Parameter estimation in linear differential-algebraic equations

Markus Gerdin, Torkel Glad and Lennart Ljung

Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: <http://www.control.isy.liu.se>
E-mail: gerdin,torkel,ljung@isy.liu.se

April 16, 2003

Report no.: LiTH-ISY-R-2515

Submitted to the 13th IFAC Symposium on System Identification

Technical reports from the Control & Communication group in Linköping are available at <http://www.control.isy.liu.se/publications>.

Parameter estimation in linear differential-algebraic equations

Markus Gerdin, Torkel Glad and Lennart Ljung

April 16, 2003

Abstract

This report describes how parameter estimation can be performed in linear DAE systems. Both time domain and frequency domain identification are examined. The results are exemplified on a small system. A potential application for the algorithms is to make parameter estimation in models generated by a modeling language like Modelica.

1 Introduction

In recent years so-called object-oriented modeling languages have become increasingly popular. Examples of such languages are Omola, Dymola and Modelica, [8, 14]. Modeling languages of this type make it possible to build models by connecting submodels in a manner that parallels the physical construction. A consequence of this viewpoint is that it is usually not possible to specify in advance what variables are inputs or outputs from a given submodel. A further consequence of this is that the resulting model is not in state space form. Instead the model is a collection of equations, some of which contain derivatives and some of which are static relations. A model of this form is sometimes referred to as a DAE (*differential algebraic equations*) model or a *descriptor model*. It can be noted that these models are a special case of the so-called behavioral models discussed in, e.g., [10]. When constructing models of this form it is common that some parameters have unknown values and have to be determined from identification experiments. It is thus of interest to examine how identification can be performed for models of this form. In the present paper some issues of principle will be addressed and a concrete example will illustrate the conclusions.

When a model has been defined in an object oriented language, one can mark the parameters that are to be identified, mark the inputs and outputs that are used in an identification experiment and give the corresponding data files. Ideally this should be enough, so that then the transformation into a form suitable for identification should be automatic.

Here only linear models will be considered. The object oriented languages discussed above are equation oriented. This means that all models consist of systems of equations (or descriptions that can be converted to systems of equations). The equations relate the physical variables and possibly first order

derivatives of them. This means that a model can be described in the form

$$E\dot{\xi}(t) = J\xi(t) + Ku(t) \quad (1a)$$

$$y(t) = L\xi(t) \quad (1b)$$

where ξ is a vector of physical variables, u is the input and y is the measured output. The matrices E , J , K and L contain physical parameters that describe the system. Some of these parameters are those that are to be estimated from an identification experiment. The model above is not in state space form but forms a DAE.

If a system is modeled by just writing down the equations describing its different parts, the matrix E is typically non-invertible so that the extraction of a state space description is not completely straight-forward. The goal of the present paper is to discuss how this model can be transformed to a form where well-known identification techniques can be directly applied. In particular we will discuss what the implications are for the noise models.

Previously, not very much work has been done on parameter identification in DAE systems. One of the works on the nonlinear case is [12].

2 DAE systems

The model structure that will be examined is thus (1). It will be assumed that $\det(sE - J)$ is not zero for all s . If the determinant were zero for all s , it can be realized by Laplace transforming the equations that $\xi(t)$ will not be uniquely determined by $u(t)$ so this is a reasonable assumption. Given this assumption, it is possible to transform (1) to a form which resembles a normal state space description. Transformations have been suggested in the literature, see for example [6, 4] and references in [15]. Here the one derived below will be used:

Lemma 1 *If $\det(sE - J) \neq 0$, the solution of (1) can be described by the normal form*

$$\dot{w}_1(t) = Aw_1(t) + B_1u(t) \quad (2a)$$

$$w_2(t) = \sum_{i=0}^{m-1} (-N)^i D_1 u^{(i)}(t) \quad (2b)$$

$$\xi(t) = T^{-1} \begin{bmatrix} w_1(t) \\ w_2(t) \end{bmatrix} \quad (2c)$$

$$y(t) = LT^{-1} \begin{bmatrix} w_1(t) \\ w_2(t) \end{bmatrix}. \quad (2d)$$

Proof. It follows e.g. from [4] that there exists invertible matrices P and T such that the transformation $PET\xi = PJT\xi + PKu$ gives the form (dependence on t is here omitted in the notation)

$$\begin{bmatrix} I & 0 \\ 0 & N \end{bmatrix} \begin{bmatrix} \dot{w}_1 \\ \dot{w}_2 \end{bmatrix} + \begin{bmatrix} -A & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} B_1 \\ D_1 \end{bmatrix} u \quad (3a)$$

$$\xi = T^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad (3b)$$

where N is nilpotent, that is $N^m = 0$ for some m . Now, if $N = 0$ then $w_2(t) = D_1 u(t)$ and the proof is done. If $N \neq 0$ the second row of (3a) is multiplied with N to get

$$N^2 \dot{w}_2(t) + N w_2(t) = N D_1 u(t). \quad (4)$$

Then (4) is differentiated and the second row of (3a) is inserted. This gives

$$w_2(t) = D_1 u(t) - N D_1 \dot{u}(t) + N^2 \ddot{w}_2(t). \quad (5)$$

If $N^2 = 0$ the proof is done, otherwise the process is continued until $N^m = 0$ (this is true for some m as N is nilpotent). The resulting expression is then

$$w_2(t) = \sum_{i=0}^{m-1} (-N)^i D_1 u^{(i)}(t) \quad (6)$$

and the proof is complete. $\square$

This result is consistent with the result in [11] that all linear systems are equivalent to a state space description, possibly with derivatives of the input added to the output. Note that the transformation in Lemma 1 may not be suitable for systems where the matrices E , J , K and L change abruptly, see further [15].

3 Discretization

When doing system identification, the measured data that is to be used to identify the parameters is usually available as sampled data in the time domain, so in this section it will be examined what the exact discrete time description for a linear DAE system is under certain assumptions on the input.

It is a standard result, e.g., [3], that the solution of a continuous-time state space system can be described by a discrete-time state space system if the input is piecewise constant, or in general is completely determined by the values in the sample points. In the DAE case the input cannot be assumed to be piecewise constant as the system according to Lemma 1 may depend on derivatives of the input. Instead it will be assumed that the $m-1$:th derivative of the input is piecewise constant and that the system is on the form (2) which is possible according to Lemma 1. This will make it possible to use the result for normal state space systems, and the first step of the derivation is to define a vector with the input and some of its derivatives:

$$w_3(t) = \begin{bmatrix} u(t)^T & \dot{u}(t)^T & \dots & u^{(m-2)}(t)^T \end{bmatrix}^T \quad (7)$$

This leads to the relation (8). (In the rest of this section dependence on the time t will be omitted if the dependence is clear.)

$$\dot{w}_3 = \underbrace{\begin{bmatrix} 0 & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & I \\ 0 & 0 & \dots & 0 \end{bmatrix}}_{\Psi_1} w_3 + \underbrace{\begin{bmatrix} 0 \\ \vdots \\ 0 \\ I \end{bmatrix}}_{\Psi_2} u^{(m-1)} \quad (8)$$

Rewriting (2) using w_3 gives

$$\dot{w}_1 = Aw_1 + \underbrace{\begin{bmatrix} B_1 & 0 & \dots & 0 \end{bmatrix}}_{\Psi_3} w_3 \quad (9a)$$

$$\begin{aligned} w_2 = & \underbrace{\begin{bmatrix} I & (-N) & \dots & (-N)^{m-2} \end{bmatrix} D_1}_{\Psi_4} w_3 \\ & + \underbrace{(-N)^{m-1} D_1}_{\Psi_5} u^{(m-1)} \end{aligned} \quad (9b)$$

$$\dot{w}_3 = \Psi_1 w_3 + \Psi_2 u^{(m-1)} \quad (9c)$$

$$y = LT^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \quad (9d)$$

If w_2 is eliminated, these equations can be seen as a normal state space equation with $u^{(m-1)}$ as the input:

$$\begin{bmatrix} \dot{w}_1 \\ \dot{w}_3 \end{bmatrix} = \underbrace{\begin{bmatrix} A & \Psi_3 \\ 0 & \Psi_1 \end{bmatrix}}_{\tilde{A}} \begin{bmatrix} w_1 \\ w_3 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \Psi_2 \end{bmatrix}}_{\tilde{B}} u^{(m-1)} \quad (10a)$$

$$y = \underbrace{LT^{-1} \begin{bmatrix} I & 0 \\ 0 & \Psi_4 \end{bmatrix}}_{\tilde{C}} \begin{bmatrix} w_1 \\ w_3 \end{bmatrix} + \underbrace{LT^{-1} \begin{bmatrix} 0 \\ \Psi_5 \end{bmatrix}}_{\tilde{D}} u^{(m-1)} \quad (10b)$$

Defining $x = \begin{bmatrix} w_1^T & w_3^T \end{bmatrix}^T$ this can be written in the compact form

$$\dot{x} = \tilde{A}x + \tilde{B}u^{(m-1)} \quad (11a)$$

$$y = \tilde{C}x + \tilde{D}u^{(m-1)}. \quad (11b)$$

Now, if it is assumed that $u^{(m-1)}$ is piecewise constant, the result for discretization of state space systems can be applied to (11) to give an exact discrete time description of the original time DAE system (1). The above discussion proves the following result:

Result 1 *Consider the DAE system (1) with the normal form (2). If $u^{(m-1)}(t)$ is constant for $T_s k \leq t < T_s k + T_s$ for constant T_s and $k = 0, 1, 2, \dots$, then $y(T_s k)$ is exactly described by the discrete time state space system*

$$x(T_s k + T_s) = \Phi x(T_s k) + \Gamma u^{(m-1)}(T_s k) \quad (12a)$$

$$y(T_s k) = \tilde{C}x(T_s k) + \tilde{D}u^{(m-1)}(T_s k). \quad (12b)$$

where

$$\Phi = e^{\tilde{A}T_s}, \Gamma = \int_0^{T_s} e^{\tilde{A}\tau} d\tau \tilde{B}. \quad (13)$$

Note that there are other assumptions on the behavior of $u^{(m-1)}$ between the sample points which also makes it possible to calculate an exact discrete time description, for example that it is piecewise linear.

4 Noise model

Systems are often affected by other factors than the known inputs. These factors can from a modeling perspective be seen as unmeasured inputs. It is also possible that the measurement $y(t)$ is corrupted by noise. These effects can be included by adding a *noise model*.

In the case of linear DAE systems a noise model can be added according to (14), where $v_1(t)$ represents the unmodeled inputs and $v_2(t)$ represents the measurement noise. M is a constant matrix.

$$E\dot{\xi}(t) = J\xi(t) + Ku(t) + Mv_1(t) \quad (14a)$$

$$y(t) = L\xi(t) + v_2(t) \quad (14b)$$

(14) can also be written as

$$E\dot{\xi}(t) = J\xi(t) + \begin{bmatrix} K & M \end{bmatrix} \begin{bmatrix} u(t) \\ v_1(t) \end{bmatrix} \quad (15a)$$

$$y(t) = L\xi(t) + v_2(t). \quad (15b)$$

Under certain assumptions, it is possible to make a transformation to a state space system as in the case without noise model. The first step to achieve this is to apply Lemma 1 to (15) get that the solution can be described by the normal form (in the rest of this section dependence of t will be omitted from the notation if the dependence is clear)

$$\dot{w}_1 = Aw_1 + B_1u + B_2v_1 \quad (16a)$$

$$w_2 = \sum_{i=0}^{m-1} (-N)^i D_1 u^{(i)} + \sum_{i=0}^{m-1} (-N)^i D_2 v_1^{(i)} \quad (16b)$$

$$y = LT^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + v_2. \quad (16c)$$

A normal assumption when E is the identity matrix (i.e. in the case of a normal state space description) is that $v_1(t)$ and $v_2(t)$ are white noise signals. Sometimes it is also assumed that they have a Gaussian distribution. The reason is usually that the state estimation problem then is easy (the solution is obtained by the Kalman filter). To estimate the states of (14) it is desirable to make the same assumption here. Time continuous white noise signals are however delicate mathematical objects, for example the integral and derivative are not well defined in the normal sense. See further [2]. Therefore one has to consider carefully what (14) means if $v_1(t)$ and $v_2(t)$ are assumed to be white noise signals. Furthermore derivatives of $v_1(t)$ cannot be allowed to occur in (16), that is it must be required that

$$ND_2 = 0. \quad (17)$$

Note that (17) is related to the so called *impulse controllability*, see for example [6], with respect to $v_1(t)$, but this viewpoint is not further pursued here.

Now, if it is assumed that the matrix M in (14) is such that (17) is fulfilled (see [13] for a discussion on the relation between (17) and M), (16) can be

written as

$$\dot{w}_1 = Aw_1 + B_1u + B_2v_1 \quad (18a)$$

$$w_2 = \sum_{i=0}^{m-1} (-N)^i D_1 u^{(i)} + D_2 v_1 \quad (18b)$$

$$y = LT^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + v_2. \quad (18c)$$

Now (18) is to be transformed to a state space description with $u^{(m-1)}$ as the input using the same method as in section 3. The vector w_3 is thus defined according to (7), which gives the description

$$\dot{w}_1 = Aw_1 + \Psi_3 w_3 + B_2 v_1 \quad (19a)$$

$$w_2 = \Psi_4 w_3 + \Psi_5 u^{(m-1)} + D_2 v_1 \quad (19b)$$

$$\dot{w}_3 = \Psi_1 w_3 + \Psi_2 u^{(m-1)} \quad (19c)$$

$$y = LT^{-1} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} + v_2. \quad (19d)$$

Eliminating $w_2(t)$ and letting $x = [w_1^T \ w_3^T]^T$ gives

$$\dot{x} = \tilde{A}x + \tilde{B}u^{(m-1)} + \underbrace{\begin{bmatrix} B_2 \\ 0 \end{bmatrix}}_{\tilde{B}_2} v_1 \quad (20a)$$

$$y = \tilde{C}x + \tilde{D}u^{(m-1)} + \underbrace{LT^{-1} \begin{bmatrix} 0 \\ D_2 \end{bmatrix}}_{\tilde{N}} v_1 + v_2 \quad (20b)$$

which can be written as

$$\dot{x} = \tilde{A}x + \tilde{B}u^{(m-1)} + \tilde{B}_2 v_1 \quad (21a)$$

$$y = \tilde{C}x + \tilde{D}u^{(m-1)} + \begin{bmatrix} \tilde{N} & I \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}. \quad (21b)$$

It is thus possible to describe the solutions of a DAE system with noise model (14) with a state space system under the assumption (17) that $ND_2 = 0$. Note however, that in the state space model, the noise at the output equation is related to the noise at the state equation through the $v_1(t)$ term. This relation can however be eliminated if the stronger assumption that $D_2 = 0$ is made. Then $\tilde{N} = 0$ so the state space description simplifies to

$$\dot{x} = \tilde{A}x + \tilde{B}u^{(m-1)} + \tilde{B}_2 v_1 \quad (22a)$$

$$y = \tilde{C}x + \tilde{D}u^{(m-1)} + v_2. \quad (22b)$$

Here, the noise at the state and output equations are related only if v_1 and v_2 are.

4.1 Discretization

Also when a noise model is included, it is interesting to examine what the discrete time description of the DAE system is. If the noise is assumed to be piecewise constant, Result 1 can be used to obtain a discrete time representation. It can also be assumed that $v_1(t)$ and $v_2(t)$ is *white noise* in (21) and (22). For this case formulas which can transform the state space descriptions (21) and (22) to discrete time are available in for example [2] and are not repeated here.

4.2 A shortcut

It may be a substantial work to examine what the corresponding state space and discrete time descriptions of (14) is. If there is no prior knowledge about the noise model, and it is desired to identify it as a black box model, it might be better to add the noise model after the conversion to the state space form (11), or even after discretization of the system. Then the noise model does not need to be converted and the assumption (17) does not have to be considered.

5 Kalman filtering

It is now established that it under some assumptions is possible to transfer a continuous time DAE system to a discrete time state space system which gives an equivalent description of the output at the sampling instants. This opens up the possibility to use a discrete time *Kalman filter* to estimate the states and make predictions. To be concrete, assume that a DAE system has been converted to the state space system (23). It is then straightforward to implement a discrete time invariant Kalman filter, e.g., [1].

$$x(t+1) = Ax(t) + Bu(t) + Nv_1(t) \quad (23a)$$

$$y(t) = Cx(t) + Du(t) + v_2(t). \quad (23b)$$

Note that a discretized DAE system just is a special case of this model with proper time scaling and redefinitions. For example $u^{(m-1)}(t)$ is renamed to $u(t)$.

It can be noted that the approach to Kalman filtering for DAE models suggested in this section is related to the approach for discrete time descriptor systems which is presented in [5].

6 Time domain identification

As it has been shown that the DAE model under some assumptions can be described by a discrete time state space system, a large amount of theory for system identification is available, e.g., [7]. The probably most common method available is the prediction error method with a quadratic criterion. This method is based on using the Kalman filter as a one-step ahead predictor and then minimizing the prediction error. It is also possible to do maximum likelihood estimation if $v_1(t)$ and $v_2(t)$ are assumed to have a Gaussian distribution. Then the Kalman filter is used to calculate the probability distribution of the output, and the parameters are estimated by maximizing the probability of the measured output.

7 Frequency domain identification

The work which has been done until now has been based on transforming the DAE system to the so called normal form. Although the normal form always exists if $\det(sE - J) \neq 0$, there may be problems to calculate it, for example one might encounter numerical difficulties. The state space form which the normal form produces might also not always be well suited for estimation purposes, especially if the input has to be differentiated numerically.

A way around these problems is identification in the frequency domain in continuous time, where the model should be specified by transfer functions according to (24).

$$Y(\omega) = G(i\omega)U(\omega) + H(i\omega)V_1(\omega) + V_2(\omega). \quad (24)$$

A linear DAE system with a noise model (14) can be transformed to the form (24) under the usual assumption that $\det(sE - J) \neq 0$. The only difference from the transfer functions of a state space system is that G and H may not be proper in the general case. Therefore $u(t)$ must be differentiable sufficiently many times, which of course is as many times as it must be differentiated in the normal form. Furthermore, if $v_1(t)$ is assumed to be white noise, the M matrix must be such that H is proper. To transform (14) to the frequency domain the equations are Fourier transformed to get

$$Ei\omega\xi(\omega) = J\xi(\omega) + KU(\omega) + MV_1(\omega) \quad (25a)$$

$$Y(\omega) = L\xi(\omega) + V_2(\omega). \quad (25b)$$

The desired description (26) is then achieved by eliminating $\xi(\omega)$.

$$\begin{aligned} Y(\omega) &= \underbrace{L(i\omega E - J)^{-1}K}_{G(i\omega)} U(\omega) \\ &\quad + \underbrace{L(i\omega E - J)^{-1}M}_{H(i\omega)} V_1(\omega) + V_2(\omega) \end{aligned} \quad (26)$$

Now, when it is established that a DAE system can be described by transfer functions, all that has to be done is to plug the transfer function into any identification algorithm for the frequency domain. Books which treat identification in the frequency domain are [7, 9]. One possible selection of identification method is to minimize the criterion

$$\sum_{k=1}^N |Y(\omega_k) - G(i\omega_k)U(\omega_k)|^2 \cdot \frac{1}{|H(i\omega_k)|^2} \quad (27)$$

with respect to the unknown parameters.

Note that the Fourier transforms of the input and output signals are needed, but these can be computed from time domain data.

8 An example

In this section the algorithms examined earlier in the paper are exemplified on a physical system. The system setup examined is a model of a generator, see Figure 1. Dependence on time is omitted in this section.

Figure 1: A model of a generator.

The model can be described as follows: The input is the angle θ_{in} on the left axis. This axis is connected to a rotating mass with inertia J which is rotated the angle θ and rotates with the angular velocity ω . The torque acting on the left side of the mass is M_1 and the torque on the right side is M_2 . The mass is then connected to a second axis which is connected to the actual generator. The variables describing the second axis and the electrical quantities are then assumed to depend on each other according to $M_2 = kI$ and $u_1 = k\omega$ for some constant k . The rest of the electrical circuit consists of two resistors and one inductor. The measured output is the voltage u_4 .

Note that it here is easy to spot that too many variables are included in the model if the torques are not interesting, as the mass then will have no impact on the system. This is to show that the identification algorithm treats this automatically. (In larger models it may be more difficult to spot that some variables are unnecessary.)

The system is now modeled in an object-oriented manner by just writing down the equations describing the different parts and the ways they are connected:

$\theta_{in} = \theta$	$\dot{\theta} = \omega$	$J\dot{\omega} = M_1 + M_2$
$M_2 = kI$	$u_1 = k\omega$	$u_2 = L\dot{I}$
$u_3 = R_1 I$	$u_4 = R_2 I$	$u_1 = u_2 + u_3 + u_4$

To apply the identification methods examined in Section 2, these equations must be written on the form (1):

$$\begin{pmatrix}
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & J & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & L & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{pmatrix}
 \begin{pmatrix}
 \dot{M}_1 \\
 \dot{M}_2 \\
 \dot{\omega} \\
 \dot{\theta} \\
 \dot{I} \\
 \dot{u}_1 \\
 \dot{u}_2 \\
 \dot{u}_3 \\
 \dot{u}_4
 \end{pmatrix}
 +$$

$$\begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -k & 0 & 0 & 0 & 0 \\
0 & 0 & -k & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & -R_1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & -R_2 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1
\end{pmatrix}
\begin{pmatrix}
M_1 \\
M_2 \\
\omega \\
\theta \\
I \\
u_1 \\
u_2 \\
u_3 \\
u_4
\end{pmatrix}
=
\begin{pmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{pmatrix}
\theta_{in}$$

$$y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix}
M_1 \\
M_2 \\
\omega \\
\theta \\
I \\
u_1 \\
u_2 \\
u_3 \\
u_4
\end{pmatrix}$$

8.1 Time domain identification

The first step when identifying in the time domain is to transform the model to the form (11) (here, no noise model is added before the transformation). In an automated identification procedure, this transformation is probably best made numerically for each value of the parameters that the identification algorithm tries during the minimization. Stable numerical algorithms for making this computation is a subject for future research.

In this simple case, it is however possible to make the transformation to a state space system by algebraic manipulations. If it is assumed that $L \neq 0$, the transformed system is

$$\dot{x} = -\frac{R_1 + R_2}{L}x + \frac{k^2(R_1 + R_2)}{L^2}\theta_{in} \quad (28a)$$

$$y = -\frac{R_1}{k}x + \frac{R_1 k}{L}\theta_{in}. \quad (28b)$$

If it is assumed that $L = 0$, the transformed system is

$$y = \frac{R_2 k}{R_1 + R_2}\dot{\theta}_{in}. \quad (29)$$

In both of these cases, the transformation leads to a description which can be used in standard identification algorithms for the time domain.

8.2 Frequency domain identification

To do identification in the frequency domain, the transfer function for the model must be calculated. This can be done using equation (26) (it is assumed that $M = 0$ in this equation, i.e., an output error model is selected). The result is

$$G(s) = \frac{R_2 k s}{L s + R_1 + R_2}. \quad (30)$$

This transfer function can be used in standard identification algorithms for the frequency domain.

9 Conclusions

This paper has treated how the parameter identification problem can be solved for linear DAE systems. In the time domain, a prediction error method or maximum likelihood identification can be implemented by constructing a Kalman filter. This requires that the descriptor system is converted to a special form. Although this form can always be calculated in theory under mild assumptions, it may be difficult to find in practice. Therefore it might be better to do identification in the frequency domain in some cases.

Future studies on the subject include to examine if there are numerically robust algorithms to find the normal form and to examine how identification can be done for nonlinear DAE models.

10 Acknowledgments

This work was supported by the Swedish Research Council and by the Foundation for Strategic Research (SSF).

References

- [1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering*. Information and System Sciences Series. Prentice-Hall inc, Englewood Cliffs, New Jersey, 1979.
- [2] K.J. Åström. *Introduction to Stochastic Control Theory*. Mathematics in Science and Engineering. Academic Press, New York and London, 1970.
- [3] K.J. Åström and B. Wittenmark. *Computer Controlled Systems, Theory and Design*. Information and System Sciences Series. Prentice-Hall, Englewood Cliffs, N.J., 1984.
- [4] K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Classics In Applied Mathematics. SIAM, Philadelphia, 1996.
- [5] L. Dai. State estimation schemes for singular systems. In *Preprints of the 10th IFAC World Congress, Munich, Germany*, volume 9, pages 211–215, 1987.
- [6] L. Dai. *Singular Control Systems*. Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin, New York, 1989.
- [7] L. Ljung. *System Identification - Theory for the User*. Information and System Sciences Series. Prentice Hall PTR, Upper Saddle River, N.J., 2. edition, 1999.
- [8] S.E. Mattsson, H. Elmqvist, and M. Otter. Physical system modeling with Modelica. *Control Engineering Practice*, 6:501–510, 1998.
- [9] R. Pintelon and J. Schoukens. *System Identification: A frequency domain approach*. IEEE Press, New York, 2001.

- [10] J.W. Polderman and J.C. Willems. *Introduction to Mathematical Systems Theory: a behavioral approach*. Number 26 in Texts in Applied Mathematics. Springer-Verlag, New York, 1998.
- [11] H. H. Rosenbrock. *State-space and Multivariable Theory*. John Wiley & Sons, Inc., New York, 1970.
- [12] K. Schittkowski. *Numerical Data Fitting in Dynamical Systems*. Kluwer Academic Publishers, Dordrecht, 2002.
- [13] T. Schön, M. Gerdin, T. Glad, and F. Gustafsson. A modeling and filtering framework for linear implicit systems. Technical Report LiTH-ISY-R-2498, Dept. of Electr. Eng., Linköpings universitet, www.control.isy.liu.se/publications/, March 2003.
- [14] M. Tiller. *Introduction to Physical Modeling with Modelica*. Kluwer, Boston, Mass., 2001.
- [15] G.C. Verghese, B.C. Lévy, and T. Kailath. A generalized state-space for singular systems. *IEEE Transactions on Automatic Control*, AC-26(4):811–831, August 1981.