This is a preprint of the Materials accepted for publication in "Theoretical Foundations of Chemical Engineering", 2014, Vol. 48, 5
www.maikonline.com

# Parameter Identification for Ordinary and Delay Differential Equations by Using Flat Inputs

**René Schenkendorf, Michael Mangold**

*Max Planck Institute for Dynamics of Complex Technical Systems*

*Sandtorstraße 1, 39106 Magdeburg, Germany*

*e-mail: mangold@mpi-magdeburg.mpg.de*

**Abstract:**

The concept of differential flatness has been widely used for nonlinear controller design. In this contribution, it is shown that flatness may also be a very useful property for parameter identification. An identification method based on flat inputs is introduced. The treatment of noisy measurements and the extension of the method to delay differential equations are discussed. The method is illustrated by two case studies: the well-known FitzHugh-Nagumo equations and a virus replication model with delays.

**Keywords:** parameter identification, differential flatness, ordinary differential equations, delay differential equations, inverse model

## 1 Introduction

Virtually all mathematical models of chemical or biochemical processes contain unknown parameters that have to be identified from experimental data. Parameter identification is therefore a central step during the development of mathematical models and a prerequisite for model based process control and process design.

In most cases, parameters are identified from experiments as shown in Figure

1 (a), see e.g. [1]. A process model $\hat{\Sigma}$ is set up to reproduce the experiments

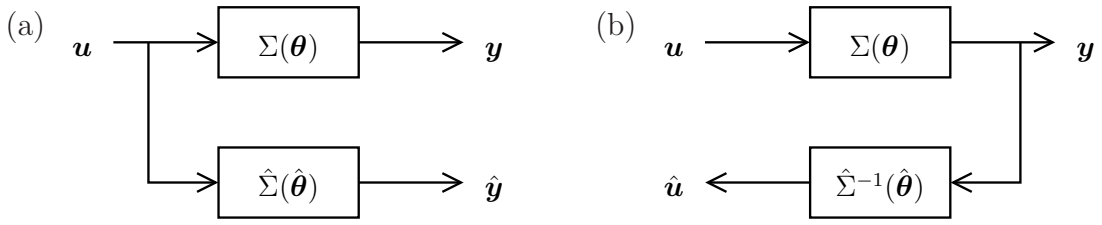



Figure 1: Alternative approaches for parameter identification: (a) by fitting a simulated system output $\hat{y}$ to the measurements $y$; (b) by fitting a reconstructed input $\hat{u}$ to the true input $u$ using an inverse process model.

in simulations, using the operation conditions or inputs $\mathbf{u}$ as in reality and an estimate $\hat{\theta}$ of the unknown model parameters $\theta$. In most cases, this requires a numerical solution of the differential equations of the model. The simulated process output or measurement $\hat{\mathbf{y}}$ is then compared to the true output $\mathbf{y}$. If there are deviations, the estimate of the parameters is refined iteratively in an optimization step. Nowadays, the numerical solution of differential or differential algebraic systems is usually not very challenging, but it may become tricky when the guesses of the parameter values or of unknown initial conditions are poor and far away from the true values. Problems arise especially, when the system equations contain delay terms, because then initial functions over a delay interval before the experiment's starting time have to be estimated and because the numerical solution of delay differential equations in general is more difficult. Further, numerical integration underlying an optimization procedure may be quite expensive and consume the biggest share of the spent computation time. Finally, the dependence of the system outputs on the model parameters is often strongly nonlinear, resulting in non-convex cost functions with local minima for parameter identification. The mentioned difficulties motivate the search for alternative approaches for parameter identification.

One possibility, which is shown in Figure 1 (b), is to look at an inverse system model $\hat{\Sigma}^{-1}$. Instead of computing simulated system outputs $\hat{\mathbf{y}}$ from given inputs $\mathbf{u}$, one could use the inverse model to compute estimated system inputs $\hat{\mathbf{u}}$ from given outputs $\mathbf{y}$ and to use the difference between $\mathbf{u}$ and $\hat{\mathbf{u}}$ for parameter fitting. This approach is rarely used and only makes sense if it is much easier to solve the inverse model $\hat{\Sigma}^{-1}$ than the usual model $\hat{\Sigma}$. But actually, there is a large class of systems, so-called differentially flat systems, with exactly this property. The concept of differential flatness was initially introduced by Fliess et al. [2]. It has received a lot of interest in control theory over the last two decades [3, 4], with the majority of applications lying in the area of tracking control of a wide range of technical systems [5, 6, 7, 8, 9, 10, 11]. Differentially flat systems have the property that the states and inputs can be expressed directly in terms of the flat outputs and a finite number of their derivatives [12]. A more formal definition is given in Section 2.1. Flatness is an attractive property for parameter identification based on the inverse model, because computing the inputs of a flat system does not require any numerical integration, but in the worst case the numerical solution of a set of algebraic equations. The topic of this paper is to study the use of flat inputs for parameter identification. Flat inputs mean input variables that turn given outputs to flat outputs and hence enable a differential parametrization of the model [13].

Vassilev et al. [14] exploited flatness properties of a precipitation reaction model in order to identify a single physical model parameter, which is hardly accessible to direct measurements. They made use of the fact that in their case the unknown parameter happens to be a flat input of the system and obtained convincing estimation results for that parameter. The drawback of the method by Vassilev et al. is that it requires a very special model structure and that it requires as

many measurement variables as there are unknown parameters, whereas the flat input method presented here is applicable to a much larger class of systems.

A related identification method was introduced by Fliess et al. [15] for systems, whose parameters can be expressed by algebraic equations depending on the inputs, the outputs and time derivatives of both. The charm of this method is that it does not require any numerical optimization, but only the solution of an algebraic set of equations, and hence is very fast. As a disadvantage, it may require a high number of derivatives of inputs and outputs, if the number of unknown parameters is large compared to the number of inputs and outputs.

The flat input method has similarities to methods of functional data analysis and principal differential analysis (PDA). Early publications in that field [16, 17] assume all model states to be measurable. The states are approximated by splines or similar functions and fitted to the measurements, introducing "nuisance" parameters in addition to the model parameters. The residuals resulting from inserting the state approximations into the differential equations of the model are minimized in order to obtain estimates of the model parameters. Over the years, the method has been more and more refined in order to cope with measurement imperfections and increase accuracy [18, 19, 20, 21, 22, 23, 24] . Powerful techniques of iterated and cascaded parameter estimation approaches have been developed in this context, which also prove to be useful for the flat input approach. The main difference, however, is that in PDA all states are parametrized independently, resulting in a quite large number of nuisance parameters, and that the technique is mainly used for systems that have hardly hidden unmeasured states. In contrast, the flat input method only introduces nuisance parameters for the measured outputs - internal system states and inputs are then automatically parametrized, as well. This reduces the number of parameters to be identified

especially for systems with many unmeasured states. Another difference between PDA and the flat input method is that in this work system inputs are reconstructed. The deviations of the reconstructed inputs from the true inputs are taken as an indicator for the quality of the parameter estimate and enter the cost function for optimization.

The flat input method for parameter identification also bears some similarity to the method of differential elimination [25, 26, 27, 28, 29]. Differential elimination uses the theory of Gröbner bases to eliminate unobserved variables from systems of differential equations that can be expressed as differential polynomials. It has been applied successfully to the estimation of parameters [25, 26] as well as to determine global identifiability of model parameters [27, 29]. Differential elimination requires reaction kinetics with certain structures (polynomial or fractional expressions), because the model equations are solved analytically for the unobserved variables. This is a difference to the flat input method that only requires implicit algebraic equations for the states and system inputs and hence is applicable to a wider class of systems.

Section 2 of this paper presents the method of flat inputs for parameter identification of differential equations with or without delays. Section 3 illustrates the properties of the method by two case studies.

# 2 Method

The next section gives a brief introduction to differentially flat systems and to the proposed identification method. Section 2.2 addresses the problem of how to find (potentially fictitious) inputs that turn a system into a flat system. Section 2.3 contains the treatment of noisy measurements in the context of the new

identification method. The extension of the method to systems with delay is discussed in Section 2.4.

## 2.1 Differentially Flat Systems and Parameter Identification

This work mainly concerns input-affine systems of the following type:

$$\dot{\mathbf{x}}(t) \;=\; \mathbf{f}\left(\mathbf{x}(t), t, \theta\right) + \sum_{i=1}^{m} \boldsymbol{\gamma}_i(\mathbf{x}(t), \theta)\, u_i(t); \quad \mathbf{x} \in \mathbb{R}^n \tag{1}$$

where $\mathbf{x}(t)$ is the state vector, $\mathbf{u}(t) = (u_1, \ldots, u_m)^T$ is the input vector, and $\theta$ is a vector of constant model parameters. A system of type (1) is called a differentially flat system, if the following conditions hold [2, 12, 5]:

1. There is a so-called flat output $\mathbf{y}(t) = (y_1(t), \ldots, y_m(t)^T$ that can be calculated from the state $\mathbf{x}(t)$, the input $\mathbf{u}(t)$, and time derivatives of $\mathbf{u}(t)$, i.e.

$$y_i(t) = h_i(\mathbf{x}(t), \mathbf{u}(t), \dot{\mathbf{u}}(t), \ldots, \mathbf{u}^{(\eta)}); \quad i = 1, \ldots, m \tag{2}$$

2. All state variables $\mathbf{x}(t)$ and all input variables $\mathbf{u}(t)$ can be calculated from $\mathbf{y}(t)$ and a finite number of their time derivatives, i.e. there are relations

$$\mathbf{x}(t) \;=\; \boldsymbol{\Psi}_x\left(\mathbf{y}(t), \dot{\mathbf{y}}(t), \ldots, \mathbf{y}^{(n-1)}(t), \theta\right) \tag{3}$$

$$\mathbf{u}(t) \;=\; \boldsymbol{\Psi}_u\left(\mathbf{y}(t), \dot{\mathbf{y}}(t), \ldots, \mathbf{y}^{(n)}(t), \theta\right) \tag{4}$$

   The second equation is an input output representation of the system, which has the special property that time derivatives of $\mathbf{u}(t)$ do not occur explicitly.

3. There does not exist any relation

$$Q\left(\mathbf{y}(t), \dot{\mathbf{y}}(t), \ldots, \mathbf{y}^{(q)}(t)\right) = 0,$$

i.e. the components of $\mathbf{y}(t)$ are differentially independent.

The third property is always fulfilled, if the number of inputs matches the number of outputs [5].

The representation (3,4) allows to determine $\mathbf{x}(t)$ and $\mathbf{u}(t)$ from $\mathbf{y}(t)$ in a very efficient way, without having to solve a differential equation. In this sense, the solution of the inverse system model (calculation of $\mathbf{x}(t)$ and $\mathbf{u}(t)$ from given $\mathbf{y}(t)$) is easier than the solution of the original model (calculation of $\mathbf{x}(t)$ and $\mathbf{y}(t)$ from given $\mathbf{u}(t)$)

The flatness property has been used frequently for controller design. In that case, the physical actuators are usually given, i.e. the functions $\boldsymbol{\gamma}_i(.)$ are determined by the physical problem. Then the question comes up, what output functions $h_i(.)$ could be used such that the resulting output $\mathbf{y}(t)$ is a flat output and flatness based control methods are applicable. The constructed flat outputs need not be identical to physical measurements, but may be fictitious quantities.

In order to exploit differential flatness for parameter identification, an opposite problem has to be solved. Now, the physical measurement variables are taken as outputs. The output equations are no longer chosen freely, but are taken from the sensor model as

$$y_i(t) = h_i(\mathbf{x}(t), \theta); \quad i = 1, \ldots, m \tag{5}$$

Then, the question is, what actuators, i.e. what input functions $\boldsymbol{\gamma}_i(.)$ are required to turn the physical outputs $\mathbf{y}(t)$ into flat outputs that enable the differential parametrization of the model. The input signals $u_i(t)$ that belong to these actuators are called flat inputs [13]. The flat inputs and corresponding actuators need not exist in the physical reality. For parameter identification, they may be purely fictitious quantities, whose nominal values in physical reality are of course identical to zero. It should be noted that the flatness property of the system needs not be fulfilled with respect to the physical control inputs available in reality. Actually, the physical control inputs are irrelevant for the parameter identification strategy presented in the following. As long as the physical control inputs do not coincide with fictitious flat inputs, they may be considered as known time dependent functions that are included in the function $\mathbf{f}\left(\mathbf{x}(t), t, \theta\right)$.

If one assumes for the moment perfect noise-free measurements, one can reconstruct the flat system inputs from (4) as

$$\hat{\mathbf{u}}(t, \hat{\theta}) = \boldsymbol{\Psi}_u \left( \mathbf{y}(t), \dot{\mathbf{y}}(t), \ldots, \mathbf{y}^{(n)}(t), \hat{\theta} \right) \tag{6}$$

where $\hat{\theta}$ is an estimate of the parameter vector. Note that due to (3) the differential equations for all states of the system are always fulfilled exactly for arbitrary choices of $\mathbf{y}$ and $\hat{\theta}$. Wrong parameter estimates only become visible in an deviation of the reconstructed flat inputs from their nominal value. Therefore, the reconstructed inputs may be used for parameter identification. Instead of minimizing the conventional least squares cost function

$$J_y(\hat{\theta}) = \int_0^T \left\| \mathbf{y}(t) - \hat{\mathbf{y}}(t, \hat{\theta}) \right\|^2 dt, \tag{7}$$

where $T$ is the duration of the experiment, one could define a new cost function

$$J_u(\hat{\theta}) = \int_0^T \left\| \mathbf{u}(t) - \hat{\mathbf{u}}(t, \hat{\theta}) \right\|^2 dt, \tag{8}$$

which also assumes a global minimum for $\hat{\theta} = \theta$. The nominal values of the flat inputs $\mathbf{u}(t)$ are identical to zero in most cases, with the only exception when a fictitious flat input happens to coincide with a physical control input. The expressions (6), (8) require some modifications, if noisy measurement data are used. This case will be discussed in Section 2.3.

*Example I.* Consider the simple linear autonomous system

$$\frac{d}{dt} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ \theta & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{9}$$

with an unknown parameter $\theta$ to be estimated from a measured output

$$y = x_1. \tag{10}$$

For the identification, we use the simulation model

$$\frac{d}{dt} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ \hat{\theta} & 0 \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} \tag{11}$$

$$\hat{y} = \hat{x}_1 \tag{12}$$

The simulation model can be turned into a flat system with flat output $\hat{y}$ by adding an input matrix $\boldsymbol{B} = (0,\, 1)^T$ and a corresponding fictitious input $\hat{u}$:

$$\frac{d}{dt}\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} -1 & 1 \\ \hat{\theta} & 0 \end{pmatrix}\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix}\hat{u} \tag{13}$$

$$\hat{y} = \hat{x}_1 \tag{14}$$

From (13), (14) one obtains

$$\hat{x}_1 = \hat{y} \tag{15}$$

$$\hat{x}_2 = \hat{y} + \dot{\hat{y}} \tag{16}$$

$$\hat{u} = \ddot{\hat{y}} + \dot{\hat{y}} - \hat{\theta}\left(\hat{y} + \dot{\hat{y}}\right), \tag{17}$$

which confirms differential flatness. Eq. (17) can now be used to test if the estimated parameter value $\hat{\theta}$ agrees with the true parameter value $\theta$, when instead of the simulated measurement $\hat{y}$ the true measurement value is inserted into (17), i.e. (17) is replaced by

$$\hat{u} = \ddot{y} + \dot{y} - \hat{\theta}\,(y + \dot{y}) \tag{18}$$

The cost function used for parameter identification is

$$J_u(\hat{\theta}) = \int_0^T \hat{u}(t)^2 dt. \tag{19}$$

In order to see that this cost function has a global minimum at $\hat{\theta} = \theta$, note that one always has

$$u = \ddot{y} + \dot{y} - \theta\,(y + \dot{y}) \equiv 0, \tag{20}$$

because the flat input does not exist in reality, Therefore (18) may also be written as

$$\hat{u} = \left(\theta - \hat{\theta}\right)(y + \dot{y}) \tag{21}$$

and (19) becomes

$$J_u(\hat{\theta}) = \left(\theta - \hat{\theta}\right)^2 \int_0^T (\dot{y}(t) + y(t))^2 \, dt \tag{22}$$

Obviously, $J_u(\hat{\theta}) = 0$ for $\hat{\theta} = \theta$, and $J_u(\hat{\theta}) > 0$ for $\hat{\theta} \neq \theta$ and $y(t) \not\equiv 0$.

The parameter identification based on the use of flat inputs may offer a number of advantages, some of which it has in common with PDA methods:

1. The flat input approach may facilitate the parameter identification, if the sensitivity of an input $u$ with respect to this parameter for a given output $y$ is larger than the sensitivity of an output $y$ with respect to the same parameter for a given input $u$.

   *Example II.* Consider the static SISO system

   $$y = f(u, \theta)$$

   with

   $$\left.\frac{\partial y}{\partial \theta}\right|_{u=\text{const.}} = \frac{\partial f}{\partial \theta}$$

   and

   $$\left.\frac{\partial u}{\partial \theta}\right|_{y=\text{const.}} = -\left(\frac{\partial f}{\partial u}\right)^{-1} \frac{\partial f}{\partial \theta}.$$

   Parameter identification based on the inverse approach may be beneficial, if $|(\partial f/\partial u)^{-1}| \gg 1$, because then the input $u$ is more sensitive towards the

parameter $\theta$ than the output $y$.

2. The flat input approach may lead to more nicely shaped cost functions for the parameter identification and thus may avoid the optimization to get stuck in a local minimum. This will be illustrated by one of the case studies in the last section of this article and by a very simple, purely academic example below.

*Example III.* Consider the static test system

$$y = f(\theta, u) := -\left(\theta - p_0(u)\right)\,\Big(\left(\theta - p_0(u)\right)^2 + p_1(u)\Big) + p_2(u)$$

with

$$\begin{aligned} p_0(u) &= u \\ p_1(u) &= -3\left(1 - |u|\right) \\ p_2(u) &= 3u. \end{aligned}$$

A graph of the system is given in Figure 2. Assume that $\theta$ is to be identified from a measurement value $y^{meas} = 10$ for $u = 0$, which corresponds to a parameter value $\theta \approx -2.7$. The conventional least squares approach results in a cost function

$$J_y(\hat{\theta}) = \left(y^{meas} - \hat{y}\right)^2 = \Big(10 - f(\hat{\theta}, 0)\Big)^2 = \Big(10 - \hat{\theta}^3 - 3\hat{\theta}\Big)^2,$$

which has a local minimum at $\hat{\theta} = 1$, as depicted in Figure 3 (a). In

contrast, the inverse model approach uses the cost function

$$J_u(\hat{\theta}) = \left(u - \hat{u}(\hat{\theta})\right)^2 = \left(f^{-1}(y^{meas}, \hat{\theta})\right)^2.$$

From Figure 2 it can be seen that $\hat{u}$ grows monotonically with $\hat{\theta}$ for $y = y^{meas}$. Therefore $J_u$ has a single global minimum at $\hat{\theta} = \theta$. Furthermore, Figure 3 (b) shows that $J_u$ is a convex function and hence well suited for numerical optimization.

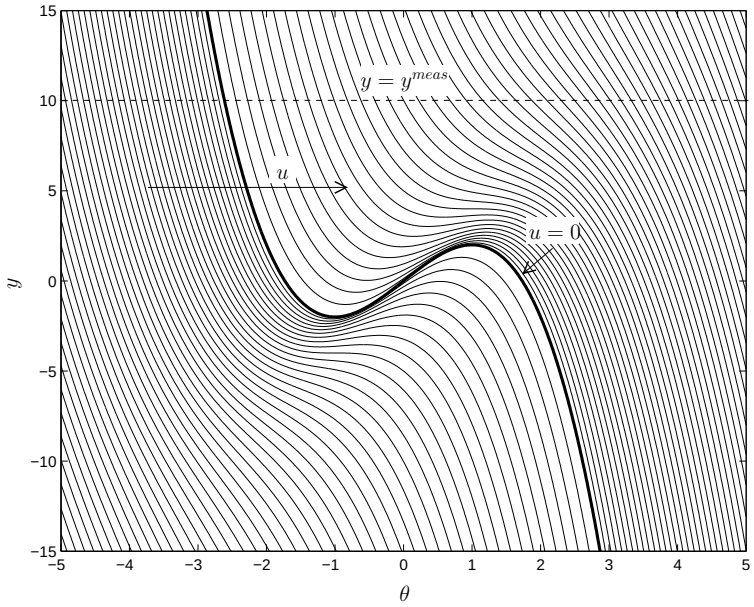



Figure 2: Visualization of the system considered in Example 2; the thick lines marks the case $y = f(\theta, 0)$, the thin lines show $y = f(\theta, u)$ for $u \neq 0$.

3. Computing flat inputs from the outputs is much cheaper than computing the outputs from given inputs. In the first case, there are often analytical expressions for the flat inputs or the flat inputs can be computed numerically from low order algebraic equations, while in the latter case a numerical integration is needed. When using the flat input method, it is also not necessary to determine or estimate initial conditions explicitly, which

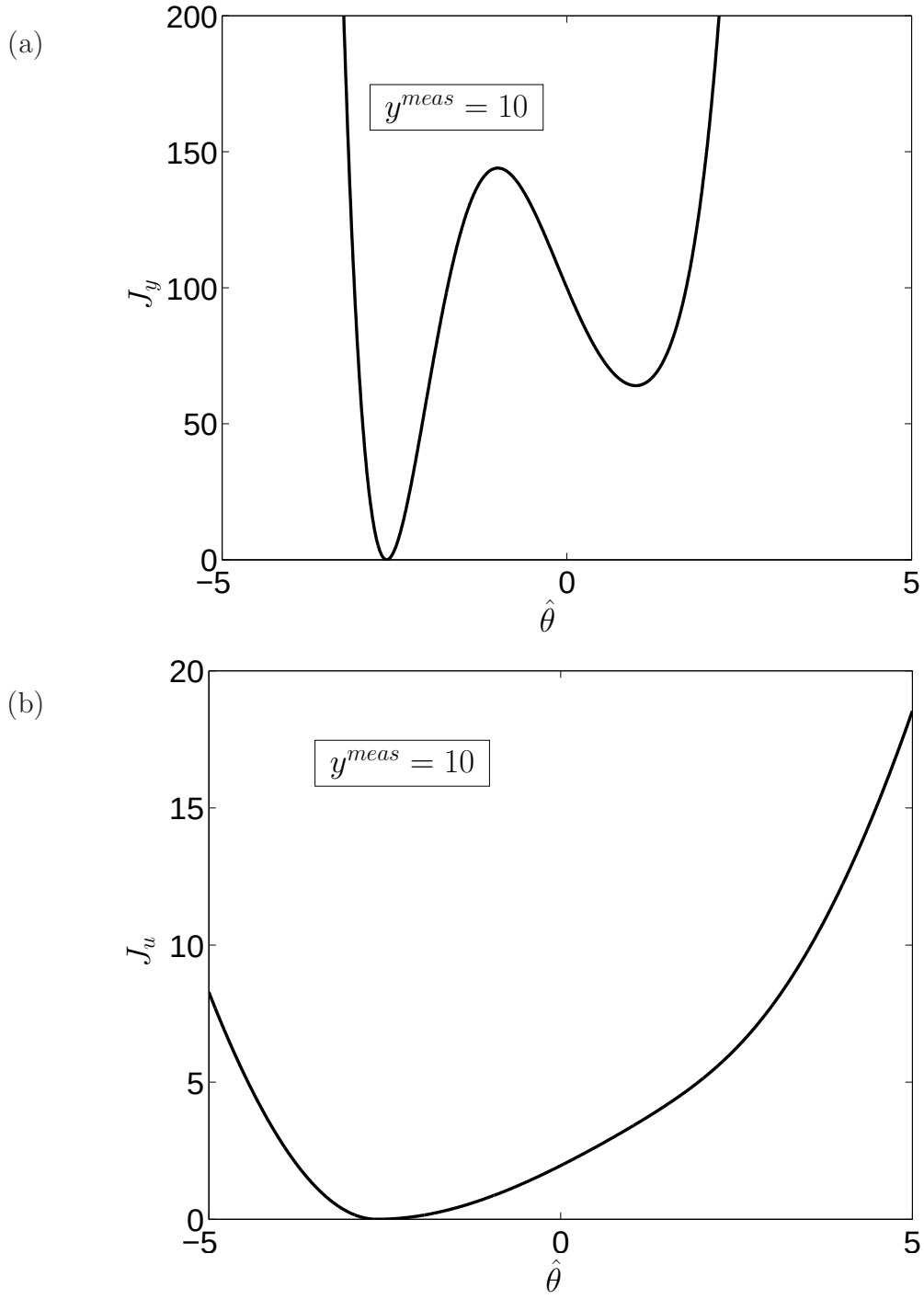



Figure 3: Objective functions for Example 2, if (a) the conventional least squares approach is used for parameter identification and (b) if the parameter $\theta$ is identified from the inverse model.

is advantageous in cases where initial conditions are not known.

4. Especially for dynamical systems with delays, whose numerical solution still is not trivial, it is much easier to use the inverse model for parameter identification and to exploit the flatness property.

## 2.2 Determination of Flat Inputs

When trying to exploit flatness properties for parameter identification, a crucial point is to determine suitable flat inputs. Sufficient and necessary conditions for the existence of flat inputs are not known in the general MIMO case [30]. However, Waldherr and Zeitz [13, 30] developed an elegant and easy-to-use method for the construction of flat inputs for the large class of observable systems, which is summarized in the following. The method makes use of the concept of observability co-distribution and observability indices, as defined below.

**Definition[30, 31]** The system (1) with output equation (5) is said to have *observability indices* $\boldsymbol{\kappa} = (\kappa_1, \ldots, \kappa_m)$ at $\mathbf{x}_0 \in \mathbb{R}^n$, if $\sum_{i=1}^{m} \kappa_i = n$, $\kappa_i \geq 0, i = 1, \ldots, m$, and there exists a neighborhood $\mathcal{X}$ of $\mathbf{x}_0$ such that the *observability co-distribution*

$$d\mathcal{O}_\kappa = \text{span}\left\{\mathrm{dL_f^j h_i}, 1 \leq \mathrm{i} \leq \mathrm{m}, 0 \leq \mathrm{j} \leq \kappa_\mathrm{i} - 1\right\}$$

is of constant dimension equal to $n$ in $\mathcal{X}$.

In the above definition, $L_f h$ denotes the Lie derivative, which is defined as

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i(\mathbf{x}),$$

$dL_f h$ denotes the gradient $\partial L_f h/\partial \mathbf{x}$, and span$\{.\}$ stands for the space spanned by these gradient vectors.

In order to construct flat inputs for a system with observability indices $\boldsymbol{\kappa}$, Waldherr and Zeitz [30] suggest to define $m$ vector fields $\tau_k(\mathbf{x})$, $k = 1, \ldots, m$ as solutions of the $m\,n$ equations

$$\begin{aligned} L_{\tau_k} L_f^r h_i(\mathbf{x}) &= 0 \quad \text{for } 0 \leq r \leq \kappa_i - 2 \\ L_{\tau_k} L_f^{\kappa_i - 1} h_i(\mathbf{x}) &= \delta_{ik} \end{aligned} \tag{23}$$

for $i = 1, \ldots, m$, with the Kronecker symbol $\delta_{ik}$, which is equal to one for $i = k$ and otherwise zero; $L_f^r h_i(\mathbf{x})$ means that the Lie derivative operator is applied $r$ times. Flat input vector fields $\gamma_j$ can then be chosen as

$$\gamma_j(\mathbf{x}) = \sum_{k=1}^{m} \alpha_{kj}(\mathbf{x}) \tau_k(\mathbf{x}),\ j = 1, \ldots, m \tag{24}$$

with arbitrary scalar functions $\alpha_{kj}(\mathbf{x})$ such that the matrix

$$\boldsymbol{A}(\mathbf{x}) = \begin{pmatrix} \alpha_{11}(\mathbf{x}) & \cdots & \alpha_{1m}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \alpha_{m1}(\mathbf{x}) & \cdots & \alpha_{mm}(\mathbf{x}) \end{pmatrix} \tag{25}$$

is non-singular [30].

*Example IV.* The construction of flat inputs is illustrated by a model for virus replication similar to Möhler et al. [32], which will be studied in more detail in Section 3.2. Uninfected Madin-Darby canine kidney (MDCK) cells $U_c(t)$ are infected by active viruses $V_i(t)$. Infected MDCK cells $I_c(t)$ release active and inactive virus particles $V_i(t)$ and $V_d(t)$, respectively. The active virus particles either infect the remaining uninfected cells or are degraded to inactive virions.

It is assumed that the concentrations of active and inactive virus particles are measurable. The preliminary model equations, which will be extended by delay terms in Section 3.2, read:

$$\begin{aligned}
\dot{U}_c(t) &= \theta_6 \frac{C_{max} - (U_c(t) + I_c(t))}{C_{max}} U_c(t) - \theta_1 U_c(t) V_i(t) \\
\dot{I}_c(t) &= \theta_1 U_c(t) V_i(t) - \theta_2 I_c(t) \\
\dot{V}_i(t) &= \theta_3 I_c(t) - \theta_4 V_i(t) - \theta_1 U_c(t) V_i(t) \\
\dot{V}_d(t) &= \theta_5 I_c(t) + \theta_4 V_i(t)
\end{aligned} \tag{26}$$

$$\begin{aligned}
y_1(t) &= h_1(U_c, I_C, V_i, V_d) = V_i(t) \\
y_2(t) &= h_2(U_c, I_C, V_i, V_d) = V_d(t)
\end{aligned} \tag{27}$$

It is easily verified that the observability matrix has full rank if the observability indices are chosen as $\kappa_1 = 2, \kappa_2 = 2$. Evaluation of condition (23) gives a set of linear equations for $\boldsymbol{\tau}_1$:

$$\begin{aligned}
L_{\tau_1} h_1 = 0 \quad &\Rightarrow \quad \tau_{1,3} = 0 \\
L_{\tau_1} L_f h_1 = 1 \quad &\Rightarrow \quad -\theta_1 V_i \tau_{1,1} + \theta_3 \tau_{1,2} - (\theta_4 + \theta_1 U_c) \tau_{1,3} = 1 \\
L_{\tau_1} h_2 = 0 \quad &\Rightarrow \quad \tau_{1,4} = 0 \\
L_{\tau_1} L_f h_2 = 0 \quad &\Rightarrow \quad \theta_5 \tau_{1,2} + \theta_4 \tau_{1,3} = 0
\end{aligned} \tag{28}$$

and for $\boldsymbol{\tau}_2$:

$$\begin{aligned}
L_{\tau_2} h_1 = 0 \quad &\Rightarrow \quad \tau_{2,3} = 0 \\
L_{\tau_2} L_f h_1 = 0 \quad &\Rightarrow \quad -\theta_1 V_i \tau_{2,1} + \theta_3 \tau_{2,2} - (\theta_4 + \theta_1 U_c) \tau_{2,3} = 0 \\
L_{\tau_2} h_2 = 0 \quad &\Rightarrow \quad \tau_{2,4} = 0 \\
L_{\tau_2} L_f h_2 = 1 \quad &\Rightarrow \quad \theta_5 \tau_{2,2} + \theta_4 \tau_{2,3} = 1.
\end{aligned} \tag{29}$$

The solutions of these equations read $\boldsymbol{\tau}_1 = (-1/(\theta_1 V_i), 0, 0, 0)^T$ and $\boldsymbol{\tau}_2 = (-\theta_3/(\theta_5 \theta_1 V_i), \theta_3/\theta_5, 0, 0)^T$. One possible choice of flat input vector fields is now

$$\begin{aligned} \gamma_1 &= (-\theta_1 V_i)\boldsymbol{\tau}_1 &&= (1,0,0,0)^T \\ \gamma_2 &= \theta_3 \boldsymbol{\tau}_1 + \frac{\theta_5}{\theta_3}\boldsymbol{\tau}_2 &&= (0,1,0,0)^T, \end{aligned} \tag{30}$$

but of course there are many degrees of freedom in this choice and hence many other possibilities. Once again, it should be noted that these inputs are purely virtual. There is no need and usually no possibility to implement them in reality. They are only needed to indicate deviations of the simulation model from the true process caused by poor parameter estimates.

As an alternative approach to the construction method by Waldherr and Zeitz, one can try to guess flat inputs from the directed graph of the system, with $n$ nodes $v_i$ for the states and edges $e_{i,j}$ from node $v_i$ to $v_j$ indicating that the right-hand side of the ODE for state $j$ depends on state $i$. The directed graph of the virus replication system is shown in Figure 4. As an heuristic rule of thumb, the flat inputs should act on those nodes that are most distant from the outputs. In the virus replication example, those are the states $I_c$ and $U_c$, which is in agreement with the result of the systematic flat input construction discussed above.

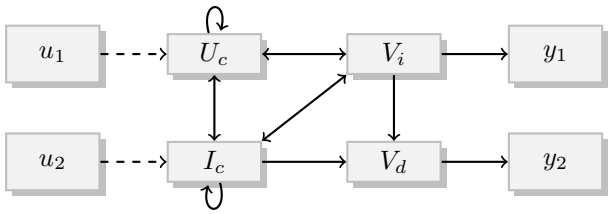



Figure 4: Directed graph of virus replication model from Example IV with nodes for states, inputs, and outputs.

## 2.3 Treatment of Measurement Noise

In most applications, the assumption of continuous noise-free measurements is unrealistic. Usually, measurement data are only available at discrete time points $t_k$ and corrupted by measurement noise. The interpolation and differentiation of noisy data is a very challenging task. A large amount of literature deals with this problem and many different approaches have been proposed, see e.g. [33, 34] and references therein. In principle, most of the existing methods could be combined with the flat input method for parameter identification. In this work, a classical interpolation approach is applied, as it is computationally less demanding and has been found efficient for many applications [34]. This approach replaces the true measurements $\mathbf{y}$ in (6) by surrogate output functions $\hat{\mathbf{y}}$ and computes $\hat{\mathbf{u}}$ from

$$\hat{\mathbf{u}}(t) = \mathbf{\Psi}_u \left( \hat{\mathbf{y}}(t), \dot{\hat{\mathbf{y}}}(t), \ldots, \hat{\mathbf{y}}^{(n)}(t), \hat{\theta} \right) \tag{31}$$

The concept of functional data analysis [19, 20, 23] provides methods for determining appropriate surrogate output functions of the form

$$\hat{\mathbf{y}}(t) = \boldsymbol{c}^T \, \Phi(t). \tag{32}$$

The time dependent functions $\Phi(t)$ are a set of known basis functions chosen before-hand. B-splines [19] are most commonly used for this purpose, but also other approaches like neural networks or wavelet concepts may be applicable. Time derivatives of $\hat{\mathbf{y}}(t)$, which are required to evaluate (31), are easily obtained by deriving (32) with respect to time. In case of B-splines, this can be done analytically.

The coefficients $\boldsymbol{c}$ in (32) are chosen in order to fit the basis functions to the

measurements in an optimal way. To make the surrogate function robust against measurement noise, a penalty term for the surrogate output's curvature is often added to the cost function of the optimization problem [18, 19, 20]:

$$\arg\ \min_{c} J_{\hat{y}}(c) = \lambda \sum_{k=1}^{K} \|\mathbf{y}(t_k) - \hat{\mathbf{y}}(t_k)\|^2 + (1-\lambda) \int_0^T \left\| \frac{d^2\hat{\mathbf{y}}(t)}{dt^2} \right\|^2 dt \qquad (33)$$

The weighting factor $\lambda \in [0, 1]$ has to be adapted in dependence of the analyzed data $y(t_k)$ and is usually determined by hand. This is a quite critical step, as the parameter $\lambda$ may influence the outcome of the surrogate function $\hat{\mathbf{y}}(t)$ strongly.

*Example V.* For the purpose of illustration, measurement data are generated by a sine function which is corrupted by additive Gaussian noise, $y(t_k) = \sin(t_k) + v_k$. Figure 5 shows that only with a deliberately chosen weighting factor ($\lambda = 0.8$) the original sine function can be approximated adequately.

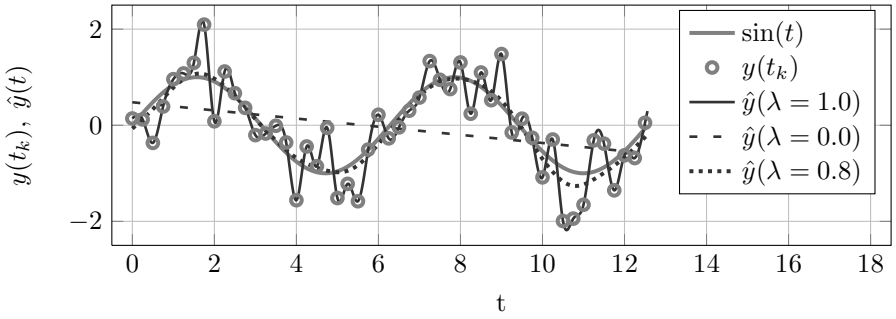



Figure 5: Influence of the additive penalty term $(1-\lambda) \int_0^T \|d^2\hat{y}(t)/dt^2\|^2 dt$ on the surrogate function $\hat{y}(t)$. In the presence of discrete noisy data, $y(t_k) = \sin(t_k) + v_k$; $t_{k+1} - t_k = 0.25$; $v_k \sim \mathcal{N}(0, 0.2)$, an appropriate function $\hat{y}(t)$ can be derived using a proper $\lambda$ value ($\lambda = 0.8$).

To avoid the tedious work of tuning $\lambda$ manually, one can incorporate the choice of $\lambda$ into the framework of parameter identification. The weighting factor $\lambda$ is then considered as another optimization variable in addition to the unknowns of the

surrogate functions $\boldsymbol{c}$ and the model parameters $\hat{\theta}$ when evaluating the cost function (8). The resulting new minimization problem for parameter identification reads

$$\arg \min_{c,\lambda,\hat{\theta}} J_u(c,\lambda,\hat{\theta}) = \int_0^T \left\| \hat{\mathbf{u}}(\hat{\mathbf{y}}(\boldsymbol{c},\lambda,t),\hat{\theta}) - \mathbf{u}(t) \right\|^2 dt, \tag{34}$$

where $\mathbf{u}(t) \equiv 0$ for an autonomous system. Although solving (34) requires the underlying solution of the minimization problem (33), this approach seems feasible, because both optimizations are not very costly. If e.g. splines are used as basis functions, very efficient solutions are available for the fitting problem (33). The solution of (34) is also quite cheap, as no differential equations have to be solved to obtain $\hat{\mathbf{u}}$.

In cases of strongly fluctuating process dynamics, a penalty term minimizing the curvature might be inappropriate. As an alternative, one can replace the penalty term in (33) by a term that explicitly incorporates the deviation of the reconstructed inputs $\hat{\mathbf{u}}$ from their nominal values:

$$\arg \min_{c,\hat{\theta}} J_u(c,\hat{\theta}) = \lambda \sum_{k=1}^{K} \|\hat{\mathbf{y}}(t_k,\boldsymbol{c}) - \mathbf{y}(t_k)\|^2 + (1-\lambda) \int_0^T \|\hat{\mathbf{u}}(t,\hat{\theta}) - \mathbf{u}(t)\|^2 dt \tag{35}$$

In this approach, the inverse model acts as a model based filter that minimizes the influence of measurement noise when determining $\hat{\mathbf{y}}(t)$. The cost function (35) has some similarity to the objective functions used for iteratively refined principal differential analysis [18, 20]. The coefficient $\lambda$ in (35) is a design parameter with values between 0 and 1. A value close to 1 means that only small differences between surrogate output and true measurements are tolerated, while accepting a larger deviation of the reconstructed input from its nominal value. Such a choice may make sense, if the measurements are considered as highly reliable,

but the process model is seen only as a crude approximation of reality. On the other hand, if one has trust in the accuracy of the model, but measurements are corrupted by a large amount of noise, a value close to 0 should be chosen for $\lambda$. By varying $\lambda$ between 0 and 1 a Pareto front of optimal solutions is obtained.

## 2.4 Treatment of Systems with Delay

The proposed parameter identification method is easily applicable to systems with delay, if these systems belong to the class of so-called $\delta$-flat systems as introduced by Mounier and Rudolph [35]. The inputs and states of $\delta$-flat systems can be expressed as algebraic functions of the outputs and their time derivatives, similar to Equation (3) and (4). The only difference is that the outputs are not only evaluated at the current time $t$, but also at times shifted towards the past or the future by multiples of the delay times, i.e. at time points $t + q\,\tau_i$, where $q$ is an integer value and $\tau_i$ are delay times appearing in the set of delay differential equations (DDE). As a consequence, the inputs $\hat{\mathbf{u}}$ may not be available during the whole time span of the experiment $t \in (0, T)$, and hence the time intervals, where $\hat{\mathbf{u}}$ is undefined, have to be excluded from the integrals in (8), (34), or (35). The inverse model approach for parameter identification of DDE systems has the big advantage that the quite challenging numerical solution of DDEs is not required. Especially, there is no need to choose reasonable initial functions of the system states for times before $t = 0$, as would be necessary when using the conventional method of parameter identification. An example will be given in Section 3.2.

# 3 Case studies

Two case studies will illustrate the properties of the proposed identification method. The first one will demonstrate advantages of the proposed inverse model method compared to the standard approach for nonlinear systems. Further, the example will be used to discuss the treatment of measurement noise. The second example will show the benefits of the proposed method for systems with delay.

## 3.1 FitzHugh-Nagumo Equations

The well-known FitzHugh-Nagumo equations [36] give a simple description of the electro-physiology of a nerve axon. The system was also used by Ramsay et al. [21] as a test example for their generalized smoothing approach for parameter identification. The model equations read

$$\begin{aligned} \dot{V} &= c\left(V - \frac{V^3}{3} + R\right) \\ \dot{R} &= -\frac{1}{c}(V - a + bR), \end{aligned} \tag{36}$$

where $V$ and $R$ represent the voltage and the recovery of the membrane. Throughout this case study, the parameters $a$ and $b$ have to be identified from measurements, whereas $c = 3$ is assumed to be known from literature. For the purpose of parameter identification, in-silico measurement data of the membrane voltage are provided as

$$y(t_k) = V(t_k) \tag{37}$$

The FitzHugh-Nagumo equations are an autonomous ODE system without a physical input. As one quantity is measured, one fictitious input has to be determined in order to obtain a flat system. It is easily seen that the flat input should

act on the equation for $R(t)$. So the system used for parameter identification is the following:

$$\begin{aligned}\dot{V}(t) &= c\left(V(t) - \frac{V^3(t)}{3} + R(t)\right)\\ \dot{R}(t) &= -\frac{1}{c}(V(t) - a + bR(t)) + u(t)\\ y(t) &= V(t)\end{aligned} \tag{38}$$

Again, $u(t)$ is a fictitious input whose nominal value is always zero.

From (38) one can derive expressions for $V(t)$, $R(t)$, $u(t)$:

$$V(t) = y(t) \tag{39}$$

$$R(t) = \frac{1}{c}\dot{y}(t) - \frac{3y(t) + y(t)^3}{3} \tag{40}$$

$$u(t) = \frac{1}{c}\ddot{y}(t) - \dot{y}(t)\left(1 - y(t)^2\right) - \frac{1}{c}\left(y(t) - a + bR(t)\right). \tag{41}$$

Because the fictitious input $u(t)$ vanishes at all times, the cost function $J_u$ for flat input based parameter identification reads in this case:

$$\begin{aligned}J_u(\hat{a}, \hat{b}) &= \int_0^T \left(\hat{u}(t) - u(t)\right)^2 dt = \int_0^T \hat{u}^2(t)\, dt\\ &= \int_0^T \left(\frac{1}{c}\ddot{y}(t) - \dot{y}(t)\left(1 - y(t)^2\right) - \frac{1}{c}\left(y(t) - \hat{a} + \hat{b}\, R(t)\right)\right)^2 dt\end{aligned} \tag{42}$$

In a first step, the conventional cost function $J_y$ from Eq. (7) and the cost function $J_u$ of the new approach given by Eq. (42) are evaluated for different parameter estimates $\hat{\theta} = (\hat{a}, \hat{b})^T$ in the range of $-1 \leq \hat{a}, \hat{b} \leq 1.5$. The nominal parameter values and initial conditions are taken as $a = b = 0.2, V(0) = -1, R(0) = 1$. The parameter ranges of $\hat{a}$ and $\hat{b}$ are discretized in 251 intervals, which leads to an

overall number of 63001 evaluations of Eq. (42) and Eq. (7), respectively. On a PC, this requires 5707 s of CPU time for Eq. (7), but only 15 s for Eq. (8), which illustrates nicely the computational efficiency of the flat input method. Another advantage of the new identification method for this example becomes visible when looking at the contour plots of $J_u$ and $J_y$ (cf. Figure 6). The shape of $J_y$ is quite complicated with several local minima that make parameter identification difficult [21]. In contrast, $J_u$ has a nice convex shape well suited for numerical optimization. It is proven in the appendix that $J_u$ is globally convex with ellipse-shaped contour lines of $J_u$, which is an improvement compared to the result of Ramsay et al. [21] where only a local smoothing of the objective function could be achieved.

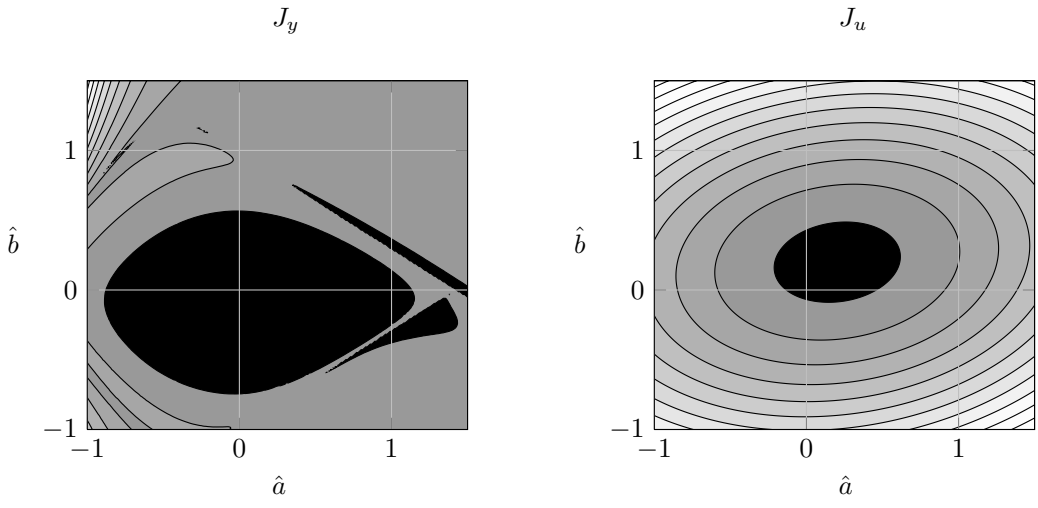



Figure 6: Contour plots of cost functions of the FitzHugh-Nagumo equations for reference parameter values $a = b = 0.2$. Left diagram: cost function $J_y$ with several local minima resulting from the standard approach; right diagram: convex cost function $J_u$ resulting from the inverse model approach.

In the following, the influence of measurement noise is analyzed. Artificial measurement data are taken from a simulation of the autonomous system ($u(t) \equiv 0$) as shown in Figure 7. The simulated measurement data for the membrane volt-

age $V(t)$ are discretized with a sampling rate of $\Delta t = 0.05$ and corrupted by additive normal distributed noise $\epsilon \sim \mathcal{N}(0, 0.04)$, which corresponds to a standard deviation of at least 10 % of the generated data. Initial guesses for the

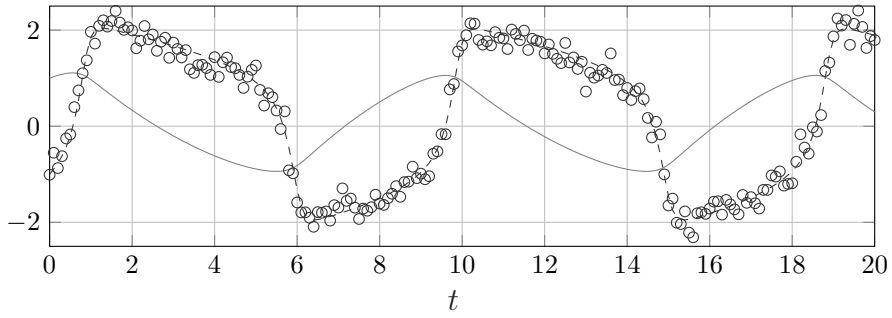



Figure 7: Simulation results of the autonomous FitzHugh-Nagumo equations; solid lines represent the membrane recovery $R$, dashed lines the voltage $V$, and circles illustrate discrete noisy measurement data of $V$.

unknown parameters are chosen randomly in the range of $0 \leq \hat{a}, \hat{b} \leq 10$. In this way, 1000 optimization problems are generated. They are solved first by the conventional approach, i.e. by minimizing $J_y$ in Eq. (7). The Matlab built in Levenberg-Marquardt algorithm lsqnonlin in its standard configuration is used for optimization. The left-hand side diagram in Figure 8 shows that the optimization results depend strongly on the initial guess of the unknown parameters. For poor initial guesses the optimizer gets stuck in a local minimum of $J_y$.

For the inverse model approach, the measurement data are approximated by a surrogate output function $\hat{y}(t)$ based on B-splines. The objective function $J_u(\boldsymbol{c}, \lambda, \hat{a}, \hat{b})$ as defined in Eq. (34) is used for optimization. As expected due to the convexity of the objective function, the parameter estimates are now much less sensitive to the initial guesses (see right-hand side diagram in Figure 8). All estimates are grouped nicely around the nominal values of $a$ and $b$.

Table 1 gives a more detailed view on the estimation results as well as on the

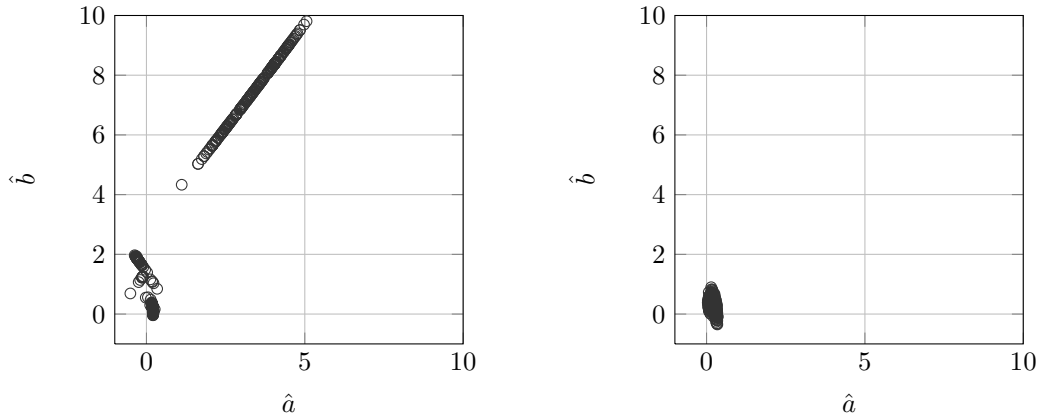



Figure 8: Estimated parameters $\hat{a}$ and $\hat{b}$ from 1000 parameter identifications; initial parameter guesses are varied randomly in the range of $0 \leq \hat{a}, \hat{b} \leq 10$ and measurement data are corrupted by normal-distributed noise; nominal parameter values are $a = b = 0.2$ left-hand side figure: result of conventional approach based on objective function $J_y$; right-hand side figure: result of alternative approach based on objective function $J_u$.

computational effort. It turns out that the flat input method is not only more accurate but also considerably faster. The new approach achieves a speed-up by a factor of 30. This is mainly because the evaluation of $J_u$ is much cheaper. The computational time for the optimization itself is also reduced, but not that strongly. Here, the advantages of the nicer shape of $J_u$ are partly compensated by the larger number of optimization variables ($\boldsymbol{c}$ and $\lambda$ in addition to $\hat{a}$ and $\hat{b}$).

| | $\mu_{\hat{a}}$ | $\sigma_{\hat{a}}$ | $\mu_{\hat{b}}$ | $\sigma_{\hat{b}}$ | $t^{cpu}$(PI)/s | $t^{cpu}(J_{y/u})$/s | calls($J_{y/u}$) |
|---|---|---|---|---|---|---|---|
| PI($J_y$) | 0.825 | 1.355 | 1.765 | 2.952 | 24394.1 | 24004.8 | 70503 |
| PI($J_u$) | 0.190 | 0.058 | 0.327 | 0.168 | 661.4 | 551.4 | 47000 |

Table 1: Parameter identification (PI) of the FitzHugh-Nagumo equations by using the conventional cost function $J_y$ or the alternative cost function $J_u$; Contents of the columns: $\mu_{\hat{a}/\hat{b}}$ and standard deviations $\sigma_{\hat{a}/\hat{b}}$ of the estimated parameters for nominal values $a = b = 0.2$, consumed CPU time for total parameter identification as well as for evaluation of objective functions, number of objective function calls.

## 3.2 Virus Replication Model

The second case study continues the investigation of the virus replication model started in Example IV. Now the model is extended in order to take into account that the infected MDCK cells $I_c$ do not release active and inactive virus particles immediately, but after some delay $\tau$, which is an additional unknown model parameter to be identified. As in Example IV, it is assumed that active and inactive virus particle concentrations are measurable. Further, the same fictitious inputs $u_1$ and $u_2$ as before are added to the model. The modified equations with fictitious inputs read:

$$\begin{aligned}
\dot{U}_c(t) &= \theta_6 \frac{C_{max} - (U_c(t) + I_c(t))}{C_{max}} U_c(t) - \theta_1 U_c(t) V_i(t) + u_1 \\
\dot{I}_c(t) &= \theta_1 U_c(t) V_i(t) - \theta_2 I_c(t) + u_2 \\
\dot{V}_i(t) &= \theta_3 I_c(t-\tau) - \theta_4 V_i(t) - \theta_1 U_c(t) V_i(t) \\
\dot{V}_d(t) &= \theta_5 I_c(t-\tau) + \theta_4 V_i(t) \\
y_1 &= V_i \\
y_2 &= V_d
\end{aligned} \tag{43}$$

System (43) is a $\delta$-flat system, because all states and inputs may be expressed by $y_1$, $y_2$ and their time derivatives in the following way:

$$
\begin{aligned}
V_i(t) &= y_1(t) \\
V_d(t) &= y_2(t) \\
I_c(t) &= \frac{1}{\theta_5}\left(\dot{y}_2(t+\tau) - \theta_4 y_1(t+\tau)\right) \\
U_c(t) &= \frac{\theta_3 I_c(t-\tau) - \theta_4 y_1(t) - \dot{y}_1(t)}{\theta_1 y_1(t)} \\
u_2(t) &= \frac{1}{\theta_5}\left(\ddot{y}_2(t+\tau) - \theta_4 \dot{y}_1(t+\tau)\right) - \theta_1 U_c(t) y_1(t) + \theta_2 I_c(t) \\
u_1(t) &= \theta_6 \frac{C_{max} - \left(U_c(t+\tau) + I_c(t+\tau)\right)}{C_{max}} U_c(t+\tau) \\
&\quad + \theta_1 U_c(t+\tau) y_1(t+\tau) - \frac{\theta_4 \dot{y}_1(t+\tau) - \ddot{y}_1(t+\tau)}{\theta_1 y_1(t+\tau)} \\
&\quad - \frac{\theta_3\left(\theta_1 U_c(t) y_1(t) - \theta_2 I_c(t) + u_2(t)\right)}{\theta_1 y_1(t+\tau)} - \frac{U_c(t+\tau)\dot{y}_1(t+\tau)}{y_1(t+\tau)}
\end{aligned}
\tag{44}
$$

In a first step, the unknown parameters $\theta_1, \ldots, \theta_6$ and $\tau$ are identified by the conventional approach, i.e. the delay differential equations are solved numerically and the resulting difference between measurements $\mathbf{y}$ and estimates $\hat{\mathbf{y}}$ is minimized. It turns out that this strategy leads to serious numerical problems. It is only successful when it is done in a two-step procedure: In an outer loop, an estimate of the delay $\hat{\tau}$ is varied over a certain interval; in an inner loop, the parameter vector $\hat{\theta}$ is determined by minimizing $J_y$ from Eq. (7) for a fixed value of $\hat{\tau}$; the value of $\hat{\tau}$ that leads to the smallest value of $J_y$ is finally assumed to be the true delay $\tau$. But even the two-step procedure works only, if the estimated $\hat{\tau}$ values are in a small range around the true time delay. This can be seen from the result of a simulation experiment displayed in Figure 9. The model parameters $\theta$ are estimated for 100 different delay parameter values equally spaced in the range of 5 to 15 hours with the true delay being $\tau = 7.5$ h. The numerical optimization, which again uses lsqnonlin, is only successful if the guess of $\tau$ is close to the reference value. For other $\tau$-values the optimization routine diverges.

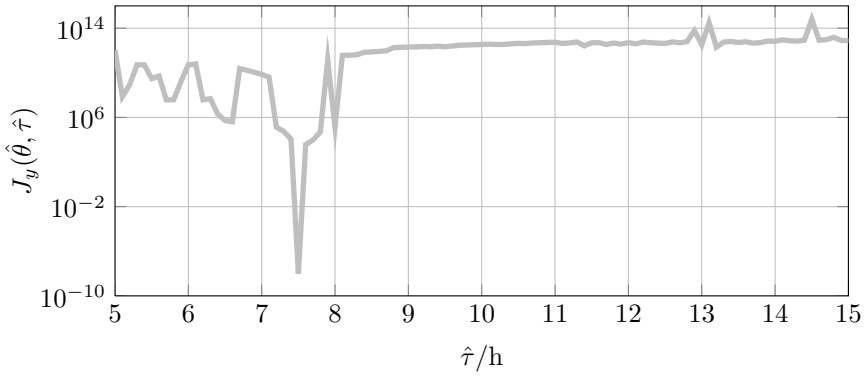



Figure 9: Minimization of the standard cost function $J_y$ for fixed estimated values $\hat{\tau}$ of the time-delay parameter, the true value being $\tau = 7.5$ h.

In a second step, the alternative cost function $J_u$ given by Eq. (34) is minimized instead of $J_y$. As the inputs are purely fictitious with $u_1(t) \equiv 0, u_2(t) \equiv 0$, the cost function reads for this example

$$J_u(\hat{\theta}_1, \ldots, \hat{\theta}_6, \hat{\tau}) = \int_{\hat{\tau}}^{T-2\hat{\tau}} \left(\hat{u}_1(t) + \hat{u}_2(t)\right)^2 \, dt, \tag{45}$$

where $\hat{u}_1(t)$ and $\hat{u}_2(t)$ are computed from (44) by substituting the true parameter values $\theta_1, \ldots, \theta_6, \tau$ with the estimates $\hat{\theta}_1, \ldots, \hat{\theta}_6, \hat{\tau}$. The limits of the integral in Eq. (45) result from the condition that the arguments of all time dependent functions the integrand depends on should lie between 0 and $T$. The approach proves to be much more benign numerically. The optimized cost function $J_u$ becomes a smooth function of $\hat{\tau}$ with a clear minimum at the reference delay $\hat{\tau}$ (see Figure 10). This even works, if the initial guesses of the $\theta$ parameters are very poor - an example is shown in Table 2, where initially all $\theta$ parameters are set to values equal to 750 times the true value and nevertheless very accurate estimates are obtained in the end. It should also be mentioned that the evaluation of $J_u$

is very efficient and that the computation of the complete curve in Figure 10 requires less than 10 s on a PC, which is about 1% of the time needed for the conventional approach.

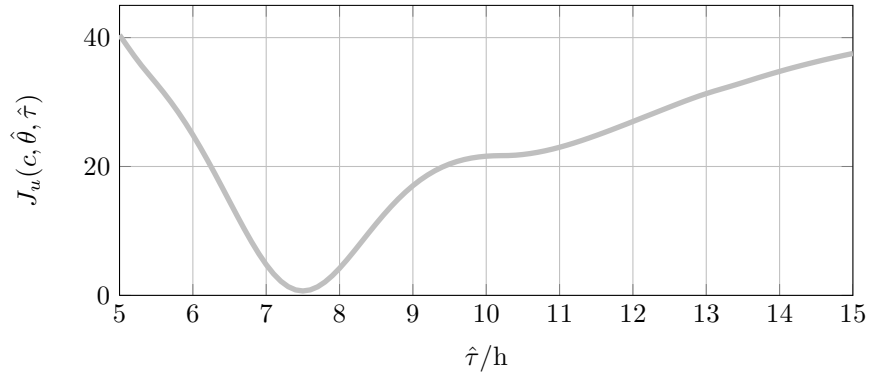



Figure 10: Minimization of the alternative cost function $J_u$ for fixed estimated values $\hat{\tau}$ of the time-delay parameter, the true value being $\tau = 7.5$ h.

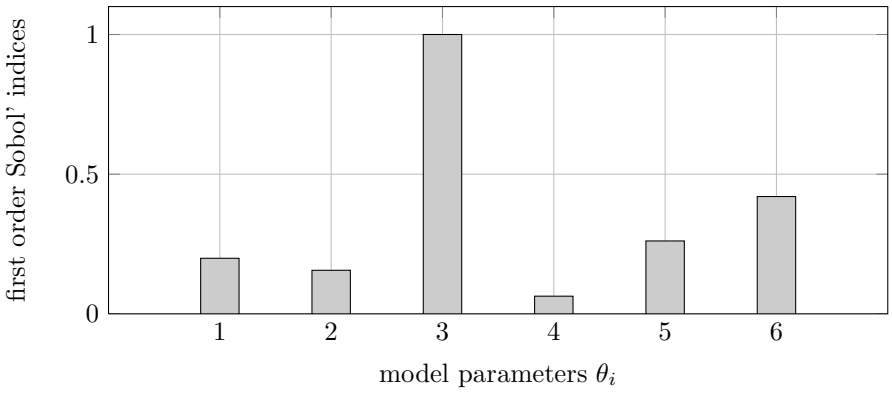
| | $\theta_1$ | $\theta_2$ | $\theta_3$ | $\theta_4$ | $\theta_5$ | $\theta_6$ |
|---|---|---|---|---|---|---|
| $\theta_{Ini}/\theta_{True}$ | 750 | 750 | 750 | 750 | 750 | 750 |
| $\theta_{Opt}/\theta_{True}$ | 1.0032 | 1.0014 | 0.9974 | 1.000 | 0.9974 | 0.9976 |

Table 2: Result of the parameter identification of model (43) when minimizing the cost function $J_u$ given in Eq. (34); first row: scaled values of the initial parameter guesses; second row: scaled results of optimization.

In Section 2 it was already mentioned that the fictitious inputs may be more sensitive against certain model parameters than the measured outputs. In order to demonstrate this for the virus replication example, first order Sobol' sensitivity indices [37] are computed for both approaches, using a numerical method presented in a previous publication [38]. The results differ strongly for the conventional approach and for the inverse model approach (see Figure 11). Parameter $\theta_4$, which is least sensitive for the conventional method, becomes the most sensitive one when the inverse model approach is applied. This is in agreement with

Table 2, where $\theta_4$ is estimated with the highest accuracy. As a consequence of the different sensitivities of $\mathbf{y}$ and $\mathbf{u}$, the information content of the measurement might be exploited in the best possible way, if the conventional identification approach based on $J_y$ and the alternative identification approach $J_u$ are combined and each approach identifies those parameters it is most sensitive against.

(a) standard approach ($J_y$):




(b) alternative approach ($J_u$):

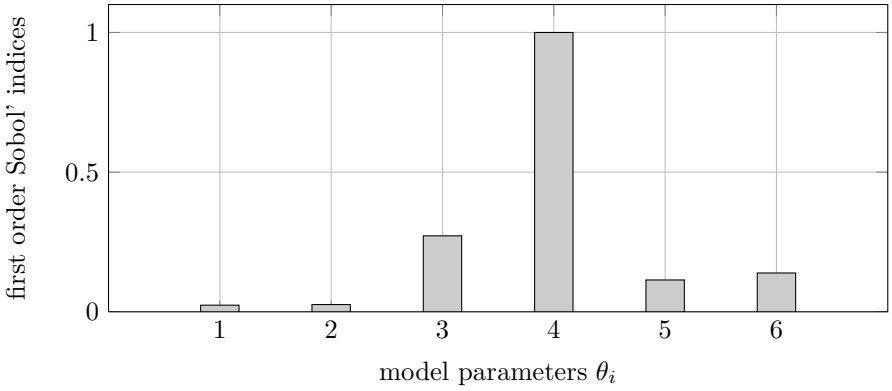



Figure 11: Integral measures of Sobol' indices normalized to the most sensitive for the two different strategies of parameter identification.

# 4 Conclusions

Exploiting the flatness property of dynamical systems for parameter identification may be beneficial in cases, where the physical measured outputs are at the same time flat outputs of the system. The most obvious advantage is that for flat systems the states and inputs are either given as analytical functions of the flat outputs and their time derivatives, or can be computed from the flat outputs by solving algebraic equations. Therefore, there is no need to solve differential equations numerically underlying the optimization problem of parameter identification. This accelerates the identification process and it circumvents the problem of determining possibly unknown initial conditions of the states. This is especially appealing for delay differential equations, whose numerical solution and feasible choice of initial functions is not trivial. Further, the flat input approach leads to a very natural treatment of hidden (unmeasurable) states without the need to introduce nuisance parameters. In the case of a completely measurable state vector, the flat input method coincides with two-stage approaches as suggested in [16, 17, 22].

Another advantage, which became visible in the example of the FitzHugh-Nagumo equations, is that the cost function $J_u$ describing the deviation of flat inputs from their reference values may have a nicer shape with less local minima. This is hardly a result that can be generalized to all classes of systems, and it may be possible to construct counter-examples where the opposite holds. But still, it seems quite typical for many systems that the flat inputs depend on the model parameters in a "less severe" nonlinear way than the outputs. The coefficients of linear systems, for example, appear as arguments of exponential functions in the expressions of the system outputs, but as polynomial or rational functions in the

expressions of flat inputs. So if one has to tackle a parameter identification problem with many local minima in the conventional approach it may be worthwhile looking at the flat inputs instead.

The obstacle one has to overcome before making use of flatness is clearly that suitable fictitious inputs must be found that turn the physical outputs into flat outputs. The good news is that there is a method in literature [13, 30] that makes this construction surprisingly simple for systems that are locally observable by the measured outputs, and that the construction of flat inputs for given outputs is much easier than the opposite problem of finding flat outputs one has to solve for flatness based control. It should also be noted that the construction of flat inputs and their application to parameter identification is not restricted to certain nonlinearities or certain types of kinetic expressions, which is another nice property of the method.

The weak spot of the flat input method is the need to form derivatives of measured outputs. In this work, this problem could be solved satisfactorily by introducing surrogate output functions based on B-splines or other approximation techniques. The additional effort due to a larger number of optimization variables could be over-compensated by the very efficient solution of the ODE/DDE model equations. But clearly, the accuracy of the method will suffer if higher order derivatives of the outputs are required and the measurements are noisy or the measurement sampling rates are low. In consequence, use of flat inputs for parameter identification is most attractive for systems whose observability indices are not too large, i.e. for systems where the number of states is not orders of magnitude larger than the number of outputs.

# Acknowledgements

The financial support of the German Ministry for Education and Research (BMBF) under contract number 0315505B is gratefully acknowledged.

# A Analysis of $J_u$ in Section 3.1

From (41) one obtains

$$\hat{u} - u = -\frac{1}{c}\left(a - \hat{a} + R\,(\hat{b} - b)\right)$$

and hence

$$\begin{aligned} J_u &= \frac{1}{c^2}\int_0^T c^2(\hat{u} - u)^2\,dt \\ &= \frac{1}{c^2}\left((\hat{a} - a)^2\,T - 2\,(\hat{a} - a)(\hat{b} - b)\int_0^T R\,dt + (\hat{b} - b)^2\int_0^T R^2\,dt\right). \end{aligned}$$

When introducing the abbreviations

$$\begin{aligned} \overline{R} &= \frac{1}{T}\int_0^T R\,dt \\ \overline{R^2} &= \frac{1}{T}\int_0^T R^2\,dt \\ \tilde{a} &= \hat{a} - a \\ \tilde{b} &= \sqrt{\overline{R^2}}(\hat{b} - b), \end{aligned}$$

$J_u$ can be written as

$$J_u = \frac{1}{c^2\,T}\left(\tilde{a}^2 - 2\frac{\overline{R}}{\sqrt{\overline{R^2}}}\,\tilde{a}\,\tilde{b} + \tilde{b}^2\right).$$

Because $\overline{R}/\sqrt{\overline{R^2}}$ is always between -1 and 1, the above equation defines an ellipse for constant values of $J_u$.

# References


[1] L. Biegler, J. Damiano, and G. Blau. Nonlinear parameter estimation: a case study. *AIChE Journal*, 32:29–45, 1986.

[2] M. Fliess, J. Levine, P. Martin, and P. Rouchon. On differentially flat systems. *C. R. Acad. Sci. Paris Sér. I Math.*, 315:619–624, 1992.

[3] H. Sira-Ramirez and S. Agrawal. *Differentially Flat Systems*, volume 17 of *Control Engineering*. Marcel Dekker, Inc., New York, Basel, 2004.

[4] J. Lévine. *Analysis and Control of Nonlinear Systems: A Flatness Based Approach*. Springer-Verlag, Berlin, Heidelberg, 2009.

[5] R Rothfuss, J Rudolph, and M Zeitz. Flatness based control of a nonlinear chemical reactor model. *Automatica*, 32(10):1433–1439, 1996.

[6] P Tsiotras and JH Luo. Control of underactuated spacecraft with bounded inputs. *Automatica*, 36(8):1153–1169, 2000.

[7] H Sira-Ramirez. Dynamic second-order sliding mode control of the hovercraft vessel. *IEEE Transactions On Control Systems Technology*, 10(6):860–865, 2002.

[8] A De Luca and G Oriolo. Trajectory planning and control for planar robots with passive last joint. *International Journal of Robotics Research*, 21(5-6):575–590, 2002.

[9] V Hagenmeyer and E Delaleau. Exact feedforward linearization based on differential flatness. *International Journal of Control*, 76(6):537–556, 2003.

[10] Alireza Payman, Serge Pierfederici, and Farid Meibody-Tabar. Energy control of supercapacitor/fuel cell hybrid power source. *Energy Conversion And Management*, 49(6):1637–1644, 2008.

[11] A. Shahin, B. Huang, J. P. Martin, S. Pierfederici, and B. Davat. New nonlinear control strategy for non-isolated DC/DC converter with high voltage ratio. *Energy Conversion and Management*, 51(1):56–63, 2010.

[12] M. Fliess, J. Lévine, P. Martin, and P. Rouchon. Flatness and defect of nonlinear systems: introductory theory and examples. *International Journal of Control*, 61:1327–1361, 1995.

[13] S. Waldherr and M. Zeitz. Conditions for the existence of a flat input. *International Journal of Control*, 81:439–443, 2008.

[14] V. Vassilev, M. Groeschel, H. Schmid, W. Peukert, and G. Leugering. Interfacial energy estimation in a precipitation reaction using the flatness based control of the moment trajectories. *Chemical Engineering Science*, 65(6):2183–2189, 2010.

[15] M. Fliess, C. Join, and H. Sira-Ramirez. Non-linear estimation is easy. *International Journal of Modelling, Identification and Control*, 4:12–27, 2008.

[16] J. Swartz and H. Bremermann. Discussion of parameter estimation in biological modelling: algorithms for estimation and evaluation of the estimates. *Journal of Mathematical Biology*, 1:241–257, 1975.

[17] J.M. Varah. A spline least squares method for numerical parameter estimation in differential equations. *SIAM J. Sci. Stat. Comput.*, 3:28–46, 1982.

[18] N. Heckman and J. Ramsay. Penalized regression with model-based penalties. *The Canadian Journal of Statistics*, 28:241–258, 2000.

[19] J. O. Ramsay and B. W. Silverman. *Functional Data Analysis*. Springer New York, 2005.

[20] A. A. Poyton, M. S. Varziri, K. B. McAuley, P. J. McLellan, and J. O. Ramsay. Parameter estimation in continuous-time dynamic models using principle differential analysis. *Computers and Chemical Engineering*, 30:698–708, 2006.

[21] J.O. Ramsay, G. Hooker, D. Campbell, and J. Cao. Parameter estimation for differential equations: a generalized smoothing approach. *J. R. Statist. Soc. B*, 69:741–796, 2007.

[22] J. Chen and H. Wu. Efficient local estimation for time-varying coefficients in deterministic dynamic models with applications to HIV-1 dynamics. *Journal of the American Statistical Association*, 103:369–384, 2008.

[23] M. S. Varziri, A. A. Pyoton, K. B. McAuley, P. J. McLellan, and J. O. Ramsay. Selecting optimal weighting factors in ipda for parameter estimation in continuous-time dynamic models. *Computers and Chemical Engineering*, 32:3011–3022, 2008.

[24] J. Cao, L. Wang, and J. Xu. Robust estimation for ordinary differential equation models. *Biometrics*, 67:1305–1313, 2011.

[25] F. Boulier. *Gröbner Bases in Symbolic Analysis*, volume 2 of *Radon Series Comp. Appl. Math.*, chapter Differential Elimination and Biological Modeling, pages 109–138. De Gruyter, Berlin, 2007, 2007.

[26] M. Nakatsui and K. Horimoto. Improvement of estimation accuracy in parameter optimization by symbolic computation. In *IEEE International Symposium on Computer-Aided Control System Design (2010)*, pages 1720–1724, Yokohama, Japan, September 8-10 2010.

[27] S. Audoly, G. Bellu, L. D'Angiò, and M.P. Saccomani. Global identifiability of nonlinear models of biological systems. *IEEE Transactions on Biomedical Engineering*, 48:55–65, 2001.

[28] M. Saccomani, S. Audoly, and L. D'Angió. Parameter identifiability of nonlinear systems: The role of initial conditions. *Automatica*, 39:619–632, 2003.

[29] G. Bellu, M.P. Saccomani, S. Audoly, and L. D'Angiò. DAISY: A new software tool to test global identifiability of biological and physiological systems. *Computer Methods and Programs in Biomedicine*, 88:52–61, 2007.

[30] S. Waldherr and M. Zeitz. Flat inputs in the MIMO case. In *8th IFAC Symposium on Nonlinear Control Systems (2010)*, pages 695–700, University of Bologna, Italy, September 1-3 2010.

[31] A. Krener and W. Respondek. Nonlinear observers with linearizable error dynamics. *SIAM J. Contr. Optim.*, 23:197–216, 1985.

[32] L. Möhler, D. Flockerzi, H. Sann, and U. Reichl. Mathematical model of influenza A virus production in large-scale microcarrier culture. *Biotechnology and Bioengineering*, 90:46–58, 2005.

[33] M. Mboup, C. Join, and M. Fliess. Numerical differentiation with annihilators in noisy environment. *Numerical Algorithms*, 50:439–467, 2009.

[34] T. Duncan, P. Mandl, and B. Pasik-Duncan. Numerical differentiation and parameter estimation in higher-order linear stochastic systems. *IEEE Transactions on Automatic Control*, 41:522–532, 1996.

[35] H Mounier and J Rudolph. Flatness-based control of nonlinear delay systems: a chemical reactor example. *International Journal Of Control*, 71(5):871–890, 1998.

[36] R. Fitzhugh. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical Journal*, 1(6):445–466, 1961.

[37] I.M. Sobol'. Sensitivity analyis for nonlinear mathematical models. *Mathematical Modeling and Computational Experiment*, 1:407–414, 1993.

[38] R. Schenkendorf and M. Mangold. Qualitative and quantitative optimal experimental design for parameter identification of a MAP kinase model. In S. Bittanti, A. Cenedese, and S. Zampieri, editors, *Preprints of the 18th IFAC World Congress*, pages 11666–11671, Milano, Italy, 2011.