

# Bond Graph Modelling and Simulation of Mechatronic Systems

## An Introduction into the Methodology

W. Borutzky

Bonn-Rhein-Sieg University of Applied Sciences, D-53754 Sankt Augustin, Germany

[borutzky@uni-koeln.de](mailto:borutzky@uni-koeln.de)

*Abstract*—This paper introduces into a graphical, computer aided modelling methodology that is particularly suited for the concurrent design of mechatronic systems, viz. of engineering systems with mechanical, electrical, hydraulic or pneumatic components including interactions of physical effects from various energy domains.

Beyond the introduction, bond graph modelling of multibody systems, as an example of an advanced topic, is briefly addressed in order to demonstrate the potential of this powerful approach to modelling mechatronic systems. It is outlined how models of multibody systems including flexible bodies can be build in a systematic manner.

*Keywords*—Bond graph modelling, object-oriented physical systems modelling, computational causalities, forms of mathematical models, mechatronic systems, multibody systems.

### I. INTRODUCTION

Bond graphs were devised by Professor H. Paynter at Massachusetts Institute of Technology (MIT) in Cambridge, Massachusetts, U. S. A. as early as 1959 [1]. His former Ph. D. students Professor Karnopp, Professor Margolis (University of California at Davis) and Professor Rosenberg (Michigan State University, East Lansing, Michigan) have elaborated this graphical model representation into a methodology that has experienced a considerable progress over the decades due to the steady work of many researchers all over the world. Since the early days, many research papers, a number of textbooks in different languages [2, 3, 4, 5], as well as a number of special journal issues on bond graph modelling have been published [6, 7, 8] with the aim of reflecting a part of contemporary research on and application of bond graph modelling in various areas. An excellent exposure of bond graph modelling can be found in the highly recognised textbook of Karnopp, Margolis and Rosenberg [9] of which the fourth edition was recently published in December 2005. Readers who quickly want to become familiar with the concepts of bond graph modelling and who are looking for a brief introduction are referred to the 28 pages *Introduction to Physical Systems Modelling with Bond Graphs* by J. F. Broenink [10]. Finally, bond graph modelling is supported by a number of advanced modelling and simulation software packages. A survey compiled by A. Samantaray is available at [www.bondgraphs.com/software.html](http://www.bondgraphs.com/software.html).

### II. FUNDAMENTAL CONCEPTS

The Bond Graph methodology clearly and intuitively starts by considering energy flows between the ports of the (conceptual) components of an engineering system. Physical effects and their interactions are considered and, in the first place, are taken into account in a *qualitative* manner. At further stages of the modelling process, details are specified as necessary so that a mathematical model can be gen-

erated and can be evaluated. If properly applied, the Bond Graph methodology enables to develop a graphical model that is consistent with the first principle of energy conservation without having the need to start with establishing and reformulating equations. The derivation of a mathematical model from the graphical description that is suitable for the purpose of a project can be rather left to appropriate software tools performing this task automatically. As a consequence, users of the methodology can focus on physical modelling of a system.

#### A. Hierarchical Bond Graph models

According to the consideration of energy flows in an engineering system, the vertices of a bond graph denote (conceptual) subsystems, system components or elements, while the edges, called power bonds or bonds for short, represent energy flows between them. The nodes of a bond graph have got so-called power *ports* where energy can enter or exit. Therefore, bond graph nodes are also termed *multiports*. Some software packages supporting bond graph modelling, e. g. 20sim<sup>TM</sup>[11], enable to make ports visible on demand by little black squares (cf. Fig. 1).

Clearly, as with other graphical modelling paradigms, bond graph models can be developed in a hierarchical combined top-down and bottom-up approach by using component models or elements from model libraries. The model hierarchy may be represented separately by a tree of component models. For each hierarchy level, the structure of the model may be represented by a bond graph. While in iconic diagrams, or in electrical or hydraulic networks application specific representations are used for the nodes of the graph, in bond graphs, nodes are presented by words enclosed by an optional ellipsis. For that reason the notion of *word* bond graphs is common in the process of a bond graph based model decomposition approach. At the lowest hierarchy level, bond graph nodes represent basic energetic processes as the delivery or storage of energy, the irreversible transformation of energy into heat, or the power conservative distribution of power. For these fundamental energetic processes, fixed types of nodes are used. For instance, the storage of kinetic energy in a rigid body, or the storage of magnetic energy in a coil is represented by a node of type I. Of course, as with other graphical representations, user defined nodes may be introduced, e. g., a node labelled *orifice* representing a mathematical model of a hydraulic orifice.

#### B. Power variables and analogies

As energy can flow back and forth between two power ports of different nodes, a *half arrow* is added to each bond

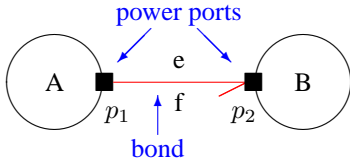


Fig. 1: Power bond connecting two power ports of two component models

indicating a *reference* direction of the energy flow.

Furthermore, it is a general observation, that the amount of power at each time instance,  $t$ , can be determined by the product of two conjugate variables, which are called *effort*,  $e$ , and *flow*,  $f$ , respectively.

$$\text{Power} = \text{Effort} \times \text{Flow}$$

In translational mechanical engineering, effort and flow can be identified as force and linear velocity. In electrical engineering, the product of the voltage drop across the two pins of a port and the joint current in both pins is the instantaneous amount of electrical power at this port. Correspondingly, in the thermal domain, these variables are the absolute temperature and the rate of change of the entropy. Table I lists the effort and flow variables in the various energy domains. The variables in the third column of Table I are the time integral of the efforts and the variables in the fourth column are time integral of the flows. They are called *energy variable* because they quantise the amount of energy in the energy stores of a model. The power variables effort and flow play an equal role with respect to each other. They are just characterised by the fact that they are a factor in the power product. Given a pair of power variables, it is a matter of preference which of them is chosen as effort and which consequently serves as the flow. This gives rise to two possible analogies. One choice could be to relate a mechanical force, or a moment to an electrical voltage drop. Consequently, velocities, or angular velocities correspond to electrical currents. This analogy has been in widely use for a long time. Therefore, occasionally it is referred to as *classical* analogy. The other possible association of a velocity with a voltage drop also makes perfectly sense from the point of measurements. It has been introduced by Firestone around 1933 and is called *mobility* analogy. If two modelling approaches only differ with respect to the analogy, the resulting bond graph models will look different as well as the mathematical models derived from the graphs. However, as to the numerical evaluation of the mathematical models, simulation results should be the same.

As a bond in a bond graph represents the energy exchange between two ports of different nodes, all edges of a bond graph carry two power variables. Bonds may be annotated by the names of these power variables. It is a convention to place the name of an effort above a horizontal bond and the flow below the bond (cf. Fig. 1). For vertical bonds the convention is to place the effort to the left and the flow to the right of the bond. For inclinations different from a multiple of  $90^\circ$ , a more sophisticated convention is to consider that variable as flow that is on the same side as the half arrow denoting the reference direction of the energy flow.

### C. Computational causalities

As each bond connecting two ports of different nodes A and B carries two power variables, one of these two may be determined by one of the two sub-models, while the other is delivered by the other model. In other words, from a computational point of view, the effort could be computed by the evaluation of model A, while the flow is computed in model B. It could be the other way around as well. The decision, in which sub-model the effort is computed is indicated by a perpendicular stroke attached to the bond and is called the assignment of *computational causality*. The end of a bond

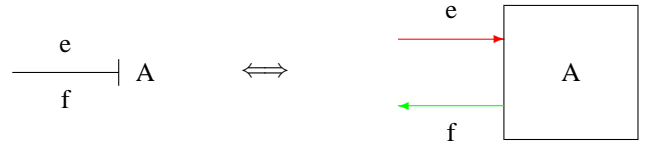


Fig. 2: Computational causality: indicating the direction of effort and flow

without perpendicular stroke indicates the model, in which the effort is computed and in which for that purpose the conjugate flow must be known. In other words, the end with perpendicular stroke point to the model, in which the effort is used in the computation of the conjugate flow. Another possible and common view is to consider both power variables of a bond as signals of opposite direction. That is, the perpendicular stroke indicates the signal direction of the effort, which implies that the end without the perpendicular stroke displays the direction of the conjugate flow. The perpendicular stroke is called *causal* stroke. A bond graph is called *causal* bond graph if a causal stroke has been added to each bond. Note that the half arrow and the causal stroke are orthogonal concepts. That is, there are four possible pattern of half arrow and causal stroke. Now, if every bond of a causal bond graph is replaced by two signal arrows of *opposite* direction, then apparently, the bond graph can be transformed into a block diagram as it is common in control engineering. However, not every block diagram can be converted into a bond graph. Bond graph modelling starts by considering energy flows in a system. Consequently, bond graph models should obey the first principle of energy conservation. In block diagrams, functional blocks represent the processing of input signals. The mathematical relations underlying a block diagram are not necessarily consistent with energy conservation. Sometimes mathematical relations between input and output signals of a signal processing block have not been derived from first principles, but curve fitting methods, for instance, have been used.

If sensors and instruments are included in the bond graph modelling of an engineering system, then the power conveyed between two ports of different components can be neglected, if the sensing of signals is of primary concern. This means that one of the two conjugate power variables associated with a bond can be dropped turning the bond into a so-called *activated* bond or reducing it to a conventional signal arrow. As a result, the ports linked by a bond that has become a signal arrow turn into signal ports.

The observation that a causal bond graph can be transformed into a block diagram implies that bond graphs com-

Energy domain	Effort $e$	Flow $f$	Generalised momentum $p$	Generalised displacement $q$
Translational mechanics	Force $F$ [N]	Velocity $v$ [m/s]	Momentum $p$ [Ns]	Displacement $x$ [m]
Rotational mechanics	Angular moment $M$ [Nm]	Angular velocity $\omega$ [rad/s]	Angular momentum $p_\omega$ [Nms]	Angle $\theta$ [rad]
Electro-	Voltage $u$ [V]	Current $i$ [A]	Linkage flux $\lambda$ [Vs]	Charge $q$ [As]
magnetic domain	Magnetomotive force $V$ [A]	Magnetic flux rate $\dot{\Phi}$ [Wb/s]	–	Magnetic flux $\Phi$ [Wb]
Hydraulic domain	Total pressure $p$ [N/m <sup>2</sup> ]	Volume flow rate $Q$ [m <sup>3</sup> /s]	Pressure momentum $p_p$ [N/m <sup>2</sup> s]	Volume $V_c$ [m <sup>3</sup> ]
Thermo-dynamic	Temperature $T$ [K]	Entropy flow rate $\dot{S}$ [J/K/s]	–	Entropy $S$ [J/K]
Chemical domain	Chemical potential $\mu$ [J/mole]	Molar flow $\dot{N}$ [mole/s]	–	Molar mass $N$ [mole]

TABLE I: Power and energy variables in various energy domains

bine features of networks and of block diagrams. With networks bond graph have in common that they represent the topological connectivity. Component sub-models are connected in the same way as the physical system components they model are linked. Furthermore, assigning causalities in a bond graph means that the bond graph is overlaid with a block diagram representing the *computational* structure of a model. On the other hand, it is well known that information about the topological structure of a system is lost in a conventional block diagram model and can hardly be reconstructed, in general, from a given block diagram.

The hierarchical development of a bond graph model and the connection of component sub-models according to the topological structure of a system implies that bond graphs of component models must be non-causal. The decision, which of the two power variables of a power port plays the role of an input signal forcing the conjugate variable being an output variable is determined by the component models a component model under consideration is connected to. That is, causal strokes cannot be assigned before the hierarchical development of an overall system model is finished and the hierarchy has been flattened. Causal strokes, or causalities

ties, at the ports of one and the same component sub-model can be different depending on the component sub-models it is connected to. If the assignment of causalities to power ports means to decide which variables are input and which of them are output variables, then all equations derived from a non-causal bond graph can only take implicit form.

#### D. Bond graph and object-oriented modelling

The concept of ports representing an interface of a component sub-model to its outside world that hides internal implementation details from its environment, the construction of bond graph models by connecting ports of sub-model components and the use of component model hierarchies, interestingly has much in common with the modern paradigm of object-oriented modelling (OOM), although bond graph modelling was developed long before object oriented modelling came into existence. For instance, an electrical diode, or a hydraulic orifice of fixed area can be considered subclasses of a more general 1-port resistor superclass, which inherit all properties of the abstract 1-port resistor class. Depending on the application, the constitutive relations are (re-)defined in the derived classes. That is, bond graph classes can be *polymorphic*. Instantiations of these subclasses are obtained by providing given parameters resulting in objects, e. g. an orifice model O1 with a given cross section area and a given discharge coefficient. In fact, the older bond graph modelling approach can be viewed as a kind of object oriented modelling approach (Borutzky [12]).

### III. BOND GRAPH ELEMENTS

Bond graph modelling uses a small set of nine basic conceptual elements representing fundamental energetic processes. They can be grouped into five categories.

#### A. Supply and absorption of energy

The supply of energy into a system is modelled by source elements. The absorption of energy flowing out of a system into its environment can be represented by sinks, which can be considered negative sources. As a power port has two variables, two kinds of sources exist. Sources may impose either an effort or a flow onto a system. A battery serving as a constant voltage source can be modelled by an effort source, while a hydraulic pump providing a constant volume flow rate can be modelled as a flow source. In bond graphs, sources and sinks respectively are denoted by the character S (Source). The type is naturally indicated by adding either the characters e or f respectively (Se or Sf). Sources may have more than one power port. Moreover, as there exist, e. g. stabilised voltage sources, or controlled hydraulic pressure pumps, sources may also have a signal port for feedback control. In this case the type identifier is prefixed by the character M standing for *modulated* source (MSe, or MSf). The signal port is located on the M side of the element.

#### B. Storage of energy

Tab. I shows that the time integral of the power variables, viz. *generalised displacement* and *generalised momentum*, have a physical meaning in many energy domains. Moreover, in many energy domains, they can be related to a

power variable resulting in the constitutive relation of a 1-port storage element. That is, one of the two power variables of the port is the rate of change of a so-called *conserved*, or stored quantity, also called a *state*, while the other power variable is an equilibrium determining variable. As either the effort or the flow can be the rate of change of the conserved quantity, two types of energy stores can be distinguished. In a C energy storage element, the flow is integrated and the resulting generalised displacement is related to the conjugate effort of the port. For the second type of energy store, the I energy storage element, the role of effort and flow is just interchanged. The effort is integrated and the resulting generalised momentum is related to the conjugate flow. In this sense, both types are dual. For instance, an electrical capacitor, or a mechanical spring can be modelled as a C-type energy store, while a rigid body storing kinetic energy, or a coil storing magnetic energy can be modelled by an I type storage element. Energy stores can be multi-port elements. In contrast to sources, however, they cannot be modulated because this would violate the principle of energy conservation.

#### C. Irreversible transformation of energy into heat

The irreversible transformation of energy into heat, e. g. in electrical resistors, or due to friction in mechanical and hydraulic systems, is often modelled as a loss of *free* energy. In bond graphs it is represented by an R element (resistive element). If the production of entropy is taken into account, a RS element is used. The character S (Source) indicates the thermal port and expresses the entropy production.

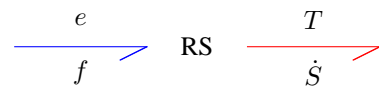


Fig. 3: Accounting for entropy production

$$e \cdot f = T \cdot \dot{S} \quad (1)$$

While the relation between the power variables of the non-thermal port may be linear, for the thermal port it is always nonlinear. According to the second law of thermodynamics, entropy production must be positive. Consequently, the graph of the constitutive relation must be within the first and third quadrant.

Like sources, resistive elements may be modulated. An electrical potentiometer, for instance, or variable hydraulic orifices controlled by the displacement of the spool in a valve may be modelled by displacement controlled R elements.

#### D. Reversible transformation of energy

In this kind of process, entropy is neither stored nor produced. Consequently, it is power conservative. There are two type of bond graph elements representing this kind process. They are denoted by the acronyms TF and GY respectively. In the most simplest case they are 2-port elements. Power conservation means that the instantaneous power at one port equals the instantaneous power at the other port. An element of type TF relates the efforts at the ports and

separately relates the flows, while an element of type GY relates the effort of one port to the flow of the other port and vice versa. In the constitutive relations of both elements, a variable is multiplied by either a constant or by a function of time. In the second case, the elements must have a signal port in addition to the power ports. This is pointed out by prefixing their acronyms with the character M (MTF, MGY). Physical components that may be modelled by a TF element are electrical transformers, mechanical gear boxes, or hydraulic displacement pumps. Examples of physical system components that can be modelled in the first place by a GY element are centrifugal pumps and electric motors. Both types of conceptual elements can be multiport elements.

#### E. Power conservative distribution of energy

Modelling of energy flows in a system means that energy is supplied by sources and conveyed and distributed between the conceptual elements of the model. As energy storage and irreversible transformation of energy have already been taken into account by energy storage elements and by resistors, distribution of energy between elements can be considered power conservative. There are two types of multiport interconnection elements called *0-junction* and *1-junction*. They both distribute power and have linear constitutive relations. For a 0-junction, the efforts of all power ports are the same and all conjugate flows sum up to zero taking into account their signs. The sign is determined by the orientation of the half arrow for the energy reference direction. If a half arrow is oriented towards a port, the flow is taken positive, otherwise the flow has a negative sign. The 1-junction plays the dual role. That is, the flows of all bonds incident the node are the same and all conjugate efforts sum up to zero by taking into account their signs. A 0-junction corresponds to an interconnection node in a network. Such a node has an effort (voltage, pressure) and according to the generalisation of Kirchhoff's current law, all flows in the branches incident to the node sum up to zero. In contrast, a 1-junction in a bond graph has no node as a counterpart in networks. The constitutive equation of a 1-junction relating all efforts is embodied implicitly in networks as it corresponds to the generalisation of Kirchhoff's voltage law. This entails some inconvenience if bond graph models are described in the widely used object-oriented modelling language Modelica, because in this framework, the description of the model structure is based on generalised networks [13]. A subgraph build of 0- and 1-junctions is called a Kirchhoff *junction structure*. If it also includes (M)TF and (M)GY elements, it is called a General Junction Structure (GJS). A bond connecting a 0- or a 1-junction to another 0- or 1-junction is called an *internal* bond. The junction structure of a bond graph is a power conservative multiport that enables the exchange of energy between all other elements, e. g., sources, sinks, energy stores and resistors as well as between component models.

#### IV. SYSTEMATIC CONSTRUCTION OF BOND GRAPHS

One of the advantages of the bond graph approach is that the topological connectivity of components in a system can guide the systematic construction of a bond graph. Two pro-

cedures can be formulated, one for the construction of bond graphs for mechanical subsystems and one for the construction of bond graphs for subsystems in energy domains other than the mechanical domain (non-mechanical subsystems).

#### Mechanical subsystems

1. Identify distinct velocities and angular velocities, represent them by a 1-junction.
2. Insert C-, R- TF- and GY-ports via a 0-junction between two 1-junctions.
3. Attach inertia 1-port elements to their respective 1-junction.
4. Attach 1-port sources and 1-port sinks to appropriate 1-junctions
5. Assign a reference direction for the energy flow to each bond (half arrow).
6. Remove all 1-junctions representing a velocity or angular velocity  $\equiv 0$  along with all incident bonds, simplify the bond graph.

#### Non-mechanical subsystems

1. Identify distinct efforts, represent them by a 0-junction.
2. Insert a power port of a source, energy store, dissipator, transformer or gyrator via a 1-junction between two proper 0-junctions.
3. Add half arrows to all bonds.
4. Choose a potential as a reference, eliminate its corresponding 0-junction along with all incident bonds. If two sub-circuits are connected via an isolating transformer, a reference potential must be chosen in each sub-circuit
5. Simplify the bond graph.

Note that there are intuitive rules for assigning power reference directions to the ports of all types of bond graph elements. The bond graph in Fig. 9 illustrates the use of half arrows. As can be seen, for sources, the half arrow points away from the element's port, for storage elements and resistors, the orientation of the incident bonds is towards the element. This is quite intuitive, sources supply energy, energy storage elements store energy temporarily and resistors transform the absorbed energy irreversibly into heat. TF and GY elements exhibit a 'through direction' of the reference direction of the energy flow. This also is intuitively understandable because energy passes through these elements without storage and without entropy production. In essence, energy reference directions are from the sources through the junction structure into energy stores, resistors and sinks. At 0-junctions in bond graphs of mechanical systems and at 1-junctions in bond graphs of non-mechanical systems, there must be a *difference* of power variables.

#### V. RULES FOR COMPUTATIONAL CAUSALITIES AT POWER PORTS

As has been explained above, at each power port of a component model, it can be decided which one of the two power variables is computed in the component model, or in other words, which of the two power variables is an outgoing signal, or an output variable in one of the constitutive relations. However, these decisions cannot be made completely deliberately. For instance, for sources there is no choice. For an effort source, the output is the effort, for a Sf



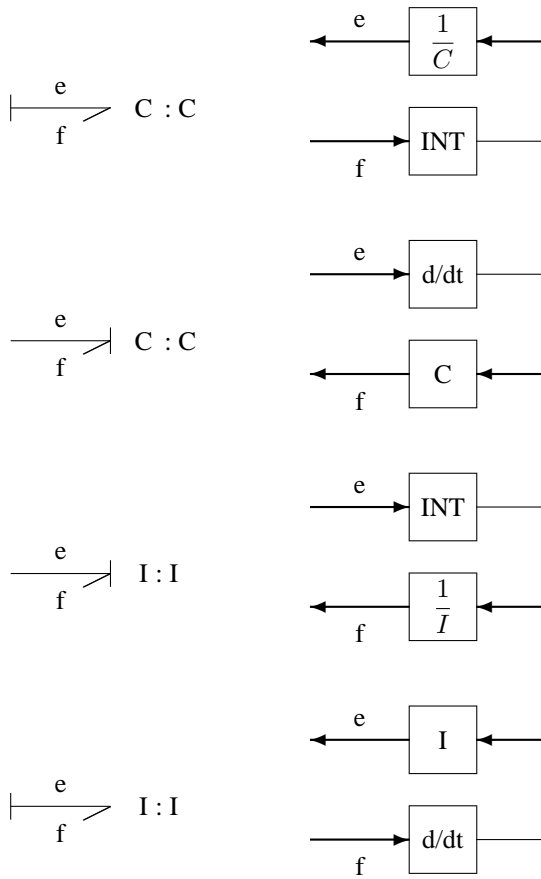


Fig. 4: Integral and derivative causality at storage ports

source, it is the flow. For storage elements so-called *integral* causality is preferred. The power variable that is integrated with respect to time is the input variable. The conjugate power variable, related to the state of the energy store, is taken as output variable. For a C energy store, this is the effort. Consequently, the causal stroke points away from the port of the C element. For the dual I element, the effort of the power port is taken as an input variable. Consequently, the causal stroke points to the port of the I element. The effort is integrated with respect to time. The flow related to the resulting state of the I energy store is its output. If, for a C energy store, the causal stroke is on the side of the power port, or if the causal stroke points away from the port of an I energy store, then this means that the output is obtained by differentiation of the conjugate power variable. In this case, so-called *derivative* causality has been assigned to the port. For resistors with linear constitutive relations, causal strokes may be either on the side of a port (conductance causality), or pointing away from the port (resistance causality). In some cases, however, there is no such choice. For a resistor representing dry friction, only the effort (force) can be the output.

At 0-junctions the effort at all incident bonds is the same. Consequently, one causal stroke can point to the junction, while all others must point away from it. For the dual 1-junctions the role of effort and flow is interchanged. That is, at one bond the causal stroke may be pointing away from the junction. At all other bonds it must point towards the 1-junction. This causal pattern reflects that one effort is equal

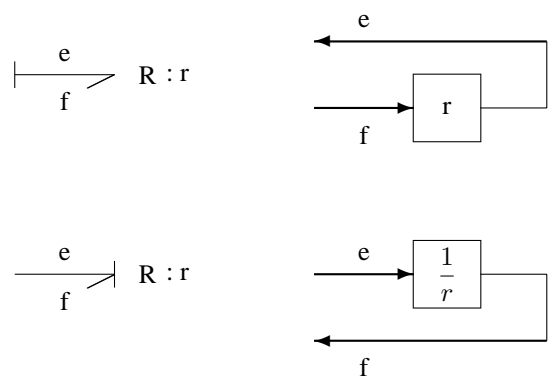


Fig. 5: Causality patterns at linear 1-port resistors

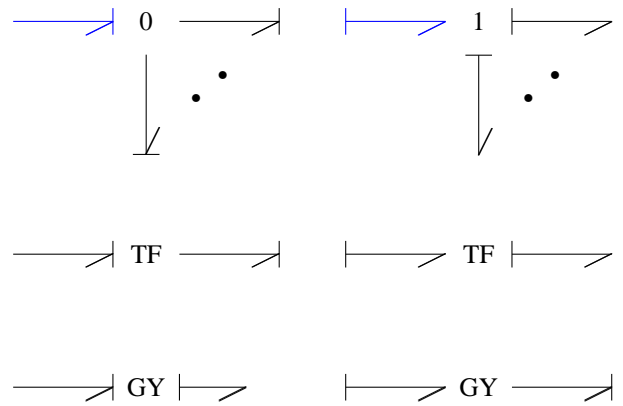


Fig. 6: Admissible causal patterns for junctions, 2-port TF and 2-port GY elements

to the sum of all other efforts and simultaneously it indicates that all flows are the same. One flow may be input to the junction, while all others are outputs. For 2-port TF and GY elements causal patterns must be as displayed in Fig. 6. A TF relates efforts. That is, if the effort at one port is an input, the effort at the other port must be an output. Since the constitutive equations of GY elements relate the effort of one port to the flow of the other one, both causal strokes must either point to the element or away from it.

Finally, causal pattern other as the admissible ones are termed *causal conflicts*. They give clear, valuable indications to consequences of modelling assumptions and may give rise to changes of the model.

## VI. SEQUENTIAL ASSIGNMENT OF COMPUTATIONAL CAUSALITIES

Once the admissible causal patterns for all bond graph elements are known, the question arises how causal strokes are to be assigned to all bonds of a bond graph. The answer to this question is the so-called *Sequential Causality Assignment Procedure* (SCAP) introduced by Karnopp and Rosenberg. Although modifications and alternatives have been proposed in the literature, this procedure has become a standard and is recalled in this tutorial.

1. Assign causality to one of the sources according to its type, *propagate* this causal information into the bond graph through its junction structure as far as possible by observing causality rules at element ports.

2. Repeat step 1 until all ports of sources are assigned an appropriate causality.
3. Assign preferred [integral](#) causality to a port of an energy store, propagate this causal information into the bond graph as far as possible. Propagation of the causality of a storage port may lead to derivative causality at power ports of other energy stores. If a causal conflict appears, e. g., at a 0- or 1-junction, it must be removed before integral causality can be assigned to the next storage port.
4. If there are any resistor ports without causality after causality has been assigned to all storage ports, then the procedure continues with assignment of causality to resistors with characteristics that do not have a unique inverse, so that there is no choice of causality.
5. Finally, if there are still resistor ports or internal bonds without causality, one resistor port or an internal bond must be chosen. Causality is arbitrarily assigned and propagated through the junction structure. This step is repeated until no causally unassigned bonds are left.

## VII. DERIVATION OF EQUATIONS FROM CAUSAL BOND GRAPHS

Once causal strokes have been added to the bonds of a bond graph, a mathematical model can be derived in a systematic manner. However, first, it must be decided for which unknowns a set of mathematical relations is to be derived. An obvious choice are the states of energy stores with integral causalities at their ports. These variables determine the energetic state of a system in the sense that they quantify the content of each energy store at all time instances  $t \geq 0$ . As the output variable of a 1-port energy store with integral causality, also called *co-energy* variable, is related to its state, it can be chosen as an alternative unknown. This choice is adopted in this paper. Note that energy stores with derivative causalities do not contribute to the system's state. Their output variable algebraically depends on the output variables of energy stores with integral causalities. In bond graph models of mechanical systems with displacement modulated elements, so-called kinematic displacements must be added to the vector of unknowns that determines the dynamic system behaviour. For further presentation of how to derive equations from a causal bond graph the notion of a *causal path* is needed. This term is frequently used in bond graph related literature.

### Causal paths:

A sequence of bonds from one power port of an element to a power port of another element is called a causal path, if there is no 2-port gyrator in between and if all bonds have their causal stroke at the same end. If there is a GY in between, the sequence of bonds is called a causal path, if all bonds on one side of the GY have their causal stroke at the same end and if all bonds on the other side of the GY have their causal stroke on the opposite end. That is, the gyrator switches the direction of efforts on one of its sides (cf. Fig. 7).

Another remarkable feature of bond graphs is that, once assignment of causalities has been completed, conclusions can be drawn with regard to the form of mathematical models that can be derived from the graph by looking for causal

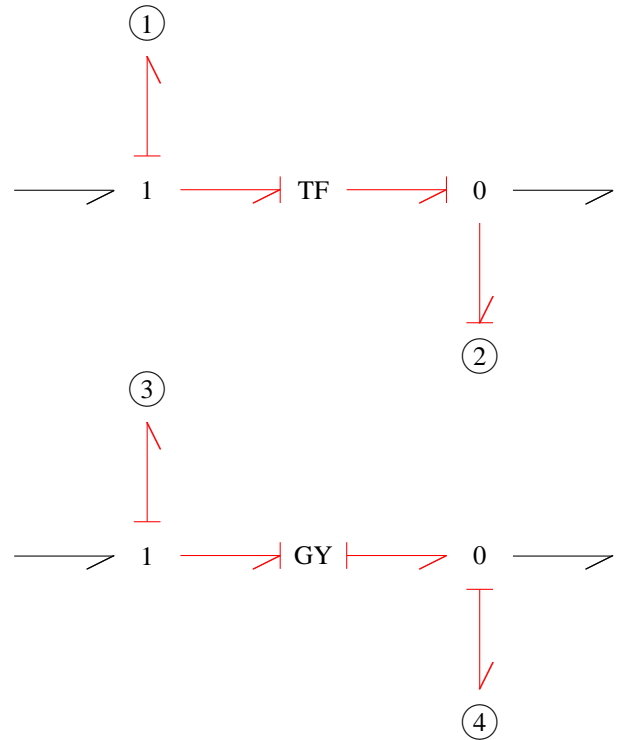


Fig. 7: Causal path between two ports

paths in causal graph. There is no need to know the actual form of nonlinear constitutive relations nor to establish and to reformulate any equations. For instance, if there are no energy storage elements with derivative causality, no causal paths between two ports of different resistors and no closed causal paths in the junction structure, then a mathematical model in the form of an explicit state space model can be derived from the graph. If, in addition, there are no elements controlled by kinematic displacements, then the order of the state space model just equals the number of storage elements.

In the following, this most simple case is considered first. One straightforward way towards the formulation of a mathematical model in a modelling language, well suited for automation, is to write the constitutive equations for all nodes of the bond graph and to have all redundancies removed symbolically. If the aim is to perform a simulation, the equations can be sorted and transformed into a programming language. If the equations are linear and if the aim is to come up with the matrices of a state space model in symbolic form to be processed by a mathematical program, e. g., the open source software package Scilab, then, clearly, all algebraic equations must be eliminated. However, auxiliary variables can be eliminated already when equations are derived from the causal bond graph by walking back and forth along causal paths. For bond graphs that are not too large, this can even be done manually in a systematic manner. The derivation of an ordered set of equations is guided by the following procedure (Borutzky [14], Wellstead [15]).

### Procedure for the derivation of equations from a causal bond graph

1. Write the constitutive equations for all independent

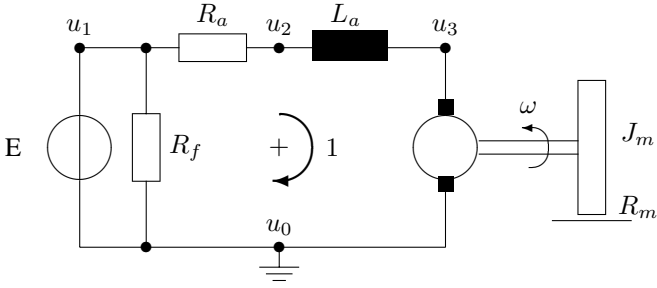


Fig. 8: Circuit schematic of a shunt motor

sources. Their outputs are given functions of time.

2. In contrast, the output of a controlled source is algebraically related to its input. If the latter is not an output of an independent source or an energy store with integral causality, then it can be expressed by means of such outputs by back propagation of causal paths in the junction structure and by eliminating intermediate variables.

3. The outputs of resistors depend algebraically on their inputs. By back propagation along causal paths through the junction structure, their outputs can be expressed by sources either independent, or controlled ones and outputs of energy stores. The outputs of dependent sources do not need to be eliminated, since they have already been determined in the previous step.

4. For storage ports, the derivative with respect to time of an output is a function of the input(s). By working back causal paths, the inputs can be expressed by outputs of other energy stores, of resistors, or sources.

If there are causal paths between resistor ports, implicit, algebraic equations will result that are likely to be nonlinear. Implicit algebraic equations also result, if there are closed causal paths in the junction structure. If there are storage elements that must accept derivative causality, then their output variables are algebraically dependent on the outputs of the storage elements in preferred integral causality and the number of states is smaller than the number of storage elements.

## VIII. A SIMPLE EXAMPLE OF A MECHATRONIC SYSTEM

What has been presented so far shall be illustrated by means of a bond graph model of the well known shunt motor. The circuit schematic and the bond graph are depicted in Figures 8 and 9 respectively. In the bond graph of the shunt motor, the 0-junction corresponds to the node with the potential  $u_1$  in the electrical circuit schematic. As the flows of bonds incident to a 0-junction add up to zero, the 0-junction corresponds to the parallel connection of the field windings and the armature windings. The armature windings has an electrical resistance and an inductance that are taken into account by the lower left R element and the left I energy store. The left-side 1-junction represents one and the same current through the resistor and the inductance. At the same time, it accounts for Kirchhoff's voltage law that the sum of all voltage drops along mesh 1 equals zero. (The voltage drop  $u_a$  is induced by the motion of the rotor.) In other words, the 1-junction corresponds to a series connection of elements with two pins. The upper 1-junction represents the current

through the field winding. As there is no I energy storage element attached to it, the inductance of the field windings has been neglected. The transformation between electrical and mechanical energy is accounted for by the gyrator as the core element in this model. The gyrator is a bilateral transducer in the sense that either electrical power can enter at one port and mechanical power exits at the other port, or vice versa. In fact, a motor could be operated as generator. As it is known, the mechanical moment of the motor acting on the shaft is a nonlinear function of the current in the field windings due to hysteresis and saturation. Accordingly, the gyrator is modulated by a variable  $\Psi$ , which for simplicity has been assumed a linear function of the current  $i_f$  through the field windings.

The 1-junction in the right-side part of the graph denotes the angular velocity of the mechanical load and simultaneously the sum of all moments acting on the load. External disturbances of the balance of moments has been taken into account by the right side effort sink.

Half arrows for the reference directions of energy flows as well as causal strokes are in accordance with the rules discussed in previous sections.

Application of the above given procedure leads to the following ordered set of equations. For simulation, these equations can be easily (automatically) formulated in a simulation language, e. g. ACSL, or in a modelling language, e. g. SIDOPS the underlying modelling language of the modelling and simulation environment 20sim<sup>TM</sup>.

### Independent sources and sinks:

$$E = f_1(t) \quad (2)$$

$$M_{load} = f_2(t) \quad (3)$$

### Dissipators:

$$u_R = R_a \cdot i_a \quad (4)$$

$$i_f = \frac{1}{R_f} E \quad (5)$$

$$M_R = R_m \cdot \omega \quad (6)$$

### Modulated gyrator:

$$\Psi = K \cdot i_f \quad (7)$$

$$u_a = \Psi \cdot \omega \quad (8)$$

$$M = \Psi \cdot i_a \quad (9)$$

### Energy stores:

$$\frac{di_a}{dt} = \frac{1}{L_a} (E - u_a - u_R) \quad (10)$$

$$\frac{d\omega}{dt} = \frac{1}{J_m} (M - M_R + M_{load}) \quad (11)$$

Right hand side expressions in these equations depend on the outputs of the storage elements with integral causality (chosen as states), on the output of the independent sources (system inputs) and on outputs of resistors previously computed. Thus, the set of equations constitute a (linear) state space model. Clearly, by substituting the constitutive relations of the resistors and of the gyrator into the differential equations of the energy stores, the matrices of the linear state space vector equation could be obtained.



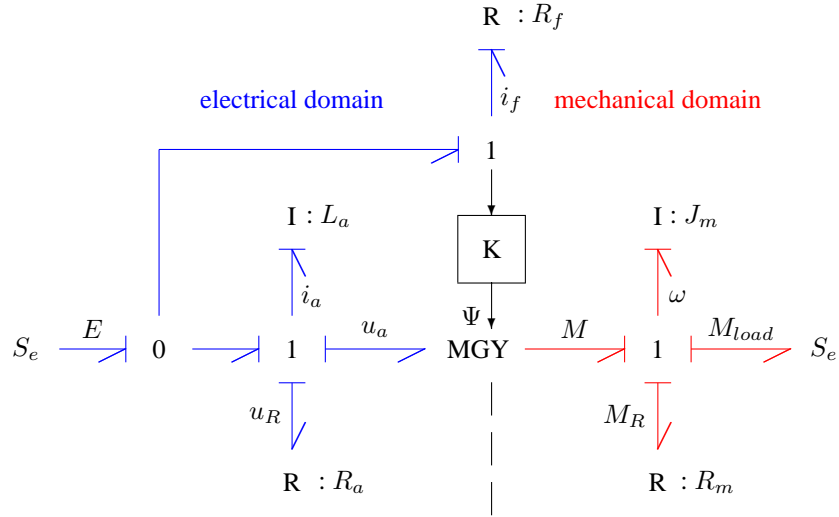


Fig. 9: Causal bond graph model of the shunt motor

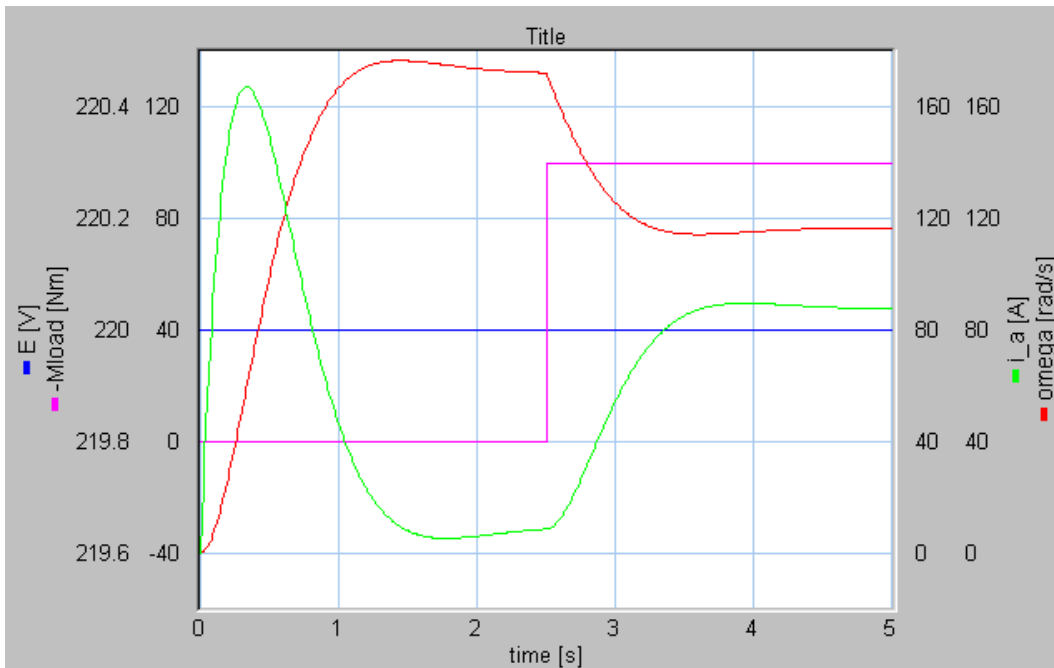


Fig. 10: Results of the shunt motor simulation

Fig. 10 displays the results of a shunt motor simulation. At  $t = 0$  s the constant voltage supply ( $E$  [V]) is switched on and immediately jumps to a value of 220 V. At  $t = 2.5$  s the motor experiences an immediate jump of the load moment from 0 to a constant level of 100 Nm. The simulation results show the time evolution of the current through the armature ( $I_a$  [A]) and of the angular velocity ( $\omega$  [rad/s]).

For high precision positioning of a mechanical load, its position and the actual angular velocity of the motor are sensed and used in feedback loops for control of the motor. In modelling and simulation environments supporting multiformalism, e. g., in 20sim, a bond graph model of the motor can be part of a block diagram of the control loop as depicted in Fig. 11. Alternatively, state space matrices in symbolic form could be derived from a linear bond graph model of the motor and passed to a block diagram based

simulation program, e. g., Simulink, or to a mathematical program, e. g., the open source software package Scilab.

Note the signal arrow directly attached to the right side 1-junction in the bond graph of the motor (Fig. 11). In the bond graph part of this combined representation, it is an activated bond which does not affect the power balance of the 1-junction. A 1-junction distributes the flow supplied at one port to all other ports. The signal arrow as an activated bond extracts this information common at all ports. The angular velocity of the load is sensed, integrated into the load's position and returned to the controller.

## IX. BOND GRAPHS - A CORE MODEL REPRESENTATION

If some storage elements in a causal bond graph must accept derivative causality and/or if causal paths exist between ports of different resistors, or if the causality assignment

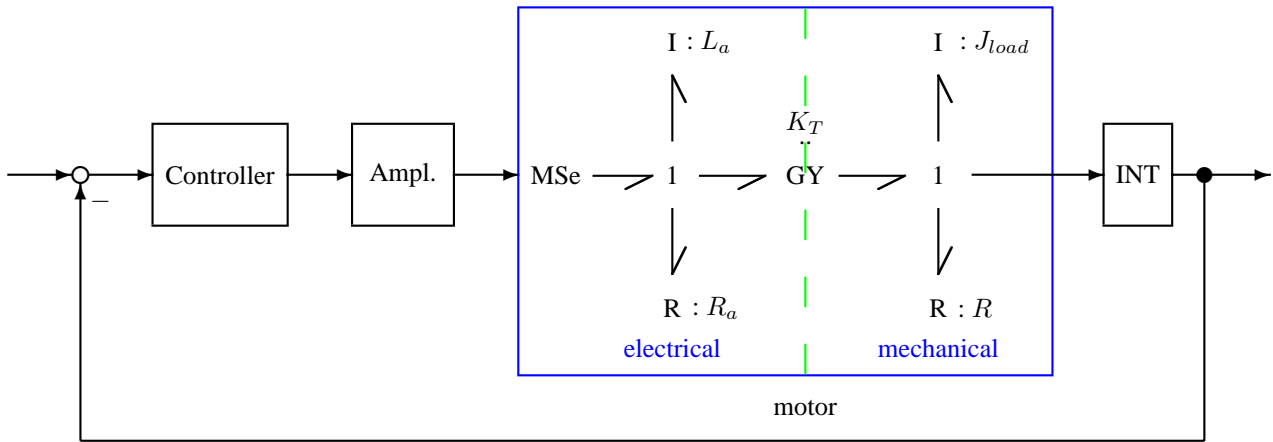


Fig. 11: Bond Graph model as part of block diagram

procedure leads to closed causal paths in the junction, then mathematical models derived from the bond graph take the form of a set of Differential Algebraic Equations (DAEs), which in general is semi-explicit and has an index  $\leq 2$ , which can be safely solved with freely available numerical codes. Bond graphs with many I energy storage elements with derivative causality easily result in the modelling of rigid multibody systems if it is assumed that rigid limbs are rigidly linked by joints. In multibody systems modelling, it is also common to use Lagrange equations of the second kind. Karnopp has introduced a procedure that enables to directly derive Lagrange equations from a causal bond graph. To that end, so-called artificial flow sources are added to the bond graph in order to identify generalised coordinates and all I energy storage elements receive derivative causality (Karnopp [16]). However, not only mathematical models in a form suitable for simulation can be derived from causal bond graphs. Brown has shown that it is not necessary to derive a signal flow diagram in order to establish transfer functions (in symbolic form) using Mason's rule. These can be directly derived from a causal bond graph (Brown [17]). Furthermore, the equations of the inverse system with respect to a pair of variables can be derived from a causal bond graph after just changing appropriately causalities and adding sources if necessary. Inverse system equations are needed for the so-called control problem, that is, system inputs are to be determined such that the behaviour of a system of known structure follows prescribed trajectories. The determination of inverse system equations is supported by so-called *bicausalities*, an extension of the causality concept introduced by Gawthrop [18]. The basic idea of bicausality is to decouple the orientation of the effort from that of the flow at a bond.

Another remarkable feature of causal bond graphs is that an inspection of causal paths can reveal information about structural controllability and structural observability of a system [19]. Further, still ongoing research is concerned with the application of bond graph modelling for fault detection and isolation and fault diagnosis (Bouamama et. al. [20], Borutzky, cf. these proceedings).

This brief survey of the capabilities of the bond graph modelling methodology shows that bond graphs can serve

as a *core* model representation, from which different information can be derived depending on the purpose of a study.

## X. BOND GRAPHS FOR MULTIBODY SYSTEMS MODELLING

Multibody systems, e. g., industrial robots, manipulators, are composed of bodies connected by certain types of joints. In modelling such systems, some bodies may be considered rigid, while for others it must be taken into account that they are flexible. Anyway, a natural approach is to develop library models for bodies and for various types of joints and to make use of the object oriented feature of bond graph modelling that enables to link component models according to the way the corresponding real world physical components are linked. A rigid body, freely moving in space, has six degrees of freedom and its motion is described by Newton-Euler equations. The number of degrees of freedom is reduced due to joints linking the body to others. One common approach in rigid multibody system modelling is to describe the translational motion of the centre of gravity with regard to a reference frame fixed in space and the body's rotation with regard to a frame located either in the body's centre of gravity or in a joint. For each direction given by an axis of the reference frame, there is a pair of power variables (velocity/force or angular velocity and moment) and a usual bond is attached to a 1-junction representing the velocity component in that direction. However, if the 1-junctions for all three components of a velocity are combined into one 1-junction and the bonds carrying power variables, say  $v_x, v_y, v_z$ , are grouped into a so-called *multi-bond* (cf. Fig. 12), then Newton-Euler equations can be represented by a concise bond graph of clear structure. In a joint, (angular) velocities are constrained by geometry and can be affected by a motor operating the joint. A joint model can be connected to a body model, but it cannot be further connected to another body model so that the joint model is sitting between two body models. The reason is that the rotation of a body is described with respect to a reference frame attached to the body. Thus, a transformation from one body fixed reference frame to another is needed. This transformation can be represented in bond graphs by a displacement modulated multiport transformer. The displacements

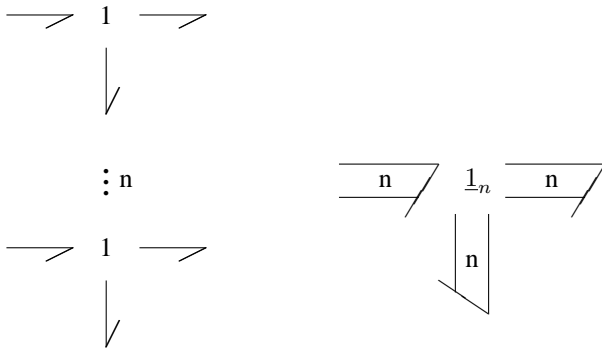


Fig. 12: Array of 1-junctions with multibonds

needed are usually Euler angles. As a result, with building blocks for a rigid body and for various types of joints and the displacement modulated multiport transformer accounting for changes between reference frames, bond graph models of rigid multibody systems can be assembled in a systematic manner. The approach briefly outlined has been introduced by Bos [21].

For some bodies of a multibody system, e. g., for long, slim limbs, the assumption of a rigid body may not be appropriate. Classical approaches to the modelling of flexible bodies are the normal modes analysis and the finite element method. Both approaches can be interpreted in terms of bond graph methodology and enable a lumped parameter model approximation of distributed parameter models. By application of the modal analysis to beams and rods, Karnopp has found a bond graph representation as early as 1968 that can be used as a basic building block for bond graph modelling of flexible mechanical structures [22]. A weak point of modal analysis is that it is a matter of engineering experience how many so-called *modal oscillators* are included. As a rule of thumb, Margolis suggests to take into account all modal frequencies up to a frequency that is twice the highest frequency of interest in the model of the overall system. A recent discussion of the finite element approach in bond graph terms can be found in [23]. A further alternative is to adopt the less widely used co-rotational formulation that can be interpreted in terms of bond graphs as well (cf. Damić in these proceedings).

## XI. CONCLUSION

The aim of this paper accompanying a tutorial, given at the ECMS 2006, has been to outline fundamental concepts of the bond graph modelling methodology and, furthermore, to indicate its potential as a powerful uniform approach to modelling, analysis, control, fault diagnosis of multidisciplinary engineering systems. By emphasising on physical principles, the methodology supports an understanding of a system's behaviour and supports the design of systems with controllers as integral part. It is hoped that readers are encouraged to have a more closer look at the vast vault of bond graph related literature and to apply this methodology and the results of ongoing research to their engineering problems.

## REFERENCES

- [1] H. M. Paynter. *Analysis and Design of Engineering Systems*. M.I.T. Press, Cambridge, Massachusetts, U. S. A., 1961.
- [2] P. C. Breedveld and J. van Amerongen. *Dynamische systemen: modelvorming en simulatie met bondgrafen*. Open universiteit, Heerlen, Nederlande, 1994.
- [3] P. Gawthrop and L. Smith. *Metamodeling: Bond Graphs and Dynamic Systems*. Prentice Hall International (UK) Limited, Hemel Hempstead, 1996. ISBN: 0-13-489824-9.
- [4] G. Dauphin-Tanguy. *Les bond graphs*. Hermes Science Europe Ltd., Paris, France, 2000. ISBN: 2-7462-0158-5.
- [5] W. Borutzky. *Bond graphs – A Methodology for Modelling Multidisciplinary Dynamic Systems*, volume FS-14 of *Frontiers in Simulation*. SCS Publishing House, Erlangen, San Diego, 2004. ISBN: 3-936150-33-8.
- [6] P. Breedveld, editor. *Journal of the Franklin Institute, Special Issue on Current Topics in Bond Graph Related Research*, volume 328/5-6. Pergamon Press, 1991.
- [7] P. Gawthrop and S. Scavarda, editors. *Proc. of the Institution of Mechanical Engineers, Journal of Systems and Control Engineering, Proceedings Part I*, volume 216/1. Professional Engineering Publishing, London, UK, 2002.
- [8] I. Troch, W. Borutzky, and P. Gawthrop, editors. *Mathematical & Computer Modelling of Dynamical Systems, Special Issue: Bond Graph Modelling*, volume 12. Taylor & Francis, April-June 2006.
- [9] D.C. Karnopp, D.L. Margolis, and R.C. Rosenberg. *System Dynamics - Modeling and Simulation of Mechatronic Systems*. John Wiley & Sons Inc., Third edition, 2000. ISBN: 0-471-33301-8.
- [10] J. F. Broenink. Introduction to Physical Systems Modelling with Bond Graphs. Technical report, University of Twente, Enschede, Netherlands, 1999. <http://www.ce.utwente.nl/bnk/papers/BondGraphsV2.pdf>.
- [11] Controllab Products. <http://www.20sim.com>.
- [12] W. Borutzky. Bond graphs and object oriented modelling – a comparison. *Proc. Instn Mech. Engrs, Part I, Journal of Systems and Control Engineering*, 216(1):25–33, 2002.
- [13] W. Borutzky, B. Barnard, and J. U. Thoma. Describing bond graph models of hydraulic components in Modelica. *Mathematics and Computers in Simulation*, 53(4-6):381–387, 2000.
- [14] W. Borutzky. Exchange and Reuse of Bond Graph Models based on XML. In J. J. Granda and F. E. Cellier, editors, *2005 International Conference on Bond Graph Modeling, and Simulation (ICBGM 2005)*, pages 129–136. SCS Publishing, 2005. Simulation Series, Vol. 37, Number 1, ISBN: 1-56555-287-3.
- [15] P. E. Wellstead. *Introduction to Physical System Modelling*. Academic Press, London, 1979.
- [16] D. Karnopp. Lagrange's equations for complex bond graph systems. *ASME Journal of Dynamic Systems, Measurement, and Control*, 99(4):300–306, December 1977.
- [17] F. T. Brown. Direct application of the loop rule to bond graphs. *Journal of Dynamic Systems, Measurement and Control*, pages 253–261, September 1992.
- [18] P. Gawthrop. Bicausal Bond Graphs. In F. E. Cellier and J. J. Granda, editors, *ICBGM'95, International Conference on Bond Graph Modeling and Simulation*, pages 83–88. SCS Publishing, 1995. Simulation Series, vol. 27, no. 1.
- [19] C. Sueur. Structural controllability/observability of linear systems represented by bond graphs. *J. of the Franklin Institute*, 326(6):869–883, 1989.
- [20] B. Ould Bouamama, A. K. Samantaray, M. Staroswiecki, and G. Dauphin-Tanguy. Derivation of Constraint Relations from Bond Graph Models for Fault Detection and Isolation. In J. J. Granda and F. Cellier, editors, *Proc. of the International Conference on Bond Graph Modeling, ICBGM'03*, pages 104–109. SCS Publishing, January 19–23 2003. Simulation Series, Vol. 35, Number 2, ISBN: 1-56555-257-1.
- [21] A. M. Bos. *Modelling Multibody Systems in Terms of Multibond Graphs with Application to a Motorcycle*. PhD thesis, Univ. of Twente, Enschede, The Netherlands, 1986.
- [22] D. Karnopp. Computer representations of continuous vibratory systems using normal modes and bond graph techniques. *Simulation*, 10(3), 1968.
- [23] A. Derkaoui, E. Bideaux, and S. Scavarda. Finite Element local structure assembly and shape functions influence on bond graph modelling. In J. J. Granda and F. Cellier, editors, *Proc. of the International Conference on Bond Graph Modeling, ICBGM'05*, pages 78–85. SCS Publishing, January 23–27 2005. Simulation Series, Vol. 37, Number 1, ISBN: 1-56555-287-3.



**Wolfgang Borutzky** is a Professor of Modelling and Simulation of Technical Systems at Bonn-Rhein-Sieg University of Applied Sciences, Germany. He received his Diploma Degree in Mathematics in 1979 and his Doctoral Degree in Mechanical Engineering in 1985, both from the Technical University of Braunschweig, Germany. He was a visiting professor at Twente University in Enschede, Netherlands (1993), at the University of Arizona in Tucson, Arizona, U.S.A (1996) and at École Centrale de Lille, France

(2001, 2002).

His main scientific interests have been modelling and simulation methodologies for multidisciplinary systems, especially Bond Graph based as well as object oriented modelling; modelling, simulation, control and fault diagnosis of mechatronic systems, modelling languages, software design for modelling and simulation of continuous, as well as hybrid systems, web-based simulation, scientific computing, numerical algorithms and software design for (parallel) continuous system simulation.

Dr. Borutzky has published extensively in major international conferences on Modelling and Simulation and in refereed scientific journals. He is author of the monograph "Bond Graphs - A Methodology for Modelling Multidisciplinary Dynamic Systems" (SCS Publishing House, Erlangen, San Diego, 2004).

Since 1990 he has served in many international scientific conferences on Modelling and Simulation in various positions, in 2005 as Assistant General Chair of the European Conference on Modelling and Simulation (EMCS) in Riga, Latvia and as General Chair of the ECMS 2006 in Sankt Augustin, Germany.

Dr. Borutzky is a member of ASIM, a senior member of SCS and currently serves on the BoD of SCS. He is also active as a member of the Editorial Board of some simulation related journals.

E-mail address: [borutzky@uni-koeln.de](mailto:borutzky@uni-koeln.de), web-page: <http://www2.inf.fh-brs.de/~wborut2m/>