Symbolic-Numeric Methods for Improving Structural Analysis of Differential-Algebraic Equation Systems

Guangning Tan, Nedialko S. Nedialkov
McMaster University

John D. Pryce
Cardiff University

May 13, 2015

Abstract

Systems of differential-algebraic equations (DAEs) are generated routinely by simulation and modeling environments such as MODELICA and MAPLESIM. Before a simulation starts and a numerical solution method is applied, some kind of structural analysis is performed to determine the structure and the index of a DAE. Structural analysis methods serve as a necessary preprocessing stage, and among them, Pantelides's algorithm is widely used. Recently Pryce's Σ-method is becoming increasingly popular, owing to its straightforward approach and capability of analyzing high-order systems. Both methods are equivalent in the sense that when one succeeds, producing a nonsingular system Jacobian, the other also succeeds, and the two give the same structural index.

Although provably successful on fairly many problems of interest, the structural analysis methods can fail on some simple, solvable DAEs and give incorrect structural information including the index. In this report, we focus on the Σ-method. We investigate its failures, and develop two symbolic-numeric conversion methods for converting a DAE, on which the Σ-method fails, to an equivalent problem on which this method succeeds. Aimed at making structural analysis methods more reliable, our conversion methods exploit structural information of a DAE, and require a symbolic tool for their implementation.

Acknowledgements

We acknowledge with thanks the support given to GT by the Ontario Research Fund (ORF), Canada, and McMaster Centre for Software Certification (McSCert), NSN by the Natural Sciences and Engineering Research Council of Canada (NSERC), and JDP by the Leverhulme Trust, the UK.

Contents

Chapter 1

Introduction

We are interested in solving initial value problems in DAEs of the general form

$$f_i(\,t, \text{the } x_j \text{ and derivatives of them}\,) = 0, \quad i = 1 : n, \tag{1.1}$$

where the $x_j(t)$ are n state variables, and t is the time variable. The formulation (1.1) includes high-order systems and systems that are jointly nonlinear in leading derivatives. Moreover, (1.1) includes ordinary differential equations (ODEs) and purely algebraic systems.

An important characteristic of a DAE is its *index*. Generally, the index measures the difficulty of solving a DAE numerically. If a DAE is of index-1, then a general index-1 solver can be used, e.g., DASSL [3], IDA of SUNDIALS [14], and MATLAB's `ode15s` and `ode23t`. If a DAE is of high index, that is, index ≥ 2, then we need a high-index DAE solver, e.g., RADAU5 for DAEs of index ≤ 3 [13] or DAETS for DAEs of any index [22]. We can also use index reduction techniques to convert the original DAE to an index-1 problem [17, 19, 33], and then apply an index-1 solver.

Structural analysis (SA) methods serve as a preprocessing stage to help determine the index. Among them is the Pantelides's method [25], which is a graph-based algorithm that finds how many times each equation needs to be differentiated. Pryce's structural analysis—the *Signature method* or *Σ-method*—is essentially equivalent to that of Pantelides [27], and in particular computes the same *structural index* when both methods succeed. However, Pantelides's algorithm can only handle first-order systems, while Pryce's can be applied to (1.1) of any order and is generally easier to apply.

This SA determines the structural index, which is often the same as the *differentiation index*, the number of degrees of freedom, the variables and derivatives that need to be initialized, and the constraints of the DAE. We give the definition of the differentiation index in §2 and that of the structural index in §3.

Nedialkov and Pryce [20, 21, 22] use the Σ-method to analyze a DAE of the form (1.1), and solve it numerically using Taylor series. On each integration step, Taylor coefficients (TCs) for the solution are computed up to some order. These coefficients are computed in a stage-wise manner. This stage by stage solution scheme, also derived from the SA, indicates at each stage which equations need to be solved and for which variables [24]. In [2, 12, 15], the Σ-method is also applied to perform structural analysis, and the resulting offset vectors are used to prescribe the computation of TCs.

Although the Σ-method provably gives correct structural information (including index) on many DAEs of practical interest [27], it can fail—whence also Pantelides's algorithm and other SA methods [34, 35] can fail—to find a DAE's true structure, producing an identically singular system Jacobian. (See §3 for the definition of system Jacobian.)

Scholz et al. [33] show that several simulation environments such as DYMOLA, OPENMODELICA and SIMULATIONX all fail on a simple, solvable 4×4 linear constant coefficient DAE; we discuss this DAE in Example 4.18. Other examples where SA fails are the Campbell-Griepentrog Robot Arm [5] and the Ring Modulator [18]. When SA fails, the structural index usually underestimates the differentiation index. In other cases, when SA produces a nonsingular system Jacobian, the structural index may overestimate the differentiation index [31]. We review in Appendix B how these DAEs in the early literature are handled so that SA reports the correct index.

SA can fail if there are hidden symbolic cancellations in a DAE; this is the simplest case among SA's failures. However, SA can fail in a more obscure way. In this case, it is difficult to understand the causes of such failures and to provide fixes to the formulation of the problem. Such deficiencies can pose limitations to the application of SA, as it becomes unreliable. Our goal is to construct methods that convert automatically a system on which SA fails into an equivalent form on which it succeeds. This report is devoted to developing such methods.

It is organized as follows. Chapter 2 overviews work that has been done to date. Chapter 3 summarizes the Σ-method and gives definitions and tools that are needed for our theoretical development. The problem of SA's failures on some DAEs is described in Chapter 4. In Chapters 5 and 6, we develop two methods, the *linear combination method* and the *expression substitution method*, respectively. We show in Chapter 7 how to apply our methods on several examples. Chapter 8 gives conclusions and indicates several research directions.

Chapter 2

Background

The index of a DAE is an important concept in DAE theory. There are various definitions of an index: differentiation index [4, 9, 10], geometric index [30, 32], structural index [7, 25, 27], perturbation index [13], tractability index [11], and strangeness index [16].

The most commonly used index is the *differentiation index*; we refer to it as d-index or ν_d. The following definition is from [1, p. 236].

Definition 2.1 *Consider a general form of a first-order DAE*

$$\mathbf{F}(t,\mathbf{x},\mathbf{x}') = \mathbf{0}, \tag{2.1}$$

where $\partial\mathbf{F}/\partial\mathbf{x}'$ may be singular. The differentiation index *along a solution $\mathbf{x}(t)$ is the minimum number of differentiations of the system that would be required to solve $\mathbf{x}'$ uniquely in terms of $\mathbf{x}$ and t, that is, to define an ODE for $\mathbf{x}$. Thus this index is defined in terms of the overdetermined system*

$$\begin{aligned} \mathbf{F}\left(t,\mathbf{x},\mathbf{x}'\right) &= \mathbf{0}, \\ \frac{d\mathbf{F}}{dt}\left(t,\mathbf{x},\mathbf{x}',\mathbf{x}''\right) &= \mathbf{0}, \\ &\vdots \\ \frac{d^p\mathbf{F}}{dt^p}\left(t,\mathbf{x},\mathbf{x}',\cdots,\mathbf{x}^{(p+1)}\right) &= \mathbf{0} \end{aligned}$$

to be the smallest integer p so that $\mathbf{x}'$ in (2.1) can be solved for in terms of $\mathbf{x}$ and t.

If a DAE (1.1) is of high-order, then one can introduce additional variables to reduce the order of the system so that it is still in the general form (2.1).

We give a definition for solution of a DAE.

Definition 2.2 *An n-vector valued function $\mathbf{x}(t)$, defined on a time interval $\mathbb{I} \subset \mathbb{R}$, is a* solution *of (1.1), if $(t,\mathbf{x}(t))$ satisfies $f_i = 0$, $i = 1:n$, pointwise for all $t \in \mathbb{I}$: that is, every f_i* vanishes *on $\mathbb{I}$.*

Reißig et al. [31] claim that a DAE of d-index 1 may have arbitrarily high structural index. They construct a class of linear constant coefficient DAEs in some specific form. On these DAEs of d-index 1, Pantelides's algorithm performs a high number of iterations and differentiations, and obtains a high structural index that far exceeds the d-index 1. A simple 3×3 linear electrical circuit example is also presented: choosing a specific node as the ground node results in a DAE of d-index 1, but of structural index 2.

Pryce [27] shows that, if the Σ-method succeeds, then the structural index ν_S is always an upper bound on the d-index. This implies that, if the structural index computed by the Σ-method is smaller than the d-index, then the method *must* fail; otherwise we would have a statement that contradicts to the above Definition 2.1. Pryce also shows that the Σ-method succeeds on one of Reißig's DAEs and produces a nonsingular system Jacobian [27]. His method also produces the same high structural index as does Pantelides's.

In [26], Pryce shows that the Σ-method fails on the index-5 Campbell-Griepentrog Robot Arm DAE—the SA produces an identically singular Jacobian. He then provides a remedy: identify the common subexpressions in the problem, introduce extra variables, and substitute them for those subexpressions. The resulting equivalent problem is an enlarged one, where the Σ-method succeeds and reports the correct structural index 5. Pryce introduces the term *structure-revealing* to conjecture that a nonsingular system Jacobian might be an effect of DAE formulation, but not of DAE's inherent nature.

Choudhry et al. [6] propose a method called symbolic numeric index analysis (SNIA). Their method can accurately detect symbolic cancellation of variables that appear linearly in equations, and therefore can deal with linear constant coefficient systems. For general nonlinear DAEs, SNIA provides a correct result in some cases, but not all. Furthermore, it is limited to order-1 systems, and it cannot handle complex expression substitution and symbolic cancellations, such as $(x\cos y)' - x'\cos y$. For the general case, their method does not derive from the original problem an equivalent one that has the correct index.

Scholz et al. [33] are interested in a class of DAEs called coupled systems. In their case, a coupled system is composed by coupling two semi-explicit d-index 1 systems. They show that the Σ-method succeeds if and only if the coupled system is again of d-index 1. As a consequence, if the coupled system is of high index, SA methods *must* fail. They develop a structural-algebraic approach to deal with such coupled systems. They differentiate a linear combination of certain algebraic equations that contribute to singularity, append the resulting equations, and replace certain derivatives with newly introduced variables. They use this *regularization* process to convert the regular coupled system to a d-index 1 problem, on which SA succeeds with nonsingular Jacobian.

Chapter 3

Summary of Pryce's structural analysis

We call this SA [27] the Σ-method, because it constructs for (1.1) an $n \times n$ *signature matrix* $\Sigma = (\sigma_{ij})$ such that

$$\sigma_{ij} = \begin{cases} \text{the order of the highest order derivative to which } x_j \text{ occurs in } f_i\text{; or} \\ -\infty \;\; \text{if } x_j \text{ does not occur in } f_i. \end{cases} \tag{3.1}$$

A *transversal* T is a set of n positions (i, j) with one entry in each row and each column. The sum of entries σ_{ij} over T, or $\sum_{(i,j)\in T} \sigma_{ij}$, is called the *value* T, written $\mathrm{Val}(T)$. We seek a *highest-value transversal* (HVT) that gives this sum the largest value. We call this number the *value of the signature matrix*, written $\mathrm{Val}(\Sigma)$.

We give a definition for a DAE's structural posed-ness.

Definition 3.1 *We say that a DAE is* structurally well-posed *(SWP) if its* $\mathrm{Val}(\Sigma)$ *is finite. That is, all entries in a HVT are finite, or equivalently, there exists some finite transversal. Otherwise, if* $\mathrm{Val}(\Sigma) = -\infty$*, then we say a DAE is* structurally ill-posed *(SIP).*

For a SWP DAE, we find *equation and variable offsets* **c** and **d**, respectively, which are non-negative integer n-vectors satisfying

$$c_i \geq 0; \quad d_j - c_i \geq \sigma_{ij} \quad \text{for all } i, j \text{ with equality on a HVT.} \tag{3.2}$$

An equality $d_j - c_i = \sigma_{ij}$ on some HVT also holds on all HVTs [29]. We refer to **c** and **d** satisfying (3.2) as *valid offsets*. They are not unique, but there exists unique **c** and **d** that are the smallest component-wise valid offsets. We refer to them as *canonical offsets*.

The *structural index* is defined by

$$\nu_S = \begin{cases} \max_i c_i + 1 & \text{if } d_j = 0 \text{ for some } j\text{, or} \\ \max_i c_i & \text{otherwise.} \end{cases}$$

Critical to the success of this method is the nonsingularity of the DAE's $n \times n$ *system Jacobian* matrix $\mathbf{J} = (\mathbf{J}_{ij})$, where

$$\mathbf{J}_{ij} = \frac{\partial f_i}{\partial x_j^{(d_j-c_i)}} = \begin{cases} \partial f_i / \partial x_j^{(\sigma_{ij})} & \text{if } d_j - c_i = \sigma_{ij}, \text{ and} \\ 0 & \text{otherwise.} \end{cases} \tag{3.3}$$

Note that $\mathbf{J} = \mathbf{J}(\mathbf{c}, \mathbf{d})$ depends on the choice of valid offsets $\mathbf{c}, \mathbf{d}$, which satisfy (3.2). That is, using different valid offsets, one may obtain different system Jacobians. However, they all have the same determinant; see Theorem 4.15. For all the examples in this report, we shall use canonical offsets and the system Jacobian derived from them.

We can use Σ and $\mathbf{c}, \mathbf{d}$ to determine a *solution scheme* for computing derivatives of the solution to (1.1). They are computed in stages

$$k = k_d, k_d + 1, \ldots, 0, 1, \ldots \quad \text{where } k_d = -\max_j d_j.$$

At each stage we solve equations

$$0 = f_i^{(c_i+k)} \quad \text{for all } i \text{ such that } c_i + k \geq 0 \tag{3.4}$$

for derivatives

$$x_j^{(d_j+k)} \qquad \text{for all } j \text{ such that } d_j + k \geq 0 \tag{3.5}$$

using the previously found

$$x_j^{(r)} \qquad \text{for all } j \text{ such that } 0 \leq r < d_j + k.$$

We refer to [24] for more details on this solution scheme; see also Example 3.2.

Throughout this report, for brevity, we write "derivatives of x_j" instead of "x_j and derivatives of it"—derivatives $v^{(l)}$ of a variable v include v itself as the case $l = 0$.

If the solution scheme (3.4–3.5) can be carried out up to stage $k = 0$, and the derivatives of each variable x_j can be uniquely determined up to order d_j, then we say the solution scheme and the SA *succeed*. The system Jacobian is nonsingular at a point

$$\left(t;\, x_1, \ldots, x_1^{(d_1)};\, x_2, \ldots, x_2^{(d_2)};\, \ldots;\, x_n, \ldots, x_n^{(d_n)}\right), \tag{3.6}$$

and there exists a unique solution through this point [20, 27, 29]. We say the DAE is *locally solvable*, and call (3.6) a *consistent point*, if derivatives $x_j^{(d_j)}$ do not occur jointly linearly in $f_i^{(c_i)}$. In the linear case, a consistent point is

$$\left(t;\, x_1, \ldots, x_1^{(d_1-1)};\, x_2, \ldots, x_2^{(d_2-1)};\, \ldots;\, x_n, \ldots, x_n^{(d_n-1)}\right). \tag{3.7}$$

For a more rigorous discussion of a consistent point, we refer the readers to [20, 24, 29].

To perform a numerical check for SA's success, or a *success check* for short, we attempt to compute numerically a consistent point at which **J** is nonsingular up to roundoff: we provide an appropriate set of derivatives of x_j's and follow the solution scheme (3.4–3.5) for stages $k = k_d : 0$. This set of derivatives is the set of *initial values* for a DAE initial value problem, and a minimal set of derivatives required for initial values is discussed in [29].

When SA succeeds, the structural index is an upper bound for the differentiation index, and often they are the same: $\nu_d \leq \nu_S$ [27]. Also, the *number of degrees of freedom* (DOF) is

$$\text{DOF} = \text{Val}(\Sigma) = \sum_j d_j - \sum_i c_i = \sum_{(i,j)\in T} \sigma_{ij}.$$

We say the solution scheme and SA *fails*, if we cannot determine uniquely a consistent point using the solution scheme defined by (3.4–3.5)—otherwise said, we cannot follow the solution scheme up to stage $k = 0$ and find a consistent point at which **J** is nonsingular. In our experience, in the failure case usually $\nu_d > \nu_S$, but not always, and the true number of DOF is overestimated by $\text{Val}(\Sigma)$. This is discussed in Examples 4.7, 4.9, 4.18, 4.19.

We illustrate the above concepts using the following example.

Example 3.2 The simple pendulum DAE (PEND) in Cartesian coordinates is

$$\begin{aligned} 0 &= f_1 = x'' + x\lambda \\ 0 &= f_2 = y'' + y\lambda - g \\ 0 &= f_3 = x^2 + y^2 - L^2. \end{aligned} \tag{3.8}$$

Here the state variables are x, y, λ; g is gravity, and $L > 0$ is the length of the pendulum.

The signature matrix and system Jacobian of this DAE are

$$\Sigma = \begin{array}{c|ccc|c} & x & y & \lambda & c_i \\ \hline f_1 & 2^\bullet & - & 0 & 0 \\ f_2 & - & 2 & 0^\bullet & 0 \\ f_3 & 0 & 0^\bullet & - & 2 \\ \hline d_j & 2 & 2 & 0 & \end{array} \quad \text{and} \quad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \end{array} \begin{array}{c} \begin{array}{ccc} x & y & \lambda \end{array} \\ \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 2x & 2y & 0 \end{bmatrix} \end{array}.$$

We write Σ in a *signature tableau*: a HVT is marked by $\bullet$; $-$ denotes $-\infty$; the canonical offsets **c**, **d** are annotated on the right of Σ and at the bottom of it, respectively.

The structural index is

$$\nu_S = \max_i c_i + 1 = c_3 + 1 = 3,$$

which is the same as the d-index. The number of degrees of freedom is

$$\text{DOF} = \sum_j d_j - \sum_i c_i = 2.$$

stage k	solve	for	using previously found
-2	$0 = f_3$	x, y	$-$
-1	$0 = f_3'$	x', y'	x, y
≥ 0	$0 = f_1^{(k)}, f_2^{(k)}, f_3^{(k+2)}$	$x^{(k+2)}, y^{(k+2)}, \lambda^{(k)}$	$x^{(<k+2)}, y^{(<k+2)}, \lambda^{(<k)}$

Table 3.1: Solution scheme for (3.8)

Since the derivatives $x_j^{(d_j)}$, $j = 1, 2, 3$, that is, x'', y'', λ, occur jointly linearly in (3.8), a consistent point is given by (t, x, x', y, y'). If we evaluate $\mathbf{J}$ at this point, then

$$\det(\mathbf{J}) = -2(x^2 + y^2) = -2L^2 \neq 0$$

(because $x^2 + y^2 = L^2$ by $f_3 = 0$) and SA succeeds [27]. The solution scheme is in Table 3.1. The notation $z^{(<r)}$ is short for $z, z', \ldots, z^{(r-1)}$.

For brevity, in the following chapters, when we give a system of equations, we write down

- the signature matrix,
- a HVT in it (marked by •),
- the canonical offsets $\mathbf{c}$, $\mathbf{d}$,
- positions (i, j) where $d_j - c_i > \sigma_{ij} \geq 0$ (marked by ▒), and
- the accompanying system Jacobian.

When we present a SA result, we omit the words

> "the signature matrix and system Jacobian are in the following."

Provided there is a finite HVT in Σ, we also show the value of the signature matrix and the determinant of the system Jacobian—Val(Σ) and det($\mathbf{J}$). For instance, after giving (3.8), we simply put Σ with Val(Σ) attached, and $\mathbf{J}$ with det($\mathbf{J}$) at the bottom.

$$\Sigma = \begin{array}{c|ccc|c} & x & y & \lambda & c_i \\ \hline f_1 & 2^\bullet & - & 0 & 0 \\ f_2 & - & 2 & 0^\bullet & 0 \\ f_3 & 0 & 0^\bullet & - & 2 \\ \hline d_j & 2 & 2 & 0 & \end{array} \quad \mathrm{Val}(\Sigma) = 2$$

$$\mathbf{J} = \begin{array}{c|ccc|} & x & y & \lambda \\ \hline f_1 & 1 & 0 & x \\ f_2 & 0 & 1 & y \\ f_3 & 2x & 2y & 0 \end{array} \quad \det(\mathbf{J}) = -2L^2$$

Similarly, if we write the signature matrix of a system as $\overline{\Sigma}$, then we write correspondingly the canonical offsets as $\overline{\mathbf{c}}$, $\overline{\mathbf{d}}$, and the Jacobian as $\overline{\mathbf{J}}$. Throughout this report, we shall show DAE problems for which our conversion methods are suitable. These methods critically depend on the SA results.

Chapter 4

Structural analysis's failure

In this chapter, we investigate how SA fails on some DAEs. That is, SA produces a singular system Jacobian, and the problem is solvable. In §4.1, we give definitions for (a) a structural zero in the system Jacobian, and (b) a structurally singular DAE, where the system Jacobian is identically singular. In §4.2 we identify two types of SA's failure.

4.1 Success check

To perform a success check for SA on a SWP DAE, we attempt to evaluate the system Jacobian $\mathbf{J}$ in (3.3). If a point (3.6) satisfies the solution scheme (3.4–3.5) at stages $k = k_d, k_d + 1, \ldots, 0$, and $\mathbf{J}$ is nonsingular, then SA succeeds.

In the definitions that follow, we let A be an $n \times n$ matrix function.

Definition 4.1 *An (i, j) position is a* structural zero *of A if A_{ij} is identically* 0*; otherwise it is a* structural nonzero.

Definition 4.2 *[20] Matrix A is* structurally singular *if every $B \in \mathbb{R}^{n\times n}$, with $B_{ij} = 0$ in A's structural zero positions, is singular—equivalently, if every transversal of A contains a structural zero. Otherwise A is* structurally nonsingular .

Definition 4.3 *Matrix A is* identically singular*, if its determinant is identically* 0*; otherwise it is* generically nonsingular.

For a matrix function, being structurally singular is a special case of being identically singular; see Example 4.4 below.

Example 4.4 Consider the following three matrix functions of variables x and y:

$$A_1 = \begin{bmatrix} x & x \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} x & x \\ y & y \end{bmatrix}, \quad \text{and} \quad A_3 = \begin{bmatrix} x & y \\ y & x \end{bmatrix}.$$

A_1 is identically singular because $\det(A_1) = 0$. It is also structurally singular, since every $B \in \mathbb{R}^{2\times 2}$ with $B_{21} = B_{22} = 0$ is singular. Here, $(2,1)$ and $(2,2)$ are structural zero positions of A, and each transversal in A contains a structural zero.

A_2 is also identically singular, as $\det(A_2) = xy - xy = 0$. It is structurally nonsingular, since a transversal does not contain a structural zero.

A_3 is structurally nonsingular. It is generically nonsingular, since $\det(A_3) = x^2 - y^2$ is not identically zero. A_3 is singular only when $x = \pm y$.

In the following, we denote (1.1) by $\mathcal{F}$ and define two concepts for it:

- a *structural zero in the system Jacobian* $\mathbf{J}$, and
- a *structurally singular DAE*.

Let $\mathcal{J}$ be the set of index-pairs

$$\mathcal{J} = \{\, (j,l) \mid j = 1:n,\ l \in \mathbb{N} \,\}. \tag{4.1}$$

Given an n-vector function $\mathbf{x} = \mathbf{x}(t)$ that is sufficiently smooth (but not necessarily a solution of $\mathcal{F}$), let

$$\mathbf{x}_{\mathcal{J}} = \{\, x_j^{(l)} \mid (j,l) \in \mathcal{J} \,\}.$$

For a finite subset J of $\mathcal{J}$, we define a $|J|$-vector $\mathbf{x}_J$ whose components are $x_j^{(l)}$ as (j,l) ranges over J. (The ordering of these components does not matter.)

Now we denote a DAE as $\mathcal{F}$. We define the *derivative set* of $\mathcal{F}$ as

$$\text{derset}(\mathcal{F}) = \{\, (j,l) \mid x_j^{(l)} \text{ occurs in } \mathcal{F} \,\}. \tag{4.2}$$

Then the derivatives occurring in $\mathcal{F}$ can be denoted concisely as $\mathbf{x}_{\text{derset}(\mathcal{F})}$.

By a *value point* we mean a $\xi \in \mathbb{R} \times \mathbb{R}^{|\text{derset}(\mathcal{F})|}$ that contains values for t and values for the derivative symbols in $\mathbf{x}_{\text{derset}(\mathcal{F})}$.

Example 4.5 In the simple pendulum DAE (3.8), the state variables x, y, λ are x_1, x_2, x_3. Let $L = 5$ and $g = 9.8$. Then

$$\text{derset}(\mathcal{F}) = \{\, (1,0), (1,2), (2,0), (2,2), (3,0) \,\}.$$

A possible value point can be

$$\xi = (t, x_1, x_1'', x_2, x_2'', x_3) = (2, 3, -3, 4, 1.6, 1),$$

which satisfies f_1 and f_3 but not f_2.

Similarly, we define the *derivative set of* $\mathbf{J}$:

$$\text{derset}(\mathbf{J}) = \left\{ (j,l) \mid x_j^{(l)} \text{ occurs in } \mathbf{J} \right\}.$$

From (3.3), a derivative occurring in $\mathbf{J}$ must also occur in $\mathcal{F}$, but not vice versa. For example, in PEND, x'', y'', λ do not appear in $\mathbf{J}$, and $\text{derset}(\mathbf{J}) = \left\{ (1,0), (2,0) \right\}$; cf. Example 3.2. The derivative set of $\mathbf{J}$ is a subset of that of $\mathcal{F}$: $\text{derset}(\mathbf{J}) \subseteq \text{derset}(\mathcal{F})$.

Definition 4.6 *An (i,j) position is a* structural zero *of* $\mathbf{J}$*, if* $\mathbf{J}_{ij}$ *is identically zero at all value points* $\xi \in \mathbb{R} \times \mathbb{R}^{|derset(\mathcal{F})|}$ *that satisfy 0 or more equations from*

$$0 = f_i^{(m)}, \quad m \geq 0, \quad i = 1:n. \tag{4.3}$$

Otherwise, (i,j) is a structural nonzero.

For the present purpose, we do not require the DAE to have a unique solution, or even *any* solution. That is, we do not consider existence and uniqueness of the DAE at this stage, while identifying structural zeros of $\mathbf{J}$ and the singularity of $\mathbf{J}$ discussed below.

Recall (3.3) that defines $\mathbf{J}$. If $d_j - c_i > \sigma_{ij}$, then $\mathbf{J}_{ij} = 0$ and thus position (i,j) is a structural zero in $\mathbf{J}$. The converse is not true; see Example 4.7.

Example 4.7 Consider an artificially modified simple pendulum DAE. We multiply the first equation f_1 by $x^2 + y^2 - L^2$ and obtain

$$\begin{aligned} 0 &= f_1 = (x'' + x\lambda)(x^2 + y^2 - L^2) \\ 0 &= f_2 = y'' + y\lambda - g \\ 0 &= f_3 = x^2 + y^2 - L^2. \end{aligned} \tag{4.4}$$

$$\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ d_j \end{array} \begin{array}{c} \begin{array}{ccc} x & y & \lambda \end{array} \\ \begin{bmatrix} 2^\bullet & \boxed{0} & 0 \\ - & 2 & 0^\bullet \\ 0 & 0^\bullet & - \end{bmatrix} \\ \begin{array}{ccc} 2 & 2 & 0 \end{array} \end{array} \begin{array}{c} c_i \\ 0 \\ 0 \\ 2 \\ \text{Val}(\Sigma) = 2 \end{array} \qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \end{array} \begin{array}{c} \begin{array}{ccc} x & y & \lambda \end{array} \\ \begin{bmatrix} \mu & 0 & x\mu \\ 0 & 1 & y \\ 2x & 2y & 0 \end{bmatrix} \\ \det(\mathbf{J}) = -2\mu(x^2 + y^2) \end{array}$$

In $\mathbf{J}$, $\mu = x^2 + y^2 - L^2$. To decide which entries in $\mathbf{J}$ are structural zeros, we notice the following.

- If we evaluate $\mathbf{J}$ at some random ξ, then μ is not identically equal zero. Hence positions (f_1, x) and (f_1, λ) are not identical zeros.

- If we evaluate $\mathbf{J}$ at some ξ that satisfies

  $$\mu = f_3 = x^2 + y^2 - L^2 = 0,$$

  then according to Definition 4.6, positions (f_1, x) and (f_1, λ) are structural zeros of $\mathbf{J}$.

We give a definition for *structural regularity* of a DAE.

Definition 4.8 *A DAE is* structurally singular *if* $\mathbf{J}$ *is identically singular at all value points* $\xi \in \mathbb{R} \times \mathbb{R}^{|derset(\mathcal{F})|}$ *that satisfy 0 or more equations from (4.3). Otherwise the DAE is* structurally nonsingular*, or* structurally regular.

Example 4.9 In the previous example, positions (f_1, x) and (f_1, λ) are structural zeros of $\mathbf{J}$ at any point that satisfies $f_3 = 0$. By Definition 4.8, (4.4) is structurally singular.

In fact, it can be shown that a solution of PEND is a solution to (4.4), but not vice versa.

Example 4.10 Consider the DAE in [1, p. 235, Example 9.2], written in (1.1) form:

$$\begin{aligned} 0 &= f_1 = -y_1' + y_3 \\ 0 &= f_2 = y_2(1 - y_2) \\ 0 &= f_3 = y_1 y_2 + y_3(1 - y_2) - t. \end{aligned} \tag{4.5}$$

$$\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ d_j \end{array} \begin{array}{c} \begin{array}{cccc} y_1 & y_2 & y_3 & c_i \end{array} \\ \left[\begin{array}{ccc} 1^\bullet & - & 0 \\ - & 0^\bullet & - \\ \boxed{0} & 0 & 0^\bullet \end{array}\right] \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \\ \begin{array}{cccc} 1 & 0 & 0 & \mathrm{Val}(\Sigma) = 1 \end{array} \end{array} \qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \end{array} \begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \left[\begin{array}{ccc} -1 & 0 & 1 \\ 0 & 1 - 2y_2 & 0 \\ 0 & y_1 - y_3 & 1 - y_2 \end{array}\right] \\ \det(\mathbf{J}) = -(1 - 2y_2)(1 - y_2) \end{array}$$

SA gives $\nu_S = 1$, and $\det(\mathbf{J})$ depends solely on y_2. From $f_2 = 0$, either $y_2 = 0$ or $y_2 = 1$. To examine if $\mathbf{J}$ is nonsingular, we consider each of the following two cases.

- If $y_2 = 0$, then $\det(\mathbf{J}) = -1$ and SA succeeds. In this case (4.5) is of d-index 1.
- If $y_2 = 1$, then $\det(\mathbf{J}) = 0$ and SA fails. This failure comes as no surprise because (4.5) is now of d-index 2 and SA underestimates its index; see the discussion in §2.

Remark 4.11 For a structurally ill-posed (SIP) DAE, there does not exist a finite transversal in its Σ—every transversal in Σ contains at least one $-\infty$. In this case, there exists no valid offsets $\mathbf{c}, \mathbf{d}$, not to mention a system Jacobian that depends on these offsets. In contrast, a structurally singular DAE has valid offsets and a system Jacobian that is identically singular. Hereby we distinguish the difference between a SIP DAE and a structurally singular DAE.

Suppose $\mathbf{J}$ is generically nonsingular. If $\mathbf{J}$ is singular when evaluated at a point along a solution, then we say the DAE is *locally unsolvable* at this point, and we call it a *singularity point*. See Example 4.12.

Example 4.12 [8] Consider

$$\begin{aligned} 0 &= f_1 = -x' + y \\ 0 &= f_2 = x + \cos(t) y. \end{aligned} \tag{4.6}$$

$$\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ d_j \end{array} \begin{array}{c} \begin{array}{cc} x & y \end{array} \\ \begin{bmatrix} 1^\bullet & 0 \\ \boxed{0} & 0^\bullet \end{bmatrix} \\ \begin{array}{cc} 1 & 0 \end{array} \end{array} \begin{array}{c} c_i \\ 0 \\ 0 \\ \mathrm{Val}(\Sigma) = 1 \end{array} \qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \end{array} \begin{array}{c} \begin{array}{cc} x & y \end{array} \\ \begin{bmatrix} -1 & 1 \\ 0 & \cos(t) \end{bmatrix} \end{array}$$

$$\det(\mathbf{J}) = -\cos(t)$$

Since $\det(\mathbf{J})$ is generically nonzero, (4.6) is structurally nonsingular. We can integrate this problem from $t = 0$ with any consistent initial value $(x(0), y(0)) = (x_0, y_0)$, and the problem is index-1 (both differentiation and structural indices) as long as $\det(\mathbf{J}) \neq 0$. However, $\mathbf{J}$ is singular at $t = t_k = (k + 1/2)\pi$, $k = 0, 1, \ldots$. Hence, we say the DAE has a singularity point at t_k.

4.2 Identifying structural analysis's failure

We give below a definition for the *true highest-order derivative* (HOD) of a variable x_j in a function u.

Definition 4.13 *The* true HOD *of x_j in u is*

$$\sigma(x_j, u) = \begin{cases} \textit{the highest order derivatives of } x_j \textit{ on which } u \text{ truly } \textit{depends; or} \\ -\infty \quad \textit{if } u \textit{ does not depend on any derivative of } x_j \textit{ (including } x_j). \end{cases}$$

By "truly" we mean that, if $r = \sigma(x_j, u) > -\infty$, then u is not a constant with respect to $x_j^{(r)}$. For example, $u = x' + \cos^2 x'' + \sin^2 x'' = x' + 1$ truly depends on x' but not x'', resulting in $\sigma(x, u) = 1$.

In practice, however, we usually find the *formal* HOD of x_j in u, denoted by $\widetilde{\sigma}(x_j, u)$, instead of the *true* HOD. By "formal" we mean the dependence of an expression (or function) on a derivative *without symbolic simplifications*. For example, $u = x' + \cos^2 x'' + \sin^2 x''$ formally depends on x'' and hence $\widetilde{\sigma}(x, u) = 2$, while $u = x' + 1$ and $\sigma(x, u) = 1$.

We denote also $\widetilde{\sigma}_{ij} = \widetilde{\sigma}(x_j, f_i)$ corresponding to σ_{ij}. The DAETS and DAESA codes implement [21, Algorithm 4.1 (Signature matrix)] for finding formal $\widetilde{\sigma}_{ij}$.

Since the formal dependence is also used in [21, §4], we can adopt the rules in [21, Lemma 4.1], which indicate how to propagate the formal HOD in an expression. The most useful rules are:

- if a variable v is a purely algebraic function of a set U of variables u, then

$$\widetilde{\sigma}\left(x_j, v\right) = \max_{u \in U} \widetilde{\sigma}\left(x_j, u\right), \tag{4.7}$$

and

- if $v = \mathrm{d}^p u / \mathrm{d}t^p$, where $p > 0$, then

$$\widetilde{\sigma}\left(x_j, v\right) = \widetilde{\sigma}\left(x_j, u\right) + p. \tag{4.8}$$

These rules are proved in [21], to which we refer for more details. We illustrate the rules in Example 4.14.

Example 4.14 Let $u = (x_1 x_2)' - x_1' x_2$. Applying (4.7) and (4.8), we derive the formal HOD of x_1 in u:

$$\begin{aligned}
\widetilde{\sigma}(x_1, u) &= \max\left\{\, \widetilde{\sigma}\left(x_1, (x_1 x_2)'\right), \widetilde{\sigma}\left(x_1, x_1' x_2\right) \right\} \\
&= \max\left\{\, \widetilde{\sigma}(x_1, x_1 x_2) + 1, \max\left\{\, \widetilde{\sigma}\left(x_1, x_1'\right), \widetilde{\sigma}(x_1, x_2) \right\} \right\} \\
&= \max\left\{\, \max\left\{\, \widetilde{\sigma}(x_1, x_1), \widetilde{\sigma}(x_1, x_2) \right\} + 1, \max\left\{\, 1, -\infty \right\} \right\} \\
&= \max\left\{\, \max\left\{\, 0, -\infty \right\} + 1, 1 \right\} \\
&= \max\left\{\, 0 + 1, 1 \right\} \\
&= 1.
\end{aligned}$$

Similarly $\widetilde{\sigma}(x_2, u) = 1$. Simplifying $u = (x_1 x_2)' - x_1' x_2$ results in $u = x_1 x_2'$. Hence, the true HOD of x_1 in u is $\sigma(x_1, u) = 0$, and that of x_2 in u is $\sigma(x_2, u) = 1$.

When such a *hidden symbolic cancellation* occurs, $\widetilde{\sigma}\left(x_j, u\right)$ can overestimate the true $\sigma\left(x_j, u\right)$. If u is an equation f_i, then the formal HOD $\widetilde{\sigma}\left(x_j, f_i\right)$ may not be the true σ_{ij}. We write $\widetilde{\sigma}_{ij} = \widetilde{\sigma}\left(x_j, f_i\right)$ corresponding to $\sigma_{ij} = \sigma\left(x_j, f_i\right)$. We call the matrix $\widetilde{\Sigma} = (\widetilde{\sigma}_{ij})$ the "formal" signature matrix. Also, let $\widetilde{\mathbf{c}}, \widetilde{\mathbf{d}}$ be any valid offsets for $\widetilde{\Sigma}$, and let $\widetilde{\mathbf{J}}$ be the resulting Jacobian defined by (3.3) with $\widetilde{\Sigma}$ and $\widetilde{\mathbf{c}}, \widetilde{\mathbf{d}}$.

If $\widetilde{\sigma}_{ij} > \sigma_{ij}$, then f_i does not depend truly on $x_j^{(\widetilde{\sigma}_{ij})}$. That is, f_i is a constant with respect to $x_j^{(\widetilde{\sigma}_{ij})}$. Then $\widetilde{\mathbf{J}}_{ij} = 0$, and (i, j) is a structural zero in $\widetilde{\mathbf{J}}$. Due to such cancellations, $\widetilde{\mathbf{J}}$ has more structural zeros than $\mathbf{J}$ does, and hence $\widetilde{\mathbf{J}}$ is more likely to be structurally singular. It is also possible that the DAE itself is structurally ill posed.

Since $\widetilde{\sigma}_{ij} \geq \sigma_{ij}$ for all $i, j = 1:n$, we can write $\widetilde{\Sigma} \geq \Sigma$ meaning "elementwise greater or equal".

We define the *essential sparsity pattern* S_{ess} of Σ to be the union of the HVTs of Σ. That is, the set of all (i, j) positions that lie on any HVT. We give two theorems below, which are Theorems 5.1 and 5.2 in [20]. In the following, we use the term "offset vector" to refer to the vector $(\mathbf{c}, \mathbf{d}) = (c_1, \ldots, c_n, d_1, \ldots, d_n)$.

Theorem 4.15 *Suppose that a valid offset vector* $(\mathbf{c},\mathbf{d})$ *for* Σ *gives a nonsingular* $\mathbf{J}$ *as defined by (3.3) at some consistent point. Then every valid offset vector gives a nonsingular* $\overline{\mathbf{J}}$ *(not necessarily the same as* $\mathbf{J}$*) at this point. All resulting* $\overline{\mathbf{J}}$*, including* $\mathbf{J}$*, are equal on* S_{ess}*, and all have the same determinant* $\det(\overline{\mathbf{J}}) = \det(\mathbf{J})$.

By "equal on S_{ess}" we mean $\overline{\mathbf{J}}_{ij} = \mathbf{J}_{ij}$ for all $(i,j) \in S_{\text{ess}}$.

Theorem 4.16 *Assume that* $\mathbf{J}$*, resulting from* Σ *and a valid offset vector* $(\mathbf{c},\mathbf{d})$*, is generically nonsingular. Let* $(\widetilde{\mathbf{c}}, \widetilde{\mathbf{d}})$ *be a valid offset vector for the formal signature matrix* $\widetilde{\Sigma}$*, and let* $\widetilde{\mathbf{J}}$ *be the Jacobian resulting from* $\widetilde{\Sigma}$ *and* $(\widetilde{\mathbf{c}}, \widetilde{\mathbf{d}})$*. In exact arithmetic, one of the following two alternatives must occur:*

(i) $Val(\widetilde{\Sigma}) = Val(\Sigma)$*. Then every HVT of* Σ *is a HVT of* $\widetilde{\Sigma}$*, and* $\widetilde{\mathbf{c}},\widetilde{\mathbf{d}}$ *are valid offsets for* Σ*. Consequently,* $\widetilde{\mathbf{J}}$ *is also generically nonsingular.*

(ii) $Val(\widetilde{\Sigma}) > Val(\Sigma)$*. Then* $\widetilde{\mathbf{J}}$ *is structurally singular.*

Theorem 4.16 shows that $\widetilde{\mathbf{J}}$, resulting from $\widetilde{\Sigma} \geq \Sigma$ and a valid offset vector $(\widetilde{\mathbf{c}}, \widetilde{\mathbf{d}})$, is *either*

(1) nonsingular, and SA is using valid, but not necessarily canonical, offsets for the true Σ; *or*

(2) structurally singular, and SA fails due to symbolic cancellations, in a way that may be detected.

In the latter case, this failure may be avoided by performing symbolic simplification on some or all of the f_i's. However, "no clever symbolic manipulation can overcome the hidden cancellation problem, because the task of determining whether some expression is exactly zero is known to be undecidable in any algebra closed under the basic arithmetic operations together with the exponential function" [20].

Example 4.17 Consider

$$\begin{aligned} 0 &= f_1 = (xy)' - x'y - xy' + 2x + y - 3 \\ 0 &= f_2 = x + y - 2. \end{aligned} \tag{4.9}$$

$$\widetilde{\Sigma} = \begin{array}{c|cc|c} & x & y & c_i \\ f_1 & 1^\bullet & 1 & 0 \\ f_2 & 0 & 0^\bullet & 1 \\ d_j & 1 & 1 & \end{array} \quad \text{Val}(\widetilde{\Sigma}) = 1 \qquad \widetilde{\mathbf{J}} = \begin{array}{c|cc} & x & y \\ f_1 & 0 & 0 \\ f_2 & 1 & 1 \end{array} \quad \det(\widetilde{\mathbf{J}}) = 0$$

Here, the signature matrix and Jacobian are the formal ones. Since $\det(\widetilde{\mathbf{J}}) = 0$, SA fails. Simplifying f_1 to $f_1 = 2x + y - 3$ reveals that (4.9) is a simple linear algebraic system:

$$\begin{aligned} 0 &= f_1 = 2x + y - 3 \\ 0 &= f_2 = x + y - 2. \end{aligned}$$

$$\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ d_j \end{array}\begin{array}{c} \begin{array}{ccc} x & y & c_i \end{array} \\ \left[\begin{array}{cc} 0^\bullet & 0 \\ 0 & 0^\bullet \end{array}\right]\begin{array}{c} 0 \\ 0 \end{array} \\ \begin{array}{cc} 0 & 0 \end{array} \end{array} \ \mathrm{Val}(\Sigma) = 0 \qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \end{array}\begin{array}{c} \begin{array}{cc} x & y \end{array} \\ \left[\begin{array}{cc} 2 & 1 \\ 1 & 1 \end{array}\right] \end{array} \quad \det(\mathbf{J}) = 1$$

Another kind of SA's failure occurs when $\mathbf{J}$ is not structurally singular, but is identically singular. Examples 4.18 and 4.19 illustrate this case.

Example 4.18 Consider the coupled DAE from [33] [1] :

$$\begin{aligned} 0 = f_1 &= -x_1' + x_3 + b_1(t) \\ 0 = f_2 &= -x_2' + x_4 + b_2(t) \\ 0 = f_3 &= \quad x_2 + x_3 + x_4 + c_1(t) \\ 0 = f_4 &= -x_1 + x_3 + x_4 + c_2(t). \end{aligned} \tag{4.10}$$

$$\Sigma = \begin{array}{c|cccc|c} & x_1 & x_2 & x_3 & x_4 & c_i \\ \hline f_1 & 1^\bullet & - & 0 & - & 0 \\ f_2 & - & 1^\bullet & - & 0 & 0 \\ f_3 & - & \boxed{0} & 0^\bullet & 0 & 0 \\ f_4 & \boxed{0} & - & 0 & 0^\bullet & 0 \\ \hline d_j & 1 & 1 & 0 & 0 & \mathrm{Val}(\Sigma) = 2 \end{array} \qquad \mathbf{J} = \begin{array}{c|cccc} & x_1 & x_2 & x_3 & x_4 \\ \hline f_1 & -1 & 0 & 1 & 0 \\ f_2 & 0 & -1 & 0 & 1 \\ f_3 & 0 & 0 & 1 & 1 \\ f_4 & 0 & 0 & 1 & 1 \end{array} \quad \det(\mathbf{J}) = 0$$

This DAE is of d-index 3, while SA finds structural index 1 and singular $\mathbf{J}$. Hence SA fails.

Example 4.19 In the following DAE, SA reports the correct d-index 2 but still fails.

$$\begin{aligned} 0 = f_1 &= -x_1' - x_3' + x_1 + x_2 + g_1(t) \\ 0 = f_2 &= -x_2' - x_3' + x_1 + x_2 + x_3 + x_4 + g_2(t) \\ 0 = f_3 &= x_2 + x_3 + g_3(t) \\ 0 = f_4 &= x_1 - x_4 + g_4(t) \end{aligned} \tag{4.11}$$

[1] We consider this DAE with parameters $\beta = \varepsilon = 1$, $\alpha_1 = \alpha_2 = \delta = 1$, and $\gamma = -1$. In [33] superscripts are used as indices, while we use subscripts instead. We also change the (original) equation names g_1, g_2 to f_3, f_4, and the (original) variable names y_1, y_2 to x_3, x_4.

$$
\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ d_j \end{array}
\begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\
\left[\begin{array}{cccc} 1^\bullet & \boxed{0} & 1 & - \\ \boxed{0} & 1^\bullet & 1 & 0 \\ - & 0 & 0^\bullet & - \\ \boxed{0} & - & - & 0^\bullet \end{array}\right] \\
\begin{array}{cccc} 1 & 1 & 1 & 0 \end{array} \end{array}
\begin{array}{c} c_i \\ 0 \\ 0 \\ 1 \\ 0 \\ \mathrm{Val}(\Sigma) = 2 \end{array}
\qquad
\mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{array}
\begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\
\left[\begin{array}{cccc} -1 & 0 & -1 & 0 \\ 0 & -1 & -1 & 1 \\ 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \\
\det(\mathbf{J}) = 0 \end{array}
$$

Using the solution scheme derived from the SA result, we would try to solve at stage $k = 0$ the linear system $0 = f_1, f_2, f_3', f_4$ for x_1', x_2', x_3', x_4, where the matrix is $\mathbf{J}$. Since it is singular, the solution scheme fails in solving (4.11) at this stage; see Table 4.1.

stage k	solve	for	using	comment
-1	$0 = f_3$	x_2, x_3	–	initialize x_1
0	$0 = f_1, f_2, f_3', f_4$	x_1', x_2', x_3', x_4	x_1, x_2, x_3	$\mathbf{J}$ is singular; solution scheme fails

Table 4.1: Solution scheme for (4.11)

Now we replace f_2 by $\overline{f}_2 = f_2 + f_3'$ to obtain

$$
\begin{aligned}
0 = f_1 &= -x_1' - x_3' + x_1 + x_2 + g_1(t) \\
0 = f_2 + f_3' = \overline{f}_2 &= x_1 + x_2 + x_3 + x_4 + g_2(t) + g_3'(t) \\
0 = f_3 &= x_2 + x_3 + g_3(t) \\
0 = f_4 &= x_1 - x_4 + g_4(t).
\end{aligned}
\tag{4.12}
$$

$$
\overline{\Sigma} = \begin{array}{c} \\ f_1 \\ \overline{f}_2 \\ f_3 \\ f_4 \\ d_j \end{array}
\begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\
\left[\begin{array}{cccc} 1 & \boxed{0} & 1^\bullet & - \\ 0^\bullet & 0 & 0 & 0 \\ - & 0^\bullet & 0 & - \\ 0 & - & - & 0^\bullet \end{array}\right] \\
\begin{array}{cccc} 1 & 1 & 1 & 1 \end{array} \end{array}
\begin{array}{c} c_i \\ 0 \\ 1 \\ 1 \\ 1 \\ \mathrm{Val}(\overline{\Sigma}) = 1 \end{array}
\qquad
\overline{\mathbf{J}} = \begin{array}{c} \\ f_1 \\ \overline{f}_2 \\ f_3 \\ f_4 \end{array}
\begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\
\left[\begin{array}{cccc} -1 & 0 & -1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & -1 \end{array}\right] \\
\det(\overline{\mathbf{J}}) = 2 \end{array}
$$

The solution scheme succeeds; see Table 4.2. The resulting DAE (4.12) is of structural index $\nu_S = 1$, which equals the differentiation index.

At stage $k = 0$, we solve $0 = f_1, \overline{f}_2', f_3', f_4'$ for x_1', x_2', x_3', x_4' using x_1, x_2, x_3, x_4. Since $\overline{f}_2' = f_2' + f_3''$, we need f_3'' to find these first-order derivatives. Therefore, the original DAE (4.11) is of differentiation index 2.

Note that by setting $f_2 = \overline{f}_2 - f_3'$ we can immediately recover the original system. It can be easily verified that a vector function

$$\mathbf{x}(t) = (x_1(t), x_2(t), x_3(t), x_4(t))^T$$

that satisfies (4.12) also satisfies (4.11), and vice versa. We explain in §5 how this conversion makes SA succeed.

stage k	solve	for	using	comment
-1	$0 = \overline{f}_2, f_3, f_4$	x_1, x_2, x_3, x_4	–	–
0	$0 = f_1, \overline{f}_2', f_3', f_4'$	x_1', x_2', x_3', x_4'	x_1, x_2, x_3, x_4	$\mathbf{J}$ is nonsingular; solution scheme succeeds

Table 4.2: Solution scheme for (4.12)

In Examples 4.18 and 4.19, $\mathbf{J}$ is not structurally singular, but is still identically singular. No symbolic cancellation occurs in the equations therein. Therefore, this kind of failure is more difficult to detect and remedy, and we wish to find techniques to deal with such failures.

We call our techniques *conversion methods*, and describe them in the upcoming chapters. We wish to convert a structurally singular DAE into a structurally nonsingular problem, provided some conditions are satisfied and allow us to perform a conversion step. The original DAE and the converted one are *equivalent* in the sense that they have (at least locally) the same solution set. We shall also elaborate on this equivalence issue.

Chapter 5

The linear combination method

In this chapter we introduce the linear combination method, or the LC method for short. We present in §5.1 some preliminary lemmas. Then we describe in §5.2 how to perform a conversion step. In §5.3 we give definitions and results about equivalence of DAEs and address how equivalence is related to the LC method.

For simplicity, throughout this report, we consider only the second type of SA's failures described in §4.2: "singular" means identically singular but not structurally singular. Based on this assumption, symbolic cancellations are not considered an issue that makes the Σ-method fail.

5.1 Preliminary lemmas

Lemma 5.1 ***(Griewank's Lemma)*** *[21, Lemma 5.1] Let v be a function of t, x_j's and derivatives of them $(j = 1:n)$. Denote $v^{(p)} = d^p v/dt^p$, where $p > 0$. If $\sigma\left(x_j, v\right) \le q$, then*

$$\frac{\partial v}{\partial x_j^{(q)}} = \frac{\partial v'}{\partial x_j^{(q+1)}}.$$

Hence

$$\frac{\partial v}{\partial x_j^{(q)}} = \frac{\partial v'}{\partial x_j^{(q+1)}} \cdots = \frac{\partial v^{(p)}}{\partial x_j^{(q+p)}}. \tag{5.1}$$

Lemma 5.2 *Let Σ and $\overline{\Sigma}$ be $n \times n$ signature matrices. Assume Val(Σ) is finite,* $\mathbf{c}, \mathbf{d}$ *are valid offsets for Σ, and $\overline{\sigma}_{ij} \le d_j - c_i$ for all $i, j = 1:n$. If a HVT in $\overline{\Sigma}$ contains a position (i, j) where $\overline{\sigma}_{ij} < d_j - c_i$, then Val$(\overline{\Sigma}) <$ Val(Σ).*

Proof. Let T be a HVT in $\overline{\Sigma}$. Then

$$\mathrm{Val}(\overline{\Sigma}) = \sum_{(i,j)\in T} \overline{\sigma}_{ij} < \sum_{j=1}^{n} d_j - \sum_{i=1}^{n} c_i = \mathrm{Val}(\Sigma). \quad \square \tag{5.2}$$

Corollary 5.3 *For a row index l, let*

$$\begin{cases} \overline{\sigma}_{ij} = \sigma_{ij} & \text{for all } i \neq l \text{ and all } j\text{, and} \\ \overline{\sigma}_{lj} < d_j - c_l & \text{for all } j. \end{cases}$$

Then $Val(\overline{\Sigma}) < Val(\Sigma)$.

Proof. Since $\overline{\sigma}_{lj} < d_j - c_l$ for all j, the intersection of a HVT in $\overline{\Sigma}$ with positions in row l is a position (l,r) with $\overline{\sigma}_{lr} < d_r - c_l$. By Lemma 5.2, $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$. □

This lemma shows that, if we replace a row l in Σ with a row with entries less than $d_j - c_l$ for each column j, then the value of this signature matrix decreases.

5.2 Conversion step

Given a SWP DAE of the form (1.1), assume that we apply the Σ-method and obtain a singular system Jacobian $\mathbf{J}$. We seek a reformulation of this DAE so that the system Jacobian $\overline{\mathbf{J}}$ of the new DAE may be generically nonsingular. We denote by Σ and $\overline{\Sigma}$ the signature matrices of the original DAE and this new DAE, respectively. Denote by $\mathbf{c}, \mathbf{d}$ the valid offsets for Σ.

We describe below how to perform a conversion step using a linear combination (LC) of equations. We call this conversion technique the *LC conversion method*, or simply the *LC method*. The main result from this conversion is that, under certain conditions, we can obtain an equation in a row, say l, such that x_j occurs in this row of order $< d_j - c_l$ for all j. Hence by Corollary 5.3, $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$.

We assume $n \geq 2$. Let u be a nonzero vector function from the null space of $\mathbf{J}^T$. Here, $\mathbf{J}$ and u are considered as functions of t, x_j's and appropriate derivatives of them.

Denote by $I(u)$ the set of indices for which the ith component of u is not identically zero

$$I(u) = \{\, i \mid u_i \neq 0 \,\}, \tag{5.3}$$

and let

$$\theta(u) = \min_{i \in I(u)} c_i. \tag{5.4}$$

Since u is nonzero and $\mathbf{J}$ is identically singular, $I(u)$ has at least two elements. Otherwise $\mathbf{J}$ has a row of identical zeros and is structurally singular.

Remark 5.4 We consider u in its simplest form in the sense that its elements do not have a common factor comprising t, x_j's, or/and derivatives of them. For instance, in Example 4.18, we do not use $u = (0,0,x_1', -x_1')^T$ though $\mathbf{J}^T u = \mathbf{0}$, but use $u = (0,0,1,-1)^T$.

Also, we do not consider u with any fractions. For example, we use $u = (0,0,x_1', x_1 x_2)^T$ instead of $(0,0,x_1^{-1}, x_2(x_1')^{-1})^T$.

The *sufficient condition* for applying the LC method is the following: for a nonzero $u \in \ker(\mathbf{J}^T)$,

$$\boxed{\sigma\left(x_j, u\right) < d_j - \theta(u) \qquad \text{for all } j = 1:n} \tag{5.5}$$

If this condition is satisfied, then we can perform a conversion step. We explain this "sufficiency" in Remark 5.8.

Denote by $L(u) \subseteq I(u)$ the set of indices l such that the lth component of u is not an identical zero and $c_l = \theta(u) = \min_{i \in I(u)} c_i$:

$$L(u) = \left\{\, l \in I(u) \mid c_l = \theta(u) \,\right\}. \tag{5.6}$$

From (5.4), there exists at least one $l \in I(u)$ such that $c_l = \theta(u)$, so $L(u) \neq \emptyset$.

We choose an $l \in L(u)$ and replace f_l by

$$\overline{f}_l = \sum_{i \in I(u)} u_i f_i^{(c_i - \theta(u))}. \tag{5.7}$$

We refer to (5.7) as a *conversion step* using the LC method and to the resulting DAE as a *converted* DAE. Critical for the success of the LC method is the following lemma.

Theorem 5.5 *For a SWP DAE with identically singular* $\mathbf{J}$*, let* u *be a nonzero* n*-vector such that* $\mathbf{J}^T u = \mathbf{0}$*. If*

$$\sigma\left(x_j, u\right) < d_j - \theta(u) \qquad \textit{for all } j = 1:n$$

and we replace f_l *by* $\overline{f}_l$ *in (5.7), then the converted DAE has* $\overline{\Sigma}$ *with* $Val(\overline{\Sigma}) < Val(\Sigma)$.

First, we illustrate with an example how to perform a conversion step, and then we prove this lemma. Since u is fixed during a conversion step, for brevity we write $I(u)$, $\theta(u)$, and $L(u)$ as I, θ, and L, respectively[1].

Example 5.6 Consider

$$\begin{aligned}
0 &= f_1 = -x_1' + x_3 \\
0 &= f_2 = -x_2' + x_4 \\
0 &= f_3 = F(x_1, x_2) \\
0 &= f_4 = x_3 F_{x_1}(x_1, x_2) + x_4 F_{x_2}(x_1, x_2) + G(x_1, x_2).
\end{aligned} \tag{5.8}$$

Here $F_{x_1}(x_1, x_2) = \partial F(x_1, x_2)/\partial x_1$, and similarly we write $F_{x_2}(x_1, x_2)$, $G_{x_1}(x_1, x_2)$, and $G_{x_2}(x_1, x_2)$.

[1]This set L is not to be confused with the constant L in the pendulum-related DAEs.

$$\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ d_j \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \left[\begin{array}{cccc} 1^\bullet & - & 0 & - \\ - & 1 & - & 0^\bullet \\ 0 & 0^\bullet & - & - \\ \boxed{0} & \boxed{0} & 0^\bullet & 0 \end{array}\right] \\ \begin{array}{cccc} 1 & 1 & 0 & 0 \end{array} \end{array} \begin{array}{c} c_i \\ 0 \\ 0 \\ 1 \\ 0 \\ \mathrm{Val}(\Sigma) = 1 \end{array} \qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \left[\begin{array}{cccc} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ F_{x_1} & F_{x_2} & 0 & 0 \\ 0 & 0 & F_{x_1} & F_{x_2} \end{array}\right] \\ \det(\mathbf{J}) = 0 \end{array}$$

Because of singular $\mathbf{J}$, the SA fails. It reports structural index 2, but the differentiation index is 3. If we take $u = \left(F_{x_1}, F_{x_2}, 1, -1\right)^T$, then $\mathbf{J}^T u = \mathbf{0}$.

We illustrate (5.3–5.6):

$$\begin{aligned} &I = \left\{\, i \mid u_i \neq 0 \,\right\} = \left\{\, 1,2,3,4 \,\right\}, \\ &\theta = \min_{i \in I} c_i = 0, \\ &L = \left\{\, l \in I \mid c_l = \theta = 0 \,\right\} = \left\{\, 1,2,4 \,\right\}. \end{aligned}$$

Then we check if condition (5.5) holds:

$$\begin{aligned} \sigma(x_1, u) &= \quad 0 \;\; < 1 = d_1 - \theta, \\ \sigma(x_2, u) &= \quad 0 \;\; < 1 = d_2 - \theta, \\ \sigma(x_3, u) &= -\infty < 0 = d_3 - \theta, \quad \text{and} \\ \sigma(x_4, u) &= -\infty < 0 = d_4 - \theta. \end{aligned}$$

Hence $\sigma\left(x_j, u\right) < d_j - \theta$ for all j.

Using (5.7) gives

$$\begin{aligned} \overline{f} &= \sum_{i \in I} u_i f_i^{(c_i - \theta)} = \sum_{i \in I} u_i f_i^{(c_i)} \\ &= F_{x_1} f_1 + F_{x_2} f_2 + f_3' - f_4 \\ &= F_{x_1}(-x_1' + x_3) + F_{x_2}(-x_2' + x_4) + (F(x_1, x_2))' - (x_3 F_{x_1} + x_4 F_{x_2} + G) \\ &= -x_1' F_{x_1} + x_3 F_{x_1} - x_2' F_{x_2} + x_4 F_{x_2} + x_1' F_{x_1} + x_2' F_{x_2} - x_3 F_{x_1} - x_4 F_{x_2} - G \\ &= -G. \end{aligned}$$

Now, with $\theta = 0$,

$$\begin{aligned} \sigma\left(x_1, \overline{f}\right) &= \quad 0 \;\; < 1 = d_1 - \theta, \\ \sigma\left(x_2, \overline{f}\right) &= \quad 0 \;\; < 1 = d_2 - \theta, \\ \sigma\left(x_3, \overline{f}\right) &= -\infty < 0 = d_3 - \theta, \quad \text{and} \\ \sigma\left(x_4, \overline{f}\right) &= -\infty < 0 = d_4 - \theta. \end{aligned}$$

That is, $\sigma\left(x_j, \overline{f}\right) < d_j - \theta$ for all j.

For each $l \in L = \left\{1,2,4\right\}$, assuming $u_l \neq 0$, we can replace f_l by $\overline{f}_l = \overline{f}$. We show in the following the three possible converted DAEs, each with $\mathrm{Val}(\overline{\Sigma}) = 0$ and generically nonsingular $\overline{\mathbf{J}}$.

- $l = 1$:

$$\begin{aligned}
0 &= \overline{f}_1 = -G(x_1,x_2) \\
0 &= f_2 = -x_2' + x_4 \\
0 &= f_3 = F(x_1,x_2) \\
0 &= f_4 = x_3 F_{x_1}(x_1,x_2) + x_4 F_{x_2}(x_1,x_2) + G(x_1,x_2)
\end{aligned} \tag{5.9}$$

$$\overline{\Sigma} = \begin{array}{c|cccc|c} & x_1 & x_2 & x_3 & x_4 & c_i \\ \hline \overline{f}_1 & 0^\bullet & 0 & - & - & 1 \\ f_2 & - & 1 & - & 0^\bullet & 0 \\ f_3 & 0 & 0^\bullet & - & - & 1 \\ f_4 & \boxed{0} & \boxed{0} & 0^\bullet & 0 & 0 \\ \hline d_j & 1 & 1 & 0 & 0 & \end{array} \quad \mathrm{Val}(\overline{\Sigma}) = 0$$

$$\overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \\ f_3 \\ f_4 \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \begin{bmatrix} -G_{x_1} & -G_{x_2} & 0 & 0 \\ 0 & -1 & 0 & 1 \\ F_{x_1} & F_{x_2} & 0 & 0 \\ 0 & 0 & F_{x_1} & F_{x_2} \end{bmatrix} \end{array} \quad \det(\overline{\mathbf{J}}) = F_{x_1}(F_{x_1}G_{x_2} - F_{x_2}G_{x_1})$$

When $u_1 = F_{x_1} \neq 0$ and $F_{x_1}G_{x_2} \neq F_{x_2}G_{x_1}$, the determinant is nonzero and the SA succeeds.

- $l = 2$:

$$\begin{aligned}
0 &= f_1 = -x_1' + x_3 \\
0 &= \overline{f}_2 = -G(x_1,x_2) \\
0 &= f_3 = F(x_1,x_2) \\
0 &= f_4 = x_3 F_{x_1}(x_1,x_2) + x_4 F_{x_2}(x_1,x_2) + G(x_1,x_2)
\end{aligned} \tag{5.10}$$

$$\overline{\Sigma} = \begin{array}{c|cccc|c} & x_1 & x_2 & x_3 & x_4 & c_i \\ \hline f_1 & 1 & - & 0^\bullet & - & 0 \\ \overline{f}_2 & 0^\bullet & 0 & - & - & 1 \\ f_3 & 0 & 0^\bullet & - & - & 1 \\ f_4 & \boxed{0} & \boxed{0} & 0 & 0^\bullet & 0 \\ \hline d_j & 1 & 1 & 0 & 0 & \end{array} \quad \mathrm{Val}(\overline{\Sigma}) = 0$$

$$\overline{\mathbf{J}} = \begin{array}{c} \\ f_1 \\ \overline{f}_2 \\ f_3 \\ f_4 \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \begin{bmatrix} -1 & 0 & 1 & 0 \\ -G_{x_1} & -G_{x_2} & 0 & 0 \\ F_{x_1} & F_{x_2} & 0 & 0 \\ 0 & 0 & F_{x_1} & F_{x_2} \end{bmatrix} \end{array} \quad \det(\overline{\mathbf{J}}) = F_{x_2}(F_{x_1}G_{x_2} - F_{x_2}G_{x_1})$$

Similarly, the SA succeeds when $u_2 = F_{x_2} \neq 0$ and $F_{x_1}G_{x_2} \neq F_{x_2}G_{x_1}$.

- $l = 4$:

$$\begin{aligned} 0 &= f_1 = -x_1' + x_3 \\ 0 &= f_2 = -x_2' + x_4 \\ 0 &= f_3 = F(x_1, x_2) \\ 0 &= \overline{f}_4 = -G(x_1, x_2) \end{aligned} \tag{5.11}$$

$$\overline{\Sigma} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ \overline{f}_4 \\ d_j \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \left[\begin{array}{cccc} 1 & - & 0^\bullet & - \\ - & 1 & - & 0^\bullet \\ 0 & 0^\bullet & - & - \\ 0^\bullet & 0 & - & - \end{array}\right] \\ \begin{array}{cccc} 1 & 1 & 0 & 0 \end{array} \end{array} \begin{array}{c} c_i \\ 0 \\ 0 \\ 1 \\ 1 \\ \mathrm{Val}(\overline{\Sigma}) = 0 \end{array} \qquad \overline{\mathbf{J}} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ \overline{f}_4 \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \end{array} \\ \left[\begin{array}{cccc} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ F_{x_1} & F_{x_2} & 0 & 0 \\ -G_{x_1} & -G_{x_2} & 0 & 0 \end{array}\right] \\ \det(\overline{\mathbf{J}}) = -F_{x_1}G_{x_2} + F_{x_2}G_{x_1} \end{array}$$

In this case, SA's success requires only $F_{x_1}G_{x_2} \neq F_{x_2}G_{x_1}$.

Using the LC method, we obtain three converted DAEs from (5.8): (5.9), (5.10), and (5.11). However, it is not guaranteed that all converted DAEs and the original DAE have exactly the same solution sets. We will cover this equivalence issue in §5.3.

Now we prove Lemma 5.5.

Proof. We show first that

$$\overline{\sigma}_{lj} = \sigma\left(x_j, \overline{f}_l\right) < d_j - c_l \qquad \text{for all } j = 1:n.$$

Since $\theta = \min_{i \in I} c_i$, $c_i - \theta \geq 0$ for $i \in I$. By (3.2), $\sigma\left(x_j, f_i\right) = \sigma_{ij} \leq d_j - c_i$. Applying Griewank's Lemma (5.1) to (3.3), with $q = c_i - \theta$, gives

$$\mathbf{J}_{ij} = \frac{\partial f_i}{\partial x_j^{(d_j - c_i)}} = \frac{\partial f_i^{(c_i-\theta)}}{\partial x_j^{(d_j - c_i + c_i - \theta)}} = \frac{\partial f_i^{(c_i-\theta)}}{\partial x_j^{(d_j-\theta)}} \qquad \text{for } i \in I. \tag{5.12}$$

Then for all $j = 1 : n$,

$$
\begin{aligned}
0 = (\mathbf{J}^T u)_j &= \sum_{i=1}^{n} u_i (\mathbf{J}^T)_{ji} = \sum_{i \in I}^{n} u_i \mathbf{J}_{ij} && \text{using } \mathbf{J}^T u = \mathbf{0} \\
&= \sum_{i \in I} u_i \frac{\partial f_i}{\partial x_j^{(d_j - c_i)}} = \sum_{i \in I} u_i \frac{\partial f_i^{(c_i - \theta)}}{\partial x_j^{(d_j - \theta)}} && \text{using (5.12)} \\
&= \frac{\partial \left(\sum_{i \in I} u_i f_i^{(c_i - \theta)} \right)}{\partial x_j^{(d_j - \theta)}} && \text{using } \sigma\left(x_j, u\right) < d_j - \theta \text{ for all } j \qquad (5.13) \\
&= \frac{\partial \overline{f}_l}{\partial x_j^{(d_j - \theta)}} && \text{using (5.7).}
\end{aligned}
$$

This shows that $\overline{f}_l$ does not truly depend on $x_j^{(d_j - \theta)}$, that is,

$$\overline{\sigma}_{lj} = \sigma\left(x_j, \overline{f}_l\right) < d_j - \theta = d_j - c_l \qquad \text{for all } j = 1 : n.$$

By Corollary 5.3, $\text{Val}(\overline{\Sigma}) < \text{Val}(\Sigma)$. □

Remark 5.7 We name this method "LC" because of the following considerations. The vector u from the null space of $\mathbf{J}^T$ that satisfies (5.5) does not comprise the leading derivatives $x_j^{(d_j - \theta)}$ for all j in equation $f_i^{(c_i - \theta)}$ for all $i \in I$. We consider here each u_i as a "constant" in

$$\overline{f}_l = \sum_{i \in I} u_i f_i^{(c_i - \theta)},$$

and $\overline{f}_l$ as a "linear combination" of equations $f_i^{(c_i - \theta)}$.

Remark 5.8 If $\sigma\left(x_j, u\right) = d_j - \theta$ for some j, then $\text{Val}(\overline{\Sigma})$ is not guaranteed $< \text{Val}(\Sigma)$. In this case, we cannot swap the sum and the differentiation operator in (5.13). Therefore, we cannot prove $\partial \overline{f}_l / \partial x_j^{(d_j - \theta)} = 0$. Then, in $\overline{\Sigma}$, it can happen that

$$\overline{\sigma}_{lj} = \sigma\left(x_j, \overline{f}_l\right) = d_j - \theta = d_j - c_l \qquad \text{for some } j,$$

giving $\leq$ instead of strictly $<$ in (5.2).

However, if $\sigma\left(x_j, u\right) < d_j - \theta$ holds for j from a particular set, we can still achieve $\text{Val}(\overline{\Sigma}) < \text{Val}(\Sigma)$. We leave this investigation for future work, consider the condition (5.5) sufficient for now, and require it to be satisfied for the LC method.

If u is a constant vector, then $\sigma\left(x_j, u\right) = -\infty$ for every x_j, and the condition (5.5) is automatically satisfied. In this case we do not need to check it. We illustrate this in the next example.

Example 5.9 Consider

$$
\begin{aligned}
0 &= f_1 = x_1 + tx_2 + t^2 x_3 + g_1(t)\\
0 &= f_2 = x_1' + tx_2' + t^2 x_3' + g_2(t)\\
0 &= f_3 = x_1'' + tx_2'' + 2t^2 x_3'' + g_3(t).
\end{aligned}
$$

$$
\Sigma = \begin{array}{c|ccc|c} & x_1 & x_2 & x_3 & c_i \\ \hline f_1 & 0^\bullet & 0 & 0 & 2\\ f_2 & 1 & 1^\bullet & 1 & 1\\ f_3 & 2 & 2 & 2^\bullet & 0\\ \hline d_j & 2 & 2 & 2 & \end{array} \quad \mathrm{Val}(\Sigma) = 3
\qquad
\mathbf{J} = \begin{array}{c|ccc|} & x_1 & x_2 & x_3 \\ \hline f_1 & 1 & t & t^2\\ f_2 & 1 & t & t^2\\ f_3 & 1 & t & 2t^2 \end{array} \quad \det(\mathbf{J}) = 0
$$

For $u = (-1, 1, 0)^T$, $\mathbf{J}^T u = 0$. Using (5.3–5.6) gives

$$
I = \{1, 2\}, \quad \theta = c_2 = 1, \quad \text{and} \quad L = \{2\}.
$$

Since u is a constant vector, condition (5.5) is satisfied. We replace f_2 by

$$
\begin{aligned}
\overline{f}_2 &= u_1 f_1^{(2-1)} + u_2 f_2^{(1-1)}\\
&= -f_1' + f_2\\
&= -\big(x_1 + tx_2 + t^2 x_3 - g_1\big)' + (x_1' + tx_2' + t^2 x_3' + g_2)\\
&= -x_2 - 2tx_3 - g_1' + g_2.
\end{aligned}
$$

The converted DAE is

$$
\begin{aligned}
0 &= f_1 = x_1 + tx_2 + t^2 x_3 + g_1\\
0 &= \overline{f}_2 = -x_2 - 2tx_3 - g_1' + g_2\\
0 &= f_3 = x_1'' + tx_2'' + 2t^2 x_3'' + g_3.
\end{aligned}
$$

$$
\overline{\Sigma} = \begin{array}{c|ccc|c} & x_1 & x_2 & x_3 & c_i \\ \hline f_1 & 0^\bullet & 0 & 0 & 2\\ \overline{f}_2 & - & 0^\bullet & 0 & 2\\ f_3 & 2 & 2 & 2^\bullet & 0\\ \hline d_j & 2 & 2 & 2 & \end{array} \quad \mathrm{Val}(\overline{\Sigma}) = 2
\qquad
\overline{\mathbf{J}} = \begin{array}{c|ccc|} & x_1 & x_2 & x_3 \\ \hline f_1 & 1 & t & t^2\\ \overline{f}_2 & 0 & -1 & -2t\\ f_3 & 1 & t & 2t^2 \end{array} \quad \det(\mathbf{J}) = -t^2
$$

If $t > \sqrt{\varepsilon}$ for a suitable ε depending on the machine precision, then $\mathbf{J}$ is computably nonsingular.

5.3 Equivalent DAEs

First, we give a definition for equivalent DAEs.

Definition 5.10 *Let $\mathcal{F}$ and $\overline{\mathcal{F}}$ denote two DAEs. They are* equivalent *(on some interval for t) if a solution of $\mathcal{F}$ is a solution to $\overline{\mathcal{F}}$ and vice versa.*

In the following context, we denote by $\mathcal{F}$ the original DAE with equations f_i, $i = 1 : n$, and singular Jacobian $\mathbf{J}$. After a conversion step using the LC method, we obtain a (converted) DAE, denoted by $\overline{\mathcal{F}}$, with equations $\overline{f}_i$, $i = 1 : n$, and Jacobian $\overline{\mathbf{J}}$, which may be singular.

Theorem 5.11 *After a conversion step using the LC method, DAEs $\mathcal{F}$ and $\overline{\mathcal{F}}$ are equivalent on some real time interval $\mathbb{I}$ for t, if $u_l \neq 0$ for all $t \in \mathbb{I}$.*

Proof. Let a solution of $\mathcal{F}$, over some interval $\mathbb{I} \subset \mathbb{R}$, be a vector-valued function

$$\mathbf{x}(t) = \big(x_1(t), \ldots, x_n(t)\big)^T$$

that satisfies (1.1) for all $t \in \mathbb{I}$.

We denote the vector used in the LC method by $u = (u_1, \ldots, u_n)$, where its ith component is of the form

$$u_i = u_i\left(t; x_1, x_1', \ldots, x_1^{(d_1-\theta-1)}; \ldots; x_n, x_n', \ldots, x_n^{(d_n-\theta-1)}\right).$$

If u is defined at $(t, \mathbf{x}(t))$ for all $t \in \mathbb{I}$, then

$$\overline{f}_l = \sum_{i \in I} u_i f_i^{(c_i-\theta)} \quad \text{and} \quad \overline{f}_i = f_i \text{ for } i \neq l$$

vanish at $(t, \mathbf{x}(t))$, and thus $\mathbf{x}(t)$ is a solution to $\overline{\mathcal{F}}$.

Conversely, assume that $\overline{\mathbf{x}}(t)$ is a solution of $\overline{\mathcal{F}}$ on $\mathbb{I}$. If u is defined at $(t, \overline{\mathbf{x}}(t))$ for all $t \in \mathbb{I}$ and $u_l \neq 0$, then

$$f_l = \frac{1}{u_l}\left(\overline{f}_l - \sum_{i \in I \setminus \{l\}} u_i \overline{f}_i^{(c_i-\theta)}\right) \quad \text{and} \quad f_i = \overline{f}_i \text{ for } i \neq l \tag{5.14}$$

vanish at $(t, \overline{\mathbf{x}}(t))$, and thus $\overline{\mathbf{x}}(t)$ is a solution to $\mathcal{F}$.

By Definition 5.10, $\mathcal{F}$ and $\overline{\mathcal{F}}$ are equivalent. □

Remark 5.12 We can see from (5.14) that, if we have a choice for l, it is desirable to choose it such that u_l is identically nonzero, e.g., a nonzero constant, $x_1^2 + 1$, or $2 + \cos^2 x_3$. In this case, $\mathcal{F}$ and $\overline{\mathcal{F}}$ are *always* equivalent—we do not need to check the *equivalence condition* $u_l \neq 0$ when we solve $\overline{\mathcal{F}}$.

Example 5.13 In Example 5.6, case $l = 1$ [resp. $l = 2$] requires $F_{x_1} \neq 0$ [resp. $F_{x_2} \neq 0$] to recover the original DAE (5.8) from (5.9) [resp. from (5.10)]. However, for case $l = 4$, $u_4 = 1$ is a nonzero constant for any t. Therefore this choice is more desirable than the other two.

Below we define an ill-posed DAE using the structural posedness defined in the DAESA papers [23, 29].

Definition 5.14 *A DAE is ill posed if it has an equivalent DAE that is structurally ill-posed (SIP); otherwise it is well posed.*

Example 5.15 Consider problem (4.4). Using $0 = f_3 = x^2 + y^2 - L^2$, we reduce f_1 to the trivial $0 = \overline{f}_1 = 0$. This is just performing a simple substitution, and is not applying the LC method. The signature matrix

$$\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \\ f_3 \end{array} \begin{array}{c} \begin{array}{ccc} x & y & \lambda \end{array} \\ \begin{bmatrix} - & - & - \\ - & 2 & 0 \\ 0 & 0 & - \end{bmatrix} \end{array} \tag{5.15}$$

does not have a finite HVT, so the resulting DAE is SIP. Hence, by Definition 5.14 , the original SWP DAE (4.4) is ill posed.

Corollary 5.16 *If a structurally well-posed DAE can be converted, by the LC method, to an equivalent DAE that is structurally ill-posed, then the original DAE is ill posed.*

Proof. This follows from Theorem 5.11 and Definition 5.14. □

Example 5.17 Consider the following SWP DAE

$$\begin{aligned} 0 &= f_1 = y''' + y'\lambda + y\lambda' \\ 0 &= f_2 = y'' + y\lambda - g \\ 0 &= f_3 = x^2 + y^2 - L^2. \end{aligned} \tag{5.16}$$

$$\Sigma = \begin{array}{c|ccc|c} & x & y & \lambda & c_i \\ f_1 & - & 3 & 1^\bullet & 0 \\ f_2 & - & 2^\bullet & 0 & 1 \\ f_3 & 0^\bullet & 0 & - & 0 \\ d_j & 0 & 3 & 1 & \end{array} \quad \mathrm{Val}(\Sigma) = 3 \qquad \mathbf{J} = \begin{array}{c|ccc|} & x & y & \lambda \\ f_1 & 0 & 1 & y \\ f_2 & 0 & 1 & y \\ f_3 & 2x & 0 & 0 \end{array} \quad \det(\mathbf{J}) = 0$$

For $u = (1, -1, 0)^T$, $\mathbf{J}^T u = 0$. Using (5.3–5.6) gives

$$I = \{1, 2\}, \quad \theta = c_1 = 0, \quad \text{and} \quad L = \{1\}.$$

Since u is a constant vector, condition (5.5) is satisfied. We replace f_1 by

$$\overline{f}_1 = f_1 - f_2' = (y''' + y'\lambda + y\lambda') - (y'' + y\lambda - g)' = 0.$$

The signature matrix of the resulting problem is exactly (5.15). Hence, by Corollary 5.16, (5.16) is ill posed.

If the Jacobian of the converted DAE is still singular, we may be able to apply the LC method iteratively, provided condition (5.5) is satisfied on each iteration. Since after each conversion step we reduce the value of the signature matrix by at least 1, the number of iterations does not exceed $\mathrm{Val}(\Sigma)$, where Σ is for the original DAE. We use Example 5.18 to show how we can iterate with the LC method.

Example 5.18 We construct the following (artificial) MODPENDA DAE from PEND (3.8):

$$\begin{aligned} 0 &= A = f_3 + f_1' = x^2 + y^2 - L^2 + (x'' + x\lambda)' \\ 0 &= B = f_1 + A'' = x'' + x\lambda + \left(x^2 + y^2 - L^2 + (x'' + x\lambda)'\right)'' \\ 0 &= C = f_2 + A''' = y'' + y\lambda - g + \left(x^2 + y^2 - L^2 + (x'' + x\lambda)'\right)'''. \end{aligned} \tag{5.17}$$

$$\Sigma^0 = \begin{array}{c|ccc|c} & x & y & \lambda & c_i \\ \hline A & 3^\bullet & 0 & 1 & 3 \\ B & 5 & 2^\bullet & 3 & 1 \\ C & 6 & 3 & 4^\bullet & 0 \\ \hline d_j & 6 & 3 & 4 & \end{array} \quad \mathrm{Val}(\Sigma^0) = 9 \qquad \mathbf{J}^0 = \begin{array}{c|ccc} & x & y & \lambda \\ \hline A & 1 & 2y & x \\ B & 1 & 2y & x \\ C & 1 & 2y & x \end{array} \quad \det(\mathbf{J}^0) = 0$$

Here, a superscript denotes an iteration number, not a power. We show how to recover the simple pendulum problem.

We find a vector in $\ker((\mathbf{J}^0)^T)$: $u^0 = (-1, 1, 0)^T$. Then

$$I^0 = \{1, 2\}, \quad \theta^0 = 1, \quad \text{and} \quad L^0 = \{2\}.$$

We replace the second equation B by

$$-A^{(3-1)} + B = -A'' + (A'' + f_1) = f_1 = x'' + x\lambda.$$

The converted DAE is

$$\begin{aligned} 0 &= A = x^2 + y^2 - L^2 + (x'' + x\lambda)' \\ 0 &= f_1 = x'' + x\lambda \\ 0 &= C = y'' + y\lambda - g + \left(x^2 + y^2 - L^2 + (x'' + x\lambda)'\right)'''. \end{aligned}$$

$$\Sigma^1 = \begin{array}{c} \\ A \\ f_1 \\ C \\ d_j \end{array}\begin{array}{ccc} x & y & \lambda \\ 3^\bullet & 0 & 1 \\ 2 & - & 0^\bullet \\ 6 & 3^\bullet & 4 \\ 6 & 3 & 4 \end{array}\begin{array}{c} c_i \\ 3 \\ 4 \\ 0 \\ \end{array} \quad \mathrm{Val}(\Sigma^1) = 6 \qquad \mathbf{J}^1 = \begin{array}{c} \\ A \\ f_1 \\ C \end{array}\begin{array}{ccc} x & y & \lambda \\ 1 & 2y & x \\ 1 & 0 & x \\ 1 & 2y & x \end{array} \quad \det(\mathbf{J}^1) = 0$$

Although $\mathrm{Val}(\Sigma^1) = 6 < 9 = \mathrm{Val}(\Sigma^0)$, matrix $\mathbf{J}^1$ is still singular. If $u^1 = (-1,0,1)^T$, then $(\mathbf{J}^1)^T u^1 = \mathbf{0}$. This gives

$$I^1 = \{1,3\}, \quad \theta^1 = 0, \quad \text{and} \quad L^1 = \{3\}.$$

We replace the third equation C by

$$-A^{(3-0)} + C = -A''' + (f_2 + A''') = f_2 = y'' + y\lambda - g.$$

The converted DAE is

$$\begin{aligned} 0 &= A = x^2 + y^2 - L^2 + (x'' + x\lambda)' \\ 0 &= f_1 = x'' + x\lambda \\ 0 &= f_2 = y'' + y\lambda - g. \end{aligned}$$

$$\Sigma^2 = \begin{array}{c} \\ A \\ f_1 \\ f_2 \\ d_j \end{array}\begin{array}{ccc} x & y & \lambda \\ 3^\bullet & \boxed{0} & 1 \\ 2 & - & 0^\bullet \\ - & 2^\bullet & \boxed{0} \\ 3 & 2 & 1 \end{array}\begin{array}{c} c_i \\ 0 \\ 1 \\ 0 \\ \end{array} \quad \mathrm{Val}(\Sigma^2) = 5 \qquad \mathbf{J}^2 = \begin{array}{c} \\ A \\ f_1 \\ f_2 \end{array}\begin{array}{ccc} x & y & \lambda \\ 1 & 0 & x \\ 1 & 0 & x \\ 0 & 1 & 0 \end{array}. \quad \det(\mathbf{J}^2) = 0$$

We have $\mathrm{Val}(\Sigma^2) = 5 < 6 = \mathrm{Val}(\Sigma^1)$, but $\mathbf{J}^2$ is still singular. We find $u = (1,-1,0)^T$ such that $(\mathbf{J}^2)^T u^2 = \mathbf{0}$. Then

$$I^2 = \{1,2\}, \quad \theta^2 = 0, \quad \text{and} \quad L^2 = \{1\}.$$

Replacing the first equation A by

$$A - f_1' = (f_3 + f_1') - f_1' = f_3 = x^2 + y^2 - L^2,$$

we recover f_1, f_2, f_3 from (5.17). This is exactly the DAE PEND (3.8), with $\mathrm{Val}(\Sigma) = 2$ and $\det(\mathbf{J}) = -2L^2$; cf. Example 3.2.

Since each u in every conversion iteration is a constant vector, each u_l we pick is a nonzero constant. By Remark 5.12, the original DAE (5.17) and PEND are *always* equivalent. Hence, we can solve (5.17) by simply solving PEND.

Chapter 6

The expression substitution method

We develop in this chapter the expression substitution conversion method. In §6.1, we introduce some notation. We describe in §6.2 how to perform a conversion step using this method and address in §6.3 the equivalence issue.

6.1 Preliminaries

A conversion using the LC method seeks a row in Σ, replaces the corresponding equation by a linear combination of existing equations, and constructs a new DAE with a signature matrix of a smaller value. Inspired by the LC method, our goal is to develop a conversion method that seeks a column in Σ, performs a change of certain variables, and constructs a new DAE with $\overline{\Sigma}$ such that $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$. We refer to this approach as the *expression substitution (ES) conversion method*, or the *ES method*.

Again, we start from a SWP DAE with a signature matrix Σ, offsets $\mathbf{c}$, $\mathbf{d}$, and identically singular Jacobian $\mathbf{J}$. To start our analysis, we give some notation below.

Let u be a vector function from the null space of $\mathbf{J}$, that is, $\mathbf{J}u = \mathbf{0}$. Denote by $L(u)$ the set of indices j for which the jth component of u is not identically zero

$$L(u) = \{\ j \mid u_j \neq 0\ \}, \tag{6.1}$$

and denote $s(u)$ by the number of elements in $L(u)$:

$$s(u) = |L(u)|. \tag{6.2}$$

Note that $s \geq 2$. Otherwise $\mathbf{J}$ has a column that is identically the zero vector, and hence $\mathbf{J}$ is structurally singular.

Let

$$I(u) = \{\ i \mid d_j - c_i = \sigma_{ij} \text{ for some } j \in L(u)\ \}. \tag{6.3}$$

Denote also

$$C(u) = \max_{i \in I(u)} c_i. \tag{6.4}$$

Now we illustrate (6.1-6.4).

Example 6.1 Consider

$$\begin{aligned} 0 = f_1 &= x_1 + e^{-x_1' - x_2 x_2''} + g_1(t) \\ 0 = f_2 &= x_1 + x_2 x_2' + x_2^2 + g_2(t). \end{aligned} \tag{6.5}$$

$$\Sigma = \begin{array}{c|cc|c} & x_1 & x_2 & c_i \\ \hline f_1 & 1^\bullet & 2 & 0 \\ f_2 & 0 & 1^\bullet & 1 \\ \hline d_j & 1 & 2 & \end{array} \quad \text{Val}(\Sigma) = 2 \qquad \mathbf{J} = \begin{array}{c|cc} & x_1 & x_2 \\ \hline f_1 & -\mu & -\mu x_2 \\ f_2 & 1 & x_2 \end{array} \quad \det(\mathbf{J}) = 0$$

In $\mathbf{J}$, $\mu = e^{-x_1' - x_2 x_2''}$. Using $u = (x_2, -1)^T$ for which $\mathbf{J}u = 0$, (6.1–6.4) become

$$\begin{aligned} L(u) &= \{1, 2\}, \quad s(u) = |L(u)| = 2, \\ I(u) &= \{1, 2\}, \\ \text{and} \quad C(u) &= \max_{i \in I(u)} c_i = c_2 = 1. \end{aligned} \tag{6.6}$$

We show later how the ES method works on this problem.

Remark 6.2 Assume that we apply the LC method to (6.5). First, we find $u = (1, \mu)^T$ from $\ker(\mathbf{J}^T)$. Using the notation in the LC method, we find $I = \{1, 2\}$, $\theta = 0$, and $k = 1$. Since

$$\sigma(x_1, u) = \sigma(x_1, \mu) = 1 = 1 - 0 = d_1 - \theta,$$

the condition (5.5) is not satisfied. After a conversion step, the resulting DAE is still structurally singular with $\text{Val}(\overline{\Sigma}) = \text{Val}(\Sigma) = 2$ and identically singular $\overline{\mathbf{J}}$. See §7.3 for more details.

6.2 A conversion step using expression substitution

We can perform a conversion step using the ES method, if the following conditions hold for some nonzero u such that $\mathbf{J}u = \mathbf{0}$:

$$\boxed{\sigma(x_j, u) \le \begin{cases} d_j - C(u) - 1 & \text{if } j \in L(u) \\ d_j - C(u) & \text{otherwise} \end{cases}} \tag{6.7}$$

and

$$\boxed{d_j - C(u) \geq 0 \qquad \text{for all } j \in L(u)} \tag{6.8}$$

We call (6.7) and (6.8) the *sufficient conditions for applying the ES method*.

Picking an $l \in L$, we show below how to perform a conversion step. Since we use the same u throughout the following analysis, we omit the argument u and simply write L, s, I, and C.

Without loss of generality, assume that the nonzero entries of u are in its first s positions:

$$u = (u_1, \ldots, u_s, 0, \ldots, 0)^T.$$

Then $L = \{\, j \mid u_j \neq 0 \,\} = \{1, \ldots, s\}$, where $s = |L|$.

We introduce s variables $y_1, \ldots, y_s$ and let

$$\begin{cases} y_j = x_j^{(d_j - C)} - \dfrac{u_j}{u_l} x_l^{(d_l - C)} & \text{for } j \in L \backslash \{\, l \,\}, \\ y_l = x_l^{(d_l - C)} & \text{for } j = l. \end{cases} \tag{6.9}$$

(The condition (6.8) guarantees that the order of x_j, $j \in L$, in (6.9) is nonnegative.)

Written in matrix form, (6.9) is

$$\begin{bmatrix} y_1 \\ \vdots \\ y_l \\ \vdots \\ y_s \end{bmatrix} = \begin{bmatrix} 1 & & -u_1/u_l & & \\ & \ddots & \vdots & & \\ & & 1 & & \\ & & \vdots & \ddots & \\ & & -u_s/u_l & & 1 \end{bmatrix} \begin{bmatrix} x_1^{(d_1 - C)} \\ \vdots \\ x_l^{(d_l - C)} \\ \vdots \\ x_s^{(d_s - C)} \end{bmatrix}.$$

This $s \times s$ square matrix is nonsingular with determinant 1.

We write the first part of (6.9) as

$$x_j^{(d_j - C)} = y_j + \frac{u_j}{u_l} x_l^{(d_l - C)} \qquad \text{for } j \in L \backslash \{l\}. \tag{6.10}$$

By (6.4), $c_i \leq C$ for all $i \in I$. Differentiating (6.10) $C - c_i \geq 0$ times yields

$$\left(x_j^{(d_j - C)}\right)^{(C - c_i)} = x_j^{(d_j - c_i)} = \left(y_j + \frac{u_j}{u_l} x_l^{(d_l - C)}\right)^{(C - c_i)}.$$

In each f_i with $i \in I$, we replace every $x_j^{(\sigma_{ij})}$ with $\sigma_{ij} = d_j - c_i$ and $j \in L \setminus \{\, l \,\}$ by

$$\left(y_j + \frac{u_j}{u_l} x_l^{(d_l - C)}\right)^{(C - c_i)}.$$

Denote by $\overline{f}_i$ the equations resulting from these substitutions. For $i \notin I$, we set $\overline{f}_i = f_i$.

From (6.9), we introduce the equations that prescribe the substitutions:

$$\begin{cases} 0 = g_j = -y_j + x_j^{(d_j - C)} - \dfrac{u_j}{u_l} x_l^{(d_l - C)} & \text{for } j \in L \backslash \{l\} \\ 0 = g_l = -y_l + x_l^{(d_l - C)} & \text{for } j = l. \end{cases} \tag{6.11}$$

We append these equations to the $\overline{f}_i$'s and construct an augmented DAE that comprises

$$\text{equations} \quad \overline{f}_1, \ldots \overline{f}_n;\ g_1, \ldots, g_s \quad \text{in variables} \quad x_1, \ldots, x_n;\ y_1, \ldots y_s.$$

We write the signature matrix and system Jacobian of this converted problem as $\overline{\Sigma}$ and $\overline{\mathbf{J}}$, respectively.

Example 6.3 For (6.5), assume we pick $l = 2$. Since

$$L \setminus \{\, l \,\} = \{\, 1,2 \,\} \setminus \{\, 2 \,\} = \{\, 1 \,\},$$

we introduce $y_j = y_1$. Since

$$d_1 = 1, \quad d_2 = 2, \quad c_1 = 0, \quad C = c_2 = 1, \quad \text{and} \quad u = (x_2, -1)^T,$$

(6.10) becomes

$$x_1^{(d_1 - C)} = x_1 = y_1 + \frac{u_1}{u_2} x_2^{(d_2 - C)} = y_1 - x_2 x_2'.$$

In f_1, we replace $x_1^{(d_1 - c_1)} = x_1^{(1-0)} = x_1'$ by

$$(y_1 - x_2 x_2')^{(C - c_1)} = (y_1 - x_2 x_2')^{(1-0)} = (y_1 - x_2 x_2')' = y_1' - x_2'^2 - x_2 x_2''.$$

Similarly, in f_2 we replace x_1 by $y_1 - x_2 x_2'$. Taking these substitutions into account and appending g_1 and g_2, we obtain

$$\begin{aligned} 0 = \overline{f}_1 &= x_1 + e^{-(y_1 - x_2 x_2')' - x_2 x_2''} + g_1(t) \\ &= x_1 + e^{-y_1' + x_2'^2} + g_1(t) \\ 0 = \overline{f}_2 &= (y_1 - x_2 x_2') + x_2 x_2' + x_2^2 + g_2(t) \\ &= y_1 + x_2^2 + g_2(t) \\ 0 = g_1 &= -y_1 + x_1 + x_2 x_2' \\ 0 = g_2 &= -y_2 + x_2. \end{aligned} \tag{6.12}$$

$$\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g_1 \\ g_2 \\ d_j \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & y_1 & y_2 \end{array} \\ \left[\begin{array}{cccc} 0 & 1^\bullet & 1 & - \\ - & 0 & 0^\bullet & - \\ 0^\bullet & 1 & \boxed{0} & - \\ - & \boxed{0} & - & 0^\bullet \end{array}\right] \\ \begin{array}{cccc} 0 & 1 & 1 & 0 \end{array} \end{array} \begin{array}{c} c_i \\ 0 \\ 1 \\ 0 \\ 0 \\ \end{array} \mathrm{Val}(\overline{\Sigma}) = 1 \qquad \overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g_1 \\ g_2 \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & y_1 & y_2 \end{array} \\ \left[\begin{array}{cccc} 1 & 2x_2'\alpha & -\alpha & 0 \\ 0 & 2x_2 & 1 & 0 \\ 1 & x_2 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{array}\right] \end{array}$$

$$\det(\overline{\mathbf{J}}) = x_2 - 2\alpha(x_2 + x_2')$$

Here $\alpha = e^{-y_1' + x_2'^2}$. If $\det(\overline{\mathbf{J}}) \neq 0$, then SA succeeds on (6.12).

Our aim is to show that $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$ after a conversion step, provided that the sufficient conditions (6.7) and (6.8) hold. Before proving this inequality, we state two lemmas related to the structure of $\overline{\Sigma}$.

Lemma 6.4 *Let $\mathbf{c} = (c_1, \ldots, c_n)$ and $\mathbf{d} = (d_1, \ldots, d_n)$ be the valid offsets for Σ. Let $\overline{\mathbf{c}}$ and $\overline{\mathbf{d}}$ be the two $(n+s)$-vectors defined as*

$$\overline{d}_j = \begin{cases} d_j & \text{if } j = 1:n \\ C & \text{if } j = n+1 : n+s, \quad \text{and} \end{cases} \tag{6.13}$$

$$\overline{c}_i = \begin{cases} c_i & \text{if } i = 1:n \\ C & \text{if } i = n+1 : n+s. \end{cases} \tag{6.14}$$

Then $\overline{\Sigma}$ is of the form in Figure 6.1.

The proof is rather technical, and we present it in Appendix A.

Theorem 6.5 *After a conversion step using the ES method, $Val(\overline{\Sigma}) < Val(\Sigma)$.*

Proof. If $\overline{\Sigma}$ does not have a finite HVT, then $\mathrm{Val}(\overline{\Sigma}) = -\infty$, while the original DAE is SWP with a finite $\mathrm{Val}(\Sigma)$. Hence $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$ holds.

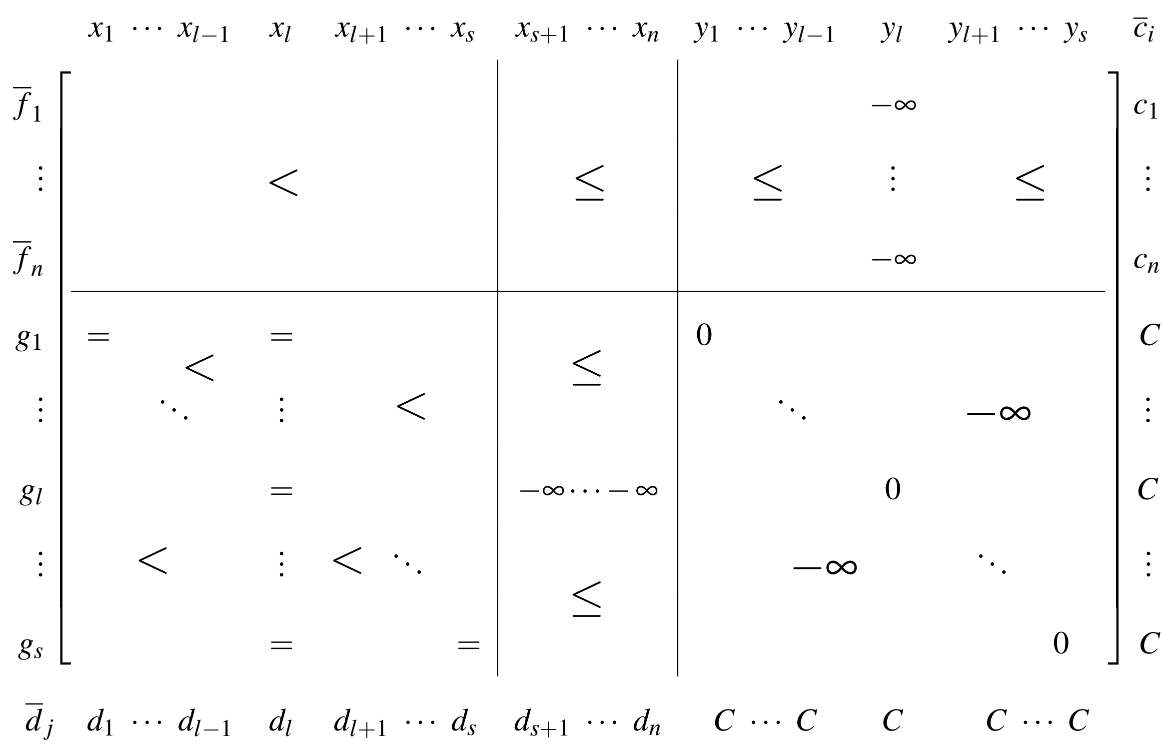

Figure 6.1: The form of $\overline{\Sigma}$ for the converted DAE using the ES method. The $<$, $\leq$, $=$ mean the relations between $\overline{\sigma}_{ij}$ and $\overline{d}_j - \overline{c}_i$, respectively. For example, an (i,j) position with $\leq$ has $\overline{\sigma}_{i,j} \leq \overline{d}_j - \overline{c}_i$.

On the other hand, if $\overline{\Sigma}$ has a finite HVT $\overline{T}$ so that $\mathrm{Val}(\overline{\Sigma}) > -\infty$, then

$$\begin{aligned}
\mathrm{Val}(\overline{\Sigma}) &= \sum_{(i,j)\in\overline{T}} \overline{\sigma}_{ij} \\
&\leq \sum_{(i,j)\in\overline{T}} (\overline{d}_j - \overline{c}_i) && \text{since } \overline{d}_j - \overline{c}_i \geq \overline{\sigma}_{ij} \text{ for all } i,j = 1:n+s \\
&= \sum_{j=1}^{n+s} \overline{d}_j - \sum_{i=1}^{n+s} \overline{c}_i \\
&= \left(\sum_{j=1}^{n} d_j + sC\right) - \left(\sum_{i=1}^{n} c_i + sC\right) && \text{using (6.13) and (6.14),} \\
&= \sum_{j=1}^{n} d_j - \sum_{i=1}^{n} c_i = \mathrm{Val}(\Sigma).
\end{aligned}$$

We prove in the following that $\mathrm{Val}(\overline{\Sigma}) = \mathrm{Val}(\Sigma)$ leads to a contradiction. Assume this equality holds. Then there exists a transversal $\overline{T}$ of $(n+s)$ positions in $\overline{\Sigma}$ such that

$$\overline{d}_j - \overline{c}_i = \overline{\sigma}_{ij} > -\infty \qquad \text{for all } (i,j) \in \overline{T}. \tag{6.15}$$

The column corresponding to y_l has only one finite entry $\overline{\sigma}_{n+l,n+l} = 0$, and therefore $(n+l, n+l) \in \overline{T}$. Consider $(i_1, 1), \ldots, (i_s, s) \in \overline{T}$. Since $(n+l, n+l) \in \overline{T}$, row numbers $i_1, \ldots, i_s$ take values among

$$1, 2, \ldots, n, n+1, \ldots, n+l-1, n+l+1, \ldots, n+s. \tag{6.16}$$

In (6.16) only $s-1$ numbers are greater than n. Hence at least one of these row numbers is among $1:n$. That is, there exists $(i_r, r) \in \overline{T}$ with $1 \le i_r \le n$ and $1 \le r \le s$. This entry is in $\overline{\Sigma}(1:n, 1:s)$[1] in Figure 6.1, so $\overline{d}_r - \overline{c}_{i_r} > \overline{\sigma}_{i_r,r}$, which contradicts to our assumption in (6.15). Therefore $\mathrm{Val}(\overline{\Sigma}) < \mathrm{Val}(\Sigma)$. □

Remark 6.6 We give several remarks about the ES method.

- After a conversion step, we perform symbolic simplifications on $\overline{f}_i$, for $i \in I$. By doing this we ensure that the $x_j^{(d_j-c_i)}$'s for $j \in L = \{1, \ldots, s\}$ disappear from these equations. That is, $\sigma\left(x_j, \overline{f}_i\right) < d_j - c_i$ for $j = 1:s$ and $i \in I$.
- Clearly, y_l appears only in g_l. We mark down the positions in $\overline{T}$ on $\overline{\Sigma}$, and then remove row $n+l$ (corresponding to g_l) and column $n+l$ (corresponding to y_l). Because $(n+l, n+l) \in \overline{T}$, the remaining marked positions still form a HVT $\widetilde{T}$ in the resulting $(n+s-1) \times (n+s-1)$ signature matrix $\widetilde{\Sigma}$.

  Since $\overline{\sigma}_{n+l,n+l} = 0$, $\mathrm{Val}(\widetilde{\Sigma}) = \mathrm{Val}(\overline{\Sigma})$. The purpose to use g_l and y_l in the above proof and analysis is for our convenience. In practice, we can exclude g_l and y_l in the resulting DAE. For consistency, after removing g_l and y_l, we still use $\overline{\Sigma}$ and $\overline{T}$ to denote the signature matrix and system Jacobian, respectively, for the resulting DAE. See Example 6.7.
- If some derivative $x_j^{(d_j-c_i)}$, for $i = 1:n$ and $j \in L \setminus \{\, l \,\}$, appears implicitly in an expression in f_i, then we need to write this expression into a form in which $x_j^{(d_j-c_i)}$ appears explicitly. See Example 6.8.

[1] Using MATLAB notation.

Example 6.7 Removing g_2 and y_2 in (6.12) results in a DAE with the following signature matrix and Jacobian

$$\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g_1 \\ d_j \end{array} \begin{array}{ccc|c} x_1 & x_2 & y_1 & c_i \\ 0 & 1 & 1^\bullet & 0 \\ - & 0^\bullet & 0 & 1 \\ 0^\bullet & 1 & \boxed{0} & 0 \\ 0 & 1 & 1 & \end{array} \quad \mathrm{Val}(\overline{\Sigma}) = 1 \qquad \overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g_1 \end{array} \begin{array}{ccc} x_1 & x_2 & y_1 \\ 1 & 2x_2'\alpha & -\alpha \\ 0 & 2x_2 & 1 \\ 1 & x_2 & 0 \end{array} \quad \det(\overline{\mathbf{J}}) = -x_2 + 2\alpha(x_2 + x_2')$$

Here $\alpha = e^{-y_1' + x_2'^2}$.

Example 6.8 Suppose that x_1''' appears implicitly in $(\sin 2x_1')''$ in f_1, and that the ES method finds

$$L = \{1, 2\}, \quad l = 2, \quad d_1 = d_2 = 3, \quad \text{and} \quad C = c_1 = 0.$$

We want to replace $x_1^{(d_1 - c_1)} = x_1'''$ by

$$\left(y_1 + \frac{u_1}{u_2} x_2^{(d_2 - C)}\right)^{(C - c_1)} = y_1 + \frac{u_1}{u_2} x_2'''.$$

To make x_1''' appear explicitly in f_1, we expand $(\sin 2x_1')''$ and write $2x_1''' \cos x_1' - 4(x_1'')^2 \sin x_1'$ instead. Now we can perform the substitution for x_1'''.

6.3 Equivalence for the ES method

We discuss here the equivalence for the ES method. Our approach below is similar to that for deriving the equivalence for the LC method.

We denote by $\mathcal{F}$ the original DAE with equations f_i, $i = 1 : n$, and a singular Jacobian $\mathbf{J}$. After a conversion step using the ES method, we obtain a (converted) DAE $\overline{\mathcal{F}}$ with equations $\overline{f}_i$, $i = 1 : n + s$, and a Jacobian $\overline{\mathbf{J}}$, which may be singular. Here $\overline{f}_{n+j} = g_j$, $j = 1 : s$.

Assume that

$$\mathbf{x}(t) = \big(x_1(t), \ldots, x_n(t)\big)^T$$

is a solution of $\mathcal{F}$ on some real time interval $\mathbb{I} \subset \mathbb{R}$. That is, every f_i, $i = 1 : n$, vanishes at $(t, \mathbf{x}(t))$. Assume also u is defined at $(t, \mathbf{x}(t))$ for all $t \in \mathbb{I}$. We can substitute $\mathbf{x}(t)$ in (6.9) to find

$$\mathbf{y}(t) = (y_1(t), \ldots, y_s(t))^T$$

such that every $\overline{f}_{n+j} = g_j$, $j = 1 : s$, in (6.11) vanishes at $(t, \mathbf{x}(t), \mathbf{y}(t))$. Using (6.9), we perform substitutions in f_i, $i \in I$, to obtain $\overline{f}_i$. We let $\overline{f}_i = f_i$ for $i \notin I$. Since these substitutions do not change the value of each equation, each $\overline{f}_i$ also vanishes at $(t, \mathbf{x}(t), \mathbf{y}(t))$. Therefore $(\mathbf{x}(t), \mathbf{y}(t))$ is a solution to $\overline{\mathcal{F}}$.

Conversely, assume that $(\overline{\mathbf{x}}(t), \overline{\mathbf{y}}(t))$ is a solution of $\overline{\mathcal{F}}$ on $\mathbb{I} \subset \mathbb{R}$. Assume also that u is defined at $(t, \overline{\mathbf{x}}(t), \overline{\mathbf{y}}(t))$ for all $t \in \mathbb{I}$. Note here u depends merely on t and $\overline{\mathbf{x}}(t)$. Since u_l is a denominator in each g_j in (6.11), this solution requires $u_l(t) \neq 0$ on $\mathbb{I}$. Given that each g_j vanishes at $(t, \overline{\mathbf{x}}(t), \overline{\mathbf{y}}(t))$, from (6.9) we have

$$y_j^{(q)} = \begin{cases} \left(x_j^{(d_j - C)} - \dfrac{u_j}{u_l} x_l^{(d_l - C)} \right)^{(q)} & j \in L \setminus \{l\} \\ \left(x_l^{(d_l - C)} \right)^{(q)} & j = l, \end{cases}$$

where $q \geq 0$. Substituting the expressions on the right-hand side for the derivatives of y_j in each $\overline{f}_i$ recovers f_i and does not change its value. Therefore, each f_i also vanishes at $(t, \overline{\mathbf{x}}(t), \overline{\mathbf{y}}(t))$, or simply $(t, \overline{\mathbf{x}}(t))$ since $\overline{\mathbf{y}}(t)$ does not appear in f_i. Then $\overline{\mathbf{x}}(t)$ is a solution to $\mathcal{F}$.

The above discussion gives

Lemma 6.9 *After a conversion step using the ES method, DAEs $\mathcal{F}$ and $\overline{\mathcal{F}}$ are equivalent if $u_l \neq 0$ for all $t \in \mathbb{I}$.*

Again, if we have a choice for l, it is desirable to choose one (whenever possible) such that u_l is identically nonzero. In this case, $\mathcal{F}$ and $\overline{\mathcal{F}}$ are *always* equivalent and we do not need to check $u_l \neq 0$ when we solve $\overline{\mathcal{F}}$.

Example 6.10 In (6.5), assume we pick $l = 1$. Then (6.10) becomes

$$x_2' = x_2^{(d_2 - C)} = y_2 + \frac{u_2}{u_1} x_1^{(d_1 - C)} = y_2 - x_1/x_2.$$

Here we use

$$d_1 = 1, \quad d_2 = 2, \quad C = 1, \quad \text{and} \quad u = (x_2, -1)^T.$$

Then we

substitute	for	in
$(y_2 - x_1/x_2)'$	x_2''	f_1
$y_2 - x_1/x_2$	x_2'	f_2

The equations g_j derived from (6.11) are

$$\begin{aligned} 0 &= g_1 = -y_1 + x_1 \\ 0 &= g_2 = -y_2 + x_2' + x_1/x_2. \end{aligned}$$

As $l = 1$, we can remove g_1 and y_1, append equation $g = g_2$, and obtain the resulting DAE

$$\begin{aligned}
0 = \overline{f}_1 &= x_1 + e^{-x_1' - x_2 \cdot (y_2 - x_1/x_2)'} + g_1(t) \\
&= x_1 + e^{-x_1' - x_2 y_2' - x_2' x_1/x_2 + x_1'} + g_1(t) \\
&= x_1 + e^{-x_2 y_2' - x_2' x_1/x_2} + g_1(t) \\
0 = \overline{f}_2 &= x_1 + x_2(y_2 - x_1/x_2) + x_2^2 + g_2(t) \\
&= x_2 y_2 + x_2^2 + g_2(t) \\
0 = g &= -y_2 + x_2' + x_1/x_2.
\end{aligned}$$

$$\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g \\ d_j \end{array} \begin{array}{ccc|c} x_1 & x_2 & y_2 & c_i \\ 0 & 1 & 1^\bullet & 0 \\ - & 0^\bullet & 0 & 1 \\ 0^\bullet & 1 & \boxed{0} & 0 \\ 0 & 1 & 1 & \end{array} \quad \mathrm{Val}(\overline{\Sigma}) = 1$$

$$\overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ \overline{f}_2 \\ g \end{array} \begin{bmatrix} x_1 & x_2 & y_2 \\ 1 - x_2'\beta/x_2 & -x_1\beta/x_2 & -x_2\beta \\ 0 & 2x_2 + y_2 & x_2 \\ 1/x_2 & 1 & 0 \end{bmatrix} \quad \det(\overline{\mathbf{J}}) = -x_2 + \beta(2x_2 + y_2 + x_2' - x_1/x_2)$$

In $\overline{\mathbf{J}}$, $\beta = \exp(-x_2 y_2' - x_2' x_1/x_2)$. If $\det(\overline{\mathbf{J}}) \neq 0$, then SA succeeds and gives structural index $\nu_S = 2$. Here $\mathrm{Val}(\overline{\Sigma}) = 1 < 2 = \mathrm{Val}(\Sigma)$.

However, the original DAE and the resulting one are equivalent only if $u_1 = x_2 \neq 0$ on some time interval $\mathbb{I}$. In practice, it is more desirable to choose $l = 2$ since $u_l = -1$ is identically nonzero; see also Example 6.3.

Chapter 7

Examples

In this chapter, we illustrate how to apply the LC method and the ES method to several structurally singular DAEs. When a conversion method *succeeds*, we obtain an equivalent structurally regular DAE with a nonsingular system Jacobian.

In §7.1, we apply both conversion methods to the 4×4 linear constant coefficient (coupled) DAE (4.10). The LC method succeeds in converting this problem to a structurally regular DAE in two iterations, reducing the value of the the signature matrix by 2. In contrast, the ES method reduces the value of the signature matrix by 1 in the first iteration. In the second iteration, the condition for applying the ES method is not satisfied, and hence it cannot be applied further.

In §7.2, we illustrate both methods on an artificially complicated problem MODPENDB derived from the simple pendulum DAE PEND (3.8). We show in §7.2.1 how the ES method succeeds in converting this problem to a structurally regular DAE, which has a relatively simple structure. In §7.2.2, the LC method is applied, but yields a considerably more complicated result.

In §7.3, we address Remark 6.2 in more detail: the condition for applying the LC method is not satisfied for (6.5). If we perform a conversion step, then the value of the signature matrix is not guaranteed to decrease.

7.1 A simple coupled DAE

Recall the 4×4 linear constant coefficient (coupled) DAE (4.10):

$$\mathcal{F}^0 : \begin{cases} 0 = f_1 = -x_1' + x_3 + b_1(t) \\ 0 = f_2 = -x_2' + x_4 + b_2(t) \\ 0 = f_3 = \quad x_2 + x_3 + x_4 + c_1(t) \\ 0 = f_4 = -x_1 + x_3 + x_4 + c_2(t). \end{cases} \tag{7.1}$$

$$
\Sigma^0 = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ d_j \end{array}
\begin{array}{cccc|c}
x_1 & x_2 & x_3 & x_4 & c_i \\
1^\bullet & - & 0 & - & 0 \\
- & 1^\bullet & - & 0 & 0 \\
- & \boxed{0} & 0^\bullet & 0 & 0 \\
\boxed{0} & - & 0 & 0^\bullet & 0 \\
1 & 1 & 0 & 0 &
\end{array}
\quad \mathrm{Val}(\Sigma^0) = 2
\qquad
\mathbf{J}^0 = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \end{array}
\begin{array}{cccc}
x_1 & x_2 & x_3 & x_4 \\
-1 & 0 & 1 & 0 \\
0 & -1 & 0 & 1 \\
0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1
\end{array}
\quad \det(\mathbf{J}^0) = 0
$$

We use[1] $\mathcal{F}^0$ to denote the original problem. We let Σ^0 and $\mathbf{J}^0$ denote the signature matrix and Jacobian of the original problem, respectively.

7.1.1 LC method

We show how to apply the LC method to this problem, and finally obtain an equivalent structurally regular DAE on which SA succeeds.

Let $u^0 = (0,0,-1,1)^T$. Then $(\mathbf{J}^0)^T u^0 = \mathbf{0}$. Using (5.3–5.6) gives

$$I^0 = \{3,4\}, \quad \theta^0 = 0, \quad \text{and} \quad L^0 = \{3,4\}.$$

We choose $l^0 = 3 \in I^0$ and replace f_3 by

$$\overline{f}_3 = u_3^0 f_3 + u_4^0 f_4 = -f_3 + f_4 = -x_1 - x_2 - c_1(t) + c_2(t).$$

The converted DAE is

$$
\mathcal{F}^1 : \begin{cases}
0 = f_1 = -x_1' + x_3 + b_1(t) \\
0 = f_2 = -x_2' + x_4 + b_2(t) \\
0 = \overline{f}_3 = -x_1 - x_2 - c_1(t) + c_2(t) \\
0 = f_4 = -x_1 + x_3 + x_4 + c_2(t).
\end{cases}
$$

$$
\Sigma^1 = \begin{array}{c} \\ f_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ d_j \end{array}
\begin{array}{cccc|c}
x_1 & x_2 & x_3 & x_4 & c_i \\
1^\bullet & - & 0 & - & 0 \\
- & 1 & - & 0^\bullet & 0 \\
0 & 0^\bullet & - & - & 1 \\
\boxed{0} & - & 0^\bullet & 0 & 0 \\
1 & 1 & 0 & 0 &
\end{array}
\quad \mathrm{Val}(\Sigma^1) = 1
\qquad
\mathbf{J}^1 = \begin{array}{c} \\ f_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \end{array}
\begin{array}{cccc}
x_1 & x_2 & x_3 & x_4 \\
-1 & 0 & 1 & 0 \\
0 & -1 & 0 & 1 \\
-1 & -1 & 0 & 0 \\
0 & 0 & 1 & 1
\end{array}
\quad \det(\mathbf{J}^1) = 0
$$

[1] We use a superscript to mean an iteration number, not a power.

Since $\mathbf{J}^1$ is still singular, we apply again the LC method. Let $u^1 = (-1,-1,1,1)^T$. Then $(\mathbf{J}^1)^T u = 0$. Now

$$I^1 = \{1,2,3,4\}, \quad \theta^1 = 0, \quad \text{and} \quad L^1 = \{1,2,4\}.$$

We choose $l^1 = 1 \in I^1$ and replace f_1 by

$$\begin{aligned}
\overline{f}_1 &= u_1^1 f_1 + u_2^1 f_2 + u_3^1 \overline{f}_3' + u_4^1 f_4 \\
&= -f_1 - f_2 + \overline{f}_3' + f_4 \\
&= -\left[-x_1' + x_3 + b_1(t)\right] - \left[-x_2' + x_4 + b_2(t)\right] + \left[-x_1 - x_2 - c_1(t) + c_2(t)\right]' \\
&\qquad + \left[-x_1 + x_3 + x_4 + c_2(t)\right] \\
&= -x_1 - b_1(t) - b_2(t) - c_1'(t) + c_2'(t) + c_2(t).
\end{aligned}$$

The converted DAE is

$$\mathcal{F}^2 : \begin{cases}
0 = \overline{f}_1 = -x_1 - b_1(t) - b_2(t) - c_1'(t) + c_2'(t) + c_2(t) \\
0 = f_2 = -x_2' + x_4 + b_2(t) \\
0 = \overline{f}_3 = -x_1 - x_2 - c_1(t) + c_2(t) \\
0 = f_4 = -x_1 + x_3 + x_4 + c_2(t).
\end{cases}$$

$$\Sigma^2 = \begin{array}{c|cccc|c}
 & x_1 & x_2 & x_3 & x_4 & \bar{c}_i \\
\hline
\overline{f}_1 & 0^\bullet & - & - & - & 1 \\
f_2 & - & 1 & - & 0^\bullet & 0 \\
\overline{f}_3 & 0 & 0^\bullet & - & - & 1 \\
f_4 & \boxed{0} & - & 0^\bullet & 0 & 0 \\
\hline
\bar{d}_j & 1 & 1 & 0 & 0 & \mathrm{Val}(\Sigma^2) = 0
\end{array}
\qquad
\mathbf{J}^2 = \begin{array}{c|cccc}
 & x_1 & x_2 & x_3 & x_4 \\
\hline
\overline{f}_1 & -1 & 0 & 0 & 0 \\
f_2 & 0 & -1 & 0 & 1 \\
\overline{f}_3 & -1 & -1 & 0 & 0 \\
f_4 & 0 & 0 & 1 & 1 \\
\hline
 & \multicolumn{4}{c}{\det(\mathbf{J}^2) = 1}
\end{array}$$

The SA succeeds on this converted DAE and gives structural index $\nu_S = 2$. Since $u_{l^0}^0$ and $u_{l^1}^1$ are nonzero constants, $\mathcal{F}^2$, $\mathcal{F}^1$, and $\mathcal{F}^0$ are always equivalent.

7.1.2 ES method

For $\mathcal{F}^0$ in (7.1), the ES method cannot convert it to a structurally regular DAE. We illustrate this argument with one particular choice of $l \in L$ in each iteration of the ES method, and do not explore all possible combinations of such choices. To handle the limitation of the ES method, further development is required, which is left as future work.

Let $u = (1,-1,1,-1)^T$. Then $\mathbf{J}^0 u = \mathbf{0}$. Using (6.1–6.4) finds

$$L = \{1,2,3,4\}, \quad s = |L| = 4, \quad I = \{1,2,3,4\}, \quad \text{and} \quad C = \max_{i\in I} c_i = 0. \tag{7.2}$$

Assume we pick $l = 3$. Using (6.9), we introduce y_j for each $j \in L \setminus \{l\} = \{1,2,4\}$:

$$\begin{aligned} y_1 &= x_1^{(d_1-C)} - (u_1/u_3)x_3^{(d_3-C)} = x_1' - x_3 \\ y_2 &= x_2^{(d_2-C)} - (u_2/u_3)x_3^{(d_3-C)} = x_2' + x_3 \\ y_4 &= x_4^{(d_4-C)} - (u_4/u_3)x_3^{(d_3-C)} = x_4 + x_3. \end{aligned} \tag{7.3}$$

From (7.3), we construct g_j's in (6.11). Since $l = 3$, we can exclude g_3 and y_3 in the converted DAE; see Remark 6.6.

By (6.10) and from (7.3), we write

$$x_1' = y_1 + x_3, \quad x_2' = y_2 - x_3, \quad \text{and} \quad x_4 = y_4 - x_3.$$

In (4.10), we

substitute	for	in
$y_1 + x_3$	x_1'	f_1
$y_2 - x_3$	x_2'	f_2
$y_4 - x_3$	x_4	f_2, f_3, f_4

The converted DAE is

$$\begin{aligned} 0 &= f_1 = -y_1 + b_1(t) \\ 0 &= f_2 = \quad y_4 - y_2 + b_2(t) \\ 0 &= f_3 = \quad x_2 + y_4 + c_1(t) \\ 0 &= f_4 = -x_1 + y_4 + c_2(t) \\ 0 &= g_1 = -y_1 + x_1' - x_3 \\ 0 &= g_2 = -y_2 + x_2' + x_3 \\ 0 &= g_4 = -y_4 + x_4 + x_3. \end{aligned} \tag{7.4}$$

$$\overline{\Sigma} = \begin{array}{c|ccccccc|c} & x_1 & x_2 & x_3 & x_4 & y_1 & y_2 & y_4 & c_i \\ \hline f_1 & - & - & - & - & 0^\bullet & - & - & 0 \\ f_2 & - & - & - & - & - & 0^\bullet & \boxed{0} & 0 \\ f_3 & - & 0\bullet & - & - & - & - & 0 & 1 \\ f_4 & 0 & - & - & - & - & - & 0^\bullet & 1 \\ g_1 & 1^\bullet & - & 0 & - & 0 & - & - & 0 \\ g_2 & - & 1 & 0^\bullet & - & - & 0 & - & 0 \\ g_4 & - & - & 0 & 0^\bullet & - & - & \boxed{0} & 0 \\ \hline d_j & 1 & 1 & 0 & 0 & 0 & 0 & 1 & \mathrm{Val}(\overline{\Sigma}) = 1 \end{array}$$

$$\overline{\mathbf{J}} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ g_1 \\ g_2 \\ g_4 \end{array} \begin{array}{c} \begin{array}{ccccccc} x_1 & x_2 & x_3 & x_4 & y_1 & y_2 & y_4 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$\det(\overline{\mathbf{J}}) = 0$$

If $u = (1,-1,1,-1,0,0,1)^T$, then $\overline{\mathbf{J}}u = \mathbf{0}$. We use (6.1–6.4) again to find

$$L = \{1,2,3,4,7\}, \quad s = |L| = 5, \quad I = \{3,4,5,6,7\}, \quad \text{and} \quad C = \max_{i \in I} c_i = 1.$$

Since $j = 3,4 \in L$ and

$$d_3 - C = d_4 - C = 0 - 1 = -1 < 0,$$

condition (6.8) is not satisfied, and thus applying the ES method to (7.4) does not guarantee a strict decrease in the value of $\overline{\Sigma}$.

7.2 Artificially modified pendulum MODPENDB

From PEND (3.8), we construct a problem MODPENDB by performing a linear transformation on the state variables:

$$\begin{bmatrix} x \\ y \\ \lambda \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}.$$

The resulting DAE is

$$\begin{aligned} 0 &= f_1 = (z_1+z_2)'' + (z_1+z_2)(z_3+z_1) \\ 0 &= f_2 = (z_2+z_3)'' + (z_2+z_3)(z_3+z_1) - g \\ 0 &= f_3 = (z_1+z_2)^2 + (z_2+z_3)^2 - L^2. \end{aligned} \tag{7.5}$$

$$\Sigma^0 = \begin{array}{c|ccc|c} & z_1 & z_2 & z_3 & c_i \\ \hline f_1 & 2 & 2 & \boxed{0} & 0 \\ f_2 & \boxed{0} & 2 & 2 & 0 \\ f_3 & 0 & 0 & 0 & 2 \\ \hline d_j & 2 & 2 & 2 & \end{array} \quad \mathrm{Val}(\Sigma^0) = 2 \qquad \mathbf{J}^0 = \begin{array}{c|ccc} & z_1 & z_2 & z_3 \\ \hline f_1 & 1 & 1 & 0 \\ f_2 & 0 & 1 & 1 \\ f_3 & 2\alpha & 2(\alpha+\beta) & 2\beta \end{array} \quad \det(\mathbf{J}^0) = 0$$

Here $\alpha = z_1 + z_2$ and $\beta = z_2 + z_3$.

7.2.1 ES method

If $u = (1,-1,1)^T$, then $\mathbf{J}^0 u = \mathbf{0}$. We apply the ES method, and (6.1–6.4) give

$$L = \{1,2,3\}, \quad s = |L| = 3, \quad I = \{1,2,3\}, \quad \text{and} \quad C = \max_{i\in I} c_i = c_3 = 2.$$

Since u is a constant vector, picking any $l \in L = \{1,2,3\}$ gives an equivalent converted DAE. We show below the conversion for case $l = 1$.

Since $L \setminus \{l\} = \{2,3\}$, we introduce variables w_2 and w_3 corresponding to z_2 and z_3, respectively. Using (6.10) gives

$$\begin{aligned} z_2 &= z_2^{(d_2-C)} = w_2 + \frac{u_2}{u_1} z_1^{(d_1-C)} = w_2 - z_1 \\ z_3 &= z_3^{(d_3-C)} = w_3 + \frac{u_3}{u_1} z_1^{(d_1-C)} = w_3 + z_1. \end{aligned} \tag{7.6}$$

To perform substitutions, we first write the derivatives z_1'', z_2'' and z_3'' in f_1 and f_2 explicitly:

$$
\begin{aligned}
0 &= f_1 = z_1'' + z_2'' + (z_1+z_2)(z_3+z_1)\\
0 &= f_2 = z_2'' + z_3'' + (z_2+z_3)(z_3+z_1) - g
\end{aligned}
$$

Then we

substitute	for	in
$w_2'' - z_1''$	z_2''	$f_1,\ f_2$
$w_3'' + z_1''$	z_3''	f_2
$w_2 - z_1$	z_2	f_3
$w_3 + z_1$	z_3	f_3

Taking (7.6) into consideration, we find the resulting DAE (with g_1 and y_1 removed as $l = 1$)

$$
\begin{aligned}
0 &= \overline{f}_1 = w_2'' + w_2(2z_1 + w_3)\\
0 &= \overline{f}_2 = (w_2+w_3)'' + (w_2+w_3)(2z_1+w_3) - g\\
0 &= \overline{f}_3 = w_2^2 + (w_2+w_3)^2 - L^2\\
0 &= g_2 = -z_2 + w_2 - z_1\\
0 &= g_3 = -z_3 + w_3 + z_1.
\end{aligned}
\tag{7.7}
$$

$$
\overline{\Sigma} = \begin{array}{c|ccccc|c}
 & z_1 & z_2 & z_3 & w_2 & w_3 & c_i\\
\hline
\overline{f}_1 & 0 & - & - & 2^\bullet & \boxed{0} & 0\\
\overline{f}_2 & 0^\bullet & - & - & 2 & 2 & 0\\
\overline{f}_3 & - & - & - & 0 & 0^\bullet & 2\\
g_2 & 0 & 0^\bullet & - & \boxed{0} & - & 0\\
g_3 & 0 & - & 0^\bullet & - & \boxed{0} & 0\\
\hline
d_j & 0 & 0 & 0 & 2 & 2 &
\end{array}
\qquad \mathrm{Val}(\overline{\Sigma}) = 2
$$

$$
\overline{\mathbf{J}} = \begin{array}{c|ccccc}
 & z_1 & z_2 & z_3 & w_2 & w_3\\
\hline
\overline{f}_1 & 2w_2 & 0 & 0 & 1 & 0\\
\overline{f}_2 & 2\mu & 0 & 0 & 1 & 1\\
\overline{f}_3 & 0 & 0 & 0 & 2(w_2+\mu) & 2\mu\\
g_2 & -1 & -1 & 0 & 0 & 0\\
g_3 & 1 & 0 & -1 & 0 & 0
\end{array}
\qquad \det(\overline{\mathbf{J}}) = -4L^2
$$

Here $\mu = w_2 + w_3$. We use equation $\overline{f}_3 = 0$ to obtain $\det(\overline{\mathbf{J}})$:

$$\det(\overline{\mathbf{J}}) = -4(2w_2^2 + 2w_2w_3 + w_3^2) = -4L^2 \neq 0.$$

Hence SA succeeds on (7.7). Because $u_1 = 1$ is a nonzero constant, (7.7) and (7.5) are always equivalent.

7.2.2 LC method

We show below how to apply the LC method to (7.5). The resulting DAE is relatively complicated, and its equivalence to the original problem requires two conditions to be satisfied.

Let $u^0 = \big(\alpha, \beta, -1/2\big)^T$. Then $(\mathbf{J}^0)^T u^0 = \mathbf{0}$. Using (5.3–5.6) gives

$$I^0 = \{1,2,3\}, \quad \theta^0 = 0, \quad \text{and} \quad L^0 = \{1,2\}.$$

Since $u_1^0 = \alpha$ and $u_2^0 = \beta$ are not identically nonzero, the converted DAE is equivalent to (7.5) only if $u_l^0 \neq 0$ for the l we pick.

Assume that $u_1^0 = \alpha = z_1 + z_2 \neq 0$. We pick $l = 1$ and replace f_1 by

$$\begin{aligned}
\overline{f}_1 &= u_1^0 f_1 + u_2^0 f_2 + u_3^0 f_3''\\
&= (z_1+z_2) f_1 + (z_2+z_3) f_2 - f_3''/2\\
&= \underline{(z_1+z_2)(z_1+z_2)''} + (z_1+z_2)^2(z_3+z_1) \underline{\underline{+(z_2+z_3)(z_2+z_3)''}} + (z_2+z_3)^2(z_3+z_1)\\
&\qquad - g(z_2+z_3) \underline{-(z_1+z_2)(z_1+z_2)''} - (z_1'+z_2')^2 \underline{\underline{-(z_2+z_3)(z_2+z_3)''}} - (z_2'+z_3')^2\\
&= \left[(z_1+z_2)^2 + (z_2+z_3)^2\right](z_3+z_1) - g(z_2+z_3) - (z_1'+z_2')^2 - (z_2'+z_3')^2\\
&= L^2(z_3+z_1) - g(z_2+z_3) - (z_1'+z_2')^2 - (z_2'+z_3')^2.
\end{aligned}$$

The resulting DAE is

$$\mathcal{F}^1 : \begin{cases} 0 = \overline{f}_1 = L^2(z_3+z_1) - g(z_2+z_3) - (z_1'+z_2')^2 - (z_2'+z_3')^2\\ 0 = f_2 = (z_2+z_3)'' + (z_2+z_3)(z_3+z_1) - g\\ 0 = f_3 = (z_1+z_2)^2 + (z_2+z_3)^2 - L^2. \end{cases}$$

$$\Sigma^1 = \begin{array}{c|ccc|c} & z_1 & z_2 & z_3 & c_i\\ \hline \overline{f}_1 & 1 & 1 & 1 & 1\\ f_2 & \boxed{0} & 2 & 2 & 0\\ f_3 & 0 & 0 & 0 & 2\\ \hline d_j & 2 & 2 & 2 & \end{array} \quad \mathrm{Val}(\Sigma^1) = 3 \qquad \mathbf{J}^1 = \begin{array}{c|ccc|} & z_1 & z_2 & z_3\\ \hline \overline{f}_1 & -2\alpha' & -2(\alpha+\beta)' & -2\beta'\\ f_2 & 0 & 1 & 1\\ f_3 & 2\alpha & 2(\alpha+\beta) & 2\beta \end{array} \quad \det(\mathbf{J}^1) = 0$$

We use α and β to denote z_1+z_2 and z_2+z_3, respectively. Let also γ denote z_3+z_1. Note that (α,β,γ) are actually (x,y,λ) in (3.8)—this notation is for simplicity only, and we shall not substitute α,β,γ for z_1+z_2, z_2+z_3, and z_3+z_1, respectively. That is, we do not use the ES method here.

Matrix $\mathbf{J}^1$ is still identically singular. We let $u^1 = \big(\alpha,\, 2\alpha\beta' - 2\beta\alpha',\, \alpha'\big)^T$. Then $(\mathbf{J}^1)^T u^1 = 0$. Using (5.3–5.6) again gives

$$I^1 = \{1,2,3\}, \quad \theta^1 = 0, \quad \text{and} \quad L^1 = \{2\}.$$

Suppose

$$\frac{u_2^1}{2} = \alpha\beta' - \beta\alpha' = (z_1+z_2)(z_2'+z_3') - (z_2+z_3)(z_1'+z_2') \neq 0.$$

Then the converted DAE is equivalent to $\mathcal{F}^1$. We replace f_2 by

$$\begin{aligned}
\overline{f}_2 &= u_1^1 f_1' + u_2^1 f_2 + u_3^1 f_3'' \\
&= \alpha f_1' + 2(\alpha\beta' - \alpha'\beta) f_2 + \alpha' f_3'' \\
&= \alpha(L^2\gamma' - g\beta' - 2\alpha'\underline{\alpha''} - 2\beta'\underline{\beta''}) + 2(\alpha\beta' - \alpha'\beta)(\underline{\beta''} + \beta\gamma - g) \\
&\qquad\quad + 2\alpha'(\alpha'^2 + \alpha\underline{\alpha''} + \beta'^2 + \beta\underline{\beta''}) \\
&= \alpha(L^2\gamma' - g\beta') + 2(\alpha\beta' - \alpha'\beta)(\beta\gamma - g) + 2\alpha'(\alpha'^2 + \beta'^2) \\
&= (z_1+z_2)\Big[L^2(z_3'+z_1') - g(z_1'+z_2')\Big] \\
&\quad + 2\Big[(z_1+z_2)(z_2'+z_3') - (z_1'+z_2')(z_2+z_3)\Big]\Big[(z_2+z_3)(z_3+z_1) - g\Big] \\
&\quad + 2(z_1'+z_2')\Big[(z_1'+z_2')^2 + (z_2'+z_3')^2\Big].
\end{aligned}$$

The resulting DAE is

$$\mathcal{F}^2 : \begin{cases}
0 = \overline{f}_1 = L^2(z_3+z_1) - g(z_2+z_3) - (z_1'+z_2')^2 - (z_2'+z_3')^2 \\
0 = \overline{f}_2 = (z_1+z_2)\Big[L^2(z_3'+z_1') - g(z_1'+z_2')\Big] \\
\qquad\qquad + 2\Big[(z_1+z_2)(z_2'+z_3') - (z_1'+z_2')(z_2+z_3)\Big]\Big[(z_2+z_3)(z_3+z_1) - g\Big] \\
\qquad\qquad + 2(z_1'+z_2')\Big[(z_1'+z_2')^2 + (z_2'+z_3')^2\Big] \\
0 = f_3 = (z_1+z_2)^2 + (z_2+z_3)^2 - L^2.
\end{cases}$$

$$\Sigma^2 = \begin{array}{c|ccc|c}
 & z_1 & z_2 & z_3 & c_i \\
\hline
\overline{f}_1 & 1 & 1 & 1^\bullet & 0 \\
\overline{f}_2 & 1^\bullet & 1 & 1 & 0 \\
f_3 & 0 & 0^\bullet & 0 & 1 \\
\hline
d_j & 1 & 1 & 1 & \mathrm{Val}(\Sigma^2) = 2
\end{array}$$

Jacobian $\mathbf{J}^2$ is complicated, and we do not show it here. Its determinant is

$$\begin{aligned}
\det(\mathbf{J}^2) &= -4L^2\,(z_1+z_2)\,\big[(z_1+z_2)(z_2'+z_3') - (z_2+z_3)(z_1'+z_2')\big] \\
&= -4\alpha L^2(\alpha\beta' - \beta\alpha') \neq 0,
\end{aligned}$$

since we already assume

$$u_1^0 = \alpha = z_1 + z_2 \neq 0 \quad \text{and} \quad u_2^1/2 = \alpha\beta' - \beta\alpha' \neq 0.$$

Therefore SA succeeds and gives structural index $\nu_S = 1$.

Now we consider the case $\alpha\beta' - \beta\alpha' = 0$. Since

$$0 = h' = 2\alpha\alpha' + 2\beta\beta' \qquad \text{and} \qquad \alpha \neq 0,$$

we have

$$0 = \alpha\beta' - \beta\alpha' = \alpha\beta' + \beta \cdot (\beta\beta')/\alpha = \beta'(\alpha^2 + \beta^2)/\alpha = \beta' L^2/\alpha.$$

So $\beta' = \alpha' = 0$. Since $u^1 = \left(\alpha, 2\alpha\beta' - 2\beta\alpha', \alpha'\right)^T = (\alpha, 0, 0)^T$, the first row in $\mathbf{J}^1$ is identically zero and the LC method is not applicable here.

Hence, the DAEs $\mathcal{F}^2$ and $\mathcal{F}^0$ are equivalent under the conditions

$$u_1^0 = \alpha = z_1 + z_2 \neq 0 \quad \text{and} \quad u_2^1 = \beta' = z_2' + z_3' \neq 0.$$

7.3 DAE (6.5) and LC method

We show below that applying the LC method to (6.5) does not convert it into a structurally non-singular DAE, because the condition for the LC method is not satisfied. Recall (6.5) and its SA result.

$$\begin{aligned} 0 &= f_1 = x_1 + e^{-x_1' - x_2 x_2''} + g_1(t) \\ 0 &= f_2 = x_1 + x_2 x_2' + x_2^2 + g_2(t). \end{aligned}$$

$$\Sigma = \begin{array}{c|cc|c} & x_1 & x_2 & c_i \\ \hline f_1 & 1^\bullet & 2 & 0 \\ f_2 & 0 & 1^\bullet & 1 \\ \hline d_j & 1 & 2 & \end{array} \qquad \mathrm{Val}(\Sigma) = 2 \qquad\qquad \mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \end{array} \!\!\begin{array}{c} \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{bmatrix} -\mu & -\mu x_2 \\ 1 & x_2 \end{bmatrix} \end{array} \qquad \det(\mathbf{J}) = 0$$

Here $\mu = e^{-x_1' - x_2 x_2''}$. If $u = (1, \mu)^T$, then $\mathbf{J}^T u = \mathbf{0}$. Using (5.3–5.6) gives

$$I = \{1, 2\}, \quad \theta = 0, \quad \text{and} \quad L = \{1\}.$$

Let $l = 1$ and replace f_1 by

$$\begin{aligned} \overline{f}_1 &= u_1 f_1 + u_2 f_2' = f_1 + \mu f_2' \\ &= x_1 + \mu + g_1(t) + \mu(x_1 + x_2 x_2' + x_2^2 + g_2(t))' \\ &= x_1 + \mu + g_1(t) + \mu\left(x_1' + x_2 x_2'' + (x_2')^2 + 2x_2 x_2' + g_2'(t)\right) \\ &= x_1 + g_1(t) + \mu\left(1 + x_1' + x_2 x_2'' + (x_2')^2 + 2x_2 x_2' + g_2'(t)\right). \end{aligned}$$

The resulting DAE is

$$
\begin{aligned}
0 &= \overline{f}_1 = x_1 + g_1(t) + e^{-x_1' - x_2 x_2''}\left(1 + x_1' + x_2 x_2'' + (x_2')^2 + 2x_2x_2' + g_2'(t)\right) \\
0 &= f_2 = x_1 + x_2x_2' + x_2^2 + g_2(t).
\end{aligned}
\tag{7.8}
$$

$$
\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \\ d_j \end{array}
\begin{array}{c} \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{bmatrix} 1^\bullet & 2 \\ 0 & 1^\bullet \end{bmatrix} \\ \begin{array}{cc} 1 & 2 \end{array} \end{array}
\begin{array}{c} c_i \\ 0 \\ 1 \\ \end{array}
\quad \mathrm{Val}(\overline{\Sigma}) = 2
\qquad
\overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \end{array}
\begin{array}{c} \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{bmatrix} -\alpha\mu & -\alpha\mu x_2 \\ 1 & x_2 \end{bmatrix} \end{array}
\quad \det(\overline{\mathbf{J}}) = 0
$$

Here $\alpha = x_1' + x_2x_2'' + (x_2')^2 + 2x_2x_2' + g_2'(t)$ and $\mu = e^{-x_1' - x_2x_2''}$.

The conversion step does not reduce the value of signature matrix nor produce a nonsingular Jacobian, because (5.5) is not satisfied:

$$\sigma(x_1, u) = \sigma(x_1, \mu) = 1 = 1 - 0 = d_1 - \theta.$$

Chapter 8

Conclusion and future work

We identified two types of structural analysis's failure. For the first type, the system Jacobian is structurally singular, and the failure is likely due to hidden symbolic cancellations. One way to handle this is to perform symbolic simplifications before applying SA.

We focused on dealing with the second type, where SA fails in a less obvious way. In this case, the Jacobian is not structurally singular but is still identically singular. We proposed two symbolic-numeric methods for converting a DAE with such singular Jacobian to an equivalent DAE on which SA succeeds with nonsingular Jacobian, provided some conditions are satisfied. Such conditions can be checked automatically. These conversion methods provably succeed and thus allow SA to handle more DAE types. Our methods enable SA to better recognize the true structure of a DAE, and thus SA is more likely to succeed and obtain correct structural information. Moreover, our methods provide insights into reasons for SA's failures, which were not well understood before.

We summarize the two conversion methods here. The LC method is more straightforward: it keeps the size of the system and replaces only one equation within a conversion step. The ES method requires more conditions to apply. It augments the system and changes several equations within a conversion step, which generally takes more symbolic operations. The common goal of both methods is to reduce the value of the signature matrix; this value is also the number of degrees of freedom reported by the SA. We also need to ensure that the converted DAE is equivalent to the original one: on some time interval, a solution of the original DAE should be a solution to the converted DAE, and vice versa. Moreover, it is desirable to choose a conversion (if possible) such that we do not need to monitor the equivalence condition ($u_l \neq 0$) when we solve the converted problem.

A practical question worth considering is how to choose the appropriate conversion method between the two for a given structurally singular DAE. For many of the examples we have studied, it is fairly common that conditions of only one method are satisfied; see §7.1 and §7.3. For some other DAEs, provided conditions of both methods are satisfied, applying one method usually requires fewer symbolic manipulations than applying the other; see MODPENDB in §7.2. In the latter case, we examine if a conversion has an identically nonzero u_l, such that the converted DAE and the original one are *always* equivalent. If applying either method guarantees such equivalence, we prefer the LC method because it changes only one equation and maintains the problem size.

Our next goal is to combine these conversion methods with block triangular forms (BTFs) of a DAE [28, 29]. If the Jacobian is identically singular and the DAE has a non-trivial BTF, that is, it has two or more diagonal blocks, then we can locate the block that leads to the singularity, and perform a conversion step on this singular block. Using this approach, we have already made some progress and managed to convert several problems, including the Campbell-Griepentrog Robot Arm and the Ring Modulator, into SA success cases. However, more careful studies are required and details need to be worked out.

Future work also includes rigorous implementation of these conversion methods in MATLAB. We currently have a prototype code, which builds upon our structural analyzer package DAESA [23] and takes advantage of MATLAB's symbolic toolbox. We can call DAESA functions to display a converted DAE's structure, perform quasilinearity analysis, and print out a solution scheme [24]. Our goal is to eventually build a complete tool for performing DAE conversions.

Another interesting direction for research is combining the dummy derivative index reduction method [17] with the conversion techniques. McKenzie et al. [19] show that dummy derivative method and the Σ-method share some similarities. The former method converts a high-index DAE into a low-index one by introducing dummy algebraic variables and augmenting the system. Suppose we adopt this methodology. When a condition for applying a conversion method is violated, applying some dummy-derivative-like strategy may help make a conversion possible.

Our conversion methods require a symbolic toolbox that is capable of performing nontrivial symbolic arithmetic operations, differentiation, and simplifications. However, we only focus on a linear combination of the equations (in the LC method) and a linear combination of derivatives of highest order (in the ES method). How to further exploit symbolic tools to develop other conversion methods should deserve some investigation.

We conclude with a conjecture here. In all our experiments, when we transform a DAE with identically singular Jacobian to an equivalent solvable DAE with generically nonsingular Jacobian, the value of the signature matrix always decreases. As Pryce points out in [26], the solvability of a DAE lies within its inherent nature, not the way it is formulated nor the method that analyzes it. We conjecture that, if a reformulation of a structurally singular DAE results in an equivalent solvable DAE, then the value of the signature matrix always decreases. However, based on our current knowledge, it seems difficult to prove this conjecture.

Appendix A

Proofs for the ES method

For readers' convenience, we recall the notation from Sections 6.1 and 6.2:

$$L = \left\{ j \mid u_j \neq 0 \right\} = \{1:s\}, \qquad s = |L|,$$
$$I = \left\{ i \mid d_j - c_i = \sigma_{ij} \ \text{ for some } j \in L \right\}, \qquad C = \max_{i \in I} c_i.$$

We also assume the conditions for applying the ES method are satisfied:

$$\sigma\left(x_j, u\right) \leq \begin{cases} d_j - C - 1 & \text{if } j \in L \\ d_j - C & \text{otherwise,} \end{cases} \tag{A.1}$$

and

$$d_j - C \geq 0 \qquad \text{for all } j \in L.$$

Denote

$$\bar{I} = \{1:n\} \setminus I = \left\{ i \mid d_j - c_i > \sigma_{ij} \ \text{ for all } j \in L \right\}, \quad \text{and}$$
$$\overline{L} = \{1:n\} \setminus L = \left\{ j \mid u_j = 0 \right\} = \{s+1:n\}.$$

In the following, we assume we pick a column index $l \in L$ in a conversion step using the ES method. We also assume $u_l \neq 0$ for all t in some $\mathbb{I} \subset \mathbb{R}$.

A.1 Preliminary results for the proof of Lemma 6.4

Lemma A.1 *Let $l \in L$. If (A.1) holds, then for an $r \in L \setminus \{l\}$,*

$$\sigma\left(x_j, y_r + \frac{u_r}{u_l} x_l^{(d_l - C)}\right) \leq \begin{cases} d_j - C - 1 & \text{if } j \in L \setminus \{l\} = \left\{1, \ldots, l-1, l+1, \ldots, s\right\} \\ d_j - C & \text{if } j \in \overline{L} \cup \{l\} = \left\{l, s+1, s+2, \ldots, n\right\}. \end{cases} \tag{A.2}$$

Proof. According to our assumptions at the beginning of §5, no symbolic cancellation occurs in a structurally singular DAE. Hence the formal HOD and the true HOD of a variable in a function are the same. Using (4.7) gives

$$\sigma\left(x_j, y_r + \frac{u_r}{u_l} x_l^{(d_l - C)}\right) \le \max\left\{\sigma\left(x_j, u\right), \sigma\left(x_j, x_l^{(d_l - C)}\right)\right\}. \tag{A.3}$$

(a) Consider the case $j = l \in L$. Using (A.1) gives $\sigma(x_l, u) \le d_l - C - 1$, so

$$\begin{aligned} \text{RHS of (A.3)} &= \max\left\{\sigma(x_l, u), \sigma\left(x_l, x_l^{(d_l - C)}\right)\right\} \\ &= \sigma\left(x_l, x_l^{(d_l - C)}\right) = d_l - C. \end{aligned} \tag{A.4}$$

(b) Consider the case $j \ne l$, that is, $j \in \{1, \ldots, l-1, l+1, \ldots, n\}$. Using (A.1) again, we have

$$\begin{aligned} \text{RHS of (A.3)} &= \max\left\{\sigma\left(x_j, u\right), \sigma\left(x_j, x_l^{(d_l - C)}\right)\right\} \\ &= \sigma\left(x_j, u\right) \le \begin{cases} d_j - C - 1 & \text{if } j \in L \setminus \{l\} = \{1, \ldots, l-1, l+1, \ldots, s\} \\ d_j - C & \text{if } j \notin L = \{l, s+1, \ldots, n\}. \end{cases} \end{aligned} \tag{A.5}$$

Combining (A.4) and (A.5) results in (A.2) and completes this proof. □

Corollary A.2 *For an* $i \in I$,

$$\sigma\left(x_j, \left(y_r + \frac{u_r}{u_l} x_l^{(d_l - C)}\right)^{(C - c_i)}\right) \le \begin{cases} d_j - c_i - 1 & \textit{if } j \in L \setminus \{l\} = \{1, \ldots, l-1, l+1, \ldots, s\} \\ d_j - c_i & \textit{otherwise}. \end{cases} \tag{A.6}$$

Proof. Since $C = \max_{i \in I} c_i$, the order $C - c_i \ge 0$ for $i \in I$. We have

$$\begin{aligned} \text{LHS of (A.6)} &= \sigma\left(x_j, y_r + \frac{u_r}{u_l} x_l^{(d_l - C)}\right) + (C - c_i) \\ &\le (C - c_i) + \begin{cases} d_j - C - 1 & \text{if } j \in L \setminus \{l\} \\ d_j - C & \text{otherwise} \end{cases} && \text{using (A.2)} \\ &= \begin{cases} d_j - c_i - 1 & \text{if } j \in L \setminus \{l\} \\ d_j - c_i & \text{otherwise} \end{cases} \\ &= \text{RHS of (A.6)}. \end{aligned}$$

□

A.2 Proof of Lemma 6.4

Proof. We write $\overline{\Sigma}$ from Figure 6.1 into the following 2×3 block form

$$\overline{\Sigma} = \left[\begin{array}{c|c|c} \overline{\Sigma}_{1,1} & \overline{\Sigma}_{1,2} & \overline{\Sigma}_{1,3} \\ \hline \overline{\Sigma}_{2,1} & \overline{\Sigma}_{2,2} & \overline{\Sigma}_{2,3} \end{array}\right].$$

We aim to prove the relations between $\overline{\sigma}_{ij}$ and $\overline{d}_j - \overline{c}_i$ in each block.

Recall (6.13) and (6.14):

$$\begin{cases} \overline{d}_j = d_j,\ \overline{c}_i = c_i & \text{if } i, j = 1 : n \\ \overline{d}_j = \overline{c}_i = C & \text{if } i, j = n+1 : n+s. \end{cases}$$

We prove for $\left[\overline{\Sigma}_{1,1} \middle| \overline{\Sigma}_{1,2}\right]$, $\overline{\Sigma}_{1,3}$, $\left[\overline{\Sigma}_{2,1} \middle| \overline{\Sigma}_{2,2}\right]$, and $\overline{\Sigma}_{2,3}$ in order.

1. Consider

$$\left[\,\overline{\Sigma}_{1,1}\,\middle|\,\overline{\Sigma}_{1,2}\,\right] = \begin{array}{c|ccccccc|ccc|c} & x_1 & \cdots & x_{l-1} & x_l & x_{l+1} & \cdots & x_s & x_{s+1} & \cdots & x_n & \overline{c}_i \\ \hline \overline{f}_1 & & & & & & & & & & & c_1 \\ \vdots & & & & < & & & & & \leq & & \vdots \\ \overline{f}_n & & & & & & & & & & & c_n \\ \hline \overline{d}_j & d_1 & \cdots & d_{l-1} & d_l & d_{l+1} & \cdots & d_s & d_{s+1} & \cdots & d_n & \end{array}.$$

We need to show that, for $i = 1 : n$,

$$\overline{\sigma}_{ij} = \sigma\left(x_j, \overline{f}_i\right) \leq \begin{cases} \overline{d}_j - \overline{c}_i - 1 & \text{if } j \in L = \{1 : s\}, \\ \overline{d}_j - \overline{c}_i & \text{if } j \in \overline{L} = \{s+1 : n\}. \end{cases} \tag{A.7}$$

We consider the cases

(a) $j \in L$ and $i \in I$,

(b) $j \in \overline{L}$ and $i \in I$, and

(c) $j \in L \cup \overline{L}$ and $i \in \overline{I}$.

Let $r \in L \backslash \{\, l \,\} = \{\, 1, \ldots, l-1, l+1, \ldots, s \,\}$.

(a) Consider $j \in L \setminus \{l\}$. By Corollary A.2, the HOD of x_j is $\le d_j - c_i - 1$ in every

$$\left(y_r + \frac{u_r}{u_l} x_l^{(d_l-C)}\right)^{(C-c_i)}$$

that replaces $x_r^{(d_r-c_i)}$ in an f_i—here $r \in L \setminus \{\,l\,\}$, which includes j. Therefore,

$$\sigma\left(x_j, \overline{f}_i\right) \le d_j - c_i - 1 \qquad \text{for } j \in L \setminus \{\,l\,\},\ i \in I. \tag{A.8}$$

Now consider $j = l \in L$. We show below that $\partial \overline{f}_i / \partial x_l^{(d_l-c_i)} = 0$, which implies $x_l^{(d_l-c_i)}$ does not appear in $\overline{f}_i$, $i \in I$. That is,

$$\sigma\left(x_l, \overline{f}_i\right) \le d_l - c_i - 1 \qquad \text{for } i \in I. \tag{A.9}$$

Using (A.1) gives

$$\sigma\left(x_l, \frac{u_r}{u_l}\right) \le \sigma\left(x_l, u\right) \le d_l - C - 1.$$

Also

$$\sigma\left(x_l, y_r + \frac{u_r}{u_l} x_l^{(d_l-C)}\right) = \max\left\{\sigma\left(x_l, u\right), \sigma\left(x_l, x_l^{(d_l-C)}\right)\right\} = d_l - C.$$

Since $C - c_i \ge 0$ for $i \in I$, we apply Griewank's lemma (5.1), with $q = C - c_i$, to

$$x_r^{(d_r-C)} = y_r + \frac{u_r}{u_l} x_l^{(d_l-C)} \tag{A.10}$$

from (6.10). Differentiating both sides of (A.10) with respect to $x_l^{(d_l-C)}$ gives

$$\frac{u_r}{u_l} = \frac{\partial x_r^{(d_r-C)}}{\partial x_l^{(d_l-C)}} = \frac{\partial x_r^{(d_r-C+C-c_i)}}{\partial x_l^{(d_l-C+C-c_i)}} = \frac{\partial x_r^{(d_r-c_i)}}{\partial x_l^{(d_l-c_i)}}. \tag{A.11}$$

Then

$$\begin{aligned}
\frac{\partial \overline{f}_i}{\partial x_l^{(d_l-c_i)}} &= \frac{\partial f_i}{\partial x_l^{(d_l-c_i)}} + \sum_{r \in L \setminus \{l\}} \frac{\partial f_i}{\partial x_r^{(d_r-c_i)}} \cdot \frac{\partial x_r^{(d_r-c_i)}}{\partial x_l^{(d_l-c_i)}} && \text{by the chain rule} \\
&= \mathbf{J}_{il} + \sum_{r \in L \setminus \{l\}} \mathbf{J}_{ir} \cdot \frac{u_r}{u_l} && \text{by (A.11)} \\
&= \frac{1}{u_l} \sum_{r \in L} \mathbf{J}_{ir} u_r = \frac{1}{u_l} (\mathbf{J} u)_i = 0 && \text{because } \mathbf{J} u = \mathbf{0}.
\end{aligned}$$

(b) Now $j \in \overline{L} = \{s+1:n\}$ and $i \in I$. None of the $x_j^{(d_j-c_i)}$ with $j \in \overline{L}$ and $i \in I$ is replaced. By Corollary A.2, the HOD of x_j is $\le d_j - c_i$ in every

$$\left(y_r + \frac{u_r}{u_l} x_l^{(d_l-C)}\right)^{(C-c_i)}$$

that replaces $x_r^{(d_r-c_i)}$. Hence

$$\sigma\left(x_j, \overline{f}_i\right) \le d_j - c_i \qquad \text{for } j \notin L,\ i \in I. \tag{A.12}$$

(c) Consider $i \in \bar{I}$. Since $d_j - c_i > \sigma_{ij}$ for every $j \in L$, we do not replace any derivative. Hence,

$$\sigma\left(x_j, \overline{f}_i\right) = \sigma\left(x_j, f_i\right) \le \begin{cases} d_j - c_i - 1 & \text{for } i \in \bar{I},\ j \in L = \{1:s\} \\ d_j - c_i & \text{for } i \in \bar{I},\ j \in \overline{L} = \{s+1:n\}. \end{cases} \tag{A.13}$$

Since $\overline{d}_j = d_j$ and $\overline{c}_i = c_i$ for $i, j = 1:n$, combining (A.8, A.9, A.12, A.13) proves (A.7).

2. For

$$\overline{\Sigma}_{1,3} = \begin{array}{c} \\ \overline{f}_1 \\ \vdots \\ \overline{f}_n \\ \overline{d}_j \end{array} \begin{array}{c} \begin{array}{ccccccc} y_1 & \cdots & y_{l-1} & y_l & y_{l+1} & \cdots & y_s \end{array} \\ \left[\begin{array}{ccc} & -\infty & \\ \le & -\vdots & \le \\ & -\infty & \end{array}\right] \\ \begin{array}{ccccccc} C & \cdots & C & C & C & \cdots & C \end{array} \end{array} \begin{array}{c} \overline{c}_i \\ c_1 \\ \vdots \\ c_n \\ \\ \end{array},$$

we need to show that, for $i = 1:n$,

$$\overline{\sigma}_{i,n+j} = \sigma\left(y_j, \overline{f}_i\right) \begin{cases} \le \overline{d}_j - \overline{c}_i & \text{if } j \in L \setminus \{l\}, \\ = -\infty & \text{if } j = l. \end{cases}$$

Here, the $(n+j)$th column corresponds to y_j.

(a) Consider $j \in L \setminus \{l\}$. For all $i \in I$,

$$\begin{aligned} \sigma\left(y_j, \left(y_j + \frac{u_j}{u_l} x_l^{(d_l-C)}\right)^{(C-c_i)}\right) &= \sigma\left(y_j, y_j + \frac{u_j}{u_l} x_l^{(d_l-C)}\right) + (C - c_i) \\ &= 0 + (C - c_i) = C - c_i. \end{aligned}$$

Then for all i,

$$\overline{\sigma}_{i,n+j} = \sigma\left(y_j, \overline{f}_i\right) = \begin{cases} C - c_i & \text{if } i \in I \text{ and} \\ & \quad x_j^{(d_j - c_i)} \text{ is replaced by } \left(y_j + \dfrac{u_j}{u_l} x_l^{(d_l - C)}\right)^{(C-c_i)} \\ -\infty & \text{otherwise.} \end{cases}$$

To combine these two cases, we can write

$$\overline{\sigma}_{i,n+j} \le C - c_i = \overline{d}_{n+j} - \overline{c}_i \qquad \text{for } j \in L \backslash \{\, l \,\} \text{ and all } i = 1:n.$$

(b) Now consider $j = l$. Since y_l does not appear in any $\overline{f}_i$,

$$\overline{\sigma}_{i,n+l} = \sigma\left(y_l, \overline{f}_i\right) = -\infty \qquad \text{for all } i = 1:n.$$

3. Consider

$$\left[\ \overline{\Sigma}_{2,1} \;\middle|\; \overline{\Sigma}_{2,2}\ \right] = \begin{array}{cc} & \\ \overline{f}_{n+1} & g_1 \\ \vdots & \vdots \\ \overline{f}_{n+l} & g_l \\ \vdots & \vdots \\ \overline{f}_{n+s} & g_s \\ & \overline{d}_j \end{array} \begin{array}{c} \begin{array}{cccccccc|ccc} x_1 & \cdots & x_{l-1} & x_l & x_{l+1} & \cdots & x_s & & x_{s+1} & \cdots & x_n \end{array} \\ \left[\begin{array}{ccccccc|ccc} = & & < & = & & & & & \le & \\ & \ddots & & \vdots & & < & & & & \\ & & & = & & & & -\infty & \cdots & -\infty \\ & < & & \vdots & < & \ddots & & & \le & \\ & & & = & & & = & & & \end{array}\right] \\ \begin{array}{cccccccccc} d_1 & \cdots & d_{l-1} & d_l & d_{l+1} & \cdots & d_s & d_{s+1} & \cdots & d_n \end{array} \end{array} \begin{array}{c} \overline{c}_i \\ C \\ \vdots \\ C \\ \vdots \\ C \\ \\ \end{array} .$$

Recall $l \in L$. Let row number $i \in \{1:s\}$. We consider the following cases.

(a) $j = l$ or $j = i$. That is, the entries in the lth column in $\overline{\Sigma}_{2,1}$ or those on the (main) diagonal of $\overline{\Sigma}_{2,1}$.

(b) $j \ne l$ and $i = l$. That is, the lth row in $\left[\overline{\Sigma}_{2,1}, \overline{\Sigma}_{2,2}\right]$ without the lth column.

(c1) $j = 1:s$ and j, l, i are distinct. This case covers all the entries with '$<$' in $\overline{\Sigma}_{2,1}$.

(c2) $j = s+1:n$ and j, l, i are distinct. This case covers all the entries with '$\le$' in $\overline{\Sigma}_{2,2}$.

We shall prove

$$\overline{\sigma}_{n+i,j} = \sigma\left(x_j, g_i\right) \begin{cases} = \overline{d}_j - \overline{c}_{n+i} & \text{if } j = l \text{ or } j = i \\ = -\infty & \text{if } j \neq l \text{ and } i = l \\ \leq \overline{d}_j - \overline{c}_{n+i} - 1 & \text{if } j = 1:s, \text{ and } j, l, i \text{ are distinct} \\ \leq \overline{d}_j - \overline{c}_{n+i} & \text{if } j = s+1:n \text{ and } j, l, i \text{ are distinct.} \end{cases} \tag{A.14}$$

Recall

$$0 = g_i = \begin{cases} -y_i + x_i^{(d_i - C)} - \dfrac{u_i}{u_l} x_l^{(d_l - C)} & \text{for } i \in L \backslash \{l\} \\ -y_l + x_l^{(d_l - C)} & \text{for } i = l. \end{cases}$$

(a) Since $x_l^{(d_l-C)}$ and $x_i^{(d_i-C)}$ (if $l = i$ then both are the same) occur in g_i,

$$\begin{aligned} \sigma\left(x_l, g_i\right) &= d_l - C = \overline{d}_l - \overline{c}_{n+i} \quad \text{and} \\ \sigma\left(x_i, g_i\right) &= d_i - C = \overline{d}_i - \overline{c}_{n+i}. \end{aligned} \tag{A.15}$$

(b) Now

$$\sigma\left(x_j, g_l\right) = \sigma\left(x_j, y_l - x_l^{(d_l - C)}\right) = -\infty \leq d_j - C - 1 = \overline{d}_j - \overline{c}_{n+l} - 1. \tag{A.16}$$

(c1, c2) Consider the last two cases together: j, l, and i are distinct. We have

$$\sigma\left(x_j, g_i\right) = \sigma\left(x_j, y_i - x_i^{(d_i - C)} + \frac{u_i}{u_l} x_l^{(d_l - C)}\right) \leq \sigma\left(x_j, u\right).$$

Using (A.1) gives

$$\sigma\left(x_j, g_i\right) \leq \sigma\left(x_j, u\right) \leq \begin{cases} d_j - C - 1 & = \overline{d}_j - \overline{c}_{n+i} - 1 & \text{if } j \in L \\ d_j - C & = \overline{d}_j - \overline{c}_{n+i} & \text{if } j \in \overline{L}. \end{cases} \tag{A.17}$$

Combining (A.15–A.17) gives (A.14).

4. For

$$\overline{\Sigma}_{2,3} = \begin{array}{cc} & \\ \overline{f}_{n+1} & g_1 \\ \vdots & \vdots \\ \overline{f}_{n+l} & g_l \\ \vdots & \vdots \\ \overline{f}_{n+s} & g_s \\ & \overline{d}_j \end{array} \begin{array}{c} \begin{array}{ccccccc} y_1 & \cdots & y_{l-1} & y_l & y_{l+1} & \cdots & y_s \end{array} \\ \left[\begin{array}{ccccc} 0 & & & & \\ & \ddots & & -\infty & \\ & & 0 & & \\ & -\infty & & \ddots & \\ & & & & 0 \end{array}\right] \\ \begin{array}{ccccccc} C & \cdots & C & C & C & \cdots & C \end{array} \end{array} \begin{array}{c} \\ C \\ \vdots \\ C \\ \vdots \\ C \\ \\ \end{array},$$

we consider $i, j = 1 : s$. Then

$$\begin{aligned} \overline{\sigma}_{n+i,n+j} = \sigma\left(y_j, g_i\right) &= \sigma\left(y_j, y_i - x_i^{(d_i-C)} + \frac{u_i}{u_l} x_l^{(d_l-C)}\right) \\ &= \sigma\left(y_j, y_i\right) \\ &= \begin{cases} 0 = C - C = \overline{d}_{n+j} - \overline{c}_{n+i} & \text{if } i = j \\ -\infty & \text{otherwise.} \end{cases} \quad \square \end{aligned}$$

Appendix B

More examples

We review how to remedy several structurally singular DAEs from the literature. We discuss the index-5 Campbell-Griepentrog Robot Arm DAE in §B.1, the transistor amplifier DAE in §B.2, and the ring modulator problem in §B.3. We also show in §B.4 how to "fix" the index overestimation problem on Reißig's family of linear DAEs, for which SA produces a nonsingular system Jacobian but overestimates the index. After we apply a technique similar to the linear combination method, SA reports the correct index $\nu_S = 1$ on this family of DAEs.

B.1 Robot Arm

We slightly simplify the two-link robot arm problem in [5] by writing the derivatives of x_1, x_2, and x_3 implicitly in the equations:

$$
\begin{aligned}
0 = f_1 &= x_1'' - \Big[2c(x_3)(x_1'+x_3')^2 + x_1'^2 d(x_3) \\
&\qquad + (2x_3 - x_2)\big(a(x_3)+2b(x_3)\big) + a(x_3)(u_1-u_2)\Big] \\
0 = f_2 &= x_2'' - \Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) \\
&\qquad + (2x_3 - x_2)\big(1-3a(x_3)-2b(x_3)\big) - a(x_3)u_1 + \big(a(x_3)+1\big)u_2\Big] \\
0 = f_3 &= x_3'' - \Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) + (2x_3 - x_2)\big(a(x_3)-9b(x_3)\big) \\
&\qquad - 2x_1'^2 c(x_3) - d(x_3)\big(x_1'+x_3'\big)^2 - \big(a(x_3)+b(x_3)\big)(u_1-u_2)\Big] \\
0 = f_4 &= \cos x_1 + \cos(x_1+x_3) - p_1(t) \\
0 = f_5 &= \sin x_1 + \sin(x_1+x_3) - p_2(t),
\end{aligned}
\tag{B.1}
$$

where

$$p_1(t) = \cos(1-e^t) + \cos(1-t) \qquad p_2(t) = \sin(1-e^t) + \sin(1-t)$$
$$a(s) = 2/(2-\cos^2 s) \qquad b(s) = \cos s/(2-\cos^2 s)$$
$$c(s) = \sin s/(2-\cos^2 s) \qquad d(s) = \sin s \cos s/(2-\cos^2 s).$$

$$\Sigma = \begin{array}{c|ccccc|c} & x_1 & x_2 & x_3 & u_1 & u_2 & c_i \\ \hline f_1 & 2 & \boxed{0} & \boxed{1} & 0^\bullet & 0 & 0 \\ f_2 & \boxed{1} & 2^\bullet & \boxed{1} & 0 & 0 & 0 \\ f_3 & \boxed{1} & \boxed{0} & 2 & 0 & 0^\bullet & 0 \\ f_4 & 0 & & 0^\bullet & & & 2 \\ f_5 & 0^\bullet & & 0 & & & 2 \\ \hline d_j & 2 & 2 & 2 & 0 & 0 & \end{array} \quad \mathrm{Val}(\Sigma) = 2$$

$$\mathbf{J} = \begin{array}{c|ccccc|} & x_1 & x_2 & x_3 & u_1 & u_2 \\ \hline f_1 & 1 & & & -a_3 & a_3 \\ f_2 & & 1 & & a_3 & -1-a_3 \\ f_3 & & & 1 & a_3+b_3 & -a_3-b_3 \\ f_4 & \frac{\partial f_4}{\partial x_1} & & \frac{\partial f_4}{\partial x_3} & & \\ f_5 & \frac{\partial f_5}{\partial x_1} & & \frac{\partial f_5}{\partial x_3} & & \end{array} \quad \det(\mathbf{J}) = 0$$

Here

$$\begin{aligned} \partial f_4/\partial x_1 &= -\sin x_1 - \sin(x_1+x_3) & a_3 &= a(x_3) = 2/(2-\cos^2 x_3) \\ \partial f_4/\partial x_3 &= -\sin(x_1+x_3) & b_3 &= b(x_3) = \cos x_3/(2-\cos^2 x_3) \\ \partial f_5/\partial x_1 &= \cos x_1 + \cos(x_1+x_3) \\ \partial f_5/\partial x_3 &= \cos(x_1+x_3). \end{aligned} \tag{B.2}$$

SA reports structural index 3, while the d-index is 5.

B.1.1 ES method

Pryce [26] fixes this failure by introducing a new variable w and substituting it for $u_1 - u_2$ in f_1 and f_3. These two equations become $\overline{f}_1$ and $\overline{f}_3$. We append $g = w - (u_1 - u_2)$ that prescribes these substitutions and obtain the converted DAE:

$$
\begin{aligned}
0 = \overline{f}_1 =& x_1'' - \Big[2c(x_3)(x_1'+x_3')^2 + x_1'^2 d(x_3) \\
&+ (2x_3 - x_2)\big(a(x_3)+2b(x_3)\big) + a(x_3)w\Big] \\
0 = f_2 =& x_2'' - \Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) \\
&+ (2x_3 - x_2)\big(1-3a(x_3)-2b(x_3)\big) - a(x_3)u_1 + \big(a(x_3)+1\big)u_2\Big] \\
0 = \overline{f}_3 =& x_3'' - \Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) + (2x_3 - x_2)\big(a(x_3)-9b(x_3)\big) \\
&- 2x_1'^2 c(x_3) - d(x_3)\big(x_1'+x_3'\big)^2 - \big(a(x_3)+b(x_3)\big)w\Big] \\
0 = f_4 =& \cos x_1 + \cos(x_1+x_3) - p_1(t) \\
0 = f_5 =& \sin x_1 + \sin(x_1+x_3) - p_2(t) \\
0 = g =& w - (u_1 - u_2).
\end{aligned}
\tag{B.3}
$$

$$
\overline{\Sigma} = \begin{array}{c|cccccc|c}
 & x_1 & x_2 & x_3 & u_1 & u_2 & w & c_i \\
\hline
\overline{f}_1 & 2 & 0^\bullet & \boxed{1} & & & 0 & 2 \\
f_2 & \boxed{1} & 2 & \boxed{1} & 0^\bullet & 0 & & 0 \\
\overline{f}_3 & \boxed{1} & 0 & 2 & & & 0^\bullet & 2 \\
f_4 & 0 & & 0^\bullet & & & & 4 \\
f_5 & 0^\bullet & & 0 & & & & 4 \\
g & & & & 0 & 0^\bullet & \boxed{0} & 0 \\
\hline
d_j & 4 & 2 & 4 & 0 & 0 & 2 & \mathrm{Val}(\overline{\Sigma}) = 0
\end{array}
$$

$$
\overline{\mathbf{J}} = \begin{array}{c|cccccc}
 & x_1 & x_2 & x_3 & u_1 & u_2 & w \\
\hline
\overline{f}_1 & 1 & a_3+2b_3 & & & & -a_3 \\
f_2 & & 1 & & a_3 & -a_3-1 & \\
\overline{f}_3 & & a_3-9b_3 & 1 & & & a_3+b_3 \\
f_4 & \frac{\partial f_4}{\partial x_1} & & \frac{\partial f_4}{\partial x_3} & & & \\
f_5 & \frac{\partial f_5}{\partial x_1} & & \frac{\partial f_5}{\partial x_3} & & & \\
g & & & & -1 & 1 &
\end{array}
$$

$$
\det(\overline{\mathbf{J}}) = -2\sin x_3(a_3^2 - 3a_3b_3 + b_3^2)
$$

$\overline{\mathbf{J}}$ is not identically singular; refer to (B.2) for the entries in it. SA reports the correct index 5, and succeeds if $\det(\overline{\mathbf{J}}) \neq 0$.

B.1.2 LC method

We replace f_3 by

$$
\begin{aligned}
\overline{f}_3 &= f_1 + \frac{a_3}{a_3+b_3} f_3 \\
&= x_1'' - \Big[2c(x_3)(x_1'+x_3')^2 + x_1'^2 d(x_3) + (2x_3 - x_2)\big(a_3+2b_3\big) + \underline{\underline{a_3(u_1-u_2)}}\Big] \\
&\qquad + \frac{a_3}{a_3+b_3} x_3'' - \frac{a_3}{\underline{\underline{a_3+b_3}}}\Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) + (2x_3-x_2)\big(a_3-9b_3\big) \\
&\qquad - 2x_1'^2 c(x_3) - d(x_3)\big(x_1'+x_3'\big)^2 \underline{\underline{-\big(a_3+b_3\big)(u_1-u_2)}}\Big] \\
&= x_1'' - \Big[2c(x_3)(x_1'+x_3')^2 + x_1'^2 d(x_3) + (2x_3-x_2)\big(a_3+2b_3\big)\Big] + \frac{a_3}{a_3+b_3} x_3'' \\
&\qquad - \frac{a_3}{a_3+b_3}\Big[-2c(x_3)(x_1'+x_3')^2 - x_1'^2 d(x_3) + (2x_3-x_2)\big(a_3-9b_3\big) \\
&\qquad - 2x_1'^2 c(x_3) - d(x_3)\big(x_1'+x_3'\big)^2\Big]
\end{aligned}
$$

(the underlined terms cancel out).

$$
\overline{\Sigma} =
\begin{array}{c}
 \\ f_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ f_5 \\ d_j
\end{array}
\begin{array}{ccccc|c}
x_1 & x_2 & x_3 & u_1 & u_2 & c_i \\ \hline
2 & 0 & 1 & 0^\bullet & 0 & 0 \\
1 & 2^\bullet & 1 & 0 & 0^\bullet & 0 \\
2 & 0^\bullet & 2 & & & 2 \\
0^\bullet & & 0 & & & 4 \\
0 & & 0^\bullet & & & 4 \\ \hline
4 & 2 & 4 & 0 & 0 & \mathrm{Val}(\overline{\Sigma}) = 0
\end{array}
\qquad
\overline{\mathbf{J}} =
\begin{array}{c}
 \\ f_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ f_5
\end{array}
\begin{array}{ccccc}
x_1 & x_2 & x_3 & u_1 & u_2 \\ \hline
 & & & -a_3 & a_3 \\
 & 1 & & a_3 & -1-a_3 \\
1 & \frac{\partial \overline{f}_3}{\partial x_2} & \frac{a_3}{a_3+b_3} & & \\
\frac{\partial f_4}{\partial x_1} & & \frac{\partial f_4}{\partial x_3} & & \\
\frac{\partial f_5}{\partial x_1} & & \frac{\partial f_5}{\partial x_3} & &
\end{array}
$$

Here

$$
\frac{\partial \overline{f}_3}{\partial x_2} = a_3 + 2b_3 + \frac{a_3}{a_3+b_3}(a_3 - 9b_3),
$$

$$
\det(\overline{\mathbf{J}}) = -2\sin x_3 (a_3^2 - 3a_3b_3 + b_3^2) a_3/(a_3+b_3).
$$

Refer to (B.2) for the other entries in $\overline{\mathbf{J}}$. Since

$$
\frac{a_3}{a_3+b_3} = \frac{2}{2+\cos x_3} \neq 0 \quad \text{for all } x_3 \in \mathbb{R},
$$

the converted DAE is always equivalent to (B.1). $\bar{\mathbf{J}}$ is not identically singular. SA reports index 5, and succeeds if $\det(\bar{\mathbf{J}}) \neq 0$.

B.2 Transistor amplifier

Below is a transistor amplifier problem originated from electrical circuit analysis [18]. It is classified in [18] as a stiff index-1 DAE consisting of 8 equations.

$$\begin{aligned}
0 = f_1 &= \quad C_1(x_1' - x_2') + \frac{x_1 - U_e(t)}{R_0} \\
0 = f_2 &= -C_1(x_1' - x_2') - \frac{U_b}{R_2} + x_2\left(\frac{1}{R_1} + \frac{1}{R_2}\right) - (\alpha - 1)g(x_2 - x_3) \\
0 = f_3 &= \quad C_2 x_3' - g(x_2 - x_3) + \frac{x_3}{R_3} \\
0 = f_4 &= \quad C_3(x_4' - x_5') + \frac{x_4 - U_b}{R_4} + \alpha g(x_2 - x_3) \\
0 = f_5 &= -C_3(x_4' - x_5') - \frac{U_b}{R_5} + x_5\left(\frac{1}{R_5} + \frac{1}{R_6}\right) - (\alpha - 1)g(x_5 - x_6) \\
0 = f_6 &= \quad C_4 x_6' - g(x_5 - x_6) + \frac{x_6}{R_7} \\
0 = f_7 &= \quad C_5(x_7' - x_8') + \frac{x_7 - U_b}{R_8} + \alpha g(x_5 - x_6) \\
0 = f_8 &= -C_5(x_7' - x_8') + \frac{x_8}{R_9},
\end{aligned} \tag{B.4}$$

where

$$\begin{aligned}
g(y) &= \beta\big(\exp(y/U_F) - 1\big) & U_e(t) &= 0.1\sin(200\pi t) & & \\
U_b &= 6.0 & R_0 &= 1000 & & \\
U_F &= 0.026 & R_k &= 9000 & &\text{for } k = 1,\ldots,9 \\
\alpha &= 0.99 & C_k &= k \times 10^{-6} & &\text{for } k = 1,\ldots,5 \\
\beta &= 10^{-6}. & & & &
\end{aligned}$$

$$
\Sigma = \begin{array}{c|cccccccc|c}
 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & c_i \\
\hline
f_1 & 1 & 1^\bullet & & & & & & & 0 \\
f_2 & 1^\bullet & 1 & \boxed{0} & & & & & & 0 \\
f_3 & & \boxed{0} & 1^\bullet & & & & & & 0 \\
f_4 & & \boxed{0} & \boxed{0} & 1 & 1^\bullet & & & & 0 \\
f_5 & & & & 1^\bullet & 1 & \boxed{0} & & & 0 \\
f_6 & & & & & \boxed{0} & 1^\bullet & & & 0 \\
f_7 & & & & & \boxed{0} & \boxed{0} & 1 & 1^\bullet & 0 \\
f_8 & & & & & & & 1^\bullet & 1 & 0 \\
\hline
d_j & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \mathrm{Val}(\Sigma) = 8
\end{array}
$$

$$
\mathbf{J} = \begin{array}{c}
 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \end{array} \\
\begin{bmatrix}
C_1 & -C_1 & & & & & & \\
-C_1 & C_1 & & & & & & \\
 & & C_2 & & & & & \\
 & & & C_3 & -C_3 & & & \\
 & & & -C_3 & C_3 & & & \\
 & & & & & C_4 & & \\
 & & & & & & C_5 & -C_5 \\
 & & & & & & -C_5 & C_5
\end{bmatrix}
\end{array}.
$$

$$\det(\mathbf{J}) = 0$$

SA reports index 1, but produces an identically singular $\mathbf{J}$. Observing its structure, we

replace	by
f_1	$\overline{f}_1 = f_1 + f_2$
f_4	$\overline{f}_4 = f_4 + f_5$
f_7	$\overline{f}_7 = f_7 + f_8$.

The new equations in the converted DAE are

$$
\begin{aligned}
0 = \overline{f}_1 &= \frac{x_1 - U_e(t)}{R_0} - \frac{U_b}{R_2} + x_2\left(\frac{1}{R_1} + \frac{1}{R_2}\right) - (\alpha - 1)g(x_2 - x_3)\\
0 = \overline{f}_4 &= \frac{x_4 - U_b}{R_4} + \alpha g(x_2 - x_3) - \frac{U_b}{R_5} + x_5\left(\frac{1}{R_5} + \frac{1}{R_6}\right) - (\alpha - 1)g(x_5 - x_6)\\
0 = \overline{f}_7 &= \frac{x_7 - U_b}{R_8} + \alpha g(x_5 - x_6) + \frac{x_8}{R_9}.
\end{aligned}
$$

$$
\overline{\Sigma} = \begin{array}{c|cccccccc|c}
 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & c_i \\
\hline
\overline{f}_1 & 0^\bullet & 0 & 0 & & & & & & 1 \\
f_2 & 1 & 1^\bullet & \colorbox{lightgray}{0} & & & & & & 0 \\
f_3 & & \colorbox{lightgray}{0} & 1^\bullet & & & & & & 0 \\
\overline{f}_4 & & 0 & 0 & 0^\bullet & 0 & 0 & & & 1 \\
f_5 & & & & 1 & 1^\bullet & \colorbox{lightgray}{0} & & & 0 \\
f_6 & & & & & \colorbox{lightgray}{0} & 1^\bullet & & & 0 \\
\overline{f}_7 & & & & & 0 & 0 & 0^\bullet & 0 & 1 \\
f_8 & & & & & & & 1 & 1^\bullet & 0 \\
\hline
d_j & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \mathrm{Val}(\overline{\Sigma}) = 5
\end{array}
$$

$$
\overline{\mathbf{J}} = \begin{array}{c|cccccccc|}
 & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\
\hline
\overline{f}_1 & R_0^{-1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} & & & & & \\
f_2 & -C_1 & C_1 & & & & & & \\
f_3 & & & C_2 & & & & & \\
\overline{f}_4 & & \frac{\partial f_4}{\partial f_2} & \frac{\partial f_4}{\partial x_3} & R_4^{-1} & \frac{\partial f_4}{\partial x_5} & \frac{\partial f_4}{\partial x_6} & & \\
f_5 & & & & -C_3 & C_3 & & & \\
f_6 & & & & & & C_4 & & \\
\overline{f}_7 & & & & & \frac{\partial f_7}{\partial x_5} & \frac{\partial f_7}{\partial x_6} & R_8^{-1} & R_9^{-1} \\
f_8 & & & & & & & -C_5 & C_5
\end{array}
$$

$$
\det(\overline{\mathbf{J}}) = C_1C_2C_3C_4C_5\left(R_0^{-1} + \frac{\partial f_1}{\partial x_2}\right)\left(R_4^{-1} + \frac{\partial f_4}{\partial x_5}\right)\left(R_8^{-1} + R_9^{-1}\right)
$$

Note that only two partial derivatives shown in $\overline{\mathbf{J}}$ contribute to $\det(\overline{\mathbf{J}})$:

$$\partial f_4/\partial x_5 = R_5^{-1} + R_6^{-1}$$
$$\partial f_1/\partial x_2 = R_1^{-1} + R_2^{-1}.$$

SA still reports index 1. Since $\overline{\mathbf{J}}$ is not identically singular, SA succeeds if $\det(\overline{\mathbf{J}}) \neq 0$.

B.3 Ring modulator

Following is a ring modulator problem originated from electrical circuit analysis [18]. By setting $C_s = 0$ in the original problem formulation, we obtain an index-2 DAE consisting of 11 differential and 4 algebraic equations:

$$\begin{aligned}
0 = f_1 &= -y_1' + C^{-1}\big(y_8 - 0.5y_{10} + 0.5y_{11} + y_{14} - R^{-1}y_1\big)\\
0 = f_2 &= -y_2' + C^{-1}\big(y_9 - 0.5y_{11} + 0.5y_{12} + y_{15} - R^{-1}y_2\big)\\
0 = f_3 &= \quad y_{10} - q(U_{D1}) + q(U_{D4})\\
0 = f_4 &= -y_{11} + q(U_{D2}) - q(U_{D3})\\
0 = f_5 &= \quad y_{12} + q(U_{D1}) - q(U_{D3})\\
0 = f_6 &= -y_{13} - q(U_{D2}) + q(U_{D4})\\
0 = f_7 &= -y_7' + C_p^{-1}\big(-R_p^{-1}y_7 + q(U_{D1}) + q(U_{D2}) - q(U_{D3}) - q(U_{D4})\big)\\
0 = f_8 &= -y_8' + -L_h^{-1}y_1\\
0 = f_9 &= -y_9' + -L_h^{-1}y_2\\
0 = f_{10} &= -y_{10}' + L_{s2}^{-1}(0.5y_1 - y_3 - R_{g2}y_{10})\\
0 = f_{11} &= -y_{11}' + L_{s3}^{-1}(-0.5y_1 + y_4 - R_{g3}y_{11})\\
0 = f_{12} &= -y_{12}' + L_{s2}^{-1}(0.5y_2 - y_5 - R_{g2}y_{12})\\
0 = f_{13} &= -y_{13}' + L_{s3}^{-1}(-0.5y_2 + y_6 - R_{g3}y_{13})\\
0 = f_{14} &= -y_{14}' + L_{s1}^{-1}(-y_1 + U_{in1}(t) - (R_i + R_{g1})y_{14})\\
0 = f_{15} &= -y_{15}' + L_{s1}^{-1}(-y_2 - (R_c + R_{g1})y_{15}),
\end{aligned} \tag{B.5}$$

where

$$\begin{aligned}
U_{D1} &= \quad y_3 - y_5 - y_7 - U_{in2}(t) & q(U) &= \gamma(e^{\delta U} - 1)\\
U_{D2} &= -y_4 + y_6 - y_7 - U_{in2}(t) & U_{in1}(t) &= 0.5\sin(2000\pi t)\\
U_{D3} &= \quad y_4 + y_5 + y_7 + U_{in2}(t) & U_{in2}(t) &= 2\sin(20000\pi t)\\
U_{D4} &= -y_3 - y_6 + y_7 + U_{in2}(t). & &
\end{aligned}$$

The parameters are

$$
\begin{aligned}
C &= 1.6 \times 10^{-8} & R_{g1} &= 36.3 \\
C_p &= 10^{-8} & R_{g2} &= 17.3 \\
R &= 25 \times 10^{3} & R_{g3} &= 17.3 \\
R_p &= 50 & R_i &= 5 \times 10 \\
L_h &= 4.45 & R_c &= 6 \times 10^{2} \\
L_{s1} &= 2 \times 10^{-3} & \gamma &= 40.67286402 \times 10^{-9} \\
L_{s2} &= 5 \times 10^{-4} & \delta &= 17.7493332 \\
L_{s3} &= 5 \times 10^{-4}.
\end{aligned}
$$

SA reports index 1 and produces the following Σ with $\mathrm{Val}(\Sigma) = 11$.

$\Sigma =$

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	c_i
f_1	$1^\bullet$							0		0	0			0		0
f_2		$1^\bullet$							0			0	0		0	0
f_3			0		0	$0^\bullet$	0			0						0
f_4				0	$0^\bullet$	0	0				0					0
f_5			$0^\bullet$	0	0		0					0				0
f_6			0	$0^\bullet$		0	0						0			0
f_7			0	0	0	0	$1^\bullet$									0
f_8	0							$1^\bullet$								0
f_9		0							$1^\bullet$							0
f_{10}	0		0							$1^\bullet$						0
f_{11}	0			0							$1^\bullet$					0
f_{12}		0			0							$1^\bullet$				0
f_{13}		0				0							$1^\bullet$			0
f_{14}	0													$1^\bullet$		0
f_{15}		0													$1^\bullet$	0
d_j	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	

We do not present **J** here. The entries that contribute to its determinant are positions (i,i) for

$i = 1,2,7:15$, where $\partial f_i / \partial y_i' = -1$, and the submatrix

$$\mathbf{J}(3:6,3:6) = \begin{array}{c} \\ f_3 \\ f_4 \\ f_5 \\ f_6 \end{array} \begin{array}{c} \begin{array}{cccc} y_3 & y_4 & y_5 & y_6 \end{array} \\ \begin{bmatrix} -s_1 - s_4 & & s_1 & -s_4 \\ & -s_2 - s_3 & -s_3 & s_2 \\ s_1 & -s_3 & -s_1 - s_3 & \\ -s_4 & s_2 & & -s_2 - s_4 \end{bmatrix} \end{array}, \quad \text{where } s_i = \gamma \delta e^{\delta U_{Di}}.$$

Since this submatrix of $\mathbf{J}$ is identically singular, so is $\mathbf{J}$.

To remedy this DAE, we replace one equation from f_3, f_4, f_5, f_6 by

$$\overline{f} = f_3 - f_4 + f_5 - f_6 = y_{10} + y_{11} + y_{12} + y_{13}.$$

Consider the case f_3. SA produces $\overline{\Sigma}$ with $\text{Val}(\overline{\Sigma}) = 10$:

$\overline{\Sigma} =$

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	c_i
f_1	$1^\bullet$							$\boxed{0}$		$\boxed{0}$	$\boxed{0}$			$\boxed{0}$		0
f_2		$1^\bullet$							$\boxed{0}$			$\boxed{0}$	$\boxed{0}$		$\boxed{0}$	0
$\overline{f}_3$										0	0	0	$0^\bullet$			1
f_4				$0^\bullet$	0	0	$\boxed{0}$				$\boxed{0}$					0
f_5			0	0	$0^\bullet$		$\boxed{0}$					$\boxed{0}$				0
f_6			$0^\bullet$	0		0	$\boxed{0}$						$\boxed{0}$			0
f_7			0	0	0	0	$1^\bullet$									0
f_8	$\boxed{0}$							$1^\bullet$								0
f_9		$\boxed{0}$							$1^\bullet$							0
f_{10}	$\boxed{0}$		0							$1^\bullet$						0
f_{11}	$\boxed{0}$			0							$1^\bullet$					0
f_{12}		$\boxed{0}$			0							$1^\bullet$				0
f_{13}		$\boxed{0}$				$0^\bullet$							1			0
f_{14}	$\boxed{0}$													$1^\bullet$		0
f_{15}		$\boxed{0}$													$1^\bullet$	0
d_j	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	

The consistent initial values for solving this converted DAE are

$$\begin{aligned} y_i &= 0 && \text{for } i = 1:15, \\ y_i' &= 0 && \text{for } i = 1,2,7:15, \end{aligned}$$

which satisfy f_i for all i and f_3'. At this consistent point, $\det(\mathbf{J}) = -1.2040 \times 10^{-14}$. Hence SA succeeds and the DAE is index-2.

B.4 An index-overestimated DAE

Reißig et al. [31] construct a family of linear constant coefficient DAEs

$$Ax'(t) + Bx(t) - q(t) = 0 \tag{B.6}$$

for which SA finds an arbitrarily high structural index $\nu_S > 1$, though the d-index is 1. Here B is an $n \times n$ identity matrix, where $n = 2k+1$ and $k \geq 1$; A has the form

$$A = \begin{pmatrix} 0 & 1 & 1 & & & & & & \\ & 1 & 1 & 0 & & & & & \\ & & 0 & 1 & 1 & & & & \\ & & & 1 & 1 & 0 & & & \\ & & & & 0 & \ddots & \ddots & & \\ & & & & & \ddots & \ddots & 0 & \\ & & & & & & 0 & 1 & 1 \\ & & & & & & & 1 & 1 \\ & & & & & & & & 0 \end{pmatrix}.$$

That is, for $i = 1:k$, equations f_{2i-1} and f_{2i} have the common expression $x_{2i}' + x_{2i+1}'$.

Pryce [27] applies the Σ-method on (B.6) with $n = 5$ and $k = 2$:

$$\begin{aligned} 0 = f_1 &= x_2' + x_3' + x_1 - q_1(t) \\ 0 = f_2 &= x_2' + x_3' + x_2 - q_2(t) \\ 0 = f_3 &= x_4' + x_5' + x_3 - q_3(t) \\ 0 = f_4 &= x_4' + x_5' + x_4 - q_4(t) \\ 0 = f_5 &= x_5 - q_5(t). \end{aligned} \tag{B.7}$$

$$
\Sigma = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ d_j \end{array}
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \\
\left[\begin{array}{ccccc}
0^\bullet & 1 & 1 & & \\
 & 1^\bullet & 1 & & \\
 & & 0^\bullet & 1 & 1 \\
 & & & 1^\bullet & 1 \\
 & & & & 0^\bullet
\end{array}\right] \\
\begin{array}{ccccc} 0 & 1 & 1 & 2 & 2 \end{array}
\end{array}
\begin{array}{c} c_i \\ 0 \\ 0 \\ 1 \\ 1 \\ 2 \\ \mathrm{Val}(\Sigma) = 2 \end{array}
\qquad
\mathbf{J} = \begin{array}{c} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & 1 & 1 & & \\
 & 1 & 1 & & \\
 & & 1 & 1 & 1 \\
 & & & 1 & 1 \\
 & & & & 1
\end{array}\right] \\
\det(\mathbf{J}) = 1
\end{array}
$$

The method succeeds with $\det(\mathbf{J}) = 1$ but reports $\nu_S = 3$ different from $\nu_d = 1$; this occurs in Pantelides's SA as well. We illustrate below how to fix this index overestimation problem for (B.7).

Observing the structure of A, we

replace	by
f_1	$\overline{f}_1 = f_1 - f_2$
f_3	$\overline{f}_3 = f_3 - f_4$.

The converted DAE is

$$
\begin{aligned}
0 &= \overline{f}_1 = x_1 - x_2 - q_1(t) + q_2(t) \\
0 &= f_2 = x_2' + x_3' + x_2 - q_2(t) \\
0 &= \overline{f}_3 = x_3 - x_4 - q_3(t) + q_4(t) \\
0 &= f_4 = x_4' + x_5' + x_4 - q_4(t) \\
0 &= f_5 = x_5 - q_5(t).
\end{aligned}
\tag{B.8}
$$

$$
\overline{\Sigma} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ f_5 \\ d_j \end{array}
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \\
\left[\begin{array}{ccccc}
0^\bullet & 0 & & & \\
 & 1^\bullet & 1 & & \\
 & & 0^\bullet & 0 & \\
 & & & 1^\bullet & 1 \\
 & & & & 0^\bullet
\end{array}\right] \\
\begin{array}{ccccc} 1 & 1 & 1 & 1 & 1 \end{array}
\end{array}
\begin{array}{c} c_i \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ \mathrm{Val}(\overline{\Sigma}) = 2 \end{array}
\qquad
\overline{\mathbf{J}} = \begin{array}{c} \\ \overline{f}_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ f_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \end{array} \\
\left[\begin{array}{ccccc}
1 & -1 & & & \\
 & 1 & 1 & & \\
 & & 1 & -1 & \\
 & & & 1 & 1 \\
 & & & & 1
\end{array}\right] \\
\det(\overline{\mathbf{J}}) = 1
\end{array}
$$

Since no $d_j = 0$, SA reports index $\nu_S = \max_i c_i = 1$. Note that here we *do not* choose the canonical offsets

$$\mathbf{c} = (0,0,1,0,1) \quad \text{and} \quad \mathbf{d} = (0,1,1,1,1),$$

which still give an overestimated structural index $\nu_S = 2$ as $d_1 = 0$ and $c_3 = c_5 = 1$.

Consider for general case $k \geq 1$. The DAE is

$$\begin{aligned} 0 = f_{2i-1} &= x'_{2i} + x'_{2i+1} + x_{2i-1} - q_{2i-1}(t) && i = 1:k \\ 0 = f_{2i} &= x'_{2i} + x'_{2i+1} + x_{2i} - q_{2i}(t) && i = 1:k \\ 0 = f_{2k+1} &= x_{2k+1} - q_{2k+1}(t). \end{aligned} \tag{B.9}$$

$$\Sigma = \begin{array}{c|cccccccc|c} & x_1 & x_2 & x_3 & x_4 & x_5 & \cdots & x_{2k} & x_{2k+1} & c_i \\ \hline f_1 & 0^\bullet & 1 & 1 & & & & & & 0 \\ f_2 & & 1^\bullet & 1 & & & & & & 0 \\ f_3 & & & 0^\bullet & 1 & 1 & & & & 1 \\ f_4 & & & & 1^\bullet & 1 & & & & 1 \\ \vdots & & & & & 0^\bullet & \ddots & & & \vdots \\ f_{2k-1} & & & & & & \ddots & 1 & 1 & k-1 \\ f_{2k} & & & & & & & 1^\bullet & 1 & k-1 \\ f_{2k+1} & & & & & & & & 0^\bullet & k \\ \hline d_j & 0 & 1 & 1 & 2 & 2 & \cdots & k-1 & k & \mathrm{Val}(\Sigma) = k \end{array}$$

$$
\mathbf{J} = \begin{array}{r} \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ \vdots \\ f_{2k-1} \\ f_{2k} \\ f_{2k+1} \end{array}
\begin{array}{c} \begin{array}{cccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & \cdots & x_{2k} & x_{2k+1} \end{array} \\
\left[\begin{array}{cccccccc}
1 & 1 & 1 & & & & & \\
 & 1 & 1 & & & & & \\
 & & 1 & 1 & 1 & & & \\
 & & & 1 & 1 & & & \\
 & & & & \ddots & \ddots & & \\
 & & & & & 1 & 1 & 1 \\
 & & & & & & 1 & 1 \\
 & & & & & & & 1
\end{array}\right] \end{array}
$$

$$\det(\mathbf{J}) = 1$$

SA succeeds and reports structural index $k+1$.

To remedy this index overestimation, we repeat the same strategy used above: for $i = 1:k$, we replace f_{2i-1} by $\overline{f}_{2i-1} = f_{2i-1} - f_{2i}$. The converted DAE is

$$
\begin{aligned}
0 &= \overline{f}_{2i-1} = x_{2i-1} - x_{2i} - q_{2i-1}(t) + q_{2i}(t) && i = 1:k \\
0 &= \quad f_{2i} = x'_{2i} + x'_{2i+1} + x_{2i} - q_{2i}(t) && i = 1:k \\
0 &= f_{2k+1} = x_{2k+1} - q_{2k+1}(t).
\end{aligned}
\tag{B.10}
$$

$$
\overline{\Sigma} = \begin{array}{r} \\ \overline{f}_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ \vdots \\ \overline{f}_{2k-1} \\ f_{2k} \\ f_{2k+1} \\ d_j \end{array}
\begin{array}{c} \begin{array}{cccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & \cdots & x_{2k} & x_{2k+1} \end{array} \\
\left[\begin{array}{cccccccc}
0^\bullet & 0 & & & & & & \\
 & 1^\bullet & 1 & & & & & \\
 & & 0^\bullet & 0 & & & & \\
 & & & 1^\bullet & 1 & & & \\
 & & & & 0^\bullet & \ddots & & \\
 & & & & & \ddots & 0 & \\
 & & & & & & 1^\bullet & 1 \\
 & & & & & & & 0^\bullet
\end{array}\right] \\
\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 1 & \cdots & 1 & 1 \end{array} \end{array}
\begin{array}{l} c_i \\ 1 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 1 \\ 0 \\ 1 \\ \mathrm{Val}(\overline{\Sigma}) = k \end{array}
$$

$$
\overline{\mathbf{J}} = \begin{array}{r} \\ \overline{f}_1 \\ f_2 \\ \overline{f}_3 \\ f_4 \\ \vdots \\ \overline{f}_{2k-1} \\ f_{2k} \\ f_{2k+1} \end{array}
\begin{array}{c}
\begin{array}{cccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & \cdots & x_{2k} & x_{2k+1} \end{array} \\
\left[\begin{array}{cccccccc}
1 & -1 & & & & & & \\
 & 1 & 1 & & & & & \\
 & & 1 & -1 & & & & \\
 & & & 1 & 1 & & & \\
 & & & & \ddots & \ddots & & \\
 & & & & & 1 & -1 & \\
 & & & & & & 1 & 1 \\
 & & & & & & & 1
\end{array}\right]
\end{array}
$$

$$
\det(\overline{\mathbf{J}}) = 1
$$

Now SA reports the correct $\nu_S = 1$ on the converted DAE (B.10). Again, we use non-canonical offsets in $\overline{\Sigma}$, while $d_1 = c_1 = 0$ in the canonical case.

Bibliography

[1] U. M. ASCHER AND L. R. PETZOLD, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia, 1998.

[2] R. BARRIO, *Performance of the Taylor series method for ODEs/DAEs*, Appl. Math. Comp., 163 (2005), pp. 525–545.

[3] K. BRENAN, S. CAMPBELL, AND L. PETZOLD, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM, Philadelphia, second ed., 1996.

[4] S. L. CAMPBELL AND C. W. GEAR, *The index of general nonlinear DAEs*, Numerische Mathematik, 72 (1995), pp. 173–196.

[5] S. L. CAMPBELL AND E. GRIEPENTROG, *Solvability of general differential algebraic equations*, SIAM J. Sci. Comput., 16 (1995), pp. 257–270.

[6] S. CHOWDHRY, H. KRENDL, AND A. A. LINNINGER, *Symbolic numeric index analysis algorithm for differential algebraic equations*, Industrial & engineering chemistry research, 43 (2004), pp. 3886–3894.

[7] I. S. DUFF AND C. W. GEAR, *Computing the structural index*, SIAM Journal on Algebraic and Discrete Methods, 7 (1986), p. 594603.

[8] D. ESTVEZ SCHWARZ AND R. LAMOUR, *Diagnosis of singular points of properly stated daes using automatic differentiation*, Numerical Algorithms, (2015), pp. 1–29.

[9] C. W. GEAR, *Differential-algebraic equation index transformations*, SIAM Journal on Scientific and Statistical Computing, 9 (1988), pp. 39–47.

[10] ——, *Differential algebraic equations, indices, and integral algebraic equations*, SIAM Journal on Numerical Analysis, 27 (1990), pp. 1527–1534.

[11] E. GRIEPENTROG AND R. MÄRZ, *Differential-algebraic equations and their numerical treatment*, in Teubner-Texte zur Mathematik [Teubner Texts in Mathematics], 88., BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1986. With German, French and Russian summaries.

[12] A. GRIEWANK AND A. WALTHER, *On the efficient generation of Taylor expansions for DAE solutions by automatic differentiation*, in Applied Parallel Computing. State of the Art in Scientific Computing, Springer, 2006, pp. 1089–1098.

[13] E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations II. Stiff and Differential– Algebraic Problems*, Springer Verlag, Berlin, 1991.

[14] A. C. HINDMARSH, P. N. BROWN, K. E. GRANT, S. L. LEE, R. SERBAN, D. E. SHUMAKER, AND C. S. WOODWARD, *Sundials: Suite of nonlinear and differential/algebraic equation solvers*, ACM Trans. Math. Softw., 31 (2005), pp. 363–396.

[15] J. HOEFKENS, *Rigorous Numerical Analysis with High-Order Taylor Models*, PhD thesis, Department of Mathematics and Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, August 2001.

[16] P. KUNKEL AND V. MEHRMANN, *Differential-Algebraic Equations — Analysis and Numerical Solution*, EMS Publishing House, Zürich, Switzerland, 2006.

[17] S. E. MATTSSON AND G. SÖDERLIND, *Index reduction in differential-algebraic equations using dummy derivatives*, SIAM J. Sci. Comput., 14 (1993), pp. 677–692.

[18] F. MAZZIA AND F. IAVERNARO, *Test set for initial value problem solvers*, Tech. Rep. 40, Department of Mathematics, University of Bari, Italy, 2003. `http://pitagora.dm.uniba.it/~testset/`.

[19] R. MCKENZIE, J. D. PRYCE, G. TAN, AND N. S. NEDIALKOV, *Structural analysis and dummy derivatives for differential-algebraic equations*, 2015. In preparation.

[20] N. S. NEDIALKOV AND J. D. PRYCE, *Solving differential-algebraic equations by Taylor series (I): Computing Taylor coefficients*, BIT Numerical Mathematics, 45 (2005), pp. 561–591.

[21] N. S. NEDIALKOV AND J. D. PRYCE, *Solving differential-algebraic equations by Taylor series (II): Computing the system Jacobian*, BIT Numerical Mathematics, 47 (2007), pp. 121–135.

[22] ——, *Solving differential algebraic equations by Taylor series (III): the DAETS code*, JNAIAM J. Numer. Anal. Indust. Appl. Math, 3 (2008), pp. 61–80.

[23] N. S. NEDIALKOV, J. D. PRYCE, AND G. TAN, *DAESA — a Matlab tool for structural analysis of DAEs: Software*, ACM Transactions on Mathematical Software, to appear (2015). 15 pages.

[24] N. S. NEDIALKOV, G. TAN, AND J. D. PRYCE, *Exploiting fine block triangularization and quasilinearity in differential-algebraic equation systems*, arXiv:1411.4128, (2014). 18 pages.

[25] C. C. PANTELIDES, *The consistent initialization of differential-algebraic systems*, SIAM. J. Sci. Stat. Comput., 9 (1988), pp. 213–231.

[26] J. D. PRYCE, *Solving high-index DAEs by Taylor Series*, Numerical Algorithms, 19 (1998), pp. 195–211.

[27] J. D. PRYCE, *A simple structural analysis method for DAEs*, BIT Numerical Mathematics, 41 (2001), pp. 364–394.

[28] J. D. PRYCE, N. S. NEDIALKOV, AND G. TAN, *Graph theory, irreducibility, and structural analysis of differential-algebraic equation systems*, arXiv:1411.4129, (2014). 24 pages.

[29] J. D. PRYCE, N. S. NEDIALKOV, AND G. TAN, *DAESA — a Matlab tool for structural analysis of DAEs: Theory*, ACM Transactions on Mathematical Software, to appear (2015). 20 pages.

[30] P. J. RABIER AND W. C. RHEINBOLDT, *A general existence and uniqueness theory for implicit differential-algebraic equations*, Differential Integral Equations, 4 (1991), pp. 563–582.

[31] G. REISSIG, W. S. MARTINSON, AND P. I. BARTON, *Differential–algebraic equations of index 1 may have an arbitrarily high structural index*, SIAM J. Sci. Comput., 21 (1999), pp. 1987–1990.

[32] W. C. RHEINBOLDT, *Differential-algebraic systems as differential equations on manifolds*, Mathematics of computation, 43 (1984), pp. 473–482.

[33] L. SCHOLZ AND A. STEINBRECHER, *A combined structural-algebraic approach for the regularization of coupled systems of DAEs*, Tech. Rep. 30, Reihe des Instituts für Mathematik Technische Universität Berlin, Berlin, Germany, 2013.

[34] R. D. P. SOARES AND A. R. SECCHI, *Structural analysis for static and dynamic models*, Mathematical and Computer Modelling, 55 (2012), pp. 1051–1067.

[35] J. UNGER, A. KRÖNER, AND W. MARQUARDT, *Structural analysis of differential-algebraic equation systems – theory and applications*, Computers & Chemical Engineering, 19 (1995), pp. 867–882.