

Chapter 2

Bond Graph Representations of Hybrid System Models

This chapter addresses the modelling abstraction of fast dynamic state transitions into instantaneous discrete state changes and its consequences and surveys various bond graph representations of hybrid system models proposed in the literature. The bond graph model-based quantitative approach to FDI in hybrid systems presented in Chap. 4 uses a bond graph representation with system mode independent computational causalities. As a result, a single set of equations in the form of a DAE system can be derived that holds for all system modes. The chapter concludes by addressing the index of the DAE system as it is an information that is relevant with regard to a symbolic and numerical processing of the DAE system.

2.1 Hybrid System Models

The way an engineering system is modelled depends on the purpose of a modelling and simulation study. Depending on the kind of problem or on the design stage modellers take different views of a system that either exists or is yet to be developed. For instance, it is well known that integrated digital circuits are modelled and simulated at different levels of abstractions. A hierarchical approach may start at the circuit level by accounting for the electrical dynamical behaviour of transistors, resistors and capacitors. One level up the hierarchy, the electrical behaviour is still of interest but transistors are abstracted into switches. State transitions are accounted for by rise and fall times. At the gate level the focus is on the logic behaviour of the circuits. The fact that a logic gate is built by means of electrical elements is taken into account by the time a signal needs to pass the logic gate. A top-down hierarchical design of complex digital circuitry starts at even higher levels of abstraction such as the register transfer level.

Besides the fast switching of transistors, electrical diodes or thyristors, fast changing phenomena such as the opening or closing of hydraulic check valves, the engagement-disengagement of clutches, stick-slip effects or stops in mechanical motion give rise to the modelling abstraction that such mode changes take place instantaneously and may be considered discrete events that either happen autonomously caused by internal continuous time variables that have crossed some

threshold values or are due to external control signals. Such a modelling abstraction is justified because the continuous time evolution of fast mode transitions during short time intervals is not relevant for the overall dynamic system behaviour. That is, there are discrete events at which switch state variables instantaneously switch between discrete values. The set of all discrete switch states $m_j(t)$, $j = 1, \dots, n_f$ characterises a system mode at time instant $t > 0$. For the time span of a system mode, the discrete switch state variables take constant values while the system behaviour is described by a continuous time model that is mode specific. Hence, if the abstraction of instantaneous mode switches is adopted then a model encompasses discrete events at which mode changes instantaneously take place and continuous time models for the dynamic system behaviour in each mode. Such models are usually called *hybrid* system models.

Adopting a hybrid system model entails a number of consequences.

- If fast switching devices are modelled as switches then parts of an overall system model are instantaneously disconnected or reconnected for a while. That is, an overall model is of variable structure. The result of such a structural change is a continuous time model that holds for the resulting system mode.
- The number of state variables of an overall model is not time-invariant but mode dependent. Due to a connection of model parts storage elements may become dependent, e.g. if a clutch is engaged or if rigid bodies stick together for a while due to Coulomb friction.

In [1], Cellier and Kofman call a model a *variable structure model* if structural properties such as the number of differential equations depend on the discrete state of some switches.

- The formulation of a mathematical model becomes mode dependent. In one mode it may be an explicit state space model. If storage elements become dependent the model turns into a set of differential-algebraic equations (DAEs).
- The numerical integration of the model equations often requires the determination of the time instant of the discrete events and a reinitialisation. Hence, the numerical computation of a hybrid system model may be viewed as the solution of a sequence of initial value problems (IVPs). Modern numerical solvers for DAE systems such as IDA [2] from the SUNDIALS suite or DASRT [3] provide a root finding feature such that the time instances of mode switches can be located.

If an *event condition* rather than the time instant of an event is known, the event is commonly denoted as a *state event*. The event condition is specified by a function of continuous variables and a state event occurs when the function crosses zero. If the time instant of a discrete event is known in advance, the event is called a *time event*. In that case, the event can be scheduled to happen by entering its time and its type into a calendar of forthcoming events.

The computation of models with variable structure is still a subject of ongoing research as latest publications show [4, 5]. With regard to a modelling by means of bond graphs the question arises how to represent a hybrid system model in a bond graph framework.

2.2 Bond Graph Representations of Hybrid System Models

Bond graph modelling starts from considering the exchange of energy between system components and energy conversion. As the exchange and the conversion of energy happen continuously with respect to time, bond graph modelling was, at first, limited to the representation of continuous time models. Bond graphs were mainly developed to obtain a mathematical model in the form of a system of ODEs or DAEs for the simulation of the dynamic behaviour of a system or to obtain transfer functions for the analysis of the frequency domain behaviour, for the analysis of structural observability and controllability, or for the design of a controller. Besides, various proposals to extend the bond graph modelling framework to capture discrete discontinuous events have evolved during the past two decades. So far, none of the reported approaches has attained common usage.

Combination of Petri Nets and Bond Graphs

One approach is to combine the advantages of established graphical representations by capturing physically feasible system modes and discrete transitions between them in a Petri net (PN) and by modelling the dynamic behaviour in each system mode by a bond graph or set of disjoint bond graphs composed of standard elements [6, 7]. Accordingly,

- the dynamic system behaviour is modelled by a set of bond graphs each of which has time-invariant computational causalities,
- each bond graph holds for the time period between two discrete events,
- for each system mode, the structure of the bond graph can be different,
- the mathematical model is a set of initial value problems.

Other approaches prefer a uniform model representation and suggest to let bond graphs capture also instantaneous discontinuous state changes although power conservation means that it takes a finite time to change the content of an energy storage element. Bond graph representations of hybrid models can be categorised into those that aim to keep computational causalities system mode independent and those accepting variable causalities.

Switching Off and On Degrees of Freedom by Means of Switched Residual Sinks

One way to keep computational causalities time-invariant as proposed in [8] is to use sinks of invariant causality in conjunction with a modulated transformer MTF : $b(t)$ that switches off and on degrees of freedom.

Example: Bouncing Ball

Figure 2.1 depicts the well known bouncing ball problem and a bond graph representation of a hybrid model.

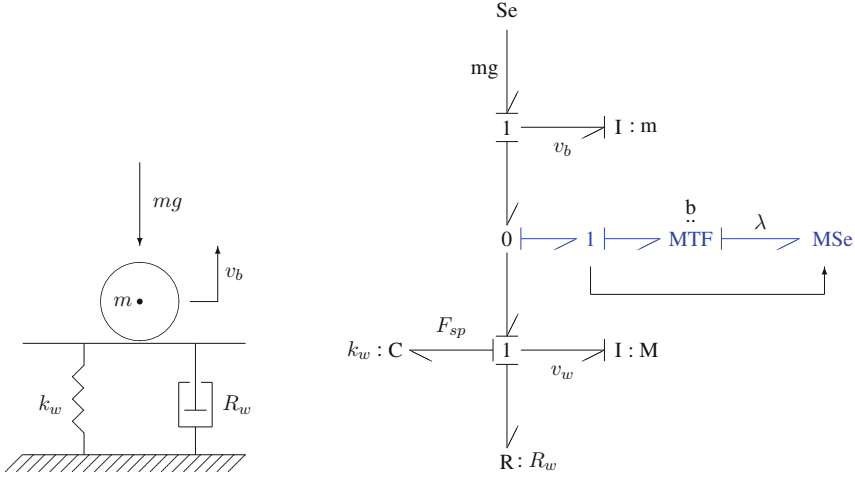
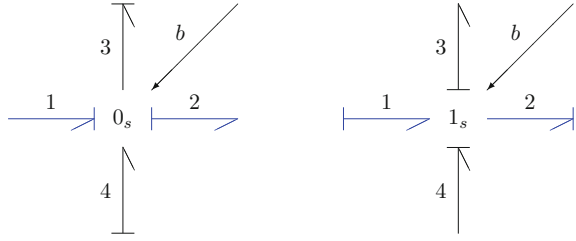


Fig. 2.1 Switching off and on degrees of freedom by sinks of invariant causality [6]

In the bond graph of Fig. 2.1, $b(t) \in \{0, 1\} \forall t \geq 0$. For $b = 0$, there is no force between the floor and the ball. The ball is freely moving in the air. If, however, the ball is in contact with the ground ($b = 1$) then the modulated effort source MSe imposes a constraint force λ which enforces that the ground and the ball move with the same velocity. Both inertia elements are in fixed integral causality. When the ball drops to the floor numerical integration restarts with new initial conditions. Modulated sinks imposing either an effort or a flow so that their input variable vanishes are termed *residual sinks*. As to the representation of the residual sink in Fig. 2.1, one might ask whether the relative velocity $v_b - v_w$ is fed into the modulated effort sink as a signal when this information is already delivered into the sink by the power bond attached to it. A standard effort sink Se does not serve the intended purpose as its output is independent of the input. In this case, a sink is needed that delivers an output so that the input vanishes. To that end, the input is sensed and fed as a signal into a modulated sink. The presentation has been adopted from [9] where it is introduced to display controlled constraint forces in multibond graphs of rigid multibody systems. Actually, these modulated sinks represent Lagrange multipliers that are not limited to mechanical systems modelling and can be used for various other purposes as well, e.g. to add tearing information into a bond graph [10]. In the context of FDI, residual sinks can be used to couple a behavioural bond graph model of a system subject to faults to a bond graph of the faultless system with nominal parameters [11]. In Chap. 4, they are used in this way to numerically determine ARR residuals. For the sake of a concise bond graph representation residual sinks are denoted by the standard symbol of a source or sink prefixed by the letter ‘r’, e.g. rSe denotes a residuals effort sink.

Fig. 2.2 Switched power junctions [6]

Switched Power Junctions (SPJs)

In order to account for the abstraction of ideal, no power consuming switching in a bond graph with invariant causalities that holds for all system modes, Umarikar extended 0- and 1-junctions by allowing for more than one bond to impose an effort on a 0-junction and more than one bond imposing a flow on a 1-junction with the constraint that only one of the causality imposing bonds is active at a time instant [12, 13]. These extensions are called *switched power junctions* and are not to be confused with *controlled junctions* to be referred to subsequently. Figure 2.2 illustrates the idea.

For a switched power junction holds

$$\forall t \geq 0 \quad \exists j \in \mathbb{N} \quad (m_j(t) = 1 \wedge m_i(t) = 0 \quad \forall i \neq j)$$

The equations for a switched power 0-junction read

$$e_3(t) = m_1(t)e_1(t) + m_2(t)e_2(t) = e_4(t) \quad (2.1a)$$

$$f_1(t) = m_1(t)(f_3(t) - f_4(t)) \quad (2.1b)$$

$$f_2(t) = m_2(t)(f_4(t) - f_3(t)) \quad (2.1c)$$

Accordingly, equations can be formulated for a switched power 1-junction.

Example: Boost Converter

The bond graph of a boost converter in Fig. 2.3 illustrates the use of these extended junctions.

Summation of efforts at the encircled left 1-junction and summation of flows at the rightmost encircled 0-junction yields mode dependent state equations (2.2a)–(2.2b)

$$1 : \quad L \frac{d}{dt} i_L = E - m_2 V_C - R_L i_L \quad (2.2a)$$

$$0 : \quad C \frac{d}{dt} V_C = m_2 i_L - \frac{V_C}{R} \quad (2.2b)$$

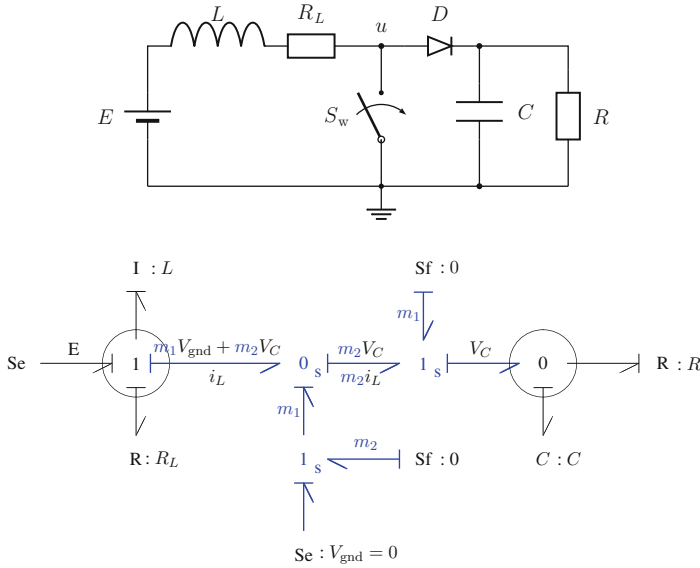


Fig. 2.3 Bond graph with SPJs of a boost converter [12]

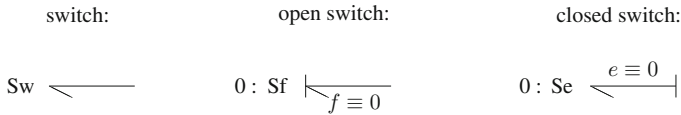


Fig. 2.4 The ideal switch as an additional bond graph element

The Ideal Switch as a Bond Graph Element

If one is willing to accept that computational causalities in bond graphs are mode dependent and if one neglects the energy loss associated with an instantaneous change of a switch state then fast switching devices may be modelled as an ideal switch [14–18] and represented by a bond graph switch element Sw or by controlled junctions [19–21]. In the case of an open switch, the bond graph switch Sw becomes a zero flow source $Se : 0$. In the case of a closed switch the bond graph switch element turns into a zero flow source $Sf : 0$. Hence, the causality at a switch port is mode dependent as displayed in Fig. 2.4.

Remark 2.1 Van der Schaft and Schuhmacher call variables such as the power variables of the bond graph switch element Sw *complementary* variables in the sense that for the two of them an inequality holds and for all times at least one of them is strictly an equality. Systems in which mode switching is determined by *complementarity conditions* they call *complementary systems* [22]. In the case that both variables can be assumed to be nonnegative, the complementarity condition is often expressed as

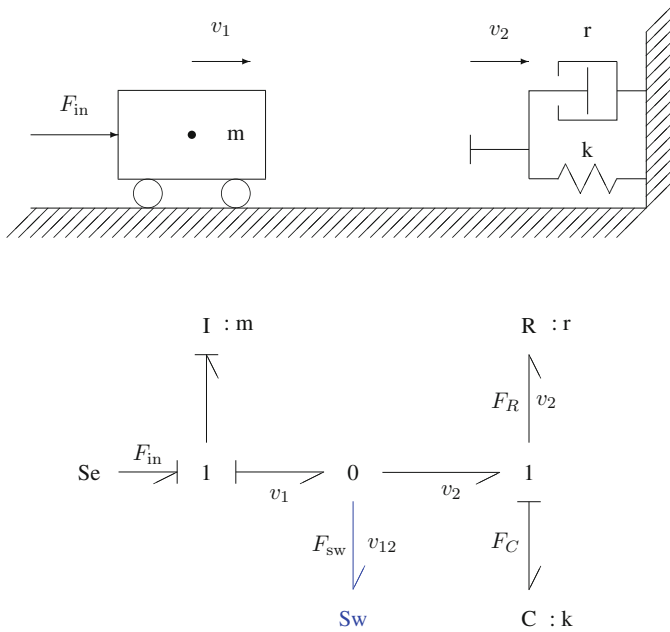


Fig. 2.5 Modelling a mechanical stop by means of an ideal switch [6, 18]

$$0 \leq f(t) \perp e(t) \geq 0 \quad \text{or} \quad 0 \leq e(t) - f(t) \geq 0 \quad (2.3)$$

In general, the complementary variables can be vectors if inequalities hold componentwise.

Let \mathbf{M} be a $k \times k$ matrix and \mathbf{q} a k -dimensional real vector. The so-called *linear complimentary problem* then consists in finding k -dimensional vectors $\boldsymbol{\lambda}$ and \mathbf{y} such that

$$\mathbf{y} = \mathbf{M}\boldsymbol{\lambda} + \mathbf{q} \quad (2.4a)$$

$$\mathbf{0} \leq \mathbf{y} - \boldsymbol{\lambda} \geq \mathbf{0} \quad (2.4b)$$

This problem has a unique solution if the matrix fulfills a certain condition [22]. \square

Example: Mechanical Stop

In the bond graph of Fig. 2.5, an ideal switch Sw is used to model an elastic mechanical stop.

Clearly, if the cart is not in contact with the stop then $F_{sw} = 0 \wedge v_{12} := v_1 - v_2 \geq 0$. If the cart is in contact with the stop then $F_{sw} \geq 0 \wedge v_{12} = 0$. That is, $0 \leq F_{sw} \perp v_{12} \geq 0$. The force F_{sw} may be thought of as a Lagrange multiplier λ which is either zero or takes a value to be computed.

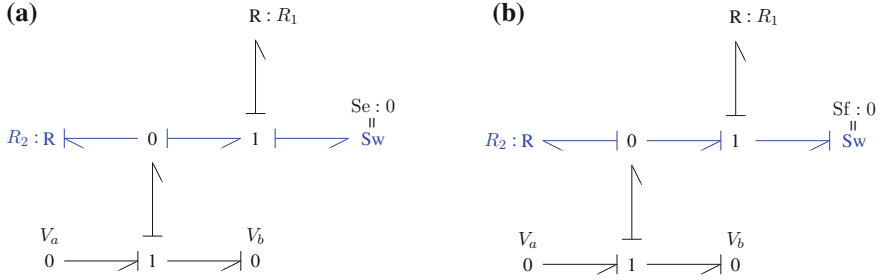


Fig. 2.7 Bond graph model of an electrical diode using an ideal switch Sw and a ‘causality resistor’ $R : R_2$. **a** Ideal switch closed. **b** Ideal switch open

Fig. 2.8 Piecewise linear model of an electrical diode

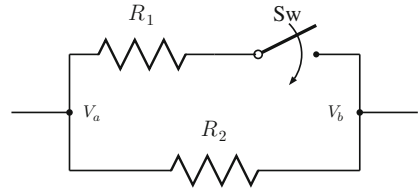


Fig. 2.7 may be a piecewise linear model of an electrical diode. The corresponding circuit schematic is displayed in Fig. 2.8.

The causality resistor $R : R_2$ clearly avoids the propagation of causality changes at the port of the ideal switch into the rest of the bond graph and captures the diode’s high resistance R_{off} in reverse mode. The resistor $R : R_1$ represents the diode’s small ON-resistance R_{on} .

Let i_D denote the current through the diode and $\Delta V := V_a - V_b$. Then from the bond graph in Fig. 2.7a the equation

$$i_D = \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \Delta V = \left(\frac{1}{R_{\text{on}}} + \frac{1}{R_{\text{off}}} \right) \Delta V \approx \frac{1}{R_{\text{on}}} \Delta V \quad (2.7)$$

is immediately deduced, where $R_{\text{off}} \gg R_{\text{on}}$. Likewise, the causal bond graph in Fig. 2.7b gives

$$i_D = i_{R_2} = \frac{1}{R_2} \Delta V = \frac{1}{R_{\text{off}}} \Delta V \quad (2.8)$$

As in the case of a mechanical stop (cf. Fig. 2.6), the bond graphs for both modes in Fig. 2.7 can be merged into one single bond graph from which equations valid for both modes can be deduced.

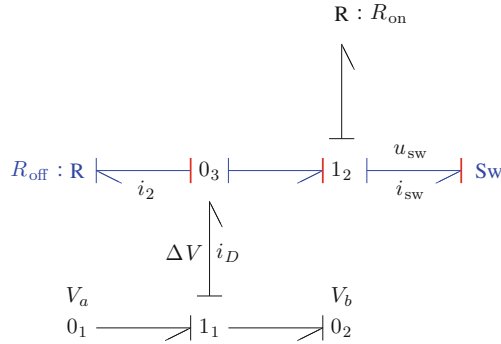


Fig. 2.9 Bond graph of the diode model capturing both modes

From the bond graph in Fig. 2.9 the following equations can be derived.

$$i_2 = \frac{\Delta V}{R_{\text{off}}} \quad (2.9a)$$

$$i_D = i_2 + i_{\text{sw}} \quad (2.9b)$$

$$\Delta V = R_{\text{on}} i_{\text{sw}} + u_{\text{sw}} \quad (2.9c)$$

The constitutive equation of the switch may be expressed by means of a variable $b(t) \in \{0, 1\} \forall t \geq 0$. Let $\bar{b} := 1 - b$. Then

$$0 = b i_{\text{sw}} + \bar{b} u_{\text{sw}}. \quad (2.10)$$

That is, $b(t) = 0$ means that the switch is closed, $b(t) = 1$ indicates that the switch is open.

From this set of equations an equation for the current i_D can be obtained that holds for both modes.

$$i_D = \left(\frac{1}{R_{\text{off}}} + \frac{\bar{b}}{\bar{b} R_{\text{on}} - b} \right) \Delta V \quad (2.11)$$

Controlled Junctions

As early as 1974, Thoma introduced the concept of *time dependent junctions* [24] in order to switch off and on connections between power ports. Mosterman picked up this idea and introduced *controlled junctions* [21]. In contrast to switched power junctions, a local control algorithm associated with a controlled junction switches off all adjacent bonds of a controlled junction when a switching device considered as an ideal switch turns off and re-activates all bonds when the switch is closed. That is, an ideal switch in ON-mode is represented by a standard 0- or 1-junction. In OFF-mode the junction is replaced by a source of value zero as shown in Fig. 2.10.

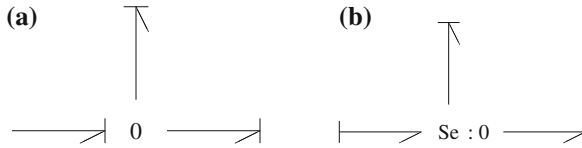


Fig. 2.10 Controlled 0-junction. **a** ON mode. **b** OFF mode

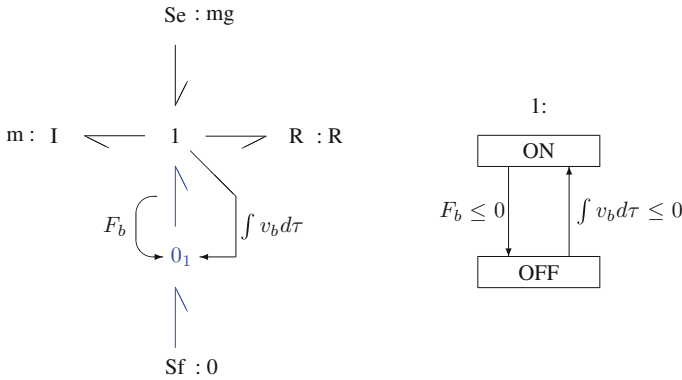


Fig. 2.11 Hybrid bond graph of the bouncing ball problem and a finite state automaton associated with the controlled 0-junction 0_1 [6, 21]

As Fig. 2.10 shows, a discrete switch state change entails a change in causality that is to be propagated into the bond graph. Bond graphs using controlled junctions are usually called *Hybrid Bond Graphs* (HBGs).

Figure 2.11 displays a hybrid bond graph for the well known bouncing ball problem given the assumption of a perfect non-elastic collision between the ball and the floor. As the focus of the bond graph methodology is to model the continuous time exchange of energy between system components and possible energy conversion and since none of the proposed extensions to include discrete events has become a widely accepted standard, there is not much bond graph software that support hybrid bond graphs. Mosterman has developed a software HYBrSIM especially suited for continuous-discrete event bond graph modelling using controlled junctions [25]. In the tool suite MoTHS, hybrid bond graphs are transformed into block diagrams by using block diagram models that can be efficiently reconfigured for those parts of a hybrid bond graph that need a re-assignment of causality after a system mode change has happened [20]. Wang and his co-workers [26] also map hybrid bond graphs onto block diagrams for simulation with Simulink¹[27].

¹ Simulink is a registered trademark of The MathWorks, 3 Apple Hill Drive, Natick, MA, USA, <http://www.MathWorks.com>.

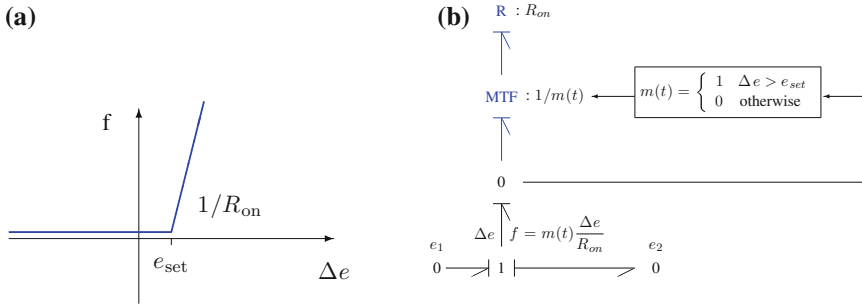


Fig. 2.12 Bond graph representation of a non-ideal switch. **a** Piecewise linear approximation of the characteristic of a diode. **b** Bond graph model of a switching device with $R : R_{on}$ in fixed conductance causality

Non-ideal Switches

Finally, another way to keep computational causalities in a bond graph system mode independent is to model fast switching devices as non-ideal switches and to account for a resistance in ON-mode. The functionality of electronic diodes or hydraulic check valves suggests to represent a non-ideal switch in a bond graph by means of a resistor with conductance causality. In [28], the nonlinear characteristic of a diode is approximated by a piecewise linear one. The switching is explicitly represented by decomposing the resistor into an MTF with a modulus $m(t) \in \{0, 1\} \forall t \geq 0$ and a resistor $R : R_{on}$ accounting for the ON-resistance as displayed in Fig. 2.12. Depending on the value of the transformer modulus $m(t)$, model parts linked by such MTFs are either connected or disconnected. For s switches in a hybrid model, there are $s_f \leq 2^s$ physically feasible switch state combinations. Each one of them constitutes a system mode.

In this book, the bond graph representation of a non-ideal switch proposed by Ducreux et al. as early as 1993 is used in a bond graph model-based approach to FDI in hybrid systems. This approach offers the following advantages.

- Switching devices, e.g. semiconductor switches in power electronic systems that are explicitly represented by a component model can be more easily identified in a bond graph in comparison to a representation that accounts for switches by controlled junctions. An initial not simplified bond graph is more close to a system schematic (cf. Chap. 4, Figs. 4.2 and 4.3).
- One single bond graph can be set up that holds for all system modes.
- The standard Sequential Causality Assignment Procedure (SCAP) can be applied without any modification resulting in time-invariant computational causalities.
- A single set of model equations as well as equations for fault indicators valid for all system modes can be (automatically) derived from the bond graph. They contain the discrete moduli $m_i(t) \in \{0, 1\}$, $i = 1, \dots, s$, of the switching MTFs. Inserting values for the moduli belonging to a feasible switch state combination gives the equations that hold for the associated system mode.

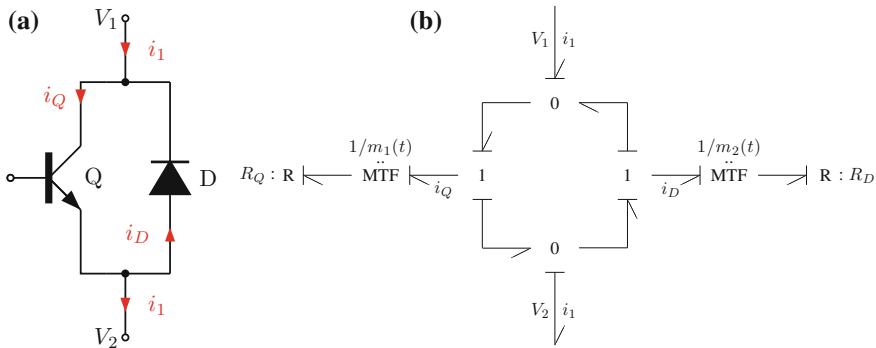


Fig. 2.13 Bipolar transistor with a reversed biased diode in parallel often used in power electronic converters and its bond graph representation. **a** Transistor-diode pair. **b** Bond graph representation

- Existing software such as SYMBOLS^{TM2} [29] could be used to generate a set of ARR.

Semiconductor switches in power electronic inverters are commonly made up of a transistor together with a diode in anti-parallel connection as depicted in Fig. 2.13a for a bipolar transistor to provide a path for an inductive load current when conducting switches are turned off and thus to avoid damage of the transistors in an inverter. MOSFET transistors have a built-in diode. Figure 2.13b shows a bond graph model of such a transistor-diode pair.

The time invariant conductance causality of the ON resistor of non-ideal switches may lead to conflicts at junctions that can be resolved by adding a storage element. Equations derived from a bond graph may be symbolically formulated in such a way that the parameter of these additional storage elements can be set to zero as well as the small ON resistance of some switches turning them into ideal switches and avoiding small time constants. That is, ideal switches may be considered the limit case of non-ideal switches. Avoiding stiff equations is of interest in the case when an explicit formulation of ARRs is not possible so that the model equations need to be solved numerically in order to determine the residuals of ARRs. In [30], Buisson et al. compare the use of ideal and non-ideal switches in a bond graph framework in the light of perturbation theory.

For the sake of a concise representation in the bond graphs of the examples in this book, a switched MTF : $m(t)$, $m \in \{0, 1\}$, in conjunction with the ON resistance $R : R_{on}$ of a switch is represented as a component model denoted by $Sw : m$ as shown in Fig. 2.14.

Instantaneous Structural Changes

If phenomena such as the engagement of a clutch or a change from slip to stick friction is considered as a discrete event then the instantaneous mode change results

² Symbols is a trademark of HighTech Consultants, STEP, I.I.T.Kharagpur - 721 302, India, <http://www.htcinfo.com/>.

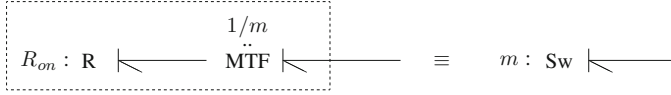


Fig. 2.14 Bond graph model of a non-ideal switch denoted as a component model $\text{Sw} : m$

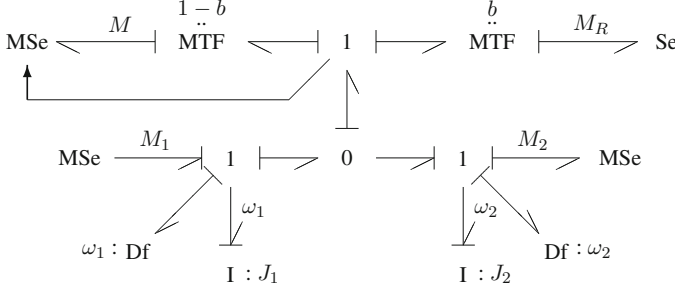


Fig. 2.15 Bond graph of a clutch with fixed mode independent causalities

in a change of the number of state variables. As long as two moving rigid bodies stick together due to friction they build a single body for some time span with one single state variable. In power electronic systems, the OFF state of a switch may cause two inductor currents to become equal. This means that in a bond graph representation of a hybrid model causality at some storage ports may change from integral to derivative causality. Preferred integral causality may, however, be preserved by using a modulated sink that either impose an effort or a flow on two dependent storage elements so that their output variables become equal. They are to be activated whenever a system mode change happens that causes two storage elements to become dependent. Their activation can be achieved by a ‘Boolean’ controlled transformer as depicted for the example of a mechanical clutch.

Example: Mechanical Clutch

Figure 2.15 depicts a bond graph representation of a hybrid model of a clutch.

The left upper modulated effort source provides a moment M that enforces that the plates of the clutch stick together when the clutch is in mode ‘engaged’, i.e. their angular velocities are equal. This moment is imposed on the two I-elements in integral causalities. At the time instant the clutch instantaneously switches from disengagement to engagement, numerical integration must be re-initialised. At this discrete event, the modulated sink is activated and the output of both I-elements jump to a common value and integration restarts from this value.

2.3 Equations Formulation for Switched LTI Systems

The use of such modulated sinks that are activated and deactivated at discrete time instances means that the underlying mathematical model is of the form of a DAE

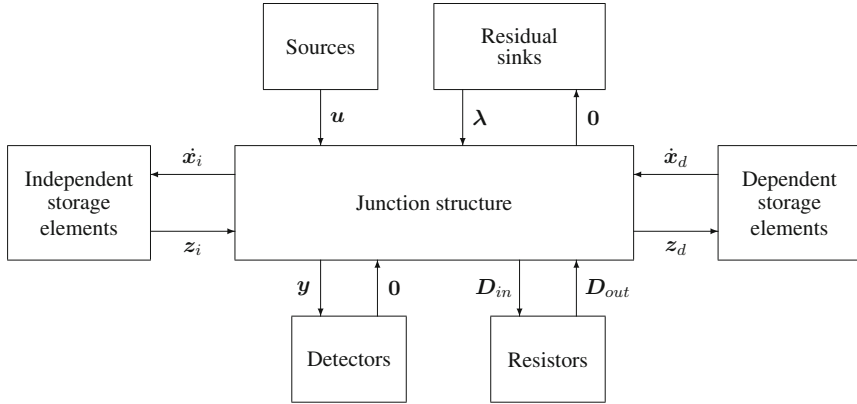


Fig. 2.16 Partitioning of a bond graph model into fields

system. These sinks enforce that algebraic constraints for the output variables of some storage elements are met but there is no differential equation for their output. When the modulated sinks are active, the DAE system is of index 2.

Let the moduli $m_j(t) \in \{0, 1\} \forall t \geq 0, j = 1, \dots, n_s$, of the n_s switching MTFs be grouped into a vector $\mathbf{m}(t)$, the outputs z_i of all storage elements in integral causality, the inputs z_d of all storage elements in derivative causality and the outputs of resistors and switches be grouped into a descriptor vector \mathbf{x} . The outputs of all modulated sinks activated at switching events are combined into a vector λ . Let \mathbf{u} denote the vector of all system inputs.

If it is assumed that the system under consideration can be described by a linear time-invariant model for the time spans between two discrete switching events then, starting from the partitioning of a bond graph model into fields as displayed in Fig. 2.16, a DAE system in the form

$$\begin{bmatrix} \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{x}(t) \\ \lambda(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}(\mathbf{m}(t)) & \mathbf{A}_{12}(\mathbf{m}(t)) \\ \mathbf{A}_{21}(\mathbf{m}(t)) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \lambda(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1(\mathbf{m}(t)) \\ \mathbf{B}_2(\mathbf{m}(t)) \end{bmatrix} \mathbf{u}(t) \quad (2.12)$$

can be established. In (2.12), the matrices are of proper dimensions and $\mathbf{E} = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{0} \end{bmatrix}$.

If, for some time intervals, no residual sinks are switched on via an MTF, then the mathematical model takes the form

$$\mathbf{E} \dot{\mathbf{x}}(t) = \mathbf{A}_{11}(\mathbf{m}(t)) \mathbf{x}(t) + \mathbf{B}_1(\mathbf{m}(t)) \mathbf{u}(t) \quad (2.13)$$

for these time intervals. If the set of algebraic constraints can be solved then the DAE system can be reduced to an explicit linear state equation.

For controlled switching elements, the time instances of the discrete state transitions are known a priori. For other events such as the transition from slip to stick friction, the value of the MTF moduli involved is the result of a local automaton evaluating constraints on model variables. In that case, the time instant of the event is to be determined and numerical integration is to be reset in general.

For the clutch example, the DAE system derived from the BG in Fig. 2.15 reads

$$\begin{aligned} \left[\begin{array}{cc|c} J_1 & 0 & 0 \\ 0 & J_2 & 0 \\ 0 & 0 & 0 \end{array} \right] \frac{d}{dt} \left[\begin{array}{c} \omega_1 \\ \omega_2 \\ M \end{array} \right] &= \left[\begin{array}{cc|c} 0 & 0 & -(1-b) \\ 0 & 0 & (1-b) \\ (1-b) & -(1-b) & 0 \end{array} \right] \left[\begin{array}{c} \omega_1 \\ \omega_2 \\ M \end{array} \right] \\ &+ \left[\begin{array}{ccc} 1 & 0 & -b \\ 0 & -1 & b \\ 0 & 0 & 0 \end{array} \right] \left[\begin{array}{c} M_1 \\ M_2 \\ M_R \end{array} \right] \end{aligned} \quad (2.14)$$

If the clutch is disengaged ($b = 1$) then the DAE system reduces to

$$\left[\begin{array}{cc} J_1 & 0 \\ 0 & J_2 \end{array} \right] \frac{d}{dt} \left[\begin{array}{c} \omega_1 \\ \omega_2 \end{array} \right] = \left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right] \left[\begin{array}{c} \omega_1 \\ \omega_2 \end{array} \right] + \left[\begin{array}{ccc} 1 & 0 & -1 \\ 0 & -1 & 1 \end{array} \right] \left[\begin{array}{c} M_1 \\ M_2 \\ M_R \end{array} \right] \quad (2.15)$$

and could be formulated as an explicit state space equation.

A different bond graph approach to hybrid system modelling recently proposed by Margetts [19] inspired by the work of Buisson [31] and of Mosterman [21] also leads to an implicit formulation of system equations that holds for all system modes and can be systematically constructed from a hybrid bond graph. This approach makes use of controlled junctions and partitions the field of storage elements and the field of resistors into a field with elements having static causalities and another field with elements having causalities that dynamically change. Margetts terms the latter dynamic causalities and expresses them on a hybrid bond graph by a dashed perpendicular causal stroke in addition to the solid line causal stroke assigned by application of the SCAP.

2.4 Index of a DAE System Derived from the Bond Graph of a Hybrid Model

In this book, switching devices such as electrical diodes and transistors, or hydraulic valves are modelled as non-ideal switches represented by a bond graph component model Sw that is composed of a switched MTF and a resistor in fixed conductance causality. The choice of fixed conductance causality is motivated by the fact that it is the flow through the element that is determined by the discrete switch state.

Structural changes such as the engagement and disengagement of a clutch or the change between slip and stick mode for two rigid bodies considered as instantaneous

discrete events cause the number of states in one system mode to change instantaneously at the advent of a discrete event. If two rigid bodies stick together for the duration of that system mode, the two of them may be considered as one body, or one inertia is dependent of the other so that one degree of freedom is temporarily lost. As presented in Sect. 2.3, such structural changes are modelled in a bond graph by means of a switched MTF and a residual sink that supplies an output variable so that its input vanishes. As a result, storage elements that have been assigned integral causality remain in integral causality for all system modes. If linear constitutive element equations are assumed, the equations derived from such a bond graph with static, system mode independent causalities can be formulated as a linear implicit DAE (2.12).

A characteristic of DAEs besides their form is their differentiation index [32]. For a definition and an example see Appendix C. It is an indicator for the problems to be encountered with the numerical solution of a set of DAEs. Systems of index > 1 are usually called higher index DAEs and the higher the index the more severe numerical difficulties can be. As the mathematical description of problems in various disciplines often leads to DAE system, they have been a research subject for more than two decades. A large body of publications and a number software programs for their numerical solution have emerged. DAE systems of index 1 can be safely numerically computed by means of the backward differentiation formula (BDF) [33, 34] implemented in solvers such as the well known and widely used DASSL code [35].

For the solution of higher index problems, index reduction techniques [34] such as the graph-based Pantelides algorithm [36] or the dummy derivative method [37] have been reported in the literature and are used in some software programs. Both methods are based on the symbolic differentiation of some of the constraint equations of a DAE system and thus introduce equations with higher order derivatives of unknown variables with respect to time. This differentiation entails a problem in its own to find a set of consistent initial values. The Pantelides algorithm aiming at the determination of a set of consistent initial conditions can and is used to determine which equations are to be differentiated how many times in order to reduce the index to one or zero. The algorithm assumes that the index does not change. A direct initialization technique for DAE systems has been presented in [38]. Reference [39] considers the problem of consistent initial conditions for switched linear passive network.

Index reduction entails another well known problem, namely that of numerical drift. Original constraint equations get lost by differentiation and cannot be taken into account in the numerical solution of the reduced index system. Hence, the numerical solution of the reduced index problem can only approximate the original constraints. This suggests to keep the original constraints and differentiated equations in the numerical solution of the reduced index problem resulting in more equations than unknowns in the reduced index problem. The dummy derivative method addresses this problem by considering the derivatives of some variables as new independent algebraic variables called *dummy derivatives* so that the number of unknowns matches the number of equations. This approach, however, requires to decide which variables are selected as dummy derivatives and which ones as states.

Hybrid models may be described by a single set of DAEs including discrete state variables, e.g. switch state variables that change their discrete values at discrete events. That is, for each set of values of the discrete state variables representing a physically feasible system mode a set of DAEs is obtained that describes the dynamic behaviour in that system mode. Such a set of n_f DAE systems

$$\mathbf{F}_i(\mathbf{x}(t), \dot{\mathbf{x}}(t), \mathbf{u}(t), \mathbf{y}(t)) = \mathbf{0}, \quad i = 1, \dots, n_f \quad (2.16)$$

together with a set of conditions $i = C(t, \mathbf{x})$ is termed a system of *hybrid* DAEs (HDAEs) where \mathbf{F} denotes a vector-valued function, t the time, \mathbf{x} the vector of unknown variables composed of state variables and algebraic variables, \mathbf{u} the vector of known inputs and \mathbf{y} the vector of outputs. Evaluation of the function C at the advent of a discrete event determines which DAE system is to be used to describe the dynamic behaviour for the duration of the next system mode. At discrete events parts of the overall DAE system are deactivated or re-activated so that the index of a DAE system for subsequent system mode may be different from the one of the DAE system for the current system mode. For the duration of a system mode, however, the DAE index remains constant. A mixed symbolic and numerical solution of HDAEs involves

- index reduction and a selection of states for the reduced index problem augmented by initial constraints and differentiated equations,
- discrete event detection during evaluation of a continuous time model for the current system mode,
- initialisation at start time and re-initialisation at discrete events.

Algorithms for these tasks have been implemented, for instance, in the open-source OpenModelica software environment [40, 41].

As problems are to be encountered with the numerical solution of higher index DAEs and as there is no general solver for them, given a causal bond graph, it is an important question of what index a DAE system derived from the bond graph is. In [42], van Dijk classifies causal paths and determines the index of bond graphs of continuous time models with a given type of causal paths. For hybrid models, a bond graph representation with time-invariant causalities is used in this book. Causal paths through switches, however, depend on the switch state.

First, it is assumed that all storage elements are in integral causality for all system modes. That is, no residual sinks are switched on at discrete events to keep some storage elements in integral causality that otherwise would become dependent and would get derivative causality accordingly. Causal paths between resistive ports are allowed. As the switch model contains a resistor in conductance causality, there may also be causal paths between a resistor and a switch or between switches. If a switch in one of these causal paths is in OFF mode, the switch and the causal path can be disregarded. Causal paths between resistors mean that their outputs are determined by a set of algebraic relations. Let \mathbf{a} denote the vector of the outputs of resistors and of switches, then the DAE of a switched LTI system is of the form

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{a}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}(\mathbf{m}(t)) & \mathbf{A}_{12}(\mathbf{m}(t)) \\ -\mathbf{A}_{21}(\mathbf{m}(t)) & \mathbf{I} - \mathbf{A}_{22}(\mathbf{m}(t)) \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{a}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1(\mathbf{m}(t)) \\ -\mathbf{B}_2(\mathbf{m}(t)) \end{bmatrix} [\mathbf{u}] \quad (2.17)$$

where $\mathbf{m}(t)$ denotes the vector of switch states $m_j(t) \in \{0, 1\}$, $j = 1, \dots, n_s$, at time instant $t > 0$. For the time interval between two discrete events, $\mathbf{m}(t)$ is constant. Differentiation of the algebraic equation

$$\mathbf{0} = -\mathbf{A}_{21}\mathbf{x} + (\mathbf{I} - \mathbf{A}_{22})\mathbf{a} - \mathbf{B}_2\mathbf{u} \quad (2.18)$$

with respect to time yields the new DAE system

$$\underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{A}_{21} & \mathbf{I} - \mathbf{A}_{22} \end{bmatrix}}_{\mathbf{E}^1} \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{a}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11}(\mathbf{m}(t)) & \mathbf{A}_{12}(\mathbf{m}(t)) \\ -\mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{a}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1(\mathbf{m}(t))\mathbf{u} \\ -\mathbf{B}_2(\mathbf{m}(t))\dot{\mathbf{u}} \end{bmatrix} \quad (2.19)$$

In [42], van Dijk has shown that the determinant of \mathbf{E}^1 , $\det(\mathbf{E}^1) = \det(\mathbf{I} - \mathbf{A}_{22})$, is non-zero for bond graphs with causal paths between resistive ports. That means that the inverse of \mathbf{E}^1 exists and that differentiation of the algebraic equation (2.18) is sufficient to transform the DAE system (2.17) into a set ODEs. Accordingly, (2.17) is a DAE system of index 1.

Example: Electrical Network with Two Independent Switches

This is checked for the example of the switched network in Fig. 2.17.

The two switches may be operated independently. That is, there are four physical feasible switch state combinations constituting four system modes. The circuit

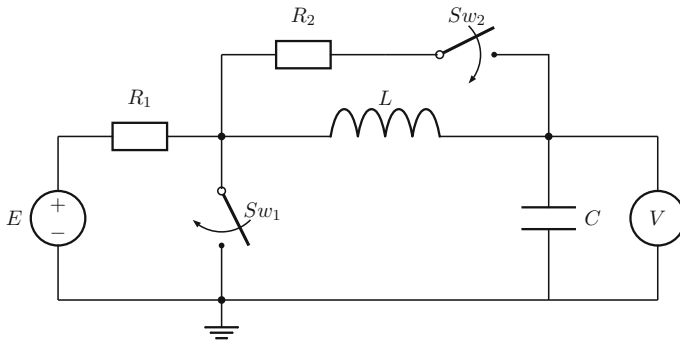


Fig. 2.17 Electrical circuit with two independent switches

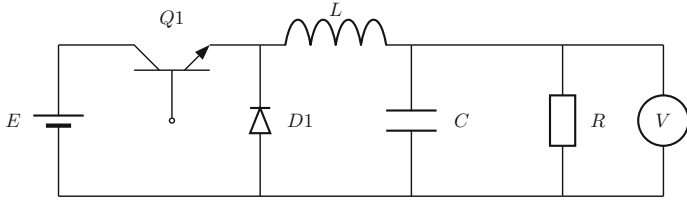


Fig. 2.19 Circuit schematic of a DC-DC buck converter

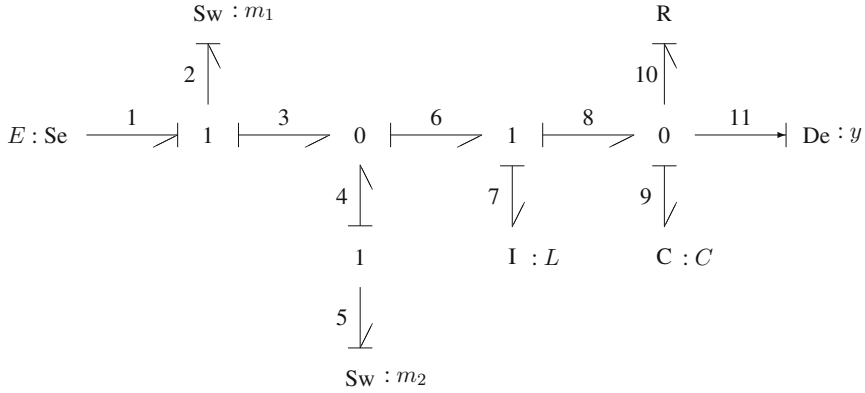


Fig. 2.20 Bond graph in preferred integral causality of the buck converter in Fig. 2.19

states $m_1(t)$, $m_2(t)$. That is, the DAE system describing the dynamic behaviour of the switched network is of index 1 for all four system modes.

Example: DC-DC Buck Converter

Figure 2.19 depicts the circuit schematic of a buck converter.

If the transistor Q_1 and the diode D_1 are modelled as non-ideal switches the bond graph in Fig. 2.20 is obtained.

The two switches commute conversely. There are two system modes, in which one switch is on while the other one is off. A third feasible mode, in which both switches are off is not considered. If one of the switches is on, conductance causality of its ON resistance can be changed. The result is a causal path between the two non-ideal switches. In Fig. 2.21, it has been assumed that $Sw : m_2$ is on.

However, as one of two switches is off, i.e. the current through that switch vanishes, the causal path can be disregarded. That is, in system modes $m_1 = 1 \wedge m_2 = 0$ and $m_1 = 0 \wedge m_2 = 1$, there are no causal paths between resistive ports and no dependent storage elements giving rise to an algebraic constraint between state variables. Hence, the dynamic behaviour in these modes is described by a set of two ODEs. In the context of DAEs, the special case of ODEs without any additional algebraic constraints is termed a DAE system of index 0.

If there are storage elements with derivative causality in all system modes then causal paths may exist to other storage elements of the same type in integral causality.

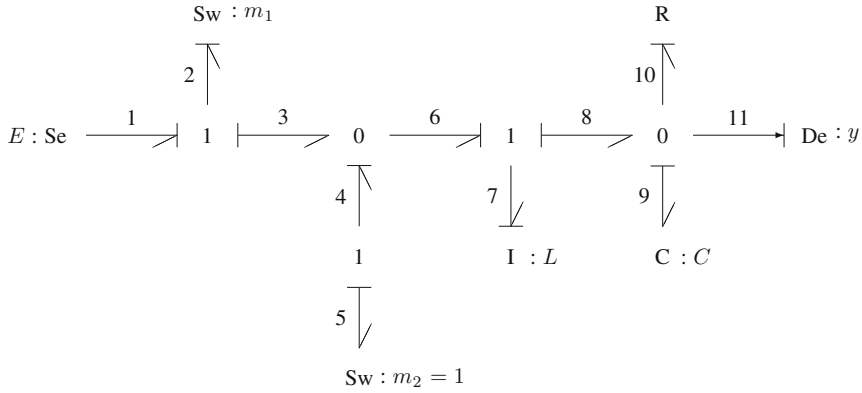


Fig. 2.21 Bond graph in preferred integral causality of the buck converter in case $Sw : m_2$ is on

This means that the state variable of a storage element in derivative causality is algebraically related to the state variables of a storage elements in integral causality. Also in this case, the DAE system for the time interval between two discrete events is of index 1 [42].

Structural Changes

For some systems with structural changes such as a clutch, storage elements may temporarily become dependent for the duration of a system mode. In such a case, a residual sink may be switched on that delivers a power variable so that the conjugate power variable vanishes and storage elements can keep integral causality. As their state variables jump to a new joint value, numerical integration has to be re-initialised at such a discrete event.

For a system mode, in which such switched residual sinks provide a non-zero output λ , the DAEs for a switched LTI system are of the form

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{22} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix} [u] \quad (2.22)$$

The entries of the matrices depend on the values of the current switch states. As the variable λ is absent in the algebraic constraint

$$\mathbf{0} = \mathbf{A}_{22}\mathbf{x} \quad (2.23)$$

differentiation of (2.23) does not give an equation for $\dot{\lambda}$ but an algebraic equation that is called a *hidden constraint*.

$$\mathbf{0} = \mathbf{A}_{22}[\mathbf{A}_{11}\mathbf{x} + \mathbf{A}_{11}\lambda + \mathbf{B}_1u] \quad (2.24)$$

If the matrix $\mathbf{A}_{22}\mathbf{A}_{11}$ is non-singular, (2.24) differentiated with respect to time can be solved for $d\lambda/dt$. According to the definition of the differentiation index, the DAE in this system mode is of index 2.

Remark 2.2

1. Initial conditions must also satisfy hidden constraints. It is these hidden constraints that pose problems for DAE-solvers.
2. The DAE form (2.22) is known as Hessenberg index-2 form. This form can be directly numerically solved by the DASKP 3.0 solver [43, 44] or the MEBDFI [45] solver available in Maple [46]. Both solvers use the BDF method. \square

As a result, if the dynamic behaviour of a system can be described by a switched LTI system, the DAE system derived from the bond graph is of index ≤ 1 as long as no structural changes occur. If a structural change modelled by a switched residual sink happens, the DAE index jumps to two.

Example: Clutch

As an example, consider the bond graph of a clutch in Fig. 2.15. There are no causal paths between resistors and no dependent storage elements. Clearly, as long as the clutch is disengaged, the DAE system is of index 0. In the case when the clutch is engaged, the unknown constraint force M between the two plates keeps their inertia elements in integral causality and at the same time ensures that the algebraic constraint

$$0 = \omega_1 - \omega_2 \quad (2.25)$$

holds. Clearly, differentiating this constraint with respect to time does not give the derivative of the unknown constraint force M . Substituting $d\omega_1/dt$, $d\omega_2/dt$ in the derivative of the constraint allows to obtain dM/dt after differentiation of the resulting hidden constraint

$$0 = -\frac{1}{J_1}M + \frac{1}{J_1}M_1 - \frac{1}{J_2}M + \frac{1}{J_2}M_2 \quad (2.26)$$

Hence, once the clutch is engaged, the DAE is of index 2.

2.5 Discrete Event Simulation of Hybrid Systems

Instead of discretising the time and using a BDF-based method for the numerical computation of a continuous-time model one may think of quantising the state variables. That is, instead of using a multistep method to compute an approximation of the value $x(t_{k+1})$ of a state variable x at time t_{k+1} , the question then is at what time the state x will deviate from its current value $x(t_k)$ by more than a given quantum ΔQ . In other words, the task is to find the smallest time step h so that

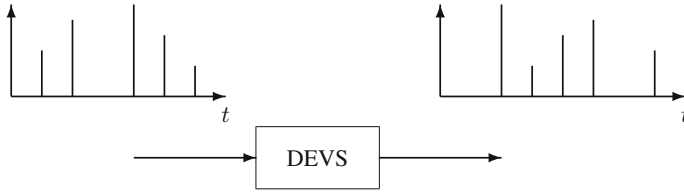


Fig. 2.22 Input/output behaviour of a DEVS model [1, 53]

$$x(t_k + h) = x(t_k) \pm \Delta Q \quad (2.27)$$

This approach replaces the traditional discrete time based computation of continuous time models by a discrete event simulation that advances the time from the time point of an event to the time of the next event which is attractive for the computation of hybrid models because discrete events, i.e. discontinuous mode changes, and the continuous time behaviour during system modes can be uniformly processed in the framework of the well-known Discrete Event System (DEVS) specification introduced by Ziegler [47, 48]. Moreover, the DEVS formalism is supported by software libraries such as adevs [49, 50] or simulation environments such as PowerDEVS [51, 52].

A DEVS model of a system takes a sequence of discrete events, i.e. of instantaneous system changes, as inputs and produces an output sequence of events according its initial conditions. Figure 2.22 displays this input/output behaviour of a DEVS model. Events can be characterised by a value and the time point of their occurrence. Accordingly, they are indicated by perpendicular strokes in Fig. 2.22. A sequence of events is called an event trajectory. It is assumed that the number of state changes in any finite time interval is finite.

An *atomic* DEVS model is defined as a tuple of sets and functions (cf. Appendix A.3). If the output events of an atomic DEVS model are converted into input events of another atomic DEVS model, i.e. if atomic DEVS models are coupled, then the result defines a new DEVS model. That is, complex systems can be modelled in the DEVS framework in a hierarchical manner. The DEVS formalism is widely used in computer science. Its application to the numerical solution of continuous-time models, however, is much less common.

The quantised state system (QSS) method introduced by Kofman [53] allows for a discrete event simulation of hybrid systems. The method starts from the observation that a piecewise constant trajectory can be represented by sequences of events. The reader is referred to the literature, e.g. references [1, 53, 54] for details. In the following, only the basic idea is outlined in a simplified manner.

Consider the state space equation

$$\dot{x}(t) = f(x(t), u(t)) \quad (2.28)$$

where $x(t)$ is the state and $u(t)$ the input given by a piecewise constant function.

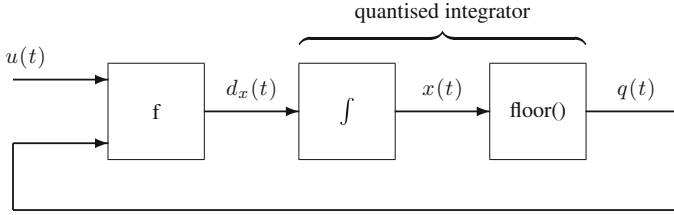


Fig. 2.23 Block diagram of a quantised state system

Let $\text{floor} : \mathbb{R} \rightarrow \mathbb{Z}^+$ denote a function that returns the integer part of its positive real-valued argument. The QSS-method then solves the approximate system

$$\dot{x}(t) = d_x(t) \quad (2.29a)$$

$$q(t) = \text{floor}(x(t)) \quad (2.29b)$$

$$d_x(t) = f(q(t), u(t)) \quad (2.29c)$$

The variable q is called a *quantised* variable and Eq. (2.29) a *quantised state system* (QSS). equations 2.29a–2.29b constitute a dynamic subsystem with a piecewise constant input trajectory $d_x(t)$ and a piecewise constant output trajectory $q(t)$. The third Eq. 2.29c is a static relation. The quantised state system (2.29) can be displayed by the block diagram in Fig. 2.23.

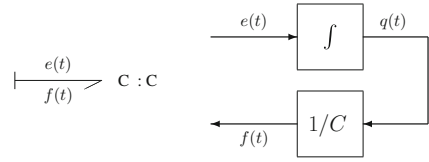
The static block and the quantised integrator can be represented by a DEVS model. Their coupling again builds a DEVS model. The DEVS models can be translated into PowerDEVS models. The software environment PowerDEVS provides a library with a number of predefined atomic and coupled DEVS models and a graphical user interface that permits to build coupled DEVS models.

Given a time instant t_k and a constant quantum ΔQ , then the next time step h that satisfies (2.27) can be determined if the derivative of the state variable in (2.29a) is approximated by a difference quotient.

$$\frac{x(t_k + h) - x(t_k)}{h} = \frac{\pm \Delta Q}{h} = d_x(t_k) = f(q(t_k), u(t_k)) \quad (2.30)$$

The QSS method outlined so far bears a problem because it cannot be guaranteed that a DEVS model resulting from the coupling of atomic DEVS models will perform a finite number of transitions in any finite time interval. DEVS models with this ability are said to be *legitimate*. For illegitimate DEVS models the simulation will get stuck. Kofman has shown that the problem can be overcome by introducing hysteresis in the relationship between a state variable x and its quantised variable q [53]. However, the method is only first-order accurate. A quantisation for which the hysteresis value equals the quantum value produces a piecewise constant output trajectory that only changes when the difference between input and output exceeds the quantum. As a result, the state variables have a piecewise linear trajectory. To improve accuracy,

Fig. 2.24 Block diagram of a bond graph C storage element in integral causality



Kofman proposed a second-order accurate QSS2 method [55]. For this method, the quantisation that does not need hysteresis produces piecewise linear output trajectories having discontinuities whenever the absolute value of the difference between a state variable and its quantised variable reaches the quantum. If the QSS2 method is applied to LTI systems then state variables have piecewise parabolic trajectories provided input trajectories are piecewise constant.

The outlined quantised state integration can be applied to systems of coupled ODEs with piecewise constant input functions. Moreover, it can be extended so that hybrid DAE systems can be solved by discrete event simulation [54].

Bond Graph Models and Discrete Event Simulation

As can be seen from Fig. 2.23, a coupled DEVS model is obtained from a continuous time model by feeding the output of an integrator into a quantising block turning the integrator into a so-called *quantised integrator* provided input trajectories are piecewise constant. This observation suggests that discrete event simulation can be applied to systems represented by a bond graph because a causal bond graph is equivalent to a block diagram in which the block diagram replacement of each energy storage element includes an integrator (Fig. 2.24).

It is sufficient to quantise the static relation between the energy variable of a storage element and its output power variable. Consider, e.g. the linear 1-port C element in integral causality in Fig. 2.24 and let $q_q(t) = \text{floor}(q(t))$ the quantisation of its displacement q . The continuous constitutive equations of a C element with the capacitance C

$$\dot{q}(t) = f(t) \quad (2.31a)$$

$$e(t) = \frac{1}{C}q(t) \quad (2.31b)$$

are then replaced by

$$\dot{q}(t) = f(t) \quad (2.32a)$$

$$e(t) = \frac{1}{C}q_q(t) = \Phi_q(q(t)) \quad (2.32b)$$

That is, the characteristic of a storage element is approximated by a piecewise constant function $\Phi_q()$ (with hysteresis). The other bond graph elements are static elements. A 2-port transformer, for instance, takes two piecewise constant inputs

and delivers two piecewise constant outputs. Bond graphs in which the characteristic of the storage elements has been quantised are called quantised bond graphs (QBGs) in [53, 56]. The elements of a quantised bond graph can be represented by DEVS models. The result of their coupling is a DEVS model of the QBG. Hence, this approximation of an initial continuous-time bond graph model can be computed by means of discrete event simulation.

Another approach that also allows to approximate each continuous time element of a bond graph by a DEVS model so that a DEVS simulation can be performed has been reported in [57, 58]. The task is to transform piecewise continuous input and output trajectories of a bond graph element into discrete event trajectories. To translate, e.g. the continuous time model of a C element in integral causality into a discrete event model, the input trajectory of the flow $f(t)$ between two time instances t_i and t_j is approximated by a linear function $f(t) = a_1t + a_0$. The output trajectory of $e(t)$ is a second order polynomial $e(t) = b_2t^2 + b_1t = (a_1/2C)t^2 + (a_0/C)t$. This polynomial is approximated by a linear function $\tilde{e}(t) = \tilde{b}_1t + \tilde{b}_0$. Its coefficients are determined by the value of $e(t)$ at the beginning and at the end of the time interval $[t_i, t_j]$.

$$\tilde{e}(t) = \frac{e(t_j) - e(t_i)}{t_j - t_i}t + e(t_i) \quad (2.33)$$

A discrete event is considered to take place whenever the linear trajectory for the time interval $[t_i, t_j]$ crosses a threshold or when there is a significant change of its slope. This threshold crossing or a gradient change determines the time instant of the next event and thus the length of the time interval of the current linear trajectory.

An attractive feature of a discrete event simulation of hybrid systems is that the simulation time advances from discrete event to discrete event. For the QSS method, discontinuities in the inputs and the quantised variables dictate the time advance. No iteration is necessary to locate the time point of a discontinuity allowing for an efficient processing of models with discontinuities. Discrete event simulation using the quantised-based integrations needs much less simulation steps than a numerical integration method of comparable accuracy based on time-discretisation. Accordingly, computational costs are saved. Nevertheless, there are still some problems to be tackled with the QSS approach as detailed in [1, Chap. 12.11].

2.6 Summary

The survey in this chapter shows that there are various bond graph approaches to hybrid system modelling having their pros and cons. A comparison of approaches using ideal switches versus those using non-ideal switches has been given in [30].

In this book, a bond graph representation of hybrid system models is chosen as basis for bond graph model-based FDI in Chap. 4. The representation is system mode independent with regard to computational causalities and allows for deriving a set of

model equations in the form of a DAE system that holds for all system modes. This is achieved by modelling fast switching devices by an MTF associated with a resistor in fixed conductance causality and by means of MTFs switching on and off residual sinks in the case of structural changes. The residual sinks allow for keeping integral causality at storage elements that become dependent by a system mode change and would take derivative causality.

If the dynamic behaviour of a system can be described by a switched LTI system, a linear implicit DAE system can be derived from the bond graph. The entries of its matrices depend on the discrete switch states. As long as no structural changes occur, i.e. no residual sinks are switched on, the DAE system is of index ≤ 1 . For system modes in which residuals sinks are switched on, the DAE system is of index 2. There are solvers available for its direct numerical computation that are based on the BDF-method. An alternative may be to perform a DEVS simulation that uses quantised based integration.

References

1. Cellier, F. E., & Kofman, E. (2006). *Continuous system simulation*. New York: Springer.
2. Hindmarsh, A., & Taylor, A. (1999). *User documentation for IDA, a differential-algebraic equations solver for sequential and parallel computers*. Lawrence Livermore National Laboratory, Center for Applied Scientific Computing. UCRL-MA-136910.
3. Brenan, K. E., Campbell, S. L., & Petzold, L. R. (1989). *Numerical solution of initial-value problems in differential-algebraic equations*. New York: North-Holland.
4. Zimmer, D. (2013). A new framework for the simulation of equation-based models with variable structure. *Simulation*, 89(8), 935–963.
5. Zimmer, D. (2010). Equation-based modeling of variable-structure systems. *Ph.D. Dissertation No. 18924*, ETH Zürich.
6. Borutzky, W. (2010). *Bond graph methodology—Development and analysis of multidisciplinary dynamic system models*. London: Springer. ISBN : 978-1-84882-881-0.
7. Borutzky, W., Broenink, J. F., & Wijbrans, K. C. J. (1993). Graphical description of physical system models containing discontinuities. In A. Pavé (Ed.), *Modelling and Simulation 1993, Proceedings of the 1993 European Simulation Multiconference* (pp. 208–214). Lyon: SCS Publishing.
8. Borutzky, W. (1995). Representing discontinuities by means of sinks of fixed causality. In F. E. Cellier & J. J. Granda (Eds.), *1995 International Conference on Bond Graph Modeling, ICBGM'95, Proceedings of the 1995 Western Simulation Multiconference* (Vol. 27, No. 1, pp. 65–72). Simulation Series. Lyon: SCS Publishing. ISBN: 1-56555-037-4.
9. Bos, A. M. (1986) Modelling multibody systems in terms of multibond graphs with application to a motorcycle. *Ph.D. Thesis*. The Netherlands: University of Twente Enschede.
10. Borutzky, W., & Cellier, F. E. (1996). Tearing algebraic loops in bond graphs. *Transactions of the SCS*, 13(2), 102–115.
11. Borutzky, W. (2009). Bond graph model-based fault detection using residual sinks. *Proceedings of the Institution of Mechanical Engineers Part I Journal of Systems and Control Engineering*, 223(3), 337–352.
12. Umarikar, A. C. (2006). Modelling of Switched mode power converters: A bond graph approach. *Ph.D. Thesis*. Bangalore: Centre for Electronics Design and Technology, Indian Institute of Science.
13. Umarikar, A. C., & Umanand, L. (2005). Modelling of switched systems in bond graphs using the concept of switched power junctions. *Journal of the Franklin Institute*, 342, 131–147.

14. Buisson, J., Cormerais, H., & Richard, P. Y. (2002). Analysis of the bond graph model of hybrid physical systems with ideal switches. *Proceedings of the Institution of Mechanical Engineers Part I: Systems and Control Engineering*, 216(1), 47–63.
15. Buisson, J. (2002). *Modélisation, Analyse et Commande des Systèmes en Commutation*. Rennes: Université de Rennes 1. l'habilitation à diriger des recherches (HDR).
16. Edström, K. (1999). Switched bond graphs: Simulation and analysis. *Ph.D. Thesis*. Linköping: Linköping University.
17. Strömberg, J. E. (1994). A mode switching modelling philosophy. *Ph.D. Thesis*. Linköping: Linköping University.
18. Strömberg, J. E., Top, J., & Södermann, U. (1993). Variable causality in bond graphs caused by discrete effects. In J. J. Granda & F. E. Cellier (Eds.), *International Conference on Bond Graph Modeling, ICBGM'93, Proceedings of the 1993 Western Simulation Multiconference* (Vol. 25, No. 2, pp. 115–119). Simulation Series. Lyon: SCS Publishing. ISBN: 1-56555-019-6.
19. Margetts, R. (2013). Modelling & analysis of hybrid dynamic systems using a bond graph approach. *Ph.D. Thesis*. University of Bath.
20. Roychoudhury, I., Daigle, M., Biswas, G., & Koutsoukos, X. (2010). Efficient simulation of hybrid systems: A hybrid bond graph approach. *Simulation: Transactions of the Society for Modeling and Simulation International*, 87(6), 467–498.
21. Mosterman, P. J. (1997). Hybrid dynamic systems: a hybrid bond graph modeling paradigm and its application in diagnosis. *Ph.D. Thesis*. Nashville: Vanderbilt University.
22. van der Schaft, A. J., & Schuhmacher, H. (2000). An introduction to hybrid dynamical systems. No. 251 in *Lecture Notes in Control and Information Sciences*. London: Springer.
23. Asher, G. M. (1993). The robust modelling of variable topology circuits using bond graphs. In J. J. Granda & F. E. Cellier (Eds.), *International Conference on Bond Graph Modeling, ICBGM'93, Proceedings of the 1993 Western Simulation Multiconference* (Vol. 25, No. 2, pp. 126–131). Simulation Series. San Diego: SCS Publishing. ISBN: 1-56555-019-6.
24. Thoma, J. U. (1975). *Introduction to bond graphs and their applications*. Oxford: Pergamon Press.
25. Mosterman, P. J. (2002). HYBrSIM—A modelling and simulation environment for hybrid bond graphs. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering Systems*, 216(1), 35–46.
26. Wang, D., Yu, M., Low, C., & Arogeti, S. (2013). *Model-based health monitoring of hybrid systems*. New York: Springer.
27. The Mathworks. *Simulink—Simulation and model-based design*, from <http://www.mathworks.com/products/simulink/>.
28. Ducreux, J. P., Dauphin-Tanguy, G., & Rombaut, C. (1993). Bond graph modelling of commutation phenomena in power electronic circuits. In J. J. Granda & F. E. Cellier (Eds.), *International Conference on Bond Graph Modeling, ICBGM'93, Proceedings of the 1993 Western Simulation Multiconference* (Vol. 25, No. 2, pp. 132–136). Simulation Series. San Diego: SCS Publishing. ISBN: 1-56555-019-6.
29. HighTec Consultants. *Symbols Sonata™*, from <http://www.htcinfo.com/>.
30. Buisson, J., Richard, P. Y., & Cormerais, H. (2000). Ideal versus non-ideal approaches in bond graph modelling of switching devices: a comparison based on singular perturbation theory. In *Proceedings of the 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic Systems (ADPM' 2000)* (pp. 257–264).
31. Buisson, J. (1993). Analysis and characterisation of hybrid systems with bond-graphs. In J. J. Granda & F. E. Cellier (Eds.), *International Conference on Bond Graph Modeling, ICBGM'93, Proceedings of the 1993 Western Simulation Multiconference* (Vol. 25, No. 2, pp. 264–269). Simulation Series. San Diego: SCS Publishing. ISBN: 1-56555-019-6.
32. Campbell, S. L., & Gear, C. W. (1995). The index of general nonlinear DAEs. *Numerische Mathematik*, 72, 173–196.
33. Gear, C. W. (1971). Simultaneous numerical solution of differential/algebraic equations. *IEEE Transactions on Circuits Theory, CT-18*(1), 89–95.

34. Hairer, E., & Wanner, G. (1996). *Solving ordinary differential equations II, stiff and differential-algebraic problems* (2nd ed.). Berlin: Springer.
35. Brennan, K. E., Campbell, S. L., & Petzold, L. R. (1996). *Numerical solution of initial-value problems in differential-algebraic equations*. Philadelphia: SIAM.
36. Pantelides, C. C. (1988). The consistent initialization of differential-algebraic systems. *SIAM Journal of Scientific and Statistical Computation*, 9, 213–231.
37. Mattsson, S., & Söderlind, G. (1993). Index reduction in differential algebraic equations using dummy derivatives. *SIAM Journal on Scientific Computing*, 14(3), 677–692.
38. Vieira, R. C., & Bisciaia, E. C, Jr. (2001). Direct methods for consistent initialization of DAE systems. *Computers and Chemical Engineering*, 25, 1299–1311.
39. Frasca, R., Camlibel, M. K., Goknar, I. C., Iannelli, I., & Vasca, F. (2010). Linear passive networks with ideal switches: Consistent initial conditions and state discontinuities. *IEEE Transactions on Circuits and Systems*, 57(12), 3138–3151.
40. Lundvall, H., Fritzson, P., & Bachmann, B. (2008). *Event handling in the openmodelica compiler and runtime system*. Series: Technical Reports in Computer and Information Science. Linköping University, Department of Computer and Information Science, PELAB—Programming Environment Laboratory. ISSN 1654–7233; 2.
41. OpenModelica Consortium. *OpenModelica*, from <https://www.openmodelica.org/>.
42. van Dijk, J. (1994). On the role of bond graph causality in modelling mechatronic systems. *Ph.D. Thesis*. The Netherlands: University of Twente. Enschede.
43. Li, S. Petzold, L. (1999). Design of new DASPK for sensitivity analysis. *UCSB*, from <http://www.engineering.ucsb.edu/~cse/>.
44. DASPK, from <http://www.cs.ucsb.edu/~cse/software.html>.
45. Cash, J. (2003). Efficient numerical methods for the solution of stiff initial-value problems and differential algebraic equations. *Proceedings of the Royal Society*, 459, 797–815.
46. Maplesoft, from <http://www.maplesoft.com/>.
47. Zeigler, B. (1976). *Theory of modeling and simulation*. New York: Wiley.
48. Zeigler, B., Tag Gon, K., & Praehofer, H. (2000). *Theory of modeling and simulation* (2nd ed.). New York: Academic Press.
49. Naturo, J. (2011). *Building software for simulation: Theory and algorithms, with applications in C++*. New York: Wiley.
50. adevs—A simulation library for discrete event and hybrid dynamics systems, from <http://sourceforge.net/projects/adevs/>.
51. Bergero, F., & Kofman, E. (1229). PowerDEVS. *A tool for hybrid system modeling and real time simulation*. Rosario: Laboratorio de Sistemas Dinámicos. FCEIA - UNR. CIFASISCONICET, from http://www.fceia.unr.edu.ar/kofman/files/bergero_kofman_pd.pdf.
52. PowerDEVS, from <http://sourceforge.net/projects/powerdevs/>.
53. Kofman, E. (2003). Discrete event based simulation and control of continuous systems. *Ph.D. Thesis*. Facultad de Ciencias Exactas, Ingeniería y Agrimensura, Rosario: Universidad Nacional de Rosario.
54. Kofman, E. (2004). Discrete event simulation of hybrid systems. *SIAM Journal on Scientific Computing*, 25(5), 1771–1797, from <http://dx.doi.org/10.1137/S1064827502418379>.
55. Kofman, E. (2002). A second order approximation for DEVS simulation of continuous systems. *Simulation*, 78(2), 76–89.
56. Kofman, E., & Junco, S. (2001). Quantized bond graphs: An approach for discrete event simulation of physical systems. In G. Dauphin-Tanguy & J. J. Granda (Eds.), *Proceedings of the 2001 International Conference on Bond Graph Modeling and Simulation (ICBGM'01)* (Vol. 33, No. 1, pp. 369–374). Simulation Series. San Diego: SCS.
57. Naamane, A., Damiba, A., & Giambiasi, N. (2001). Bond ggraphs and gdevs. In G. Dauphin-Tanguy & J. J. Granda (Eds.), *Proceedings of the 2001 International Conference on Bond Graph Modeling and Simulation (ICBGM'01)* (Vol. 33, No. 1, pp. 375–380). Simulation Series. San Diego: SCS.
58. D'Abreu, M. C., & Wainer, G. A. (2003). Models for continous and hybrid system simulation. In S. Chick, P. J. Sánchez, D. Ferrin, & D. J. Morrice (Eds.), *Proceedings of the 2003 Winter Simulation Conference* (pp. 641–649). New Orleans: IEEE Press.

Bond Graph Model-based Fault Diagnosis of Hybrid
Systems

Wolfgang, B.

2015, XV, 277 p. 172 illus., 98 illus. in color., Hardcover

ISBN: 978-3-319-11859-8