θ_i^* constitute the decision variables in the flatness-based optimal experimental design framework for model selection, which is introduced in the following section.

3. Robust Design of Experiments for Model Selection

To maximize the information provided by an experiment for the aim of discriminating between different model hypotheses, we incorporate the inverse modeling approach into the design of experiments setting. First, a generic scheme for a standard MBDoE strategy for model selection is given in Figure 2. Starting with the model-building process, several competing model hypotheses and candidates may exist. Thus, the MBDoE technique predicts operating conditions and experimental data that ensure better model selection and falsification. Next, the optimized experimental setup is implemented, and new informative data are recorded. With these new data, the model parameters of all model candidates are refined and updated. Finally, based on the recalculated mismatch of experimental data and simulation results, the difference in the competing model candidates may become more

significant, and some model candidates might be excluded in terms of falsification. As indicated in Figure 2, the MBDoE workflow is an iterative process; that is, the prediction of improved experiments, the data acquisition, the adaptation of the model parameters, and the assessment of the remaining model candidates is repeated until the most suitable model candidate is identified.

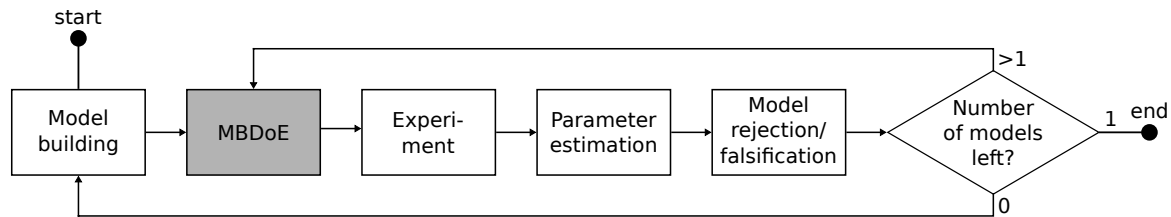


Figure 2. MBDoE workflow for model selection adapted from [20].

Mathematically, the MBDoE approach can be stated as an *optimal control problem* (OCP) of the form

$$\begin{aligned}
 & \max_{\boldsymbol{\phi}(t) \in \Phi} \mathcal{J}(\boldsymbol{\phi}(t_f)) \\
 \text{s.t.} \quad & \text{Process model : Equation (1),} \\
 & \mathbf{x}(t_0) = \mathbf{x}_0, \\
 & \mathbf{g}_{\text{eq}}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}) = 0, \\
 & \mathbf{g}_{\text{ineq}}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}) \leq 0, \\
 & \boldsymbol{\Phi}_{\min} \leq \boldsymbol{\Phi} \leq \boldsymbol{\Phi}_{\max}.
 \end{aligned} \tag{8}$$

\mathcal{J} is the scalar objective function aiming to maximize the difference between the trajectories of the output functions of competing model candidates, and $\boldsymbol{\phi}(t) = \boldsymbol{\phi}(\mathbf{u}(t), \mathbf{x}_0)$ is the design vector within the design space Φ specified by the control $\mathbf{u}(t)$ and the initial states \mathbf{x}_0 . Please note that $t \in [t_0, t_0 + t_f]$, where $t_0 = 0$ is the initial time, and t_f is the time duration of the experiment. The objective function is given in *Mayer* form without loss of generality as the *Lagrange* and *Bolza* problem types can be converted to the former [18]. Both equality constraints $\mathbf{g}_{\text{eq}} : \mathbb{R}^{n \times m \times p} \rightarrow \mathbb{R}^{n_{\text{eq}}}$ and inequality constraints $\mathbf{g}_{\text{ineq}} : \mathbb{R}^{n \times m \times p} \rightarrow \mathbb{R}^{n_{\text{ineq}}}$ can be considered in the optimization problem. Exemplarily, they might be introduced to avoid negative values of system states or to restrict a species' concentration to a certain limit. $[\boldsymbol{\Phi}_{\min}, \boldsymbol{\Phi}_{\max}]$ are the upper and lower boundaries for the design variables.

Solving the dynamic optimization problem (Equation (8)) can be performed following two different discretization strategies of the time-dependent functions in Equation (8): *Optimize then Discretize*, also called the *indirect* or *variational approach*, or *Discretize then Optimize*, its *direct* counterpart. The first strategy focuses on the solution of the first-order optimality conditions for the OCP, resulting in a boundary value problem (BVP). The difficulties associated with solving BVPs led to research activities dealing with the conversion of the OCP to constrained, finite-dimensional problems to exploit state-of-the-art large-scale *nonlinear program* (NLP) solvers [18]. Methods applying NLP solvers can be classified further into sequential and simultaneous strategies. In the sequential approach referred to as *single shooting*, the control vector is parameterized, and the resulting NLP problem is solved using control vector parameterization (CVP) methods [60,61]. In contrast, the simultaneous approach *multiple shooting* divides the OCP into smaller subproblems requiring next to CVP initial states in the individual problems and serves as a bridge to the *direct transcription* approach, where all variables, i.e., states and controls, are discretized [18]. The discretization of the states and controls is usually realized using collocation and control parameterization techniques that lead to complex optimization problems for which efficient solvers and great computational power are needed [11].

Various scalar objective functions \mathcal{J} for the MBDoE approach (Equation (8)) have been proposed in the literature [7,62–66]. The general goal of the optimal control problem for model selection (8) is to maximize some measure on the distance between the different model candidates by searching for an

optimal control profile while meeting the specified constraints, including the solving of the underlying ODE system (Equation (1)).

The proposed flatness-based approach for the MBDoE performs differently. It literally operates in an inverse manner. Optimized flat output trajectories maximize the difference between the model outputs of the competing model candidates. The related controls \mathbf{u} and the initial states \mathbf{x}_0 are reconstructed from the flat model \mathcal{M}^{-1} and have to be identical for all model candidates and their optimized flat outputs \mathbf{y}^{flat} . Using the Euclidean distance as a discrepancy measure between model outputs and assuming M model candidates, the original OCP given in Equation (8) becomes

$$\begin{aligned} \max_{\mathbf{y}^{\text{flat}}} \quad & \int_{t_0}^{t_f} \sum_{i=1}^{M-1} \sum_{j=i+1}^M \left[\mathbf{y}^i(\mathbf{y}^{\text{flat},i}, \dot{\mathbf{y}}^{\text{flat},i}, \dots, \mathbf{y}^{\text{flat},i(r)}) \right. \\ & \left. - \mathbf{y}^j(\mathbf{y}^{\text{flat},j}, \dot{\mathbf{y}}^{\text{flat},j}, \dots, \mathbf{y}^{\text{flat},j(r)}) \right]^2 dt \\ \text{s.t.} \quad & \Delta \mathbf{u}(\mathbf{y}^{\text{flat}}) = 0, \\ & \Delta \mathbf{x}_0(\mathbf{y}^{\text{flat}}) = 0, \\ & \mathbf{g}_{\text{eq}}(\mathbf{y}^{\text{flat}}) = 0, \\ & \mathbf{g}_{\text{ineq}}(\mathbf{y}^{\text{flat}}) \leq 0. \end{aligned} \tag{9}$$

The inputs, states and outputs in the optimization formulation are expressed according to Equations (2), (4), and (5). The differences between the recalculated inputs and the initial states for all M model candidates are measured by the delta function $\Delta(\cdot)$ that might be the Euclidean distance alike. For the flat outputs are of functional form, time-dependent empirical basis functions $\mathbf{y}^{\text{flat},i} = \mathbf{y}^{\text{flat},i}(\boldsymbol{\theta}^{*,i}, t) \forall i \in \{1, \dots, M\}$ with meta-parameters $\boldsymbol{\theta}^{*,i}$ have to be specified; see also Section 2.3. The optimal control problem stated in Equation (8) is readily transformed in an algebraic nonlinear optimization problem as the system inputs and outputs are available in closed form, where the need for integrating the underlying ODE system and the need for control and or state vector parameterization are dropped. As a last step, the resulting optimal control profiles and initial conditions are employed to conduct a new experiment that is expected to deliver additional informative data regarding model falsification.

After conducting the optimally designed experiment, gathering of corresponding data, and the model parameters update, the model assessment is performed. Criteria for model assessment and selection are subject to different dimensions: falsifiability, explanatory adequacy, interpretability, faithfulness, goodness-of-fit, complexity/simplicity and generalizability [35]. For instance, a well-known and commonly used criterion is the Akaike information criterion (AIC) or its small-sample analog, the corrected Akaike information criterion AIC_c [67], which evaluates the model response residuals, as well as the model complexity in terms of the dimension of the model parameter vector. In general, however, there is no master strategy for model discrimination and therefore, no universal criterion. The method applied depends on the stage of modeling, the type of uncertainty expected, and the strategy chosen [68]. For instance, the so-called overlap approach emphasizes the fact that in reality, model parameters are uncertain, which is in turn not regarded in many common discrimination criteria as the AIC or the Bayesian information criterion BIC [31]. Assuming additive measurement noise and according to the Doob–Dynkin lemma [69], the identified model parameters $\hat{\boldsymbol{\theta}}$ can be considered to be random variables, where the probability space (Ω, \mathcal{F}, P) contains the sample space Ω , the σ -algebra \mathcal{F} , and the probability measure P . Thus, the overlap approach accounts for parameter uncertainties by considering the trajectory confidence intervals, which is in contrast to the AIC, where only the point estimate at the maximum likelihood and the

corresponding model responses are used [68]. Adjusted to the requirements at hand, the overlap of two models considering parametric uncertainties is

$$OVL = \sum_{k=0}^T \frac{2\sqrt{\sigma_{1,k}^2\sigma_{2,k}^2}}{\sigma_{1,k}^2 + \sigma_{2,k}^2} \exp \frac{-0.5(\mu_{1,k} - \mu_{2,k})^2}{\sigma_{1,k}^2 + \sigma_{2,k}^2}, \quad (10)$$

where $\mu_{i,k}$ and $\sigma_{i,k}$ are the expected value and the variance of the output function of model i at time point k resulting from the stochastic nature of the parameters, respectively. In the next subsection, we illustrate how these statistical quantities in Equation (10) can be quantified efficiently with the point estimate method (PEM).

3.1. Point Estimate Method

The mean and variance occurring in the overlap formula (Equation (10)) are known as the first raw and second central moments of a continuous random variable and are defined as

$$\mu = \mathbb{E}[k(\boldsymbol{\theta})] = \int_{\Omega} k(\boldsymbol{\theta})P(\boldsymbol{\theta}) d\boldsymbol{\theta}, \quad (11a)$$

$$\sigma^2 = \mathbb{V}[k(\boldsymbol{\theta})] = \int_{\Omega} (k(\boldsymbol{\theta}) - \mu)^2 P(\boldsymbol{\theta}) d\boldsymbol{\theta}, \quad (11b)$$

where $P(\boldsymbol{\theta})$ is the probability density function related to the random model parameter vector $\boldsymbol{\theta}$ of a process model \mathcal{M} (Equations (1) and (2)). $k(\cdot)$ represents a generic nonlinear function. Usually, no closed solutions of the integrals on the right side of Equations (11a) and (11b) are available, and must be evaluated numerically. For accurate results, Monte Carlo simulations can be performed that draw many samples from the probability density functions. Thus, in particular, in the case of complex functions $k(\boldsymbol{\theta})$ and many model parameters, the evaluation of the integral becomes computationally expensive or even prohibitive due to the so-called *curse of dimensionality*. Alternatively, the PEM, initially developed for generic multi-dimensional integration problems over symmetrical regions, is a credible and practical method for uncertainty propagation with low computational cost; see [33] and references within. In process systems engineering, the PEM has been successfully applied to robustify various optimization problems, including non-symmetrical probability density functions, correlated model parameters, and imprecise parameter uncertainties [3,70,71].

The PEM approximates the integrals in Equation (11) by summing over n_{PEM} weighted sampling points

$$\int_{\Omega} k(\boldsymbol{\theta})P(\boldsymbol{\theta}) d\boldsymbol{\theta} \approx \sum_{l=1}^{n_{PEM}} w_l k(\boldsymbol{\theta}_l), \quad (12)$$

with weight factors w_l and parameter vector realizations $\boldsymbol{\theta}_l$. In detail, assuming a nominal parameter vector $\boldsymbol{\theta}_0$, dedicated model parameter vector realizations, $\boldsymbol{\theta}_l$, form a parameter vector set, $\boldsymbol{\theta}_l \in \mathcal{O} := \{\boldsymbol{\theta}_0, \mathcal{O}_1, -\mathcal{O}_1, \mathcal{O}_2, -\mathcal{O}_2, \mathcal{O}_3, -\mathcal{O}_3\}$, where

$$\mathcal{O}_1 := \{\boldsymbol{\theta}_0[i] + \vartheta, \forall i \in \{1, \dots, p\}\},$$

$$\mathcal{O}_2 := \{\boldsymbol{\theta}_0[(i, j)] + [+ \vartheta, + \vartheta], \forall i, j \in \{1, \dots, p\}, j > i\},$$

$$\mathcal{O}_3 := \{\boldsymbol{\theta}_0[(i, j)] + [- \vartheta, + \vartheta], \forall i, j \in \{1, \dots, p\}, j > i\}.$$

Here, $\boldsymbol{\theta}_0[i]$ means that the i th element of the nominal parameter vector, $\boldsymbol{\theta}_0$, is permuted via the spreading parameter, ϑ , and $\boldsymbol{\theta}_0[(i, j)]$ that the i th and the j th elements are modified, respectively. Note that the weight factors, w_l , as well as the spreading parameter, ϑ , are determined via a corresponding algebraic equation system; the interested reader is referred to [33] and references therein. Based on the parameter samples, $\boldsymbol{\theta}_l$, we can approximate the statistics of a given nonlinear

function. For instance, the approximations of the expected value and the variance as stated in Equations (11a) and (11b) read as

$$\mu \approx w_0 k(\theta_0) + w_1 \sum_l^{|\mathcal{O}_{w_1}|} k(\theta_l) + w_2 \sum_l^{|\mathcal{O}_{w_2}|} k(\theta_l), \tag{14a}$$

$$\sigma^2 \approx w_0 (k(\theta_0) - \mu)^2 + w_1 \sum_l^{|\mathcal{O}_{w_1}|} (k(\theta_l) - \mu)^2 + w_2 \sum_l^{|\mathcal{O}_{w_2}|} (k(\theta_l) - \mu)^2, \tag{14b}$$

with $\mathcal{O}_{w_1} := \{\mathcal{O}_1, -\mathcal{O}_1\}$ and $\mathcal{O}_{w_2} := \{\mathcal{O}_2, -\mathcal{O}_2, \mathcal{O}_3, -\mathcal{O}_3\}$. Please note that the overall parameter sample number, n_{PEM} , used in Equations (14a) and (14b) scales quadratically to the dimension of uncertain model parameters:

$$n_{PEM} = 2p^2 + 1. \tag{15}$$

At the cost of accuracy (i.e., approximation error introduced via Equation (13)), the required sample numbers in Equations (14a) and (14b) can be reduced according to:

$$\mu \approx w_0 k(\theta_0) + w_1 \sum_l^{|\mathcal{O}_{w_1}|} k(\theta_l), \tag{16a}$$

$$\sigma^2 \approx w_0 (k(\theta_0) - \mu)^2 + w_1 \sum_l^{|\mathcal{O}_{w_1}|} (k(\theta_l) - \mu)^2, \tag{16b}$$

where the overall parameter sample number, n_{PEM} , scales linearly to the dimension of uncertain model parameters:

$$n_{PEM} = 2p + 1. \tag{17}$$

Equations (14) and (16) are exact for monomials of up to degrees five and three [33]. Thus, in what follows, we use the terms PEM5 and PEM3 to distinguish between the two PEM approximation schemes. In Table 1, we summarize the values for the weights, w_l , and the spread parameter, ϑ , assuming a standard Gaussian distribution. In the case of PEM3, the spread parameter, ϑ , can be considered to be a design parameter, which is frequently set to the corresponding PEM5 value; that is, $\vartheta = \sqrt{3}$.

Table 1. Weights and spread parameter of the PEM for a standard Gaussian distribution [33].

Approximation Scheme	w_0	w_1	w_2	ϑ
PEM5 (Equations (14a) and (14b))	$1 + \frac{p^2-p}{18}$	$\frac{4-p}{18}$	$\frac{1}{36}$	$\sqrt{3}$
PEM3 (Equations (16a) and (16b))	$1 - \frac{p}{\vartheta^2}$	$\frac{1}{2\vartheta^2}$	-	ϑ

Please note that any parametric or non-parametric probability distribution of relevant model parameters can be considered via a (non)linear transformation step, including parameter correlations [70].

3.2. Robust Flatness-Based Objective Function

For the robust formulation of the nominal optimization problem (Equation (9)), we aim at introducing an additional term in the objective function that penalizes the propagated uncertainty that is quantified by the variances \mathbb{V} of the models' inputs \mathbf{u} . In this vein, we expect the calculated overlap (Equation (10)) to decrease as the confidence bands of the trajectories tighten. The objective function is balanced by a weight factor $\alpha \in [0, 1]$ leading to Pareto optimal points for its different

values. The optimization problem for a flatness-based robust design of experiments for model selection is then stated as

$$\begin{aligned} \max_{\boldsymbol{\theta}^*} & \sum_{k=0}^T \left(\alpha \sum_{i=1}^{M-1} \sum_{j=i+1}^M \left[\mathbf{y}^i(\boldsymbol{\theta}_i^*, t_k) - \mathbf{y}^j(\boldsymbol{\theta}_j^*, t_k) \right]^2 \right. \\ & \left. (1 - \alpha) \sum_{i=1}^M \mathbb{V}[\mathbf{u}_i(\boldsymbol{\theta}_i^*, t_k)] \right) \\ \text{s.t.} & \quad \Delta \mathbf{u}(\boldsymbol{\theta}^*, t_k) = 0, \\ & \quad \Delta \mathbf{x}_0(\boldsymbol{\theta}^*, t_k) = 0, \\ & \quad \mathbf{g}_{\text{eq}}(\boldsymbol{\theta}^*, t_k) = 0, \\ & \quad \mathbf{g}_{\text{ineq}}(\boldsymbol{\theta}^*, t_k) \leq 0. \end{aligned} \quad (18)$$

The decision variables of the optimization problem are the meta-parameters present in the basis functions for the flat outputs, i.e., the control vectors of the B-spline curves. As previously in the nominal optimization problem (Equation (9)), the robust formulation is a programming problem for which all occurring functions are algebraically available. Thus, the nominal version (Equation (9)) and the robust version (Equation (18)) are readily stated as an NLP without any use of parameterization methods. Furthermore, the possibility of deriving functions for exact gradients is given as opposed to the application of automatic differentiation or finite differences methods. A flow chart for the proposed robust MBDoE strategy using differential flatness is given in Figure 3.

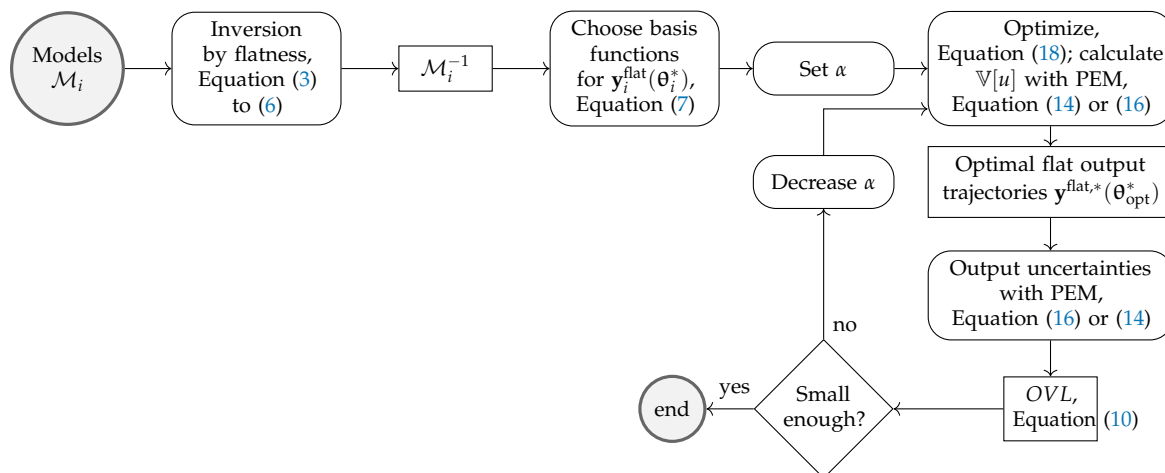


Figure 3. Robust MBDoE strategy.

3.3. Computational Methods

The whole program is coded in Julia [72]. The derivation of flat systems is based on the comprehensive Julia suite on differential equations [73]. For symbolic calculation, the Python package *SymPy* is used where its functions can be accessed from Julia via an interface. As modeling language for the optimization problems, Julia-based *JuMP* was chosen [74]. The nonlinear programming problems were solved with the open-source large-scale interior point solver *IPOPT* [75]. All calculations were run on a standard desktop machine and finished within an hour, where the bulk of the computational load was spent in symbolic calculations and the quantification of the propagated uncertainty.

4. Case Study

Biocatalytic reactions appear in numerous syntheses of natural products and active pharmaceutical ingredients. For example, chiral hydroxy ketones are important building blocks in the pharmaceutical

industry and can be produced from aldehydes via enzymatic carbonylation [22]. A lack of mechanistic kinetic models for biocatalytic carbonylation and simultaneous inactivation of the benzaldehyde lyase was studied in [76] using MBDoe, where the model quality of the different model candidates was analyzed using the AIC. Motivated by the biocatalytic carbonylation of ketones from aldehydes, in this simulation study, we consider two models simulating the enzymatic step from benzaldehyde to benzoin [22]. In the enzymatic reaction chain, the forward reaction forming the intermediate is significantly larger than its reverse counterpart leading to the two model candidates in Tables 2 and 3.

Table 2. Reaction scheme of model \mathcal{M}_1 .

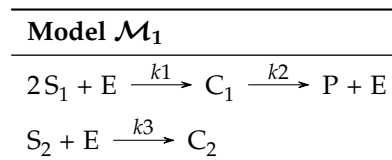
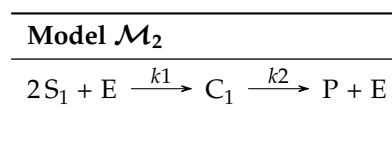


Table 3. Reaction scheme of model \mathcal{M}_2 .



The two model candidates differ in that way that model candidate \mathcal{M}_1 suffers from a loss reaction where a second substrate S_2 is irreversibly inhibiting. Additionally, substrate S_2 is expected to be constant via proper control actions over the experimental run. The corresponding ODE systems of the two model candidates are

$$\mathcal{M}_1 = \begin{cases} \frac{dx_1}{dt} = -2k_1 x_1^2 x_3 + u_1 \\ \frac{dx_2}{dt} = k_1 x_1^2 x_3 - k_2 x_2 \\ \frac{dx_3}{dt} = -k_1 x_1^2 x_3 + k_2 x_2 - k_3 [S_2] x_3 + u_2 \\ \frac{dx_4}{dt} = k_2 x_2 \end{cases}, \quad (19)$$

$$\mathcal{M}_2 = \begin{cases} \frac{dx_1}{dt} = -2k_1 x_1^2 x_3 + u_1 \\ \frac{dx_2}{dt} = k_1 x_1^2 x_3 - k_2 x_2 \\ \frac{dx_3}{dt} = -k_1 x_1^2 x_3 + k_2 x_2 + u_2 \\ \frac{dx_4}{dt} = k_2 x_2 \end{cases}, \quad (20)$$

where the indices are assigned according to $\{1, 2, 3, 4\} = \{S_1, C_1, E, P\}$, and substrate S_1 and enzyme E can be dosed over the course of the experiment. Please note that in model \mathcal{M}_1 , the differential equations of the second substrate S_2 , which is assumed to be constant, and the loss product C_2 are not specified, as information about their time behavior does not influence the other differential equations, and therefore, is not relevant to the problem at hand. The measured concentrations are the concentrations of the substrate and the final product; that is, $\mathbf{y} = (x_1, x_4)^T$. The kinetic constants have the following expected values: For model \mathcal{M}_1 $(k_1, k_2, k_3) = (0.215 \text{ L}^2/(\text{mmol}^2 \text{ min}), 91.475 \text{ min}^{-1}, 0.019 \text{ L}/(\text{mmol min}))$, and for model \mathcal{M}_2

$(k_1, k_2) = (0.204 \text{ L}^2 / (\text{mmol}^2 \text{ min}), 103.344 \text{ min}^{-1})$. The variances of the parameters equal 5 % of their expected values in the absence of parameter correlations.

5. Results

For the case study given in Section 4, we analyzed the differential flatness property, and then, we designed discriminating input controls based on the proposed MBDoE approach, which makes use of the differential flatness concept. The effect of model parameter uncertainties was studied as well, and the robust MBDoE approach was applied as an appropriate countermeasure.

5.1. Differential Flatness Property

We derive the flat outputs for both models following heuristic methods to (1) obtain a flat output candidate according to Equation (3) and (2), with the help of graph theory, to show that the candidate fulfills the differential flatness conditions given Equations (4)–(6). Drawing the directed graphs (digraphs) for model \mathcal{M}_1 and model \mathcal{M}_2 , we observe that they look alike. The corresponding adjacency matrices are

$$A_u = \begin{matrix} & u_1 & u_2 \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \end{matrix}, \quad A_x = \begin{matrix} & x_1 & x_2 & x_3 & x_4 \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix} & \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{matrix}, \quad A_y = \begin{matrix} & y_1 & y_2 \\ \begin{matrix} x_1 & x_2 & x_3 & x_4 \end{matrix} & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \end{matrix},$$

and the resulting digraph is shown in Figure 4a. The digraph is composed of 8 vertices, $V = \{u_1, u_2\} \cup \{x_1, x_2, x_3, x_4\} \cup \{y_1, y_2\}$, and 13 edges corresponding to the non-zero entries in the adjacency matrices A_u , A_x , and A_y . The self-loops of $\{x_1, x_2, x_3\} \in V$ are related to the non-zero diagonal elements of A_x .

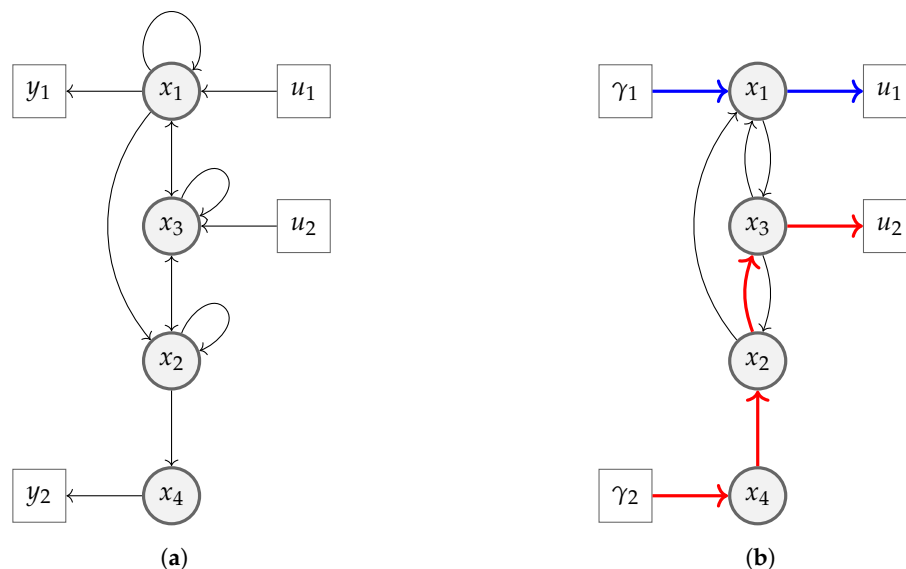


Figure 4. Digraph and explicit expressibility graph to study the differential flatness property. (a) Digraph for model \mathcal{M}_1 and \mathcal{M}_2 ; (b) Explicit expressibility graph.

The control vector dimension in both models is 2. Therefore, to comply with condition given in Equation (6), the dimension of the flat output vector must be 2 as well. Inspecting the system of differential equations in Equation (19) one might observe that x_4 does not appear on the right side of any of the differential equations. The corresponding node in the digraph (see Figure 4a) represents a dead end (i.e., no edge from $x_4 \in V$ to $\{x_1, x_2, x_3\} \in V$), and therefore, must be part of the flat output.

Obviously, the second element of the flat output vector should be chosen from the concentrations left as they have direct physical meaning for the study case at hand. We consider the flat output candidate

$$\mathbf{y}^{\text{flat}} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_4 \end{pmatrix} \quad (21)$$

that evidently satisfies Equation (3).

The explicit expressibility graph, which can be readily obtained from the digraph, is shown in Figure 4b. Please note that in comparison to the digraph (Figure 4a), the self-loops are omitted, and the edge directions are reversed. In Figure 4b, two disjoint paths are drawn that cover all vertices related to the inputs, system states, and flat outputs. Therefore, the flat output vector (Equation (21)) is supposed to form a differentially flat system which we will confirm using its definition in the following. From the ODE system, after several reformulations and substitutions, we obtain the inverse models:

$$\mathcal{M}_1^{-1} = \begin{cases} x_1 & = \gamma_1 \\ x_2 & = \frac{\dot{\gamma}_2}{k_2} \\ x_3 & = \frac{\frac{\dot{\gamma}_2}{k_2} - \gamma_2}{k_1 \gamma_1^2} \\ x_4 & = \gamma_2 \\ u_1 & = \dot{\gamma}_1 + 2\gamma_1^2 \frac{\frac{\dot{\gamma}_2}{k_2} - \gamma_2}{\gamma_1^2} \\ u_3 & = \mathcal{F}_1(\gamma_1, \dot{\gamma}_1, \gamma_2, \dot{\gamma}_2, \ddot{\gamma}_2) \end{cases} \quad (22)$$

and

$$\mathcal{M}_2^{-1} = \begin{cases} x_1 & = \gamma_1 \\ x_2 & = \frac{\dot{\gamma}_2}{k_2} \\ x_3 & = \frac{\frac{\dot{\gamma}_2}{k_2} - \gamma_2}{k_1 \gamma_1^2} \\ x_4 & = \gamma_2 \\ u_1 & = \dot{\gamma}_1 + 2\gamma_1^2 \frac{\frac{\dot{\gamma}_2}{k_2} - \gamma_2}{\gamma_1^2} \\ u_3 & = \mathcal{F}_2(\gamma_1, \dot{\gamma}_1, \gamma_2, \dot{\gamma}_2, \ddot{\gamma}_2) \end{cases} \quad (23)$$

Please note that we omitted to write out the equations for the second element of the output vector u_2 for model \mathcal{M}_1 and model \mathcal{M}_2 due to readability reasons. However, keep in mind that the functions \mathcal{F}_1 and \mathcal{F}_2 in Equations (22) and (23) differ. The inverse model (Equation (22)) with its flat output (Equation (21)) satisfies the set containing all three conditions for differential flatness (Equations (4)–(6)). Accordingly, the inverse model of model \mathcal{M}_2 (Equation (23)) fulfills the conditions for differential flatness. Thus, both models are differentially flat models. For prescribed trajectories $\gamma_1^* = x_1^*$ and $\gamma_2^* = x_4^*$, the necessary experimental conditions for the controls are readily available by differentiation; for an example, see [20].

5.2. Non-Optimized Experimental Design

We first consider the non-optimized experimental design, i.e., the initial situation before any optimization iteration. Please note that a batch process is assumed, and thus, the controls are zero and

not explicitly shown. Plots for the measured states and the measured states with confidence bands are shown in Figure 5 on the left side and on the right side, respectively.

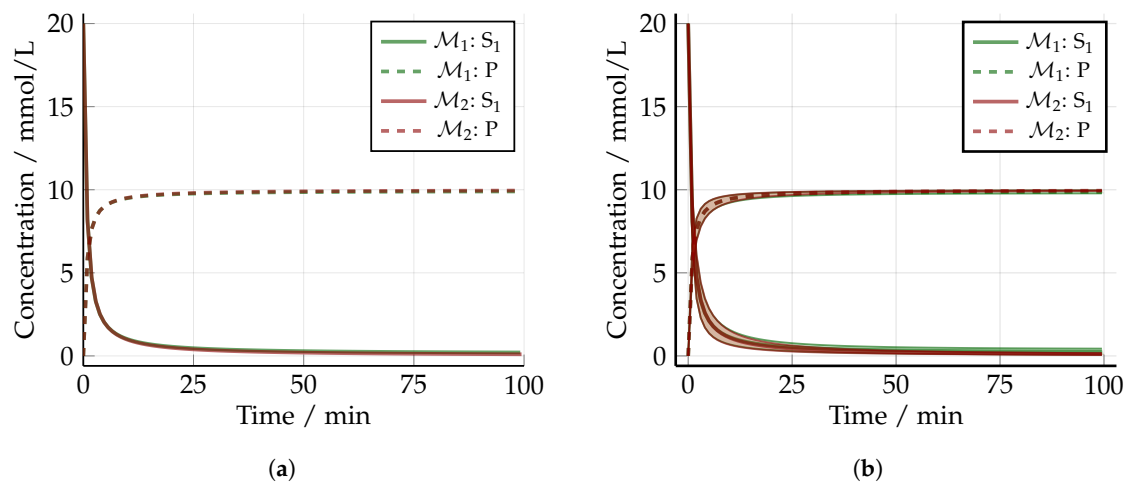


Figure 5. Measured states before applying the MBDoE approach. (a) Output trajectories neglecting parameter uncertainties; (b) Confidence intervals due to parameter uncertainties.

For the nominal case in Figure 5a, it is impossible to distinguish the two models from each other, even if the experimental data were close to one of the models. In Figure 5b, it is well observable that the uncertainties in the measured states given as confidence intervals are low. To better compare the results with the expected results from the optimization part, we normalize the overlap measure (Equation (10)) to the overlap corresponding to Figure 5b; that is, $OVL_N = 1.0$.

5.3. Optimized Experimental Design without Uncertainty: The Nominal Case

Before optimizing, suitable B-spline types must be chosen. In contrast to the classical application of splines in approximation or interpolation of available data, they are used to express smooth control and state trajectories along the time axis, and are created from their definition as opposed to a fitting process. Thus, common problems experienced in spline approximation and interpolation, like cusps and loops [77], are less likely to occur. From the different methods for splines, we choose the *uniform* method where the data points are uniformly distributed over the domain range. Furthermore, for each element of the flat output vectors, a B-spline curve of order 6 with 12 control points and clamped end conditions was used, resulting in a 48-dimensional decision space for the optimization problem. In the first optimization step, we follow the classical deterministic optimization by setting $\alpha = 1$ in the setting (Equation (18)). The results for the measured states of the nominal case are shown in Figure 6, again, without uncertainty and with uncertainty vis-à-vis. The corresponding control curves are displayed in Figure 7 on the left side.

In the plot without propagated uncertainties, it is clearly visible that the expected values of the measured states are significantly driven apart. However, a look at the plot on the right reveals that (1) the uncertainties have substantially enlarged compared to the initial situation, and (2) the confidence bands of both models overlap noticeably. The overlap values are given in Table 4. The overlap has decreased versus the initial situation, i.e., it is 6 percentage points lower compared to before the optimization. However, we also came across simulation runs where the overlap worsened in the sense of a larger overlap value if the nominal optimization state was compared with the non-optimized. In this context, the question arises if we can perform better, i.e., further lower the overlap by inherently taking care of the parametric uncertainties in the optimization setting. Therefore, we robustify the optimization by considering the variances in the objective function as described in Section 3.1.

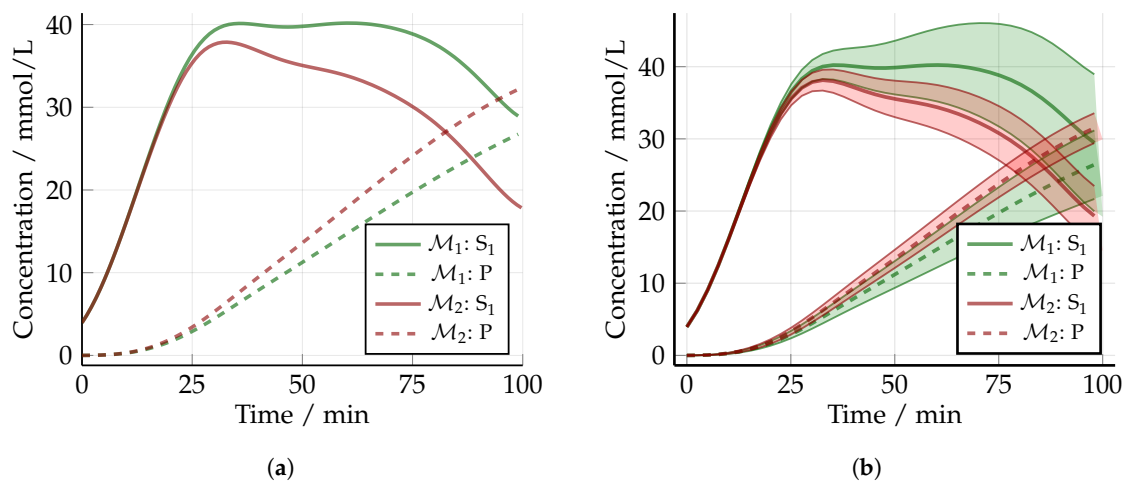


Figure 6. Measured states for the nominal design. (a) Output trajectories neglecting parameter uncertainties; (b) Confidence intervals due to parameter uncertainties.

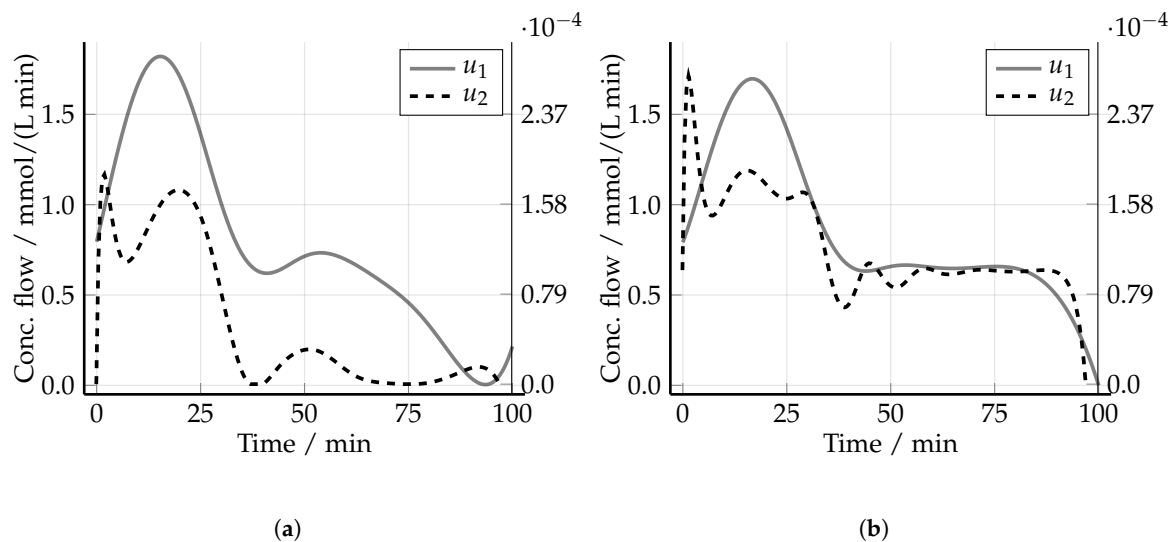


Figure 7. Optimized input controls (u_1 : y-axis on the left, u_2 : y-axis on the right). (a) Nominal design; (b) Robust design (PEM5).

5.4. Robust Experimental Design

For the robust MBDoe setting (Equation (18)), we set $\alpha = 0.5$; that is, the Euclidean distance of the flat outputs of model \mathcal{M}_1 and \mathcal{M}_2 as well as the resulting uncertainties in the recalculated model inputs are equally weighted. Compared with the nominal optimization, the computational time to solve reaching satisfying convergence increases multiple-fold despite the low number of sampling points that the PEM is using. In particular, when PEM5 is employed, the computational time increases considerably due to the higher sample number; see Equations (15) and (17). The concentration curves for the resulting states of the robust flatness-based design are shown in Figure 8.

The trajectories for the expected values in the left subfigure clearly drift apart, but less strongly than in the nominal optimization case; see Figure 6. Considering 100 equally spaced points, the norms of the model output distances in the robust case are roughly half as large as in the nominal case. The presumption that the overlap has decreased can be confirmed when looking at the overlap values in Table 4.

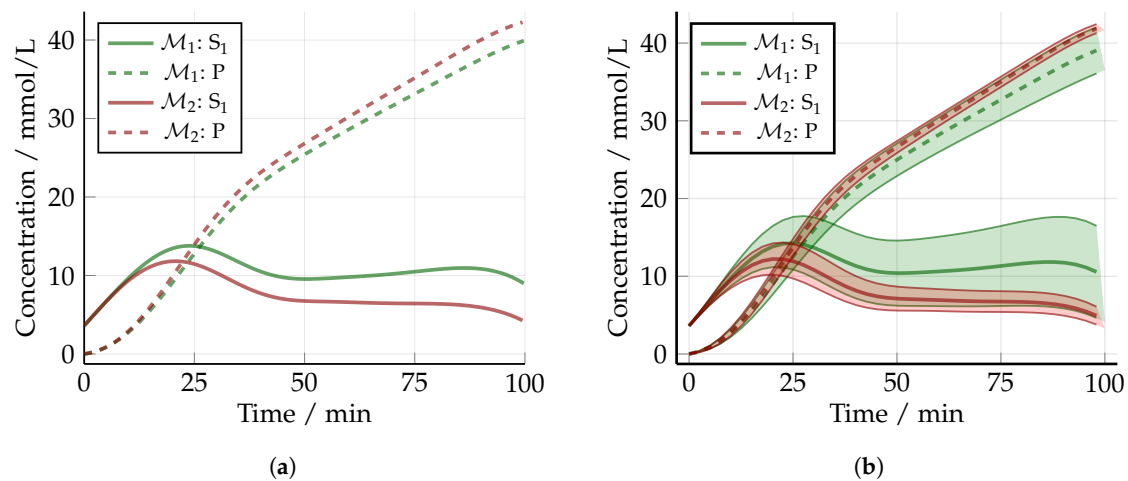


Figure 8. Measured states for the robust flatness-based design. (a) Output trajectories neglecting parameter uncertainties; (b) Confidence intervals due to parameter uncertainties.

Table 4. Results for the overlap.

Design	None	Nominal	PEM3	PEM5
OVL_N	1.0	0.94	0.81	0.80

The overlap value has decreased by 19 percentage points and by 13 percentage points when compared with the initial situation and the nominal optimization, respectively. The propagated uncertainties can be extracted from the graphs shown in Figure 9 where the variances for both outputs of model 1 are considerably lower when the nominal case on the left is compared with the robust one on the right. Furthermore, the approximation of the propagated uncertainty using the PEM is sufficiently good, as the results are in proximity to the Monte Carlo results with 10^4 samples. Thus, the inclusion of the propagated uncertainty as a penalty in the objective function has delivered the desired outcome. Prospectively, a different choice of α or the drawing of a Pareto front might further decrease the overlap between the two model outcomes. The corresponding optimal controls for the substrate and enzyme are drawn in Figure 7 on the right side. Please note that the realization of the profiles in a laboratory setup is constrained by the available pump devices. These constraints could be included in the robust MBDoE framework (Equation (18)), but the technical realization of these input controls is beyond the scope of this study.

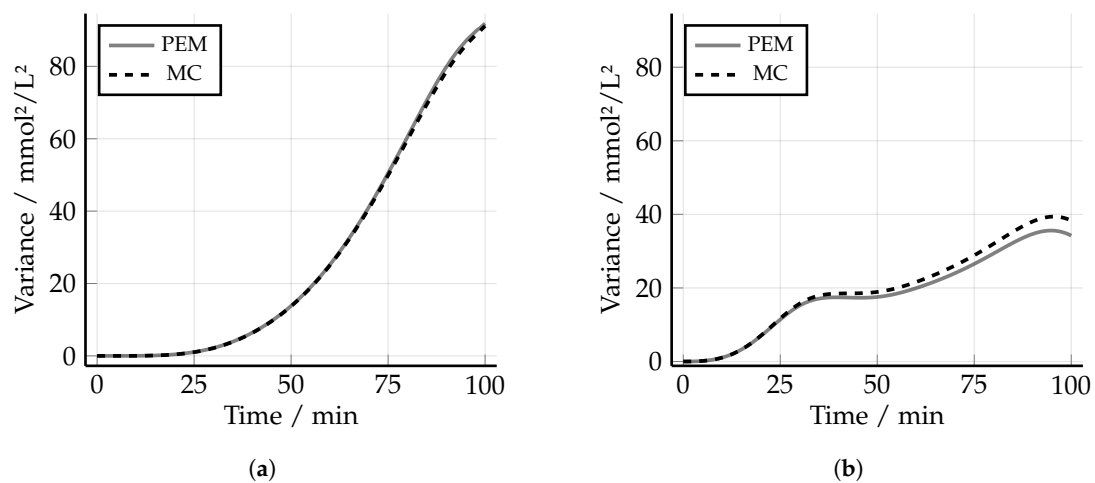


Figure 9. Cont.

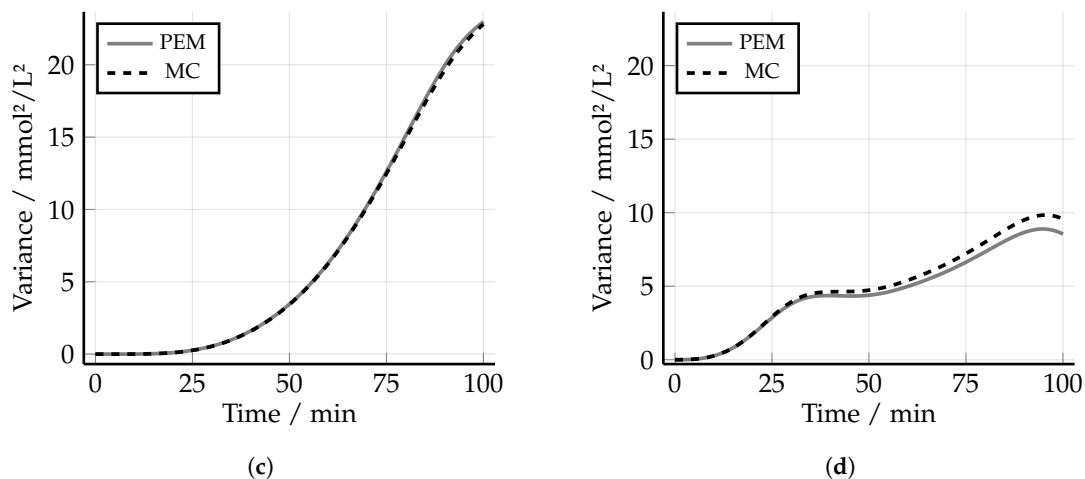


Figure 9. Variances of model outputs for model \mathcal{M}_1 . (a) Nominal design, y_1 ; (b) Robust design, y_1 ; (c) Nominal design, y_2 ; (d) Robust design, y_2 .

6. Conclusions

We have extended a novel approach to the model-based design of experiments for model selection using the differential flatness property of the systems by robustifying the optimization problem against parametric uncertainty. In comparison with commonly used methods for model discrimination, the resulting optimal control problem does not require approximation methods for solving the underlying system of differential equations as both controls and states can be derived analytically in differentially flat systems without numerical integration. Likewise, parameterization of the controls and potentially the states is obsolete for we directly arrive at a nonlinear programming problem. Moreover, the possibility to provide analytic gradients to the optimization solver, even for highly nonlinear systems, is given. The robustification strategy comprised the consideration of propagated uncertainties; that is, the variances of the reconstructed model inputs were added as a penalty term in the objective function. The corresponding uncertainty quantification was performed using the point estimate method, a computationally cheap but reasonably accurate method compared with standard Monte Carlo simulations for uncertainty quantification. We successfully applied the strategy to a nonlinear enzymatic reaction using B-splines as the parameterization technique for expressing the flat outputs, and thus, obtained optimal experimental input controls for the subsequent experiment. We showed that a nominal optimization without the consideration of parameter uncertainties might result in unreliable optimal controls, i.e., not leading to the possibility of selecting one model over another after having performed the subsequent experiment. In contrast, if parametric uncertainties were inherently considered in the optimal experimental design problem, more reliable MBDofE results were identified. We emphasize the importance of including the quantification of uncertainties over classic deterministic optimization where the differential flatness concept shows promising characteristics for advanced system identification strategies in terms of precise parameter estimates and reliable model selection.

Author Contributions: Conceptualization, R.S.; methodology, M.S. and R.S.; software, M.S.; validation, M.S. and R.S.; writing—original draft preparation, M.S.; funding acquisition, R.S. All authors have read and agreed to the published version of the manuscript.

Funding: We acknowledge support from the German Research Foundation and the Open Access Publication Funds of the Technische Universität Braunschweig.

Acknowledgments: M.S. acknowledges the support from the International Max Planck Research School for Advanced Methods in Process and Systems Engineering, MPI Magdeburg. Moreover, we thank Elsevier for the permission to reproduce Figure 2: “This figure was published in Flatness-Based Design of Experiments for Model Selection, Moritz Schulze, René Schenkendorf, IFAC-PapersOnLine, Volume 51, Issue 15, 2018, Pages 233–238, Copyright Elsevier.”

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Su, Q.; Ganesh, S.; Moreno, M.; Bommireddy, Y.; Gonzalez, M.; Reklaitis, G.V.; Nagy, Z.K. A Perspective on Quality-by-Control (QbC) in Pharmaceutical Continuous Manufacturing. *Comput. Chem. Eng.* **2019**, *125*, 216–231. [[CrossRef](#)]
2. Taylor, D. The Pharmaceutical Industry and the Future of Drug Development. In *Issues in Environmental Science and Technology*; Hester, R.E., Harrison, R.M., Eds.; Royal Society of Chemistry: Cambridge, UK, 2015; pp. 1–33.
3. Emenike, V.N.; Xie, X.; Schenkendorf, R.; Spiess, A.C.; Krewer, U. Robust Dynamic Optimization of Enzyme-Catalyzed Carboligation: A Point Estimate-Based Back-off Approach. *Comput. Chem. Eng.* **2019**, *121*, 232–247. [[CrossRef](#)]
4. Kroll, P.; Hofer, A.; Ulonska, S.; Kager, J.; Herwig, C. Model-Based Methods in the Biopharmaceutical Process Lifecycle. *Pharm. Res.* **2017**, *34*, 2596–2613. [[CrossRef](#)] [[PubMed](#)]
5. Pörtner, R.; Barradas, O.P.; Frahm, B.; Hass, V. 16–Advanced Process and Control Strategies for Bioreactors. In *Current Developments in Biotechnology and Bioengineering*; Larroche, C., Sanromán, M.Á., Du, G., Pandey, A., Eds.; Elsevier: Amsterdam, The Netherlands, 2017; pp. 463–493.
6. Abt, V.; Barz, T.; Cruz, N.; Herwig, C.; Kroll, P.; Möller, J.; Pörtner, R.; Schenkendorf, R. Model-Based Tools for Optimal Experiments in Bioprocess Engineering. *Curr. Opin. Chem. Eng.* **2018**, *22*, 244–252. [[CrossRef](#)]
7. Stamati, I.; Logist, F.; Akkermans, S.; Fernández, E.N.; Impe, J.V. On the Effect of Sampling Rate and Experimental Noise in the Discrimination between Microbial Growth Models in the Suboptimal Temperature Range. *Comput. Chem. Eng.* **2016**, *85*, 84–93. [[CrossRef](#)]
8. Telen, D.; Logist, F.; Van Derlinden, E.; Van Impe, J. Approximate Robust Optimal Experiment Design in Dynamic Bioprocess Models. In Proceedings of the 20th Mediterranean Conference on Control & Automation (MED), Barcelona, Spain, 3–6 July 2012; pp. 157–162.
9. Skanda, D.; Lebedz, D. An Optimal Experimental Design Approach to Model Discrimination in Dynamic Biochemical Systems. *Bioinformatics* **2010**, *26*, 939–945. [[CrossRef](#)]
10. Franceschini, G.; Macchietto, S. Model-Based Design of Experiments for Parameter Precision: State of the Art. *Chem. Eng. Sci.* **2008**, *63*, 4846–4872. [[CrossRef](#)]
11. Biral, F.; Bertolazzi, E.; Bosetti, P. Notes on Numerical Methods for Solving Optimal Control Problems. *IEEJ J. Ind. Appl.* **2016**, *5*, 154–166. [[CrossRef](#)]
12. Rodrigues, D.; Billeter, J.; Bonvin, D. Maximum-Likelihood Estimation of Kinetic Parameters via the Extent-Based Incremental Approach. *Comput. Chem. Eng.* **2018**, *122*, 152–171. [[CrossRef](#)]
13. Wang, Y.; Biegler, L.T.; Patel, M.; Wassick, J. Parameters Estimation and Model Discrimination for Solid-Liquid Reactions in Batch Processes. *Chem. Eng. Sci.* **2018**, *187*, 455–469. [[CrossRef](#)]
14. Barz, T.; Bournazou, M.N.C.; Körkel, S.; Walter, S.F. Real-Time Adaptive Input Design for the Determination of Competitive Adsorption Isotherms in Liquid Chromatography. *Comput. Chem. Eng.* **2016**, *94*, 104–116. [[CrossRef](#)]
15. Liu, J.; Mendoza, S.; Li, G.; Fathy, H. Efficient total least squares state and parameter estimation for differentially flat systems. In Proceedings of the American Control Conference, Boston, MA, USA, 6–8 July 2016; pp. 5419–5424. [[CrossRef](#)]
16. Schenkendorf, R.; Mangold, M. Parameter Identification for Ordinary and Delay Differential Equations by Using Flat Inputs. *Theor. Found. Chem. Eng.* **2014**, *48*, 594–607. [[CrossRef](#)]
17. Fliess, M.; Lévine, J.; Martin, P.; Rouchon, P. Sur Les Systèmes Non Linéaires Différentiellement Plats. *C. R. Acad. Sci. Paris* **1992**, *619*–624.
18. Biegler, L.T. *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*; MOS-SIAM Series on Optimization; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2010.
19. Franke, M.; Robenack, K. On the Computation of Flat Outputs for Nonlinear Control Systems. In Proceedings of the 2013 European Control Conference (ECC), Zurich, Switzerland, 17–19 July 2013; pp. 167–172.
20. Schulze, M.; Schenkendorf, R. Flatness-Based Design of Experiments for Model Selection. *IFAC-PapersOnLine* **2018**, *51*, 233–238. [[CrossRef](#)]
21. Schenkendorf, R.; Reichl, U.; Mangold, M. Parameter Identification of Time-Delay Systems: A Flatness Based Approach. *IFAC Proc. Vol.* **2012**, *45*, 165–170. [[CrossRef](#)]

22. Hildebrand, F.; Köhl, S.; Pohl, M.; Vasic-Racki, D.; Müller, M.; Wandrey, C.; Lütz, S. The Production of (R)-2-Hydroxy-1-Phenyl-Propan-1-One Derivatives by Benzaldehyde Lyase from *Pseudomonas fluorescens* in a Continuously Operated Membrane Reactor. *Biotechnol. Bioeng.* **2007**, *96*, 835–843. [[CrossRef](#)]
23. Švarc, A.; Findrik Blažević, Z.; Vasić-Rački, Đ.; Szekrenyi, A.; Fessner, W.D.; Charnock, S.J.; Vrsalović Presečki, A. 2-Deoxyribose-5-phosphate Aldolase from *Thermotoga maritima* in the Synthesis of a Statin Side-chain Precursor: Characterization, Modeling and Optimization. *J. Chem. Technol. Biotechnol.* **2019**, *94*, 1832–1842. [[CrossRef](#)]
24. Hampel, S.; Steitz, J.P.; Baierl, A.; Lehwald, P.; Wiesli, L.; Richter, M.; Fries, A.; Pohl, M.; Schneider, G.; Dobritzsch, D.; et al. Structural and Mutagenesis Studies of the Thiamine-Dependent, Ketone-Accepting YerE from *Pseudomonas protegens*. *ChemBioChem* **2018**, *19*, 2283–2292. [[CrossRef](#)]
25. Galvanin, F.; Barolo, M.; Bezzo, F.; Macchietto, S. A backoff strategy for model-based experiment design under parametric uncertainty. *AIChE J.* **2009**, *56*, 2088–2102. [[CrossRef](#)]
26. Jost, F.; Sager, S.; Le, T.T.T. A Feedback Optimal Control Algorithm with Optimal Measurement Time Points. *Processes* **2017**, *5*, 10. [[CrossRef](#)]
27. Buzzi-Ferraris, G.; Forzatti, P. A new sequential experimental design procedure for discriminating among rival models. *Chem. Eng. Sci.* **1983**, *38*, 225–232. [[CrossRef](#)]
28. Donckels, B.M.; Pauw, D.J.D.; Baets, B.D.; Maertens, J.; Vanrolleghem, P.A. An anticipatory approach to optimal experimental design for model discrimination. *Chemom. Intell. Lab. Syst.* **2009**, *95*, 53–63. [[CrossRef](#)]
29. Violet, L.; Loubière, K.; Rabion, A.; Samuel, R.; Hattou, S.; Cabassud, M.; Prat, L. Stoichio-kinetic model discrimination and parameter identification in continuous microreactors. *Chem. Eng. Res. Des.* **2016**, *114*, 39–51. [[CrossRef](#)]
30. Schenkendorf, R.; Xie, X.; Rehbein, M.; Scholl, S.; Krewer, U. The Impact of Global Sensitivities and Design Measures in Model-Based Optimal Experimental Design. *Processes* **2018**, *6*, 27. [[CrossRef](#)]
31. Schöniger, A.; Wöhling, T.; Samaniego, L.; Nowak, W. Model Selection on Solid Ground: Rigorous Comparison of Nine Ways to Evaluate Bayesian Model Evidence. *Water Resour. Res.* **2014**, *50*, 9484–9513. [[CrossRef](#)]
32. Brunetti, C.; Linde, N. Impact of petrophysical uncertainty on Bayesian hydrogeophysical inversion and model selection. *Adv. Water Resour.* **2018**, *111*, 346–359. [[CrossRef](#)]
33. Lerner, U.N. Hybrid Bayesian Networks for Reasoning about Complex Systems. Ph.D. Thesis, Stanford University Stanford, Stanford, CA, USA, 2002.
34. Xie, X.; Schenkendorf, R.; Krewer, U. Toward a Comprehensive and Efficient Robust Optimization Framework for (Bio)chemical Processes. *Processes* **2018**, *6*, 183. [[CrossRef](#)]
35. Walter, E.E.; Pronzato, L. *Identification of Parametric Models from Experimental Data*; Springer: Berlin, Germany, 1997; p. 413.
36. Buchholz, J.; Grünhagen, W.V. *Inversion Impossible*; GRIN Publishing GmbH: Munich, Germany, 2007.
37. Lu, L.; Murray-Smith, D.J.; Thomson, D. Issues of numerical accuracy and stability in inverse simulation. *Simul. Model. Pract. Theory* **2008**, *16*, 1350–1364. [[CrossRef](#)]
38. Lu, L. Inverse Modelling and Inverse Simulation for System Engineering and Control Applications. Ph.D. Thesis, University of Glasgow, Glasgow, UK, 2007.
39. Wey, T. *Nichtlineare Regelungssysteme: Ein Differentialalgebraischer Ansatz; mit 13 Tabellen*, 1st ed.; OCLC: 76389242; Teubner: Stuttgart, Germany, 2002.
40. Adamy, J. *Nichtlineare Systeme und Regelungen*; Springer Vieweg: Berlin/Heidelberg, Germany, 2014.
41. Waldherr, S.; Zeitz, M. Conditions for the Existence of a Flat Input. *Int. J. Control.* **2008**, *81*, 439–443. [[CrossRef](#)]
42. Kaminski, Y.J.; Lévine, J.; Ollivier, F. Intrinsic and Apparent Singularities in Differentially Flat Systems, and Application to Global Motion Planning. *Syst. Control. Lett.* **2018**, *113*, 117–124. [[CrossRef](#)]
43. Rigatos, G.G. Nonlinear Control and Filtering Using Differential Flatness Approaches. In *Studies in Systems, Decision and Control*; Springer International Publishing: Cham, Switzerland, 2015; Volume 25.
44. Fliess, M.; Mounier, H.; Rouchon, P.; Rudolph, J. A Distributed Parameter Approach to the Control of a Tubular Reactor: A Multivariable Case. In Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No. 98CH36171), Tampa, FL, USA, 18 December 1998; pp. 439–442.

45. Andrej, J.; Meurer, T. Flatness-Based Constrained Optimal Control of Reaction-Diffusion Systems. In Proceedings of the 2018 Annual American Control Conference (ACC), Milwaukee, WI, USA, 27–29 June 2018; p. 6.
46. Kolar, B.; Diwold, J.; Schöberl, M. Necessary and Sufficient Conditions for Difference Flatness. *arXiv* **2019**, arXiv:1909.02868.
47. Wagner, M.O.; Meurer, T.; Kugi, A. Trajectory Planning for Semilinear PDEs Modeling a Countercurrent Heat Exchanger. *IFAC Proc. Vol.* **2010**, *43*, 593–598. [[CrossRef](#)]
48. Vollmer, U.; Raisch, J. Control of Batch Crystallization—A System Inversion Approach. *Chem. Eng. Process. Process. Intensif.* **2006**, *45*, 874–885. [[CrossRef](#)]
49. Mahadevan, R.; Agrawal, S.K.; Doyle III, F.J. Differential Flatness Based Nonlinear Predictive Control of Fed-Batch Bioreactors. *Control. Eng. Pract.* **2001**, *9*, 889–899. [[CrossRef](#)]
50. Liu, J.; Li, G.; Fathy, H.K. An Extended Differential Flatness Approach for the Health-Conscious Nonlinear Model Predictive Control of Lithium-Ion Batteries. *IEEE Trans. Control. Syst. Technol.* **2017**, *25*, 1882–1889. [[CrossRef](#)]
51. Kolar, B.; Kaldmäe, A.; Schöberl, M.; Kotta, Ü.; Schlacher, K. Construction of Flat Outputs of Nonlinear Discrete-Time Systems in a Geometric and an Algebraic Framework. *IFAC-PapersOnLine* **2016**, *49*, 796–801. [[CrossRef](#)]
52. Victor, S.; Melchior, P.; Lévine, J.; Oustaloup, A. Flat Output Computation for Fractional Linear Systems: Application to a Thermal System. *IFAC Proc. Vol.* **2014**, *47*, 2891–2896. [[CrossRef](#)]
53. Waldherr, S.; Zeitz, M. Flat Inputs in the MIMO Case. *IFAC Proc. Vol.* **2010**, *43*, 695–700. [[CrossRef](#)]
54. Richard, P.; Buisson, J.; Cormerais, H. Analysis of Flatness Using Bond Graphs and Bicausality. *IFAC Proc. Vol.* **2002**, *35*, 25–30. [[CrossRef](#)]
55. Csercsik, D.; Szederkényi, G.; Hangos, K.M. Determining Flat Outputs of MIMO Nonlinear Systems Using Directed Graphs. In Proceedings of the 8th Portuguese Conference on Automatic Control (CONTROLO), Vila Real, Portugal, 21–23 July 2008; p. 6.
56. Schizas, C.; Evans, F. A graph theoretic approach to multivariable control system design. *Automatica* **1981**, *17*, 371–377. [[CrossRef](#)]
57. Reinschke, K.J. *Multivariable Control: A Graph-Theoretic Approach*; Springer: Berlin, Germany, 1988.
58. Dion, J.M.; Commault, C.; van der Woude, J. Generic properties and control of linear structured systems: A survey. *Automatica* **2003**, *39*, 1125–1144. [[CrossRef](#)]
59. de Boor, C. *A Practical Guide to Splines*; Number 27 in Applied Mathematical Sciences; Springer: New York, NY, USA, 2001.
60. Vassiliadis, V.; Sargent, R.; Pantelides, C. Solution of a class of multistage dynamic optimization problems. 1. Problems without path constraints. *Ind. Eng. Chem. Res.* **1994**, *33*, 2111–2122. [[CrossRef](#)]
61. Vassiliadis, V.; Sargent, R.; Pantelides, C. Solution of a class of multistage dynamic optimization problems. 2. Problems with path constraints. *Ind. Eng. Chem. Res.* **1994**, *33*, 2123. [[CrossRef](#)]
62. Kremling, A.; Fischer, S.; Gadkar, K.; Doyle, F.J.; Sauter, T.; Bullinger, E.; Allgöwer, F.; Gilles, E.D. A benchmark for methods in reverse engineering and model discrimination: Problem formulation and solutions. *Genome Res.* **2004**, *14*, 1773–1785. [[CrossRef](#)] [[PubMed](#)]
63. Alberton, A.L.; Schwaab, M.; Lobão, M.W.N.; Pinto, J.C. Experimental design for the joint model discrimination and precise parameter estimation through information measures. *Chem. Eng. Sci.* **2011**, *66*, 1940–1952. [[CrossRef](#)]
64. Schenkendorf, R.; Mangold, M. Online model selection approach based on Unscented Kalman Filtering. *J. Process. Control.* **2013**, *23*, 44–57. [[CrossRef](#)]
65. Vanlier, J.; Tiemann, C.A.; Hilbers, P.A.; van Riel, N.A. Optimal experiment design for model selection in biochemical networks. *BMC Syst. Biol.* **2014**, *8*, 20. [[CrossRef](#)] [[PubMed](#)]
66. Martin-Casas, M.; Mesbah, A. Discrimination Between Competing Model Structures of Biological Systems in the Presence of Population Heterogeneity. *IEEE Life Sci. Lett.* **2016**, *2*, 23–26. [[CrossRef](#)]
67. Burnham, K.P.; Anderson, D.R. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*, 2nd ed.; OCLC: ocm48557578; Springer: New York, NY, USA, 2002.
68. Lorenz, S.; Diederichs, E.; Telgmann, R.; Schütte, C. Discrimination of Dynamical System Models for Biological and Chemical Processes: Discrimination of Dynamical System Models. *J. Comput. Chem.* **2007**, *28*, 1384–1399. [[CrossRef](#)]

69. Rao, M.M.; Swift, R.J. *Probability Theory with Applications*; Springer: Berlin, Germany, 2006.
70. Xie, X.; Krewer, U.; Schenkendorf, R. Robust Optimization of Dynamical Systems with Correlated Random Variables Using the Point Estimate Method. *IFAC-PapersOnLine* **2018**, *51*, 427–432. [[CrossRef](#)]
71. Xie, X.; Schenkendorf, R. Robust Process Design in Pharmaceutical Manufacturing under Batch-to-Batch Variation. *Processes* **2019**, *7*, 509. [[CrossRef](#)]
72. Bezanson, J.; Edelman, A.; Karpinski, S.; Shah, V.B. Julia: A Fresh Approach to Numerical Computing. *SIAM Rev.* **2017**, *59*, 65–98. [[CrossRef](#)]
73. Rackauckas, C.; Nie, Q. Differential Equations. J1—A Performant and Feature-Rich Ecosystem for Solving Differential Equations in Julia. *J. Open Res. Softw.* **2017**, *5*, 15. [[CrossRef](#)]
74. Dunning, I.; Huchette, J.; Lubin, M. JuMP: A Modeling Language for Mathematical Optimization. *SIAM Rev.* **2017**, *59*, 295–320. [[CrossRef](#)]
75. Wächter, A.; Biegler, L.T. On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. *Math. Program.* **2006**, *106*, 25–57. [[CrossRef](#)]
76. Ohs, R.; Leipnitz, M.; Schöpping, M.; Spiess, A.C. Simultaneous Identification of Reaction and Inactivation Kinetics of an Enzyme-catalyzed Carboligation. *Biotechnol. Prog.* **2018**, *34*, 1081–1092. [[CrossRef](#)]
77. Fang, J.J.; Hung, C.L. An Improved Parameterization Method for B-Spline Curve and Surface Interpolation. *Comput.-Aided Des.* **2013**, *45*, 1005–1028. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).