Semantics of multi-mode DAE systems

Albert Benveniste, Timothy Bourke, Benoît Caillaud, Marc Pouzet

D.4.1.1 – Semantics of multi-mode DAE systems

Task description: Provides the formal definition and semantics of multi-mode DAE systems. The first version of this deliverable is the basis for the prototypes in WP4.2. The second version is an improved form taking into account the experience with the prototypes.

WP.4 – Systems with multiple operating modes

Version 0.1

Date 07/08/2013

Authors

Albert Benveniste	Inria
Timothy Bourke	Inria
Benoît Caillaud	Inria
Marc Pouzet	ENS Ulm

Executive summary

Hybrid systems modelers exhibit a number of difficulties related to the mix of continuous and discrete dynamics and sensitivity to the discretization scheme. Modular modeling, where subsystems models can be simply assembled with no rework, calls for using Differential Algebraic Equations (DAE). In turn, DAE are strictly more difficult than ODE. They require sophisticated pre-processing using various notions of index before they can be submitted to a solver.

In this report we study some fundamental issues raised by the modeling and simulation of hybrid systems involving DAEs. The objective of this work is to serve for the evolution and the design of future releases of the Modelica language for such systems. We focus on the following questions:

- What is the proper notion of index for a hybrid DAE system?
- What are the primitive statements needed for a DAE hybrid systems modeler?

The differentiation index for DAE explicitly relies on everything being differentiable. Therefore, generalizations to hybrid systems must be done with caution. We propose relying on non-standard analysis for this. Non-standard analysis formalizes differential equations as discrete step transition systems with infinitesimal time basis. We can thus bring hybrid DAE systems to their nonstandard form, where the notion of difference index can be firmly used.

From this study, general hints for future releases of Modelica can be drawn.

Summary

1. Introduction

Modern hybrid systems modelers aim at adressing systems involving:

- Nonsmooth dynamics with bilateral (equality) and unilateral (inequality) constraints, thus giving raise to systems with multiple modes of operation—depending on which inequality constraint is active;
- Differential Algebraic Equations (DAE), which requires supporting index reduction techniques;
- The handling of multiple modes with its associated mode switching and their use in synchronizing different sub-systems---for instance, the switching from free to contact motion in multi-body dynamics causing a change in the control law.

1.1. Issues raised by hybrid systems modelers

Existing modelers can exhibit, for some (non pathological) examples spurious behaviors, such as:

- Unwanted coupling between otherwise noninteracting subsystems;
- Simulation results that are highly nonrobust, depending on the discretization scheme, configuration, or even parameter values used;
- Wrong scheduling of reset operations at mode changes.

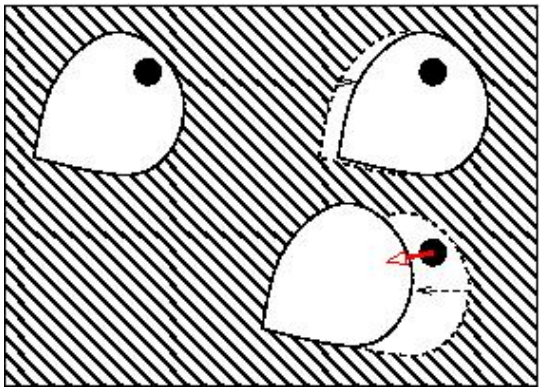
1.2. Some difficult examples from the physics

The above pictures shows three samples of systems that are difficult for modelers. The first hit in American billiard game exhibits a cascade of shocks whose outcome is highly unpredictable and thus difficult to simulate. The same holds for the circuit breaker. The ABS system for brakes operates with an on-and-off mode with very fast dynamics and is an instance of so-called sliding mode control. All these examples are difficult in that they exhibit a dense cascade of events. This cascade may be finite (e.g., the first two examples) or possess a positive duration (e.g., the ABS system). Events generally cause ODE/DAE solvers to stop for reset operations. For some classes of systems, however, stopping is not required as we show in the next section.

1.3. Nonsmooth Systems

Such systems arise in multi-body mechanics or analog circuits involving ideal diods. The simplest instance of this class is the *Moreau Sweeping Process*, shown on the next figure.

The figure shows a rectangular table---from above---carrying two configurations of a two-body system comprising a white, convex cavity that contains an unimpeded black ball. The cavity can freely move and change its shape, whereas the ball only moves in response to the forces at the boundary of the cavity. In the configuration at left, the ball is not moving since it is not touching the cavity boundary. In the configuration at right, however, the ball is subject to contact forces normal to the boundary that keep it within the cavity. A near-Zeno situation occurs when the ball approaches the corner of the cavity. Moreau's Sweeping Processes are global discretization schemes that ignore mode change events, as when the ball hits or leaves the boundary. Variations on the basic scheme have recently been developed by V. Acary and colleagues[1] for the modeling of hair.



The principles behind these schemes can be explained with reference to the above figure. Suppose the system starts in an initial state where the ball is strictly inside the cavity and thus motionless. Now, if a step occurs in which the cavity moves, there are two possible cases. In the first, the ball is still inside the cavity and thus the step is complete. In the second, the ball would now be outside the cavity; a situation which must be corrected by projecting the ball on the cavity. Events of hitting are not computed by this scheme. It may be that the ``exact trajectory'' hits or leaves the boundary several times during the discretization step. Still, it can be proved that this scheme converges to a unique trajectory, for this two-body system, provided the cavity moves smoothly enough.

1.4. Objectives of this work

A number of reasons explain the artifacts mentioned in Section 1.1. We believe, however, that the main reasons for these troubles are the following:

- Lack of a fine typing discrete-time/continuous-time for signals and subsystems;

[1] http://bipop.inrialpes.fr/people/acary/

- Lack of precise causality analysis and synthesis, resulting in a weak mastering of the possible scheduling for the different discrete operations handled by the modeler (detecting mode changes and handling resets);

- Use of a single, centralized, solver for ODEs and DAEs, thus resulting in unwanted coupling between the different time-scales and bandwidths of the subsystems.

In this work we aim at addressing the above issues. We also advocate the technique of *slicing*, in which the simulation engine is structured as a set of off-the-shelf solvers for continuous-time dynamics, coordinated in discrete-time by a synchronous language engine. In our previous work restricted to hybrid systems that are input/output functions involving only ODEs, slicing relied on a typing continuous/discrete for the variables and modules, and the hand was given to the synchronous discrete engine each time an event was created by some zero-crossing. Since some advanced solvers for nonsmooth systems are able to jump over such events without stopping, see Section 1.3, the above mentioned typing discrete/continuous must be revisited for using the latest technology for DAE solvers. We provide first hints for this.

2. The full document is attached

The extended document is attached in the coming pages.

D4.1.1 – Semantics of multi-mode DAE systems

Albert Benveniste
INRIA Rennes

Timothy Bourke
INRIA Paris

Benoit Caillaud
INRIA Rennes

Marc Pouzet
UPMC and ENS, Paris

***Abstract*—Hybrid systems modelers exhibit a number of difficulties related to the mix of continuous and discrete dynamics and sensitivity to the discretization scheme. Modular modeling, where subsystems models can be simply assembled with no rework, calls for using Differential Algebraic Equations (DAE). In turn, DAE are strictly more difficult than ODE.[1] They require sophisticated pre-processing using various notions of *index* before they can be submitted to a solver.**

In this report we study some fundamental issues raised by the modeling and simulation of hybrid systems involving DAEs. The objective of this work is to serve for the evolution and the design of future releases of the Modelica language for such systems. We focus on the following questions:

- **What is the proper notion of *index* for a hybrid DAE system?**
- **What are the primitive statements needed for a DAE hybrid systems modeler?**

The differentiation index for DAE explicitly relies on everything being differentiable. Therefore, generalizations to hybrid systems must be done with caution. We propose relying on *non-standard analysis* for this. Non-standard analysis formalizes differential equations as discrete step transition systems with infinitesimal time basis. We can thus bring hybrid DAE systems to their non-standard form, where the notion of *difference index* can be firmly used.

From this study, general hints for future releases of Modelica can be drawn.[2]

CONTENTS

[1] C.W. Gear said: DAEs are *not* ODEs.

[2] This work was supported by the SYNCHRONICS "action d'envergure" of Inria and by ITEA/Modrio project.

[3] Private communication by H. Elmqvist and S.E. Mattson.

I. Introduction

Modern hybrid systems modelers aim at adressing systems involving:

1) Nonsmooth dynamics with bilateral (equality) and unilateral (inequality) constraints, thus giving raise to systems with multiple modes of operation—depending on which inequality constraint is active;
2) Differential Algebraic Equations (DAE), which requires supporting index reduction techniques;
3) The handling of multiple modes with its associated mode switching and their use in synchronizing different subsystems—for instance, the switching from free to contact motion in multi-body dynamics causing a change in the control law.

A. Issues raised by hybrid systems modelers

As illustrated in [2], [3], [6], existing modelers can exhibit, for some (non pathological) examples spurious behaviors, such as:

- Unwanted coupling between otherwise noninteracting subsystems;
- Simulation results that are highly nonrobust, depending on the discretization scheme, configuration, or even parameter values used;
- Wrong scheduling of reset operations at mode changes.

B. Some difficult examples from the physics

Figure 1. Some difficult examples from the physics

Figure 1 shows three samples of systems that are difficult for modelers. The first hit in american billiard game exhibits a cascade of shocks whose outcome is highly unpredictable and thus difficult to simulate. The same holds for the circuit breaker. The ABS system for brakes operates with an on-and-off mode with very fast dynamics and is an instance of so-called sliding mode control. All these examples are difficult in that they exhibit a dense cascade of events. This cascade may be finite (e.g., the first two examples) or possess a positive duration (e.g., the ABS system). Events generally cause ODE/DAE solvers to stop for reset operations. For some classes of systems, however, stopping is not required as we show in the next section.

C. Nonsmooth Systems

In this section we briefly recall the class of so-called *nonsmooth systems* arising, e.g., in multi-body mechanics or analog circuits involving ideal diods.

The simplest instance of this class is the *Moreau Sweeping Process*, shown on Figure 2. The figure shows a rectangular

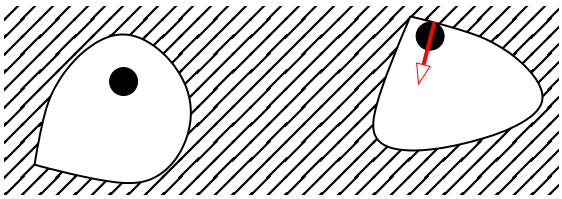


Figure 2. Moreau's Sweeping Process [9], [1]. The red inward pointing arrow indicates the normal contact force applied to the ball.

table—from above—carrying two configurations of a two-body system comprising a white, convex cavity that contains an unimpeded black ball. The cavity can freely move and change its shape, whereas the ball only moves in response to the forces at the boundary of the cavity. In the configuration at left, the ball is not moving since it is not touching the cavity boundary. In the configuration at right, however, the ball is subject to contact forces normal to the boundary that keep it within the cavity. A near-Zeno situation occurs when the ball approaches the corner of the cavity. Moreau's Sweeping Processes [9], [1] are global discretization schemes that ignore mode change events, as when the ball hits or leaves the boundary. Variations on the basic scheme have recently been developed by V. Acary and colleagues [4] for the modeling of hair.

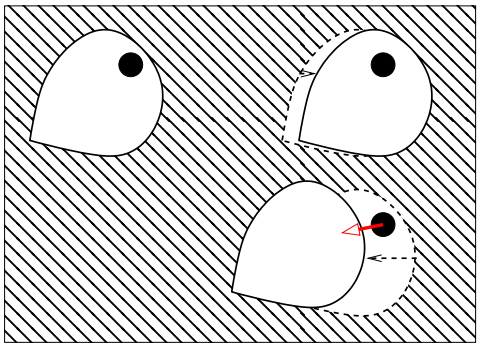


Figure 3. Moreau's Sweeping Process [9], [1]. Left: original position. Top right: first case, shifting the cavity from dashed to solid. Bottom right: second case, shifting the cavity from dashed to solid; projection is needed.

The principles behind these schemes can be explained with reference to Figure 3. Suppose the system starts in an initial state where the ball is strictly inside the cavity and thus motionless. Now, if a step occurs in which the cavity moves, there are two possible cases. In the first, the ball is still inside the cavity and thus the step is complete. In the second, the ball would now be outside the cavity; a situation which must be

[4] http://bipop.inrialpes.fr/people/acary/

corrected by projecting the ball on the cavity. Events of hitting are not computed by this scheme. It may be that the "exact trajectory" hits or leaves the boundary several times during the discretization step. Still, it can be proved [9], [1] that this scheme converges to a unique trajectory, for this two-body system, provided the cavity moves smoothly enough.

D. Objectives of this work

A number of reasons explain the artifacts mentioned in Section I-A. We believe, however, that the main reasons for these troubles are the following:

1) Lack of a fine typing discrete-time/continuous-time for signals and subsystems;
2) Lack of precise causality analysis and synthesis, resulting in a weak mastering of the possible scheduling for the different discrete operations handled by the modeler (detecting mode changes and handling resets);
3) Use of a single, centralized, solver for ODEs and DAEs, thus resulting in unwanted coupling between the different time-scales and bandwidths of the subsystems.

In this work we aim at addressing the above issues 1–3.

We also advocate the technique of *slicing*, in which the simulation engine is structured as a set of off-the-shelf solvers for continuous-time dynamics, coordinated in discrete-time by a synchronous language engine. In our previous work restricted to hybrid systems that are input/output functions involving only ODEs, slicing relied on a typing continuous/discrete for the variables and modules, and the hand was given to the synchronous discrete engine each time an event was created by some zero-crossing [4]. Since some advanced solvers for nonsmooth systems are able to jump over such events without stopping, see Section I-C, the typing discrete/continuous proposed in [3], [6] must be revisited for using the latest technology for DAE solvers. We provide first hints for this.

The paper is organized as follows. In Section II we recall the background on the differentiation index and index reduction for DAE. Difference index is the counterpart of differentiation index, for discrete time difference Algebraic Equations (dAE); this is developed in Section III. To properly study the notion of index for hybrid DAE systems we map the problem to nonstandard analysis, for which we recall the basics in Section IV. We revisit the notion of index using nonstandard analysis in Section V. The central section of the paper is Section VI, where we use our previous material to introduce the notion of index for hybrid DAE systems. Practical considerations are discussed in Section VII and an example is developed. Finally, hints for future evolutions of Modelica (and any hybrid system modeling language) are drawn in Section VIII.

II. Background on index reduction for DAE

The basic references are the works of Campbell and Gear [11] and Mattson and Söderlin [12]. Let $\mathbb{R}$ denote the set of reals, $\mathbb{Z}$ the set of positive or negative integers, and $\mathbb{N}$ the set of non-negative integers.

In this section we consider DAE problems of the following form:[5]

$$F(x,\dot{x}) \quad = \quad 0 \tag{1}$$

where x takes its values in $\mathbb{R}^n$ and F in $\mathbb{R}^m$. In the sequel, F_x and $F_{\dot{x}}$ denote the partial derivatives of F with respect to the first and second variables of F, respectively.

Definition 1: [11] DAE (1) is *solvable*[6] in the connected open set $\Omega \subset \mathbb{R}^{2n}$ if there are connected open sets $\Lambda \subset \mathbb{R}^{\rho}$ and $\mathcal{I} \subset \mathbb{R}$ and a function $(t,\lambda) \to \Phi(t,\lambda)$ such that:

1) $\Theta(t,\lambda) = (t, \Phi(t,\lambda))$ is a diffeomorphism of $\mathcal{I}\times\Lambda$ into $\mathbb{R}^{n+1}$.
2) $\Phi(t,\lambda)$ is a solution of (1) for each value of λ.
3) $(\Phi(t,\lambda), \frac{d}{dt}\Phi(t,\lambda)) \in \Omega$ for every $\lambda \in \Lambda$ and $t \in \mathcal{I}$.
4) If $x(t)$ is a solution of (1) such that $(x(t), \dot{x}(t)) \in \Omega$ for some $t \in \mathcal{I}$, then it holds that $x(t) = \Phi(t,\lambda)$ for some $\lambda \in \Lambda$. A pair (t,x) is called *consistent* if $x = \Phi(t,\lambda)$ holds for some λ. □

Condition 2) expresses that λ acts as a daemon solving the possible nondeterminism. Condition 4) expresses that λ expresses all the nondeterminism. Indeed, λ parameterizes consistent initial conditions, which, in turn, determine solutions of (1).

Systems with exogeneous inputs, of the form, e.g.:

$$F(x,\dot{x},u) \quad = \quad 0 \tag{2}$$

are used in control and when composing subsystems to form larger systems. Systems of the form (2) are a specialization of (1) by putting $y = (x,u)$ and reformulating it as a DAE with state y. Systems of the form (2) leave generally some freedom on exogeneous u (subject to the constraints) when selecting solution $\Phi(t,\lambda)$.

A. The differentiation index

The *differentiation index* for DAE (1) is defined as follows. The *kth derivative array* associated to (1) is:

$$\begin{bmatrix} F(x,\dot{x}) \\ \frac{d}{dt}F(x,\dot{x}) \\ \vdots \\ \frac{d^k}{dt^k}F(x,\dot{x}) \end{bmatrix} \quad =_{\text{def}} \quad F_k(x,\dot{x},w) \;=\; 0 \tag{3}$$

$$\text{where } w \quad =_{\text{def}} \quad (x^{(2)},\ldots,x^{(k+1)}) \tag{4}$$

where we recall that $\frac{d}{dt}F(x,\dot{x}) = F_x(x,\dot{x})\dot{x} + F_{\dot{x}}(x,\dot{x})\ddot{x}$, and so on for higher degree derivatives. In (4) $x^{(1)} = \dot{x}, x^{(2)} = \ddot{x}, x^{(3)} = \ldots$ denote the successive derivatives of x.

The reason for considering the kth derivative array equations is the following. Adding $\frac{d}{dt}F(x,\dot{x}) = 0$ to the original DAE adds new equations and new variables, namely some components of $x^{(2)}$ involved in these new equations. As the subsequent examples show, some of the new equations may not bring fresh variables, but only reuse previous variables,

[5] We may also consider $F(t,x,\dot{x}) = 0$, but dependence on time t can always be removed by making t an additional variable obeying $\dot{t} = 1$.

[6] The term used in [11] is *geometrically solvable.*

which they further constrain. In this case, *hidden constraints* get revealed.

Following again [11], a value x is called *consistent* for (3) if there exists (v, w) such that

$$F_k(x, v, w) \quad = \quad 0 \tag{5}$$

seen as an algebraic equation. Given a consistent value x for (3), algebraic equation (5) will generally have a set of solutions for (v, w).

Definition 2: Assume that DAE (1) is solvable. The *differentiation index* of this DAE, denoted by ν_D, is the smallest index k such that v is uniquely determined by the algebraic equation (5) for any consistent value x for (3). □

That is, the map

$$x \quad \rightarrow \quad \exists w. F_{\nu_D}(x, v, w) = 0 \tag{6}$$

defines v as a deterministic function of x. Since (3) is equivalent to the original DAE, (6) determines $\dot{x}$ and is, therefore, sort of an ODE that can be solved for x.

Comment 1: [11] If the considered DAE is solvable, and $k \geq \nu_D$, then v gives the vector field defined by the solutions on the manifold formed by the solutions. □

Comment 2: Computing the differentiation index is, strictly speaking, a numerical problem. It becomes a structural problem, however, if we wish to compute the *structural* differentiation index, that is, the essential minimum (minimum up to some zero Lebesgue measure set) of the differentiation index when the non-zero parameters of (1) vary over some neighborhood. □

Finally, since the additional rows $\frac{d}{dt}F$, etc., are linear in the derivatives of degree ≥ 2, matrices play a central role in finding the differentiation degree. We thus recall some basic material in the next section.

B. Structurally nonsingular matrices

Rectangular $m \times n$-matrix A is called *structurally nonsingular* if it remains almost everywhere[7] nonsingular when its non-zero entries vary over some neighborhood. Square $m \times m$-matrix P is a *permutation matrix* iff $p_{ij} = Id_{i\sigma(i)}$, where Id is the m-identity matrix and σ is a permutation. Pre- and post-multiplication of a matrix A by a permutation matrix results in permuting the rows and columns of this matrix. It is shown that A is structurally nonsingular iff PA has a nonzero diagonal (all entries of the diagonal are nonzero) for some permutation matrix P; the entries of this nonzero diagonal yield an *output set* for A. As an example, since

$$\underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{P} \underbrace{\begin{bmatrix} 0 & b \\ c & d \end{bmatrix}}_{A} = \begin{bmatrix} c & d \\ 0 & b \end{bmatrix}$$

$\{c, b\}$ is an output set for A. This terminology will be justified when using output sets for Jacobians in DAE.

We shall furthermore use *Block Lower Triangular* (BLT) partitioning of matrix A by applying simultaneous row and column permutations over PA, which yields $Q^T PAQ$, where Q is another square permutation matrix; $Q^T PAQ =_{\text{def}} P'AQ$ has same diagonal as PA. As an example,

$$\underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{Q^T} \begin{bmatrix} c & d \\ 0 & b \end{bmatrix} \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{Q} = \begin{bmatrix} b & 0 \\ d & c \end{bmatrix}$$

BLT is useful to decompose a problem into subproblems, as we shall see.

These definitions apply to the nonlinear equation $F(x) = 0$ by considering the Jacobian $\nabla F(x)$ at a solution of $F(x) = 0$. An output set then defines which variables can be taken as outputs in the system $F(x) = 0$, see the pendulum example below.

C. The pendulum example

Consider the pendulum example (T is an unkown constant):

$$\begin{array}{rcl} \ddot{x} & = & Tx \\ \ddot{y} & = & Ty - g \\ L^2 & = & x^2 + y^2 \end{array} \tag{7}$$

Form (1) for (7) is:

$$\begin{array}{rcl} \dot{x} & = & u \\ \dot{u} & = & Tx \\ \dot{y} & = & v \\ \dot{v} & = & Ty - g \\ L^2 & = & x^2 + y^2 \end{array} \tag{8}$$

This is not index 0 since the Jacobian with respect to $\dot{x}, \dot{u}, \dot{y}, \dot{v}, T$ is singular:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -x \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -y \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

So we must differentiate the system. Differentiating the third equation twice yields:

$$\begin{array}{rcll} \dot{x} & = & u & (i1) \\ \dot{u} & = & Tx & (i2) \\ \dot{y} & = & v & (ii1) \\ \dot{v} & = & Ty - g & (ii2) \\ L^2 & = & x^2 + y^2 & (iii) \\ 0 & = & \dot{x}x + \dot{y}y & (iv) \\ 0 & = & \dot{u}x + \dot{x}^2 + \dot{y}^2 + \dot{v}y & (v) \end{array} \tag{9}$$

Unknowns of highest derivative order are $\dot{x}, \dot{u}, \dot{y}, \dot{v}, T$. Rewriting all equations $(i$–$v)$ in the form $0 = \ldots$ yields the following Jacobian for the equations involving $\dot{x}, \dot{u}, \dot{y}, \dot{v}, T$:

[7] "Almost everywhere" means "outside a set of zero Lebesgue measure".

$$
\begin{bmatrix}
\frac{\partial(i1)}{\partial\dot{x}} & \frac{\partial(i1)}{\partial\dot{u}} & \frac{\partial(i1)}{\partial\dot{y}} & \frac{\partial(i1)}{\partial\dot{v}} & \frac{\partial(i1)}{\partial T} \\
\frac{\partial(i2)}{\partial\dot{x}} & \frac{\partial(i2)}{\partial\dot{u}} & \frac{\partial(i2)}{\partial\dot{y}} & \frac{\partial(i2)}{\partial\dot{v}} & \frac{\partial(i2)}{\partial T} \\
\frac{\partial(ii1)}{\partial\dot{x}} & \frac{\partial(ii1)}{\partial\dot{u}} & \frac{\partial(ii1)}{\partial\dot{y}} & \frac{\partial(ii1)}{\partial\dot{v}} & \frac{\partial(ii1)}{\partial T} \\
\frac{\partial(ii2)}{\partial\dot{x}} & \frac{\partial(ii2)}{\partial\dot{u}} & \frac{\partial(ii2)}{\partial\dot{y}} & \frac{\partial(ii2)}{\partial\dot{v}} & \frac{\partial(ii2)}{\partial T} \\
\frac{\partial(iv)}{\partial\dot{x}} & \frac{\partial(iv)}{\partial\dot{u}} & \frac{\partial(iv)}{\partial\dot{y}} & \frac{\partial(iv)}{\partial\dot{v}} & \frac{\partial(iv)}{\partial T} \\
\frac{\partial(v)}{\partial\dot{x}} & \frac{\partial(v)}{\partial\dot{u}} & \frac{\partial(v)}{\partial\dot{y}} & \frac{\partial(v)}{\partial\dot{v}} & \frac{\partial(v)}{\partial T}
\end{bmatrix}
= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -x \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & -y \\
x & 0 & y & 0 & 0 \\
2x & x & 2y & y & 0
\end{bmatrix}
$$

which, by reordering the rows, yields the Jacobian:

$$
\begin{bmatrix}
x & 0 & y & 0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & -x \\
0 & 0 & 1 & 0 & 0 \\
2x & x & 2y & y & 0 \\
0 & 0 & 0 & 1 & -y
\end{bmatrix} \tag{10}
$$

Under the condition $y \neq 0$, removing the red row yields a structurally nonsingular Jacobian. Hence, $\dot{x}, \dot{u}, \dot{y}, \dot{v}, T$ is determined as a function of other variables—when y is small, then, due to constraint (iii), x is not small and we can exchange the roles of x and y. Hence, the index was found equal to 2.

In principle, Definition 2 requires that we not only differentiate (iii) twice, but also $(i1$–$ii2)$. This would, however, introduce fresh variables $x^{(2)}, x^{(3)}, u^{(2)}, u^{(3)}, y^{(2)}, y^{(3)}, v^{(2)}, v^{(3)}$, which enter the w of (6); eliminating this w is simply achieved by ignoring the differentiation of $(i1$–$ii2)$. Observe that this was a structural reasoning.

Following (5), a consistent value for the tuple x, u, y, v must satisfy the following equations, obtained by substituting $\dot{x}$ using $(i1)$ and $\dot{y}$ using $(ii1)$ in (iv)—this accounts for the red row in (10):

$$
\begin{array}{rcll}
L^2 & = & x^2 + y^2 & (iii) \\
0 & = & ux + vy & (iv)
\end{array} \tag{11}
$$

The remaining equations form a DAE of index 0 (i.e., is equivalent to an ODE) with highest order derivatives $\dot{x}, \dot{u}, \dot{y}, \dot{v}, T$, since the Jacobian is structurally nonsingular (outside a neighborhood of $y = 0$):

$$
\begin{array}{rcll}
\dot{x} & = & u & (i1) \\
\dot{u} & = & Tx & (i2) \\
\dot{y} & = & v & (ii1) \\
\dot{v} & = & Ty - g & (ii2) \\
0 & = & \dot{u}x + u^2 + v^2 + \dot{v}y & (v)
\end{array} \tag{12}
$$

The combination (11,12) is a DAE of index 1, for which good solvers exist. Still, these solvers require projecting derivatives at each integration step to maintain the constraints (11).

D. Mattson-Söderlin "dummyfication"

One can get rid of the difficulty mentioned at the end of Section II-C by using a method due to Mattson and Söderlin [12]. We develop it here by first discussing the pendulum example.

1) The pendulum example: Replace, in (12), $\dot{x}, \dot{u}$ by so-called *dummy derivatives* x', u', which are fresh variables. Doing so yields:

$$
\begin{array}{rcll}
L^2 & = & x^2 + y^2 & (iii) \\
0 & = & ux + vy & (iv) \\
x' & = & u & (i1) \\
u' & = & Tx & (i2) \\
 & & & \\
\dot{y} & = & v & (ii1) \\
\dot{v} & = & Ty - g & (ii2) \\
0 & = & u'x + u^2 + v^2 + \dot{v}y & (v)
\end{array} \tag{13}
$$

The first observation of Mattson and Söderlin is that

Property 1: Problem (13) is equivalent to original pendulum problem (7) in that the respective solutions coincide on the triple (x, y, T) of variables. □

We could have equally well turned, in (9), $\dot{y}, \dot{v}$ into dummy derivatives while leaving $\dot{x}, \dot{u}$ unchanged. This is useful, e.g., when y becomes too close to 0—note that in this case we have x nonzero unless the pendulum has zero-length.

The second and key observation in Mattson-Söderlin method is about the execution scheme of (13).

Execution Scheme 1: The execution scheme for (13) is:

1) given y, v, T on the solution at some instant;
2) use the first four equations to compute x, x', u, u';
3) using ODE $(ii1, ii2, v)$ *autonomously*, evaluate y, v, T for the next discretization instant and repeat. □

By doing so, static invariants are preserved without the need for any projection in the discretization scheme of this autonomous ODE. The key idea is to "dummyfy" the right number of state variables of the index 1 DAE system so that:

1) Property 1 holds, and
2) Pivoted execution scheme 1 applies.

In the next section we present a sketch of the general method of [12] and refer the reader to that reference for details.

2) The general case: (1) is in state space form (derivatives of order at most one). By eliminating spurious state coordinates x_j of the form $\dot{x}_i = x_j$ we rewrite (1) in operator form

$$
\mathcal{F}x = 0 \tag{14}
$$

where the remaining variables appear algebraically or differentiated up to some finite order.

We now introduce some notations. $D = d/dt$ is the differentiation operator. Tuple $\nu = (\nu_1 \dots \nu_n)$ is a multi-index of integers and $D^\nu = \mathrm{diag}(D^{\nu_1}, \dots, D^{\nu_n})$. $\mu(\mathcal{F}) = (\mu_1 \dots \mu_n) \in \mathbb{N}^n$ is the multi-index such that $D^{\mu(\mathcal{F})}$ collects the highest-order derivatives appearing in (14), i.e., $x_j^{(\mu_j)}$ is the highest-order derivative of x_j that appears in (14). Applying Block-Lower-Triangular (BLT) partitioning yields:

$$
P\mathcal{F}^\nu Q = PD^\nu P^T P\mathcal{F}Q = D^{P\nu} P\mathcal{F}Q \tag{15}
$$

DAE system (14) is called structurally nonsingular if there is an output set when we consider x as unknown and we do not distinguish algebraic and differentiated occurrences of a same variable.[8]

Consider a structurally nonsingular DAE $\mathcal{F}x = 0$. Pantelides algorithm finds the minimal multi-index ν such that $\mathcal{G}x =_{\text{def}} D^{\nu}\mathcal{F}x = 0$ is structurally nonsingular with respect to the highest-order derivatives $D^{\mu(\mathcal{G})}x$; this algorithm also provides a corresponding output set. By definition, problem $\mathcal{F}x = 0$ has index 0 if it uniquely determines the highest-order derivatives $D^{\mu(\mathcal{F})}x$ with all $\mu_j(\mathcal{F}) > 0$, as continuous functions of time t and lower derivatives. If the same condition holds with some $\mu_j(\mathcal{F}) = 0$, it has index 1. These are only sufficient conditions, as shown by the following index-1 problem, which does not satisfy these criteria: $\dot{x} + \dot{y} = 1\,,\ x - y = 0$. It may thus be needed to remove unnecessary differentiations resulting from Pantelides algorithm.

The *index reduction procedure* consists of the following steps, for $\mathcal{F}x = 0$ structurally nonsingular:

1) *Differentiation:* Use Pantelides algorithm to obtain
   a) a multi-index $\nu(\mathcal{F})$
   b) $\mathcal{G}x =_{\text{def}} \mathcal{F}^{\nu}x = 0$ with $\mathcal{F}^{\nu} =_{\text{def}} D^{\nu(\mathcal{F})}\mathcal{F}$
   c) an output set for $\mathcal{G}x = 0$ w.r.t. its highest-order derivatives $D^{\mu(\mathcal{G})}x$
2) *Permutation:* using (15), BLT partitioning $\mathcal{G}x = 0$ w.r.t. unknowns $D^{\mu(\mathcal{G})}x$ yields
   a) the static problem $\mathcal{H}y = 0$ with $\mathcal{H} = P\mathcal{F}Q$ and $y = Q^T x$
   b) the differentiated problem $\mathcal{H}^{P\nu}y = 0$ with $\mathcal{H}^{P\nu} = P\mathcal{F}^{\nu}Q$, so that $\nu(\mathcal{H}) = P\nu(\mathcal{F})$ and $\mathcal{H}$ is BLT w.r.t. its highest-order derivatives $D^{Q^T\mu(\mathcal{G})}y$
3) *Result:* after permutation, we obtain a problem, still denoted by $\mathcal{G}x = 0$,
   a) which is in BLT form, and such that, for each block g with corresponding variables z for this block:
      i) the equations have been sorted in descending order w.r.t. the number of differentiations: $\nu_1(g) \geq \nu_2(g) \geq \ldots$
      ii) the Jacobian $\partial g/\partial z$ evaluated at the current point on the solution has full rank. (See the paper for the singular case.)
4) *Index reduction:* replace by dummy ones the derivatives pointed in "Result" (subsequent details are omitted).

Theorem 1: [12] The so obtained problem has index one. It is equivalent to the original problem in that the undifferentiated variables x coincide for both. □

The execution scheme for the resulting problem is a replica of Execution Scheme 1. It is illustrated on Figure 4.

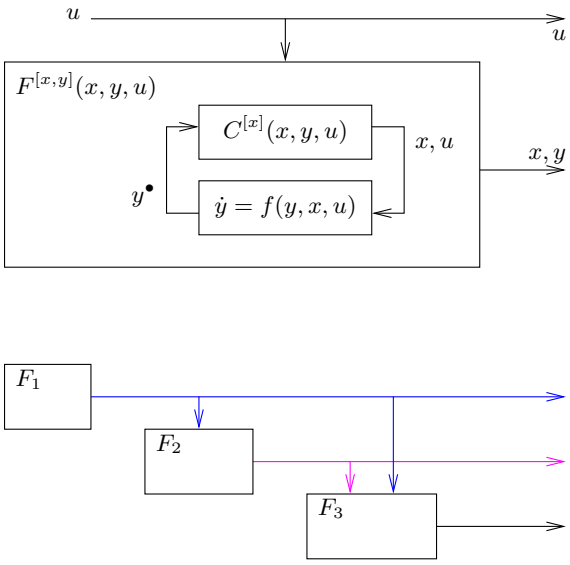


Figure 4. Execution scheme for the result of index reduction, In this figure, the diagram of the top illustrates Execution Scheme 1, with an additional exogenous input u. $C^{[x]}(x,y,u)$ means that the static constraint C involves the variables (x,y,u) and is solved for x as unknown. Thus, x,u is fed to the ODE, which is activated for one discretization step and returns the next value for the solution, denoted by $y^\bullet$. Overall, this defines a DAE problem $F^{[x,y]}(x,y,u)$ with exogenous input u, whose solution is x,y. The bottom diagram depicts the general cascaded architecture as derived from BLT partitioning. It involves a chain of cascaded DAE problems F_i of the generic form shown on top diagram.

III. Index of Difference Algebraic Equations

Difference Algebraic Equations (dAE) are the discrete time counterpart of DAEs:

$$(x, x^\bullet) \in C \quad \text{where} \quad C \subset D_x \times D_x \tag{16}$$

where D_x is the domain of tuple of variables x and tuple $x^\bullet$ has the same domain as x.

Definition 3: A *solution* of dAE (16) is any sequence $\{x_k \mid k \in \mathbb{Z}\}$ satisfying

$$\forall k \in \mathbb{Z} \quad : \quad (x_k, x_{k+1}) \in C$$

and (16) is *solvable* if solutions for it exist. □

Definition 3 expresses that

$$x^\bullet \text{ is the } \textit{forward shifted} \text{ version of } x\text{: } x^\bullet_k = x_{k+1} \tag{17}$$

For convenience, we write in the sequel

$$C(x, x^\bullet) \tag{18}$$

instead of (16). Referring to (4), we define the *kth difference array equations* associated to (18):

$$\begin{bmatrix} C(x,x^\bullet) \\ C^\bullet(x,x^\bullet) \\ \vdots \\ C^{\bullet k}(x,x^\bullet) \end{bmatrix} =_{\text{def}} \quad C_k(x, x^\bullet, w) \tag{19}$$

$$\text{where } w =_{\text{def}} (x^{\bullet 2}, \ldots, x^{\bullet k+1}) \quad \text{and } x^{\bullet k+1} =_{\text{def}} (x^{\bullet k})^\bullet \tag{20}$$

A value x is called *consistent* for (19) if there exists (v,w) such that

$$C_k(x, v, w) \tag{21}$$

[8] Verbatim from [12] but unclear.

seen as an algebraic equation. Given a consistent value x for (19), algebraic equation (21) will generally have a set of solutions for (v, w). Writing w_k instead of w in (19), the chain of sets $\mathcal{V}_k =_{\text{def}} \{v \mid \exists w_k : C_k(x, v, w_k)\}$ is decreasing for set inclusion. Having finite index for the considered dAE means that this chain becomes a singleton for some finite value of k, and then remains so.

Definition 4: Assume that dAE (16) is solvable. The *difference index* of this dAE, denoted by ν_d, is the smallest index k such that v is uniquely determined by the algebraic equation (21) for any consistent value x for (19). □

That is, the map

$$x \quad \rightarrow \quad \exists w.C_{\nu_d}(x, v, w) \tag{22}$$

defines v as a deterministic function of x. Since (19) is equivalent to the original dAE, (22) determines $x^\bullet$ and is, therefore, an OdE (Ordinary Difference Equation), i.e., a transition system that can be directly executed. Mattson-Söderlin method translates to dAE as well.

Index for dAE is conceptually simpler than DAE index. It is, in turn, computationally more complicated because, unlike the computation of $\frac{d}{dt}F$, no linearization occurs as a result of shifting equations forward.

In the sequel we establish a link between the differentiation index for DAE and the difference index for dAE. This will be useful for investigating hybrid DAE systems and is achieved by making use of non-standard analysis, which we briefly recall now.

IV. A SHORT PRIMER ON NON-STANDARD ANALYSIS

The background material of this section is used in proofs, so the reader can skip it for a first reading. The text is borrowed verbatim from [4].

A. Motivation and intuitive introduction

We begin with an intuitive introduction to the construction of the non-standard reals. The goal is to augment $\mathbb{R} \cup \{\pm\infty\}$ by adding, to each x in the set, a set of elements that are "infinitesimally close" to it. We will call the resulting set ${}^\star\mathbb{R}$. Another requirement is that all operations and relations defined on $\mathbb{R}$ should extend to ${}^\star\mathbb{R}$.

A first idea is to represent such additional numbers as convergent sequences of reals. For example, elements infinitesimally close to the real number zero are the sequences $u_n = 1/n$, $v_n = 1/\sqrt{n}$ and $w_n = 1/n^2$. Observe that the above three sequences can be ordered: $v_n > u_n > w_n > 0$ where 0 denotes the constant zero sequence. Of course, infinitely large elements (close to $+\infty$) can also be considered, e.g., sequences $x_u = n$, $y_n = \sqrt{n}$, and $z_n = n^2$.

Unfortunately, this way of defining ${}^\star\mathbb{R}$ does not yield a total order since two sequences converging to zero cannot always be compared: if u_n and u'_n are two such sequences, the three sets $\{n \mid u_n > u'_n\}$, $\{n \mid u_n = u'_n\}$, and $\{n \mid u_n < u'_n\}$ may even all be infinitely large. The beautiful idea of Lindstrøm is to enforce that *exactly one of the above sets is important and the other two can be neglected*. This is achieved by fixing once and for all a finitely additive positive measure μ over the set $\mathbb{N}$ of integers with the following properties:[9]

1) $\mu : 2^{\mathbb{N}} \rightarrow \{0, 1\}$;
2) $\mu(X) = 0$ whenever X is finite;
3) $\mu(\mathbb{N}) = 1$.

Now, once μ is fixed, one can compare any two sequences: for the above case, exactly one of the three sets must have μ-measure 1 and the others must have μ-measure 0. Thus, say that $u > u', u = u'$, or $u < u'$, if $\mu(\{n \mid u_n > u'_n\} = 1)$, $\mu(\{n \mid u_n = u'_n\}) = 1$, or $\mu(\{n \mid u_n < u'_n\}) = 1$, respectively. Indeed, the same trick works for many other relations and operations on non-standard real numbers, as we shall see. We now proceed with a more formal presentation.

B. Construction of non-standard domains

For I an arbitrary set, a *filter* $\mathcal{F}$ over I is a family of subsets of I such that:

1) the empty set does not belong to $\mathcal{F}$,
2) $P, Q \in \mathcal{F}$ implies $P \cap Q \in \mathcal{F}$, and
3) $P \in \mathcal{F}$ and $P \subset Q \subseteq I$ implies $Q \in \mathcal{F}$.

Consequently, $\mathcal{F}$ cannot contain both a set P and its complement P^c. A filter that contains one of the two for any subset $P \subseteq I$ is called an *ultra-filter*. At this point we recall Zorn's lemma, known to be equivalent to the axiom of choice:

Lemma 1 (Zorn's lemma): Any partially ordered set $(X, \leq)$ such that any chain in X possesses an upper bound has a maximal element.

A filter $\mathcal{F}$ over I is an ultra-filter if and only if it is maximal with respect to set inclusion. By Zorn's lemma, any filter $\mathcal{F}$ over I can be extended to an ultra-filter over I. Now, if I is infinite, the family of sets $\mathcal{F} = \{P \subseteq I \mid P^c \text{ is finite}\}$ is a *free* filter, meaning it contains no finite set. It can thus be extended to a free ultra-filter over I:

Lemma 2: Any infinite set has a free ultra-filter.

Every free ultra-filter $\mathcal{F}$ over I uniquely defines, by setting $\mu(P) = 1$ if $P \in \mathcal{F}$ and otherwise 0, a finitely additive measure[10] $\mu : 2^I \mapsto \{0, 1\}$, which satisfies

$$\mu(I) = 1 \text{ and, if } P \text{ is finite, then } \mu(P) = 0.$$

Now, fix an infinite set I and a finitely additive measure μ over I as above. Let $\mathbf{X}$ be a set and consider the Cartesian product $\mathbf{X}^I = (x_i)_{i \in I}$. Define $(x_i) \approx (x'_i)$ iff $\mu\{i \in I \mid x_i \neq x'_i\} = 0$. Relation $\approx$ is an equivalence relation whose equivalence classes are denoted by $[x_i]$ and we define

$${}^\star\mathbf{X} = \mathbf{X}^I / \approx \tag{23}$$

$\mathbf{X}$ is naturally embedded into ${}^\star\mathbf{X}$ by mapping every $x \in \mathbf{X}$ to the constant tuple such that $x_i = x$ for every $i \in I$. Any algebraic structure over $\mathbf{X}$ (group, ring, field) carries over to ${}^\star\mathbf{X}$ by almost point-wise extension. In particular, if

[9] The existence of such a measure is non trivial and is explained later.

[10] Observe that, as a consequence, μ cannot be sigma-additive (in contrast to probability measures or Radon measures) in that it is *not* true that $\mu(\bigcup_n A_n) = \sum_n \mu(A_n)$ holds for an infinite denumerable sequence A_n of pairwise disjoint subsets of $\mathbb{N}$.

$[x_i] \neq 0$, meaning that $\mu\{i \mid x_i = 0\} = 0$ we can define its inverse $[x_i]^{-1}$ by taking $y_i = x_i^{-1}$ if $x_i \neq 0$ and $y_i = 0$ otherwise. This construction yields $\mu\{i \mid y_i x_i = 1\} = 1$, whence $[y_i][x_i] = 1$ in ${}^\star\mathbf{X}$. The existence of an inverse for any non-zero element of a ring is indeed stated by the formula: $\forall x\,(x = 0 \vee \exists y\,(xy = 1))$. More generally:

Lemma 3 (Transfer Principle): Every first order formula is true over ${}^\star\mathbf{X}$ iff it is true over $\mathbf{X}$.

C. Non-standard reals and integers

The above general construction can simply be applied to $\mathbf{X} = \mathbb{R}$ and $I = \mathbb{N}$. The result is denoted ${}^\star\mathbb{R}$; it is a field according to the transfer principle. By the same principle, ${}^\star\mathbb{R}$ is totally ordered by $[u_n] \leq [v_n]$ iff $\mu\{n \mid v_n > u_n\} = 0$. We claim that, for any finite $[x_n] \in {}^\star\mathbb{R}$, there exists a unique $st([x_n])$, call it the *standard part* of $[x_n]$, such that

$$st([x_n]) \in \mathbb{R} \quad \text{and} \quad st([x_n]) \approx [x_n]\,. \tag{24}$$

To prove this, let $x = \sup\{u \in \mathbb{R} \mid [u] \leq [x_n]\}$, where $[u]$ denotes the constant sequence equal to u. Since $[x_n]$ is finite, x exists and we only need to show that $[x_n] - x$ is infinitesimal. If not, then there exists $y \in \mathbb{R}, y > 0$ such that $y < |x - [x_n]|$, that is, either $x < [x_n] - [y]$ or $x > [x_n] + [y]$, which both contradict the definition of x. The uniqueness of x is clear, thus we can define $st([x_n]) = x$. Infinite non-standard reals have no standard part in $\mathbb{R}$.

It is also of interest to apply the general construction (23) to $\mathbf{X} = I = \mathbb{N}$, which results in the set ${}^\star\mathbb{N}$ of *non-standard natural numbers*. The non-standard set ${}^\star\mathbb{N}$ differs from $\mathbb{N}$ by the addition of *infinite natural numbers,* which are equivalence classes of sequences of integers whose essential limit is $+\infty$.

D. Integrals and differential equations: the standardization principle

Any sequence (g_n) of functions $g_n : \mathbb{R} \mapsto \mathbb{R}$ point-wise defines a function $[g_n] : {}^\star\mathbb{R} \mapsto {}^\star\mathbb{R}$ by setting

$$[g_n]([x_n]) = [g_n(x_n)] \tag{25}$$

A function ${}^\star\mathbb{R} \to {}^\star\mathbb{R}$ so obtained is called *internal*. Properties of and operations on ordinary functions extend point-wise to internal functions of ${}^\star\mathbb{R} \to {}^\star\mathbb{R}$. The *non-standard version* of $g : \mathbb{R} \to \mathbb{R}$ is the internal function ${}^\star g = [g, g, g, \ldots]$. The same notions apply to sets. An internal set $A = [A_n]$ is called *hyperfinite* if $\mu\{n \mid A_n \text{ finite}\} = 1$; the *cardinal* $|A|$ of A is defined as $[|A_n|]$.

Now, consider an infinite number $N \in {}^\star\mathbb{N}$ and the set

$$T = \left\{0, \frac{1}{N}, \frac{2}{N}, \frac{3}{N}, \ldots \frac{N-1}{N}, 1\right\} \tag{26}$$

By definition, if $N = [N_n]$, then $T = [T_n]$ with

$$T_n = \left\{0, \frac{1}{N_n}, \frac{2}{N_n}, \frac{3}{N_n}, \ldots \frac{N_n - 1}{N_n}, 1\right\}$$

hence $|T| = [|T_n|] = [N_n + 1] = N + 1$. Now, consider an internal function $g = [g_n]$ and a hyperfinite set $A = [A_n]$. The *sum* of g over A can be defined:

$$\sum_{a \in A} g(a) =_{\text{def}} \left[\sum_{a \in A_n} g_n(a)\right]$$

If t is as above, and $f : \mathbb{R} \to \mathbb{R}$ is a standard function, we obtain

$$\sum_{t \in T} \frac{1}{N} {}^\star f(t) = \left[\sum_{t \in T_n} \frac{1}{N_n} f(t_n)\right] \tag{27}$$

Now, f continuous implies $\sum_{t \in T_n} \frac{1}{N_n} f(t_n) \to \int_0^1 f(t)dt$, so,

$$\int_0^1 f(t)dt = st\left(\sum_{t \in T} \frac{1}{N} {}^\star f(t)\right) \tag{28}$$

Under the same assumptions, for any $t \in [0, 1]$,

$$\int_0^t f(u)du = st\left(\sum_{u \in T, u \leq t} \frac{1}{N} {}^\star f(t)\right) \tag{29}$$

Now, consider the following ODE:

$$\dot{x} = f(x, t), \quad x(0) = x_0 \tag{30}$$

Assume (30) possesses a solution $[0, 1] \ni t \mapsto x(t)$ such that the function $t \mapsto f(x(t), t)$ is continuous. Rewriting (30) in its equivalent integral form $x(t) = x_0 + \int_0^t f(x(u), u)du$ and using (29) yields

$$x(t) = st\left(x_0 + \sum_{u \in T, u \leq t} \frac{1}{N} {}^\star f(x(u), u)\right) \tag{31}$$

The substitution in (31) of $\partial = 1/N$, which is positive and infinitesimal, yields $T = \{t_n = n\partial \mid n = 0, \ldots, N\}$. The expression in parentheses on the right hand side of (31) is the piecewise-constant right-continuous function ${}^\star x(t), t \in [0, 1]$ such that, for $n = 1, \ldots, N$:

$$\begin{array}{rcl} {}^\star x(t_n) & = & {}^\star x(t_{n-1}) + \partial \times {}^\star f({}^\star x(t_{n-1}), t_{n-1}) \\ {}^\star x(t_0) & = & x_0 \end{array} \tag{32}$$

By (31), the solutions x, of ODE (30), and ${}^\star x$, as computed by algorithm (32), are related by $x = st({}^\star x)$. Formula (32) can be seen as a *non-standard semantics* for ODE (30); one which depends on the choice of infinitesimal step parameter ∂. Property (31), though, expresses the idea that all these non-standard semantics are equivalent from the standard viewpoint regardless of the choice made for ∂. This fact is referred to as the *standardization principle*.

V. Index reduction and non-standard semantics

In this section we use the nonstandard semantics of DAE systems to relate the differentiation index and the difference index of the nonstandard semantics of a DAE system.

A. Non-standard semantics of DAEs

As a time domain for our non-standard semantics we use

$$\mathbb{T} \;=\; \{k\partial \mid k \in {}^{\star}\mathbb{Z}\}$$

where ∂ is an infinitesimal time step and ${}^{\star}\mathbb{Z}$ is the set of non-standard integers. Elements of $\mathbb{T}$ will be denoted by the symbol τ. The *non-standard semantics* of a DAE is obtained by applying the following substitution rules:

$$\begin{array}{lcl} \dot{x} & \leftrightarrow & \frac{1}{\partial}(x^{\bullet} - x) \\ \ddot{x} & \leftrightarrow & \frac{1}{\partial^2}(x^{\bullet 2} - 2\,x^{\bullet} + x) \\ x^{(3)} & \leftrightarrow & \frac{1}{\partial^3}(x^{\bullet 3} - 3\,x^{\bullet 2} + 3\,x^{\bullet} - x) \\ x^{(4)} & \leftrightarrow & \ldots \end{array} \tag{33}$$

where $\tau^{\bullet} =_{\text{def}} \tau + \partial$, $x^{\bullet}_{\tau} =_{\text{def}} x_{\tau^{\bullet}}$, and $x^{\bullet m} = (x^{\bullet m-1})^{\bullet}$. Applying this to the pendulum example (8) yields:

$$\begin{array}{lcl} x^{\bullet} & = & x + \partial \times u \\ u^{\bullet} & = & u + \partial \times Tx \\ y^{\bullet} & = & y + \partial \times v \\ v^{\bullet} & = & v + \partial \times (Ty - g) \\ L^2 & = & x^2 + y^2 \end{array} \tag{34}$$

B. Index reduction in non-standard semantics

We now establish a link between index reduction for DAE and index reduction for dAE. Return to the pendulum example. Highest degree shifted variables are $x^{\bullet}, u^{\bullet}, y^{\bullet}, v^{\bullet}, T$. Corresponding Jacobian is singular, thus the difference degree is strictly positive. Forward shifting the last equation two times yields:

$$\begin{array}{lcll} x^{\bullet} & = & x + \partial \times u & (i1) \\ u^{\bullet} & = & u + \partial \times Tx & (i2) \\ y^{\bullet} & = & y + \partial \times v & (ii1) \\ v^{\bullet} & = & v + \partial \times (Ty - g) & (ii2) \\ L^2 & = & x^2 + y^2 & (iii) \\ L^2 & = & (x^{\bullet})^2 + (y^{\bullet})^2 & (iv) \\ L^2 & = & (x^{\bullet 2})^2 + (y^{\bullet 2})^2 & (v) \end{array} \tag{35}$$

Substituting, in (iv, v), $x^{\bullet}$ and $y^{\bullet}$ by using $(i1)$ and $(ii1)$, and reorganizing the result yields:

$$\begin{array}{lcll} x^{\bullet} & = & x + \partial \times u & (i1) \\ u^{\bullet} & = & u + \partial \times Tx & (i2) \\ y^{\bullet} & = & y + \partial \times v & (ii1) \\ v^{\bullet} & = & v + \partial \times (Ty - g) & (ii2) \\ L^2 & = & (x^{\bullet} + \partial \times u^{\bullet})^2 + (y^{\bullet} + \partial \times v^{\bullet})^2 & (v) \\ & & & \\ L^2 & = & x^2 + y^2 & (iii) \\ L^2 & = & (x + \partial \times u)^2 + (y + \partial \times v)^2 & (iv) \end{array} \tag{36}$$

The first group of equations has structurally nonsingular Jacobian with respect to $x^{\bullet}, u^{\bullet}, y^{\bullet}, v^{\bullet}, T$, and is thus a dAE of index 0. dAE system (34) has thus index 2. Finally, Mattson-Söderlin dummyfication applies as well. And all of this extends to general DAE. Informally speaking, we have the following "equation":

$$\text{index}(\text{NS}(\text{DAE})) \quad \overset{\text{structurally}}{=} \quad \text{index}(\text{DAE}) \tag{37}$$

where NS(DAE) denotes the non-standard interpretation of DAE, seen as a dAE.

VI. Hybrid DAE

A. Mode dependent dynamics

The basic form for a hybrid DAE system is

$$\begin{array}{lcl} 0 & = & \sum_{i\in I} b_i(x,\dot{x}) \times F_i(x,\dot{x}) \\ 1 & = & \sum_{i\in I} b_i(x,\dot{x}) \end{array} \tag{38}$$

where I is some finite index set, x denotes a n-tuple of real variables, the F_i's are real-valued and smooth, and the b_i's are smooth $\{0,1\}$-valued functions representing boolean predicates over the listed variables. The second equation expresses that one and exactly one predicate must be valid at any time. Thus set I indexes the different system *modes*. The first equation specifies that, in mode i, DAE $F_i = 0$ must hold; observe that this is a fixpoint equation. Thus, b_i is the guard of mode i and $F_i = 0$ its dynamics.

Examples: We discuss here a few examples, showing the flexibility of generic form (38):

1) A first example of system of the form (38) is obtained by considering a DAE system with unilateral constraint:
$$0 \;\leq\; F(x,\dot{x}) \tag{39}$$
where x and F are as above. Then, (39) can be put in the form (38) with two modes, where, for the first mode, $b(x,\dot{x}) = [0 \geq F(x,\dot{x})]$ is the boolean predicate expressing that unilateral constraint (39) is *active*, whereas in the second mode with guard $1-b$, $\bar{F} \equiv 0$, expressing that $(x,\dot{x})$ is unconstrained when the inequality is strict in (39).
2) So-called DAE systems with a *complementarity conditions* are a second example:
$$\begin{array}{r} U(x)\geq 0 \text{ and } V(x)\geq 0 \text{ and } U(x)V(x) = 0 \\ F(x,\dot{x}) = 0 \end{array} \tag{40}$$
Such systems are encountered, e.g., in electric circuits with perfect diods. Some massaging can bring (40) to the generic form (38).
3) A third example is
$$\begin{array}{lcl} 0 & = & (1-b)\times F(x,y,\dot{x}) + b\times C(x,y) \\ b & = & \mathbf{1}_{P(y)} \end{array} \tag{41}$$
where $P(y)$ holds true at the *zero-crossings* of some smooth function $g(y)$, i.e., at any instant when g crosses zero from below, and $C(x,y) = 0$ yields a consistent value for DAE $F(x,y,\dot{x}) = 0$. In words, $C(x,y) = 0$ specifies the reset of DAE $F(x,y,\dot{x}) = 0$ at the zero-crossings of g. This example shows that, in some cases, (38) specializes to systems that are not fixpoint.

Definition 1 must be adapted to define what a solution of (38) is. Let $\mathbf{B} = \{0,1\}$ be the Boolean domain, represented by the two values 0 and 1. Set $\mathbb{S} = \mathbb{R} \times \mathbb{N}$, equipped with the lexicographic order defined by: $(t,k) < (t',k')$ if and only if, either $t < t'$, or $t = t'$ and $k < k'$; elements of $\mathbb{S}$ are denoted by the symbol s, or explicitly as pairs $s = (t,k)$

whenever needed. Time set $\mathbb{S}$ defines the so-called *super-dense time*, see [7], [8]. It allows defining solutions for (38) in which finite (but possibly unbounded) cascades of mode changes can occur.

Definition 5: Hybrid DAE (38) is *solvable* if there exists a pair of functions $(s,\lambda) \to (\Phi(s,\lambda),\beta(s,\lambda))$, from $\mathbf{S}\times\Lambda$ into $(\mathbb{R}^n \times \mathbf{B}^I) \cup \{\bot\}$, where $\bot$ is the *undefined* value and Λ is some nonempty open set of $\mathbb{R}^\rho$, satisfying the following conditions, where $\beta_i, i{\in}I$ denote the components of β:

1) The function $t{\to}\beta((t,0),\lambda)$ satisfies the constraint $\sum_i \beta_i((t,0),\lambda) = 1$;
2) For each $\lambda \in \Lambda$, there exists an increasing sequence $T(\lambda) = \{t_k(\lambda) \mid k \in \mathbb{Z}\}$ of instants of $\mathbb{R}$ such that $\lim_{k\to\pm\infty} t_k(\lambda) = \pm\infty$ and $(\Phi(s,\lambda),\beta(s,\lambda)) = \bot$ for $s = (t,k)$ with $t \notin T(\lambda)$ and $k > 0$. Write t_k instead of $t_k(\lambda)$ when no confusion results.
3) Regarding the mode i:
   - The function $t{\to}\beta((t,0),\lambda)$ is constant over each interval $(t_k, t_{k+1}]$ with $b \in \mathbf{B}^I$ the corresponding value, and $\beta((t_k, n_{t_k}),\lambda) = b$.
   - For $t \in T(\lambda)$, there exists some integer $n_t = n_t(\lambda) > 0$ such that $\beta((t,k),\lambda) = \bot$ for $k > n_t$ and $k \to \beta((t,k),\lambda)$ varies in $\mathbf{B}^I$ for $k \le n_t$.
4) Regarding the state x:
   - For each $\lambda \in \Lambda$ and each open interval (t_k, t_{k+1}), then $t{\to}\Phi((t,0),\lambda)$ is a diffeomorphism from (t_k, t_{k+1}) into $\mathbb{R}^n$, and, if $\beta_i((t,0),\lambda) = 1$, then $F_i(\Phi((t,0),\lambda), \frac{d}{dt}\Phi((t,0),\lambda) = 0$ holds for every $t \in (t_k, t_{k+1})$.
   - For each $\lambda \in \Lambda$ and each $t{\in}T(\lambda)$, then
     - $\Phi((t,0),\lambda) = \lim_{t'<t, t'\to t} \Phi((t',0),\lambda)$;
     - $\Phi((t,n_t),\lambda) = \lim_{t'>t, t'\to t} \Phi((t',0),\lambda)$; and,
     - for $k{>}0$, $\Phi((t,k),\lambda)$ is a consistent value for $F_i = 0$ if $\beta_i((t,k),\lambda) = 1$.
5) If $s{\to}(x_s, b_s)$ satisfies conditions 3) regarding β and 4) regarding x for some increasing sequence $T = \{t_k \mid k \in \mathbb{Z}\}$ satisfying condition 2, then it holds that $(x_s, b_s) = (\Phi(s,\lambda),\beta(s,\lambda))$ for some λ. □

Albert: We are missing the counterpart of statement 1) in Definition 1. The difficulty lies in the need to account for the jumps in trajectory when talking about diffeomorphisms. Need to correct this.

Some comments are in order regarding Definition 5:

1) As for DAE systems, parameter λ serves to represent the choice of some initial condition (in fact, any consistent value can be set at some time of the DAE; it may not be the initial instant).
2) Mode changes can occur in cascades, indexed by the second component k of $s = (t,k)$. Cascades must be finite but need not be bounded. Different cascades are isolated.
3) The state dynamics is set by each mode following the specification (38).
4) This condition expresses that λ does capture all the non-determinism.

Our definition of a solution for a hybrid DAE system generalizes the classical definition for (ODE based) hybrid systems. Observe that conditions for existence and/or uniqueness of solutions are delicate, particularly so because we have ruled out Zeno behaviors.

The theory of DAE differentiation index recalled in Section II deeply relies on differentiability, so it does not apply as such to (38). In contrast, the notion of difference index for dAE does not require differentiability. To circumvent the lack of differentiability of (38), we propose to move to its non-standard semantics. As a matter of fact, the non-standard semantics of a DAE hybrid system is simple and clean defining.

B. Non-standard semantics of hybrid DAE

Using (33), the non-standard semantics of (38) is:

$$C_{[b_i,F_i]}(x,x') \tag{42}$$
$$=_{\text{def}} \quad \begin{cases} 0 = \sum_{i\in I} b_i(x,x') \times F_i(x,x') \\ 1 = \sum_{i\in I} b_i(x,x') \end{cases}$$

$$\text{where} \quad x'_\tau = \frac{x_{\tau+\partial} - x_\tau}{\partial} \text{ i.e., } x' = \frac{x^\bullet - x}{\partial} \tag{43}$$

In the following reasoning, concepts and notations of Section IV are used.

Theorem 2: Assume that hybrid DAE (38) is solvable. Then, every solution of (38) is the standardisation of some solution of (42). □

Proof: Consider the standard hybrid DAE system (38) and let $(x_s, b_s), s = (t,k) \in \mathbb{S}$, be a solution for it in the sense of Definition 5. Then, let λ be the parameter representing its initial condition. We show that this solution is the standardisation of some solution of hybrid dAE (42). Two cases can occur, see condition 2) of Definition 5:

Case 1: $t \notin T(\lambda)$. Then, by condition 2) of Definition 5, we can assume that $s = (t,0)$. Suppose that $P(x_{(t,0)}) = \mathrm{T}$ holds—the opposite case is handled similarly. We then have $F(x_{(t,0)}, \dot{x}_{(t,0)}) = 0$. Pick $\mathbb{T} \ni \tau \approx t$. We can thus assume $\tau = [t_n]$ for some sequence t_n of reals converging to t. Hence, there exists an integer N such that, for $n{\ge}N$, $t_n \notin T(\lambda)$ follows, and thus, if $x_{(t_n,0)}$ is the x-component of the solution of (38) at time $(t_n,0)$, then it results that $P(x_{(t_n,0)}) = \mathrm{T}$. Therefore, setting $x_\tau = [x_{(t_n,0)}]$, we get $P(x_\tau) = P([x_{(t_n,0)}]) = [P(x_{(t_n,0)})] = \mathrm{T}$. Similarly, since F is smooth and the solution of $F(x,\dot{x}) = 0$ is assumed to be infinitely differentiable, we get $F(x_{(t_n,0)}, \dot{x}_{(t_n,0)}) = 0$ for $n{\ge}N$, whence $F(x_\tau, x'_\tau) \approx 0$ follows as well. For this case, we thus proved the existence of a solution (x_τ, b_τ) for (42) such that $(x_{(t,0)}, b_{(t,0)}) = st(x_\tau, b_\tau)$.

Case 2: $t \in T(\lambda)$, meaning that one or more successive mode changes occur at t, so that the super-dense instants for consideration are $(t,0),(t,1),\ldots,(t,n_t(\lambda))$, on which $b_{(t,k)}$ alternate in $\mathbf{B}$. Suppose $b_{(t,0)} = \mathrm{T}$, whence $x_{(t,0)}$ is a consistent value for $F = 0$. By condition 3 of Definition 5, we also have $b_{(t_n,0)} = \mathrm{T}$ for t_n any increasing sequence converging to t and $n{\ge}N$. Consider $\mathbb{T} \ni \tau = [t_n]$, we have $st(\tau) = t$. Setting $x_\tau =_{\text{def}} [x_{(t_n,0)}]$ yields a consistent value for $F(x,x') = 0$

and complementing it with $b_\tau = \mathtt{T}$ extends the solution of (42) at the considered τ. We further extend this solution for the subsequent non-standard instants $\tau+\partial, \ldots, \tau+n_t\partial$ as follows. First, observe that $\tau+\partial \approx \ldots \approx \tau+n_t\partial \approx t$. Then, we simply extend the solution of (42) by setting $b_{\tau+k\partial} = b_{(t,k)}$ and $x_{\tau+k\partial} = x_{(t,k)}$, where $(x_{(t,k)}, b_{(t,k)})$ is the given (standard) solution of (38). This proves the theorem. ■

C. Non-standard hybrid DAE index

In this section we study the difference index of dAE (42). Accordingly, using (43) we regard F as a function of the pair $(x, x^\bullet)$. To simplify the notations, when no confusion can result, we write F for short instead of $F(x, x^\bullet)$ and similarly for $\bar{F}$ and P. With this convention, we have

$$C^{\bullet k}_{[b_i,F_i]}(x, \underbrace{x^\bullet}_{v}, w) = \begin{cases} 0 & = \sum_{i\in I} b_i^{\bullet k}(x,x') \times F_i^{\bullet k}(x,x') \\ 1 & = \sum_{i\in I} b_i^{\bullet k}(x,x') \end{cases} \quad (44)$$

where

$$w \quad = \quad (x^{\bullet 2}, \ldots, x^{\bullet k+1})$$

and the difference array of dAE (42) is

$$C_{k,[b_i,F_i]}(x,v,w) \quad =_{\text{def}} \quad \begin{bmatrix} C_{[b_i,F_i]}(x,v,w) \\ C^{\bullet}_{[b_i,F_i]}(x,v,w) \\ \vdots \\ C^{\bullet k}_{[b_i,F_i]}(x,v,w) \end{bmatrix} \quad (45)$$

At this point, we would like to relate the difference index of (42) to the indexes of the F_i's (by (37) we can either consider the differential index, or the difference index of the corresponding nonstandard semantics). Unfortunately, we have a problem near instants where the mode changes. Consider for example an instant τ such that $b_{i,\tau} = 1$ whereas $b_{j,\tau+\partial} = \ldots = b_{j,\tau+k\partial} = 1$ for some $j \neq i$. This causes the arrays of F_i and F_j to get mixed in (45), at that instant. At such an instant τ we thus have

$$C_{k,[b_i,F_i]}(x_\tau, v_\tau, w_\tau) \quad = \quad \begin{bmatrix} F_i(x_\tau, v_\tau, w_\tau) \\ F_j^{\bullet}(x_\tau, v_\tau, w_\tau) \\ \vdots \\ F_j^{\bullet k}(x_\tau, v_\tau, w_\tau) \end{bmatrix}$$

This observation leads to introducing the following family of arrays:

$$A^F_{\ell,k} \quad =_{\text{def}} \quad \begin{bmatrix} F_{\ell(0)} \\ F^{\bullet}_{\ell(1)} \\ \vdots \\ F^{\bullet k}_{\ell(k)} \end{bmatrix} \quad (46)$$

where $F = \{F_i \mid i \in i\}$, k is a nonnegative integer, and $\ell : \{0 \ldots k\} \to I$ is a map. The following results follows from the above observation:

Theorem 3: Let ν_d be the difference index of dAE (42). Then, $\nu_d \leq \nu_d(F)$, where $\nu_d(F)$ is the smallest index k such that, for any consistent x and any ℓ,

$$\text{the map } x \to \exists w. A^F_{\ell,k}(x,v,w) = 0 \text{ uniquely defines } v. \quad (47)$$

□

The following theorem is central:

Theorem 4: Assume that hybrid DAE (38) is solvable and the length of cascades of mode changes is bounded by some finite integer K. Then:

$$\nu_d(F) \quad \leq \quad 2 \times \max_{i\in I} \nu_D(F_i) + K - 1 \quad (48)$$

where $\nu_D(F_i)$ is the differentiation index of DAE $F_i = 0$. □

Proof: Due to the assumption, two successive cascades of mode changes must be separated by intervals of time of positive duration where the mode does not change. Accordingly, the worst k for (47) to hold is depicted on Figure 5. Time

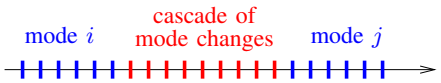


Figure 5. The worst k for (47) to hold

progresses from left to right. The current instant is the first one shown. The system stays in mode i too short a duration for v being uniquely defined by the array associated to $F_i = 0$: this causes the "$\max_{i\in I} \nu_D(F_i) - 1$" contribution to the bound (48). Then we have a cascade of maximal length K. And, finally, a long enough interval is spent in mode j for v being uniquele determined by the array associated to $F_j = 0$, giving another "$\max_{i\in I} \nu_D(F_i)$" contribution to the bound. ■

VII. Practical algorithms

A. Implementing index reduction for hybrid DAE systems

Based on the previous analysis we propose the following approach for index reduction of hybrid DAE systems:

(a) *Inside a mode:* perform index reduction for the DAE system in force in this mode.

(b) *At a cascade of mode changes:* switch to the nonstandard semantics and compute on-the-fly the dAE index reduction following Section VI-C.

The handling of mode changes (task (b) above) remains to be developed, with algorithms of acceptable complexity. In the remainder of this section we develop tools for mode-dependent index reduction in task (a) above.

B. The guarded Pantelides graph

For his algorithm [10], Pantelides introduces a graphical method for finding the differentiation index, i.e., solving equation (6). The method consists in pivoting. To prepare for this we introduce the graphical statement

$$E \;\text{—}[]\text{—}\; (\mathit{othervar} \,|\, \mathit{outvar}) \quad (49)$$

where E denotes an equation, and

- $\mathit{othervar} \cup \mathit{outvar}$ is the set of variables involved in E;

- *outvar* is the set of variables that are among the candidate outputs for E, and
- set *othervar* collects other variables, i.e., those variables which cannot serve as an output for E.

Abstraction (49) is coarse in that it does not take into account the actual numerical coefficients involved in the different constraints—strictly speaking, singularity is a numerical property, not a graphical one.

Now, in our hybrid systems involving DAE, DAE systems are mode-dependent, see Sections VI-C and VII-A. Based on the analysis developed in Section VI-C, for each mode we must develop a Pantelides search. We will reuse the ideas behind clock-and-dependency calculus of Signal [5] by introducing *guarded Pantelides graphs* whose branches have the form

$$E \ -[guard]- \ (othervar \,|\, outvar) \tag{50}$$

meaning that (49) if and only if *guard* is true. Guarded branches of the form (50) avoid enumerating the different modes and fold the different Pantelides algorithms into a single "guarded-Pantelides".

C. A simple circuit

Referring to Figure 6, we consider the simple example consisting of a the second vertical branch of the circuit shown, consisting of a perfect diod followed by a capacitor:

$$\begin{array}{lcccl} D_1 & : & 0 & \leq & i \\ D_2 & : & 0 & \geq & u \\ D_3 & : & 0 & = & iu \\ C & : & i & = & Cv' \end{array} \tag{51}$$

This circuit has two modes, characterized by the conditions $i > 0$ and $u < 0$, respectively—we rule out the trivial mode $0 = u = i$. We study this simple circuit by using the theory developed in Section VI.

1) Mode-dependent index: We begin with index analysis while the system being in each different mode:

mode $i > 0$: (51) boils down to

$$\begin{array}{lcccl} D_1 & : & 0 & < & i \\ D_2 & : & 0 & = & u \\ C & : & i & = & Cv' \end{array} \tag{52}$$

We interpret the derivative operator v' as in (43), namely:

$$v' \ = \ \frac{v^\bullet - v}{\partial}$$

By doing so, (52) abstracts as

$$\begin{array}{l} D_2 \ -[i > 0]- \ (\ |\, u\ ) \\ C \ -[\quad\quad]- \ (i \,|\, v^\bullet) \end{array} \tag{53}$$

which has index 0 (it is an ODE system).

mode $u < 0$: (51) boils down to

$$\begin{array}{lcccl} D_1 & : & 0 & = & i \\ D_2 & : & 0 & < & u \\ C & : & i & = & Cv' \end{array} \tag{54}$$

which abstracts as

$$\begin{array}{l} D_1 \ -[u < 0]- \ (\ |\, i\ ) \\ C \ -[\quad\quad]- \ (i \,|\, v^\bullet) \end{array} \tag{55}$$

which again has index 0 (it is an ODE system), but is singular in that it does not determine u.

2) Handling the mode changes: It remains to perform index reduction at the mode changes, from $i > 0$ to $u < 0$ and vice-versa. For the first one, the following non-standard difference array is considered:

$$\left[\begin{array}{lcccl} D_1 & : & 0 & < & i \\ D_2 & : & 0 & = & u \\ C & : & i & = & Cv' \\ D_1^\bullet & : & 0 & = & i^\bullet \\ D_2^\bullet & : & 0 & > & u^\bullet \\ C^\bullet & : & i^\bullet & = & C(v')^\bullet \end{array}\right] \tag{56}$$

Eliminating $v^{\bullet 2}$ in (56) amounts to discarding the last equation of this array, which uniquely determines the next values $v^\bullet$ and $i^\bullet$ for use as reset values in the next mode $u < 0$. Observe that this array does not determine u at all, which is consistent with the analysis of mode $u < 0$. The latter result can be found by replacing array (56) by its graph abstraction

$$\left[\begin{array}{c} D_2 \ -[i > 0]- \ (\ |\, u) \\ C \ -[]- \ (i \,|\, v^\bullet) \\ D_1^\bullet \ -[u^\bullet < 0]- \ (\ |\, i^\bullet) \\ C^\bullet \ -[]- \ (i^\bullet \,|\, v^{\bullet 2}) \end{array}\right] \tag{57}$$

Array (56) provides the consistent reset conditions for the new mode.

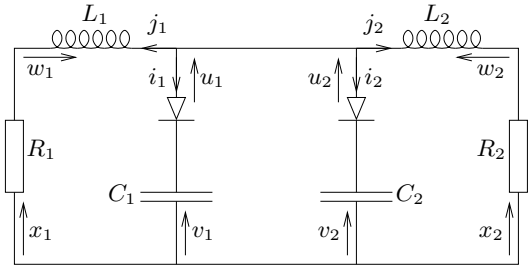
D. The Elmqvist-Mattson circuit example[11]

In this section we use the material introduced in Section VII-B. We analyze the RLDC circuit involving two parallel ideal diods shown in Figure 6. To avoid notational problems with symbols i, j we use x' to mean $\dot{x}$, etc. We show in Figure 6 both the DAE system and the corresponding non-standard semantics using shift operator (43), which we repeat for the sake of readability of the example:

$$x' \ =_{\text{def}} \ \frac{x^\bullet - x}{\partial}$$

1) Mode-dependent index: We begin with the index analysis for each given mode. The Pantelides abstraction is shown on Figure 7, left, where $D^=$ refers to the saturated form of unilateral constraint D. We now perform index and causality analysis. We first consider the L and C systems. In all modes of the system, they require the j's and v's to be their respective outputs; this in turn force the same signals to be inputs in other equations—we use this color coding in the sequel while

[11] Private communication by H. Elmqvist and S.E. Mattson.



$$
\begin{array}{rl}
\text{Kirchhof laws} : & \left\{ \begin{array}{lcl} K_1 : 0 = & j_1 + j_2 + i_1 + i_2 \\ & x_1 + w_1 \\ K_2 : = & u_1 + v_1 \\ K_3 : = & u_2 + v_2 \\ K_4 : = & x_2 + w_2 \end{array} \right. \\
\text{Resistors} : & \left\{ \begin{array}{lcl} R_1 : x_1 = & R_1 j_1 \\ R_2 : x_2 = & R_2 j_2 \end{array} \right. \\
\text{Inductors (L)} : & \left\{ \begin{array}{lcl} L_1 : w_1 = & L_1 j_1' \\ L_2 : w_2 = & L_2 j_2' \end{array} \right. \\
\text{Capacitors} : & \left\{ \begin{array}{lcl} C_1 : i_1 = & C_1 v_1' \\ C_2 : i_2 = & C_2 v_2' \end{array} \right. \\
\text{Ideal Diods} : & \left\{ \begin{array}{lcl} D_1 : 0 \leq & i_1 \\ D_2 : 0 \geq & u_1 \\ D_3 : 0 = & i_1 u_1 \\ D_4 : 0 \leq & i_2 \\ D_5 : 0 \geq & u_2 \\ D_6 : 0 = & i_2 u_2 \end{array} \right.
\end{array}
$$

Figure 6. The RLDC circuit: diagram and equations.

$$
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_1^= \ -[0<u_1]- \ (\,|\,i_1) \\ D_2^= \ -[I_1]- \ (\,|\,u_1) \\ \text{F} = [0 < u_1] \wedge [0 < i_1] \\ D_4^= \ -[0<u_2]- \ (\,|\,i_2) \\ D_5^= \ -[0<i_2]- \ (\,|\,u_2) \\ \text{F} = [0 < u_2] \wedge [0 < i_2] \end{array} \right.
\qquad
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_1^= \ -[0<u_1]- \ (\,|\,i_1) \\ D_2^= \ -[I_1]- \ (\,|\,u_1) \\ \text{F} = [0 < u_1] \wedge [0 < i_1] \\ D_4^= \ -[0<u_2]- \ (\,|\,i_2) \\ D_5^= \ -[0<i_2]- \ (\,|\,u_2) \\ \text{F} = [0 < u_2] \wedge [0 < i_2] \end{array} \right.
$$

Figure 7. Left: abstraction. Right: first stage of causality synthesis.

performing causality synthesis, which consists in picking in a consistent way exactly one variable in each set *outvar*. We proceed by successive steps. In a first step we focus on the L's and C's and then on the R's. The result is shown on Figure 7, right.

At this point we are stuck and we must analyze the different modes separately. We first investigate the mode $[0 < u_1] \wedge [0 < i_2]$. So we remove the branches whose guard is violated and remove the trivial guards of the remaining branches. Then we propagate backward the status of output. For completeness we recall the guard for this mode. The result is shown in Figure 8, left. A consistent causality results since every equation determines exactly one output and all variables

$$
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_1^= \ -[]- \ (\,|\,i_1) \\ D_5^= \ -[]- \ (\,|\,u_2) \end{array} \right.
\qquad
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_2^= \ -[]- \ (\,|\,u_1) \\ D_4^= \ -[]- \ (\,|\,i_2) \end{array} \right.
$$

Figure 8. Left column: mode $[0 < u_1] \wedge [0 < i_2]$;
notation $-[]-\ (.\,|\,.)$ stands for $-[0 < u_1 \wedge 0 < i_2]-\ (.\,|\,.)$.
Right column: mode $[0 < u_2] \wedge [0 < i_1]$;
notation $-[]-\ (.\,|\,.)$ stands for $-[0 < u_2 \wedge 0 < i_1]-\ (.\,|\,.)$.

are listed as the output of some equation (the system is closed). In addition, the state of every ODE is an output of that ODE. This system is index 1 and regular. It obeys the execution scheme of Figure 4. The symmetric mode $[0 < u_2] \wedge [0 < i_1]$ is handled similarly, see Figure 8, right.

We next investigate the mode $[0 < i_1] \wedge [0 < i_2]$, see Figure 9, left. No consistent causality can be found here. This

$$
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_2^= \ -[]- \ (\,|\,u_1) \\ D_5^= \ -[]- \ (\,|\,u_2) \end{array} \right.
\qquad
\left\{ \begin{array}{l} K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\ K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\ K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\ K_3^\bullet \ -[]- \ (\,|\,u_1^\bullet, v_1^\bullet, u_2^\bullet, v_2^\bullet) \\ K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\ R_1 \ -[]- \ (\,|\,x_1, j_1) \\ R_2 \ -[]- \ (\,|\,x_2, j_2) \\ L_1 \ -[]- \ (w_1 \,|\, j_1^\bullet) \\ L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\ C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\ C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\ D_2^= \ -[]- \ (\,|\,u_1) \\ D_2^{=\bullet} \ -[]- \ (\,|\,u_1^\bullet) \\ D_5^= \ -[]- \ (\,|\,u_2) \\ D_5^{=\bullet} \ -[]- \ (\,|\,u_2^\bullet) \end{array} \right.
$$

Figure 9. Left: the mode $[0 < i_1] \wedge [0 < i_2]$.
Right: shifting some equations forward.

reflects that the system is not index 1. Thus index reduction must be applied. Shifting forward equations $K_3, D_2^=, D_5^=$ yields the system shown on Figure 9, right. At this point we change the status of $v_1^\bullet, u_1^\bullet, u_2^\bullet$ and replace them by dummy static variables $\widehat{v}_1, \widehat{u}_1, \widehat{u}_2$. The result is shown on Figure 10, left. This is a regular index 1 system in Mattson-Söderlin form. It is equivalent to the original system in the considered mode. Its standard DAE counterpart is obtained by replacing backward shifting by differentiation and the dummy variables are $\widehat{v}_1, \widehat{u}_1, \widehat{u}_2$, see the right column of Figure 10. Observe that we have kept the causality that results from applying the rules of nonstandard semantics in constructing the sets *outvar* and *othervar*. As a result, the equation defining $\widehat{v}_1$ has v_2' in its right hand side. This is acceptable here because we have an expression for v_2' that can be substituted for this derivative.

Comment 3: Can the above be generalized? □

The symmetric mode $[0 < u_1] \wedge [0 < u_2]$ raises similar

$$
\begin{array}{ll}
K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) & i_2 = -i_1 - j_1 - j_2 \\
K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) & w_1 = -x_1 + u_1 + v_1 \\
K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) & v_1 = -u_1 + u_2 + v_2 \\
K_3^\bullet \ -[]- \ (\,|\,\widehat{u}_1, \widehat{v}_1, v_2^\bullet, \widehat{u}_2) & \widehat{v}_1 = -\widehat{u}_1 + \widehat{u}_2 + v_2' \\
K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) & w_2 = -x_2 + u_2 + v_2 \\
R_1 \ -[]- \ (\,|\,x_1, j_1) & x_1 = R_1 j_1 \\
R_2 \ -[]- \ (\,|\,x_2, j_2) & x_2 = R_2 j_2 \\
L_1 \ -[]- \ (w_1, \,|\, j_1^\bullet) & L_1 j_1' = w_1 \\
L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) & L_2 j_2' = w_2 \\
C_1 \ -[]- \ (\,|\,i_1, \widehat{v}_1, v_1) & i_1 = C_1 \widehat{v}_1 \\
C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) & C_2 v_2' = i_2 \\
D_2^= \ -[]- \ (\,|\,u_1) & u_1 = 0 \\
D_2^{=\bullet} \ -[]- \ (\,|\,\widehat{u}_1) & \widehat{u}_1 = 0 \\
D_5^= \ -[]- \ (\,|\,u_2) & u_2 = 0 \\
D_5^{=\bullet} \ -[]- \ (\,|\,\widehat{u}_2) & \widehat{u}_2 = 0
\end{array}
$$

Figure 10. Left: introducing dummy variables in mode $0 < i_1 \wedge 0 < i_2$; notation $-[]-\ (.\,|\,.)$ stands for $-[0 < i_1 \wedge 0 < i_2]-\ (.\,|\,.)$. Right: corresponding DAE system.

difficulties, we report it in Figure 11.

$$
\begin{array}{l}
K_1 \ -[]- \ (\,|\,j_1, j_2, i_1, i_2) \\
K_1^\bullet \ -[]- \ (\,|\,\widehat{j}_1, \widehat{i}_1, \widehat{i}_2, j_2^\bullet) \\
K_2 \ -[]- \ (\,|\,x_1, w_1, u_1, v_1) \\
K_3 \ -[]- \ (\,|\,u_1, v_1, u_2, v_2) \\
K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) \\
R_1 \ -[]- \ (\,|\,x_1, j_1) \\
R_2 \ -[]- \ (\,|\,x_2, j_2) \\
L_1 \ -[]- \ (\,|\,w_1, \widehat{j}_1, j_1) \\
L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) \\
C_1 \ -[]- \ (i_1 \,|\, v_1^\bullet) \\
C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) \\
D_1^= \ -[]- \ (\,|\,i_1) \\
D_1^{=\bullet} \ -[]- \ (\,|\,\widehat{i}_1) \\
D_4^= \ -[]- \ (\,|\,i_2) \\
D_4^{=\bullet} \ -[]- \ (\,|\,\widehat{i}_2)
\end{array}
$$

Figure 11. The symmetric mode $[0 < u_1] \wedge [0 < u_2]$; notation $-[]-\ (.\,|\,.)$ stands for $-[0 < u_1 \wedge 0 < u_2]-\ (.\,|\,.)$.

2) Handling mode changes: Here we use again the analysis developed in Section VI-C. For our example, cascades of mode changes have length one. Albert: Too complex an example to investigate the mode changes.

3) The whole model: The whole abstract model is summarized in Figure 12, left. Each equation in this model indicates the mode predicate in which it holds. For each such mode predicate, the corresponding line indicates, on the right, the corresponding standard static or ODE equation. In each mode the system has Mattson-Söderlin form.

VIII. Hints for a hybrid system language and compiler

In this section we formulate hints for the design a DAE hybrid system language and its compilation scheme.

A. Have the right primitives

The language may be rich, but it should be built on top of a small number of primitive statements, for which a precise semantics should be developed. Richer constructs should inherit their semantics by macro expansion—this does

$$
\begin{array}{ll}
K_1 \ -[I_1 \wedge I_2]- \ (\,|\,j_1, j_2, i_1, i_2) & j_1 = -j_2 - i_1 - i_2 \\
K_1 \ -[U_1 \wedge I_2]- \ (\,|\,j_1, j_2, i_1, i_2) & i_2 = -i_1 - j_1 - j_2 \\
K_1 \ -[I_1 \wedge U_2]- \ (\,|\,j_1, j_2, i_1, i_2) & i_1 = -i_2 - j_1 - j_2 \\
K_1 \ -[U_1 \wedge U_2]- \ (\,|\,j_1, j_2, i_1, i_2) & j_1 = -j_2 - i_1 - i_2 \\
K_1^\bullet \ -[U_1 \wedge U_2]- \ (\,|\,\widehat{j}_1, \widehat{i}_1, \widehat{i}_2, j_2^\bullet) & \widehat{j}_1 = -j_2' - \widehat{i}_1 - \widehat{i}_2 \\
K_2 \ -[\neg(U_1 \wedge U_2)]- \ (\,|\,x_1, w_1, u_1, v_1) & w_1 = -x_1 + u_1 + v_1 \\
K_2 \ -[U_1 \wedge U_2]- \ (\,|\,x_1, w_1, u_1, v_1) & u_1 = -v_1 + x_1 + w_1 \\
K_3 \ -[I_1 \wedge U_2]- \ (\,|\,u_1, v_1, u_2, v_2) & u_2 = -v_2 + u_1 + v_1 \\
K_3 \ -[U_1 \wedge U_2]- \ (\,|\,u_1, v_1, u_2, v_2) & u_2 = -v_2 + u_1 + v_1 \\
K_3 \ -[U_1 \wedge I_2]- \ (\,|\,u_1, v_1, u_2, v_2) & u_1 = -v_1 + u_2 + v_2 \\
K_3 \ -[I_1 \wedge I_2]- \ (\,|\,u_1, v_1, u_2, v_2) & v_1 = -u_1 + u_2 + v_2 \\
K_3^\bullet \ -[I_1 \wedge I_2]- \ (\,|\,\widehat{u}_1, \widehat{v}_1, \widehat{u}_2, v_2^\bullet) & \widehat{v}_1 = -\widehat{u}_1 + \widehat{u}_2 + v_2' \\
K_4 \ -[]- \ (\,|\,u_2, v_2, x_2, w_2) & w_2 = -x_2 + u_2 + v_2 \\
 & \\
R_1 \ -[]- \ (\,|\,x_1, j_1) & x_1 = R_1 j_1 \\
R_2 \ -[]- \ (\,|\,x_2, j_2) & x_2 = R_2 j_2 \\
 & \\
L_1 \ -[\neg(U_1 \wedge U_2)]- \ (w_1 \,|\, j_1^\bullet) & L_1 j_1' = w_1 \\
L_1 \ -[U_1 \wedge U_2]- \ (\,|\,w_1, \widehat{j}_1, j_1) & w_1 = L_1 \widehat{j}_1 \\
L_2 \ -[]- \ (w_2 \,|\, j_2^\bullet) & L_2 j_2' = w_2 \\
 & \\
C_1 \ -[\neg(I_1 \wedge I_2)]- \ (i_1 \,|\, v_1^\bullet) & C_1 v_1' = i_1 \\
C_1 \ -[I_1 \wedge I_2]- \ (\,|\,i_1, \widehat{v}_1, v_1) & i_1 = C_1 \widehat{v}_1 \\
C_2 \ -[]- \ (i_2 \,|\, v_2^\bullet) & C_1 v_2' = i_2 \\
 & \\
D_1^= \ -[U_1]- \ (\,|\,i_1) & i_1 = 0 \\
D_1^{=\bullet} \ -[U_1]- \ (\,|\,\widehat{i}_1) & \widehat{i}_1 = 0 \\
D_2^= \ -[I_1]- \ (\,|\,u_1) & u_1 = 0 \\
D_2^{=\bullet} \ -[I_1]- \ (\,|\,\widehat{u}_1) & \widehat{u}_1 = 0 \\
\text{F} = U_1 \wedge I_1 & \\
D_4^= \ -[U_2]- \ (\,|\,i_2) & i_2 = 0 \\
D_4^{=\bullet} \ -[U_2]- \ (\,|\,\widehat{i}_2) & \widehat{i}_2 = 0 \\
D_5^= \ -[I_2]- \ (\,|\,u_2) & u_2 = 0 \\
D_5^{=\bullet} \ -[I_2]- \ (\,|\,\widehat{u}_2) & \widehat{u}_2 = 0 \\
\text{F} = U_2 \wedge I_2 &
\end{array}
$$

Figure 12. Left: the whole model; I_1 denotes the guard $i_1 > 0$ and we use a similar convention for I_2, U_1, U_2. Right: for each mode the corresponding standard equation.

not mean that macro expansion must be part of the compilation process.

The basic primitive statement is (38), which can be reformulated as

$$\textbf{on } P(x) : F(x, \dot{x}) = 0 \tag{58}$$

As we have seen in Section VI-A, (58) specializes to a number of useful constructs:

(a) mode changes due to zero-crossings, followed by resets;
(b) unilateral constraints;
(c) complementarity conditions.

This suggests that *modes* are defined by boolean predicates over system states and variables and *events* are defined as changes in mode configuration. Considerations of numerical quality, however, mitigate this option as predicates take their values in the boolean domain, which prevents from assessing the risk of seing an event soon. Higher-level constructs to express modes such as (a)-(c) above, are preferred from this point of view.

Nevertheless, events should be mode changes and easily identifiable from the syntax. This disciplin will make it easier to develop Discrete/Continuous typing.

B. Make Discrete/Continuous typing easy

The language should make it easy to separate between

- the duty of the DAE/ODE continuous-time solver(s), and
- the duty of the event-based discrete coordinator.[12]

In our previous work and the Zelus prototype [6], typing was about Discrete/Continuous (D/C). By decree, we decided that discrete events could only result from *zero-crossings,* i.e., some real-valued signal of type C crossing zero upward.[13] Such zero-crossings could only be created by using a particular statement of the Zelus language. With this disciplin, D/C typing became feasible and even relatively simple. Any periodic clock could be seen as a macro on top of former primitive statements, where a tick occurs at certain level-crossings of the C signal "time", solution of $\dot{t} = 1$—bypasses are, of course, highly desirable for efficiency issues but their semantics should be that of the above macro expansion.

As seen from the introductory discussion of Section I-C, however, not all events should be deferred to the discrete coordinator. It is therefore highly advisable to offer different statements for declaring

- mode changes that remain local to the (sub)system in consideration and are supported by DAE/ODE solvers;
- mode changes that serve to synchronize the reset of other subsystems, and, thus, need to be visible outside the (sub)system in consideration.

C. Have Pantelides graphs in the language

Guarded Pantelides graphs were extensively used in the Elmqvist-Mattson example of Section VII-D. Such graphs are useful in many respects:

- They can be used to implement index reduction;
- They support causality analysis and synthesis.

Hence, they are instrumental in deriving all possible scheduling of the atomic actions performed by the modeler at run time. Concurrency may be exhibited in this way and used to run multiple solvers, therefore avoiding unwanted coupling, see the introductory discussion of Section I-A.

An important remark is that guarded Pantelides graphs are constraints in some abstract domain represented by the two statuses $\{candidate\_output, cannot\_be\_output\}$. Viewing Pantelides graphs as contraints pave the way toward compositional techniques for index reduction and causality synthesis in hybrid DAE systems. This technique was not used in the example of Section VII-D and remains to be developed.

D. Develop a nonstandard semantics

It is advisable to develop a nonstandard semantics for the language. DAE systems, as we know, require smoothness conditions for them to behave properly and possess solutions in the sense of Definition 1. This becomes worse for hybrid DAE systems, since, in addition, Zeno/nonZeno issues come into play. For this reason, the tradition has been in the modeling and simulation community, that no attention is paid by the tool regarding smoothness conditions—they are under the responsibility of the engineer developing the model.

Computer scientists know the benefit of being able to reject "spurious programs", whatever they be. The lenghty and verbose warnings returned by Simulink are a good example of an attempt to do this. We believe that the use of the nonstandard semantics allows to sort out between two kinds of spuriousness, namely:

- Vicious mix and wild combination of Discrete and Continuous signals and operators—these should be preferrably detected at compile time;
- Lack of needed smoothness, excessive stiffness, near-Zeno condition—these cannot be handled by the compiler and must be addressed using a proper understanding of the physics.

[12] We think of it as a slightly adapted synchronous language engine.

[13] See the discussion on page 391, section 6 of [4] regarding this definition of "discrete".

References

[1] Acary Vincent and Brogliato Bernard, *Numerical Methods for Nonsmooth Dynamical Systems*, ser. Lecture Notes in Applied and Computational Mechanics. Springer-Verlag, 2008, vol. 35.

[2] A. Benveniste, T. Bourke, B. Caillaud, and M. Pouzet, "Divide and recycle: types and compilation for a hybrid synchronous language," in *LCTES*, J. Vitek and B. D. Sutter, Eds. ACM, 2011, pp. 61–70.

[3] ——, "A hybrid synchronous language with hierarchical automata: static typing and translation to synchronous code," in *EMSOFT*, S. Chakraborty, A. Jerraya, S. K. Baruah, and S. Fischmeister, Eds. ACM, 2011, pp. 137–148.

[4] ——, "Non-standard semantics of hybrid systems modelers," *J. Comput. Syst. Sci.*, vol. 78, no. 3, pp. 877–910, 2012.

[5] A. Benveniste, P. Caspi, S. A. Edwards, N. Halbwachs, P. L. Guernic, and R. de Simone, "The synchronous languages 12 years later," *Proceedings of the IEEE*, vol. 91, no. 1, pp. 64–83, 2003.

[6] T. Bourke and M. Pouzet, "Zélus: a synchronous language with odes," in *HSCC*, C. Belta and F. Ivancic, Eds. ACM, 2013, pp. 113–118.

[7] E. Lee and H. Zheng, "Operational semantics of hybrid systems," in *HSCC*, 2005, pp. 25–53.

[8] E. A. Lee and H. Zheng, "Leveraging synchronous language principles for heterogeneous modeling and design of embedded systems," in *EMSOFT*, 2007, pp. 114–123.

[9] K. M. and M. M.D.P., "An introduction to Moreau's sweeping process," in *Impacts in Mechanical Systems – Analysis and Modelling*, ser. Lecture Notes in Physics, B. B., Ed., vol. 551. Springer-Verlag, 2000, pp. 1–60.

[10] C. Pantelides, "The consistent initialization of differential-algebraic systems," *SIAM J. Sci. Stat. Comput.*, vol. 9, no. 2, pp. 213–231, 1988.

[11] Stephen L. Campbell and C. William Gear, "The index of general nonlinear DAEs," *Numer. Math.*, vol. 72, pp. 173–196, 1995.

[12] Sven Erik Mattson and Gustaf Söderlin, "Index reduction in Differential-Algebraic Equations using dummy derivatives," *Siam J. Sci. Comput.*, vol. 14, no. 3, pp. 677–692, 1993.