TECHNICAL NOTES

The Modelica Multi-bond Graph Library

Dirk Zimmer, François E. Cellier, ETH Zürich, Switzerland
{dzimmer, fcellier}@inf.ethz.ch

Bond graphs have established themselves as a reliable tool for modeling physical systems. Multi-bonds are a bond graphic extension that provides a general approach to modeling all kinds of multi-dimensional processes in continuous physical systems. This paper presents a Modelica library for modeling multi-bond graphs and their application to three-dimensional mechanical systems. A set of bond graphic models for ideal mechanical components is provided that enables a fully object-oriented modeling of mechanical systems. The wrapping of the bond graphic models and their representation by meaningful icons gives the mechanical models an intuitive appeal and makes them easy to use. The resulting mechanical systems can be efficiently simulated. Additionally, the continuous mechanical models were extended to hybrid models that allow discrete changes to be modeled that occur in mechanical system as a result of hard impacts.

1 Introduction

1.1 Introduction to Bond Graphs

If a physical system is subdivided into small components, we observe that these components all exhibit specific behavior with respect to power and energy: Certain components *store* energy like a thermal capacitance; other elements *dissipate* energy like a mechanical damper. An electric battery can be considered a *source* of energy. The power that is flowing between components is distributed along different types of junctions. This perspective offers a general modeling approach for physical systems: bond graphs: [3, 8].

Bond graphs are a domain-neutral modeling tool for continuous system modeling in the field of physics. The actual graph represents the power flows between the elements of a physical system. The edges of the graph are the bonds themselves. A bond is represented by a "harpoon" and carries two variables: the flow, f, written on the plain side of the bond, and the effort, e, denoted on the other side of the bond [3].

The product of effort and flow is defined to be power. Hence a bond is denoting a power flow from one vertex element to another.

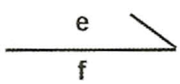

Figure 1: Representation of a bond

Domain	Effort	Flow
electrical	voltage (u)	current (i)
translational mechanics	force (f)	velocity (v)
rotational mechanics	torque (t)	angular velocity (ω)
acoustics / hydraulics	pressure (p)	volumetric flow (Φ)
thermodynamics	temperature (T)	entropy flow ($\dot{S}$)
chemical	chemical potential (μ)	molar flow (ν)

Table 1: Domain-specific effort/flow pairs

The assignment of effort and flow to a pair of physical variables determines the modeling domain. Table 1 below lists the effort/flow pairs for the most important physical domains.

The vertex elements are denoted by a mnemonic code corresponding to their behavior with respect to energy

Name	Code	*Equation*
resistance	**R**	$e = R \cdot f$
source of effort	**Se**	$e = e0$
source of flow	**Sf**	$f = f0$
capacitance	**C**	$f = C \cdot \mathrm{der}\,(e)$
inductance	**I**	$e = I \cdot \mathrm{der}\,(f)$
0-junction	**0**	all efforts equal $\mathrm{sum}\,(f) = 0$
1-junction	**1**	all flows equal $\mathrm{sum}\,(e) = 0$

Table 2: Mnemonic code of bondgraphic elements

SNE 17/3-4, December 2007

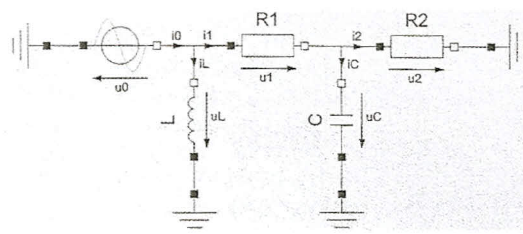

Figure 2: Schematic of electric circuit

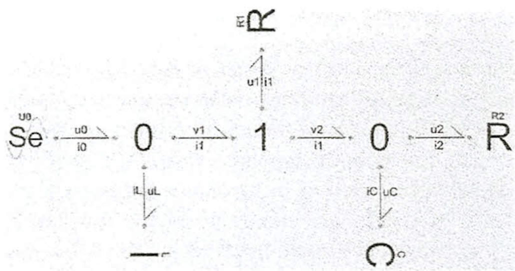

Figure 3: Bond graph of electric circuit

and power. Table 2 lists the most important bond graphic elements. The mnemonic code is borrowed from the electrical domain.

Using the bond graphic methodology, one can e.g. model the electric circuit of Figure 2 by the corresponding bond graph of Figure 3.

1.2 The Modelica BondLib

To conveniently model with bond graphs using Dymola, a Modelica library called BondLib [4] has been developed by F.E. Cellier and his students. Using this library, bond graphs can be created in an object-oriented fashion. The library provides a complete set of basic bondgraphic elements as atomic (equation) models. These can be composed graphically to form more complex composite models.

The basic bondgraphic elements and the bonds themselves can be placed on the screen by drag and drop. They can then easily be connected with each other. Further specifications can be added by means of parameter menus.

Although BondLib is a general Modelica library, its usage is strongly linked to the graphical modeling environment of Dymola [5]. Since bond graphs are a graphical modeling tool, it may be much less desirable to use this library in a purely alphanumerical modeling environment.

1.3 Introduction to Multi-bond Graphs

Multi-bond graphs (sometimes also called vector-bond graphs) are a vectorial extension of the regular

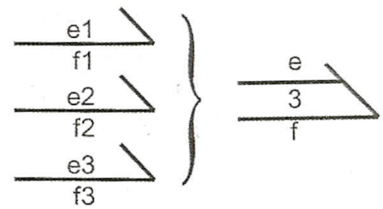

Figure 4: Composition of a multi-bond

bond graphs [2]. They are especially well suited for modeling multi-dimensional processes.

A multi-bond is composed of a certain number of bonds of either the same domain or at least closely related domains. It is represented by a (blue) double half-arrow as shown in Figure 4. The number shown at the center of the multi-bond denotes its cardinality, i.e., the number of individual bonds included.

This paper presents multi-bond graphs of mechanical systems, which is their primary application. However, multi-bond graphs can be used to model all kinds of multi-dimensional processes. Diffusion processes like heat distribution in a planar electric circuit may be another meaningful application. Yet another interesting field of application could be the modeling of chemical reaction dynamics [2]. Multi-bonds may also be applicable in the field of general relativity [7].

1.4 Advantages of Bondgraphic Modeling

Concerning the modeling of complex physical systems, bond graphs offer a suitable balance between specificity and generality. The interdisciplinary concept of energy and power flows creates a semantic level for bond graphs that is independent of the modeling domain. Thus, basic concepts of physics such as the first rule of thermodynamics can always be verified in a bond graph, independent of its application. This is particularly helpful for intra-domain models that operate in multiple energy domains. In addition, the semantic level helps the modeler avoid many types of modeling errors and find an adequate solution for his or her task. Modeling by equations is far more flexible and therefore leaves more room for mistakes.

Another advantage of bond graphs is their graphical approach to modeling. Relations can be expressed more naturally by two-dimensional drawings than in a one-dimensional equation code. Also the limitations of the screen (or drawing area) force the modeler to split his model into simple, easily understandable elements.

To us, bond graphs offer also a perfect approach to gaining a profound understanding of the basic principles covering all of physics and to organizing the knowledge concerning specific models. This makes bond graphs extremely valuable and useful for teaching purposes. However, bond graphs are a modeling tool like any other. Everything that can be modeled by bond graphs can also be modeled by other modeling paradigms. Some researchers will find bond graphs a convenient means to organizing their knowledge, whereas other researchers won't.

2 The MultiBondLib

The original BondLib only contains models for regular bond graphs. An additional library is needed to conveniently create models of multi-bond graphs. This paper presents a Modelica library for multi-bondgraphic modeling. It is called MultiBondLib and was designed to bear a strong resemblance to the existing BondLib in structure and composition. All users already familiar with BondLib should therefore be able to quickly acquaint themselves with the new MultiBondLib.

The MultiBondLib features also domain-specific sub-libraries for mechanical systems. These are introduced further on in Section 4. Although mechanics are the major field of application, the basic multi-bondgraphic elements provide a general solution to modeling all kinds of physical processes. These basic elements shall be briefly discussed in the following paragraphs.

Each vertex element of a regular bond graph has its multi-bondgraphic counterpart. The elementary equations of the basic bondgraphic elements remain the same. A transformation to the multi-bondgraphic terminology simply extends the scalar equations to vectorial form. The MultiBondLib provides these multi-bondgraphic counterparts and much more.

Just like the regular BondLib, the MultiBondLib features also causal bonds that are helpful for determining the computational causality of a model and for analyzing computational problems. A stroke denotes the side of the bond at which the flow vector is being computed. However, there is no need to ever use causal bonds in Modelica since the computational causality is determined automatically otherwise. Causal multi-bonds are also less convenient than their single-bondgraphic counterparts because the two causal multi-bonds do not cover all possible cases: mixed causality is possible as well, but a standard notation for mixed causality is missing and would probably be more confusing than helpful.

Figure 5: The two causal multi-bonds

In addition, MultiBondLib provides certain elements that are specific to multi-bond graphs. These elements handle the composition, decomposition and permutation of multi-bonds. Also special converter elements have been designed to allow a combination with the classic BondLib.

The cardinality of a multi-bond and its connected elements can be defined for each individual element separately. Consequently, it is possible to create models containing multi-bonds of different cardinality without any additional effort. Yet in many cases, the same cardinality is being used for the entire model. To afford a good usability in such cases, a default model has been developed. The default model is an outer model for all multi-bondgraphic components on the same level of the modeling hierarchy or below that defines the default cardinality of all underlying bondgraphic elements.

3 Mechanical MultiBond Graphs

In the mechanical domain, the bondgraphic effort is identified with the force, f, and the bondgraphic flow is identified with the velocity, v. The corresponding effort/flow pair of the rotational domain is torque, t, and angular velocity, ω. These assignments define the semantic meaning of the bondgraphic elements.

The inductance implements the fundamental equation $f = m \cdot dv / dt$ and represents storage of kinetic energy. The capacitance represents the storage of potential energy. It can be used to model a spring. A linear bondgraphic resistor models an ideal damper. All of these elements can be rigidly connected using a 1-junction, whereas a force interaction between two neighboring elements is modeled by a 0-junction.

Based upon this standard, various tools for the multi-bondgraphic modeling of mechanical systems have been developed since their invention by Breedveld in 1984 [2]. Unfortunately, most of these tools are incomplete, e. g. only suitable for the description of linear systems; and many are outdated.

SNE 17/3-4, December 2007

To understand the specific difficulties that arise when using a bondgraphic approach to modeling a mechanical system, it is important to note that such systems often exhibit holonomic constraints, i.e., constraints resulting from the topology of the system. No two mechanical bodies can occupy the same space at the same time, and frequently, different mechanical bodies are related to each other by a distance constraint, e. g. they may be connected to each other by a mass-less bar. Holonomic constraints are constraints based on location. However, bond graphs operate on velocities and forces only. Hence mechanical systems cannot be modeled by bond graphs alone. There is a need for additional graphical modeling tools for expressing holonomic constraints between bodies.

The MultiBondLib offers such a link to other modeling tools through sensor elements. Sensors elements "measure" certain bondgraphic variables like effort, flow, momentum (integrated effort) or position (integrated flow). In the MultiBondLib, the signal emitted by the sensor elements is a-causal and simply represents an equality equation (not an assignment). Hence sensor elements are not limited to a single usage. They may serve a number of different purposes:

- to measure bondgraphic variables,
- to convert bondgraphic variables to non-bondgraphic signals, and
- to establish algebraic relationships between bondgraphic elements.

It is this latter form of usage that enables us to state holonomic constraint equations: The positional state is derived through sensor elements and influences the dynamical behavior through modulations. It is important to note that sensor elements as well as modulations are neutral with respect to power and energy. Hence the entire energy flow is described by the bond graph alone.

The following paragraphs present two examples that include the applications of these elements in the field of planar mechanical systems. In such a system, the world is restricted to two dimensions. Models for planar mechanic systems are therefore a lot simpler than three-dimensional models. They may serve as a good introduction to multi-dimensional mechanics since such models are much more easily understandable. One major simplification in planar mechanics is the fact that the rotational inertance, J, is a constant scalar for all possible orientations, because the axis of rotation is fixed. Thus everything can be comfortably

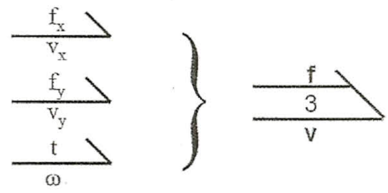

Figure 6: Composition of a planar multi-bond

computed using the coordinate vectors of the inertial system. There is no need for transformations between different coordinate systems as is the case in 3D-mechanics. All effort and flow variables of a planar mechanical bond graph can be conveniently resolved in the inertial system.

Hence planar mechanical systems can be described by multi-bond of cardinality three. Figure 6 depicts the planar mechanical multi-bond. The first two bonds belong to the translational domain, whereas the third bond belongs to the rotational domain. The effort vector of a planar multi-bond then is $(f_x, f_y, t)^{\mathrm{T}}$, whereas the corresponding flow vector is $(v_x, v_y, \omega)^{\mathrm{T}}$. Figure 7 presents the bondgraphic model of a simple planar pendulum.

The translational position is fixed at the revolute joint by a source of zero flow, *Sf*, of cardinality two. The revolute joint is free to rotate, i.e., it does not experience any torque. Hence it is modeled by a source of zero effort, *Se*, of cardinality one. The two multi-bonds are amalgamated to form a general multi-bond of planar mechanical systems of cardinality three.

The mass itself is represented by the 1-junction shown at the bottom of Figure 7. In a 1-junction, the flow variables (velocities) are equal, whereas the effort variables (forces) add up to zero. Hence the 1-junction represents the d'Alembert principle applied to the mass. The forces acting on the mass are the

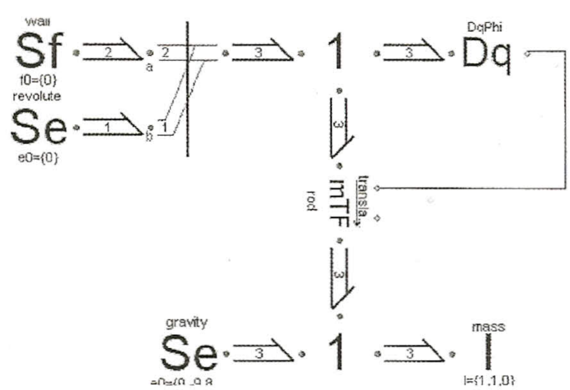

Figure 7: Bond graph of a planar pendulum

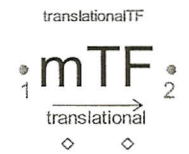

Figure 8: Icon of the modulated transformer. It provides two bondgraphic connectors (denoted by light blue circles) and two signal connectors (denoted by blue diamonds)

inertial force, *I*, and the gravitational force, which can be represented by another source of effort, *Se*, pulling in the negative *y*-direction.

The mass element is connected to the revolute joint by a mass-less rod describing a positional translation between the body element and the revolute joint. This rod is modeled by a modulated transformer element, *mTF*, that transforms the angular velocity into a translational velocity, and vice-versa.

This transformation is dependent on the current angle of the revolute joint. Therefore the transformer is modulated by the orientation angle of the revolute joint that is measured by the sensor element, *Dq*. The a-causal signal connecting the sensor element in combination with the transformer implements the holonomic constraint.

Let us take a closer look at the *mTF* element presented in Figure 8 to get a more profound understanding. The element inherits the bondgraphic effort and flow vectors (please remember the vector composition presented in Figure 6) $(\boldsymbol{e}_1, \boldsymbol{f}_1)$ and $(\boldsymbol{e}_2, \boldsymbol{f}_2)$, from its two bondgraphic connectors. These two connectors represent the hinges of a mass-less rod. Hence, the transformation between the variable pairs is specified by the parameter vector, ***d***, that represents the distance vector between the two hinges. A modulation of the transformation is determined by two signals: the current orientation φ and an optional elongation factor *ampl*. The actual model then defines an additional auxiliary variable, ***r***, and implements the balance equations of a lever:

$$\boldsymbol{r} = \begin{pmatrix} -\sin(\varphi) & \cos(\varphi) \\ -\cos(\varphi) & -\sin(\varphi) \end{pmatrix} \cdot \boldsymbol{d} \cdot ampl \qquad (1)$$

$$f_2[1;2] = f_1[1;2] + \boldsymbol{r}\, f_1[3]$$
$$f_2[3] = f_1[3]$$
$$e_1[1;2] = e_2[1;2]$$
$$e_2[3] = e_2[3] + \boldsymbol{r}^T e_2[1:2]$$

This modulated transformer, *mTF*, is a highly specialized element that hardly makes any sense outside the planar mechanical domain. Such specialized elements are thus provided in corresponding domain-specific sub-libraries, rather than listing them among the basic multi-bond graph elements of the MultiBondLib.

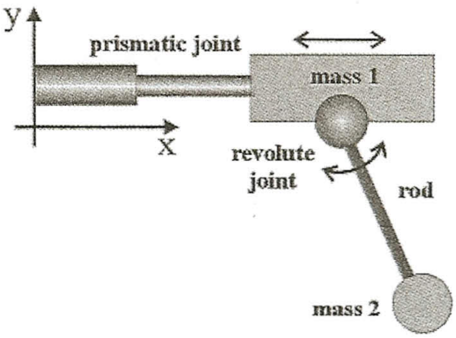

Figure 9: Schematic diagram of a crane crab

The large bond graph of Figure 10 represents the model of a second example: a simple model of a crane crab. The corresponding schematic diagram is presented in Figure 9.

Let us refrain from offering a detailed explanation of the model. Instead we shall take a look at the overall structure. The reader may observe the a-causal signals in Figure 10 that flow alongside the actual bond graph. These signals contain the variables for the current position and orientation. Whereas the multi-bond graph models the dynamics of the system, the signals handle the system's positional state.

Furthermore, multi-bond graphs of mechanical systems tend to become very large, since the sheer generality of the bondgraphic approach does not allow a more specific representation of larger mechanical elements. This makes the resulting bond graph hard to read and understand. It is therefore helpful to separate the bond graph into specific mechanically meaningful subparts as indicated by the gray rectangular frames of Figure 10. These subparts can then be represented by composite models that contain a wrapped version of the underlying subsystem multi-bond graph. The resulting model is presented in Figure 11 and is now easily understandable.

4 The Mechanical Sub-libraries

Two libraries for the modeling of mechanical systems have been developed, one for planar mechanics: "PlanarMechanics," and the other for 3D-Mechanics: "Mechanics3D". Both libraries are included as subpackages within the MultiBondLib. The following discussion will now focus on the components of the Mechanics3D library. However, most of the subse-

SNE 17/3-4, December 2007

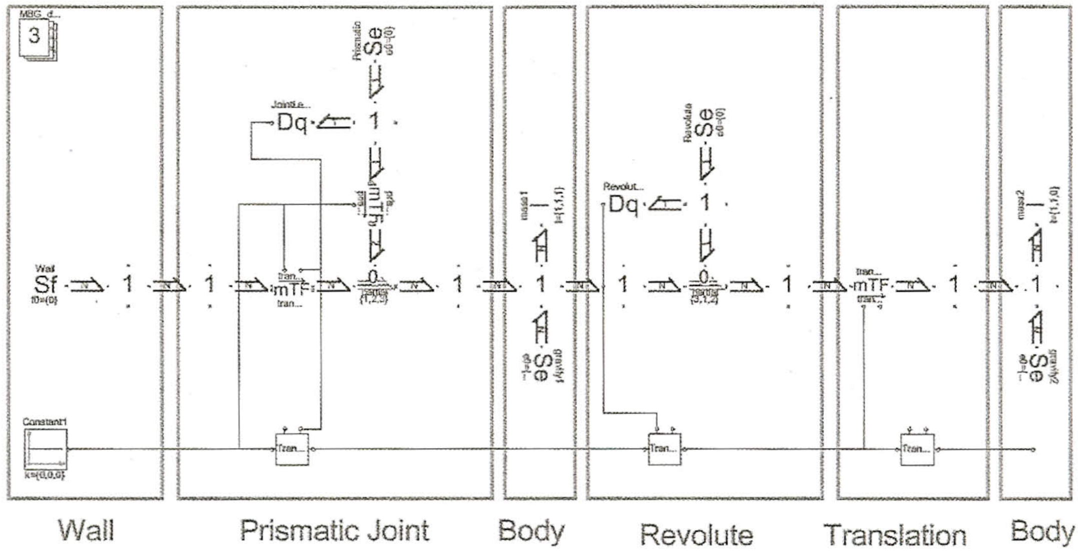

Figure 10: Multi-bond graph of a crane crab

quent descriptions hold for the planar mechanical library as well.

Similar to the Multi-body systems library of M. Otter [9], the Mechanics3D library offers models for an extensive set of mechanical components. It contains models for rigid parts such as bodies and rods, as well as models for different kinds of joints such as revolute joints or prismatic joints. A spring and a damper are typical examples of force elements for which models are offered in the library as well. In addition, models of ideal rolling objects such as wheels or marbles are provided. All of these elements can be further specified by parameter menus and feature a suitable animation.

Due to the similarity in structure and design, users of the standard Modelica MultiBody library will find the Mechanics3D library very easy to use. The Mechanics3D library represents both a subset and a superset of the MultiBody library. Yet in contrast to the standard MultiBody library, all mechanical models of the new Mechanics3D library are based upon wrapped bondgraphic models. A look inside the models reveals a bondgraphic explanation.

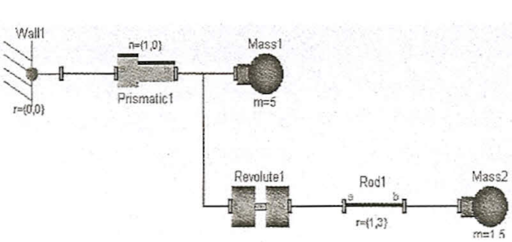

Figure 11: Multi-body diagram of a crane crab

4.1 Connector Types

In this section, we discuss the selection of connector variables for the mechanical components in Mechanics3D. This selection is defined by the process of wrapping. Therefore all bondgraphic variables have to be part of the connector. These are:

- the force vector, $\boldsymbol{f}$, (flow variables),
- the torque vector, $\boldsymbol{t}$, (flow variables),
- the velocity vector, $\boldsymbol{v}$, (potential variables),
- the angular velocity vector, $\boldsymbol{\omega}$, (potential variables);

where the variables of the rotational domain are resolved with respect to their corresponding body system. All other variables are resolved in the inertial system. Also the variables of the positional signals are part of the connector:

- the position vector, $\boldsymbol{x}$, (potential variables),
- the orientation matrix, $\boldsymbol{R}$, (potential variables).

Please note that the connector contains redundant information. The variables $\boldsymbol{v}$ and $\boldsymbol{\omega}$ are derivatives of $\boldsymbol{x}$ and $\boldsymbol{R}$. Also $\boldsymbol{R}$ itself is 3×3 orientation matrix and thus a redundant way of expressing the current orientation.

Although the Mechanics3D library and the MultiBody library are very similar, the Mechanics3D library defines its own slightly different connectors. Hence it is not possible to combine the components of these two libraries without additional tools.

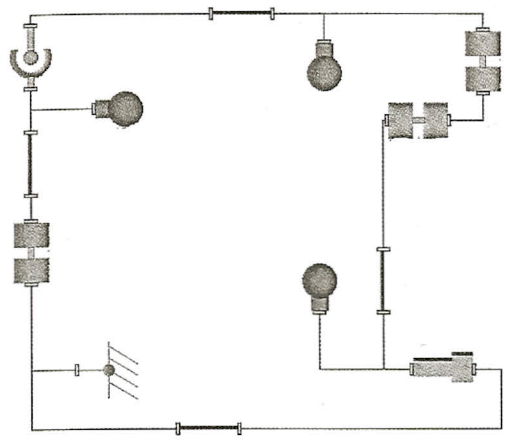

Figure 12: Multi-body diagram of a 3D kinematic loop

4.2 Kinematic Loops

The redundancy of the connector causes problems when components are connected in a circular fashion, as this is often the case in kinematic loops. A standard connection statement leads then to a singularity in the model. To overcome these difficulties, a second alternative connection statement is needed that contains only non-redundant information. For each kinematic loop, one standard connection has then to be replaced by its non-redundant counterpart.

However, the user of the library does not need to worry about these details, because the replacement is achieved automatically. To implement the necessary preprocessing of the model, a set of special Modelica functions[1] has been used. These are the same methods that have also been applied in the case of the standard MultiBody library (see [9]).

Figure 12 shows the Mechanics3D model of a 3D kinematic loop. No special cut joints have been used, and the elimination of redundant equations and redundant states takes place automatically. Whereas the automatic elimination algorithm works in most cases, planar loops within a 3D environment still require special treatment. A special cut joint has been developed for this purpose.

4.3 An Example: The Bicycle

This example was created by components of Mechanics3D and presents the model of an uncontrolled ideal rolling bicycle. Such a bicycle has seven degrees of freedom on the level of position and three degrees of freedom on the level of velocity. It is a partially stable system for a specific range of driving velocities. A nice description of the dynamic behavior of a bicycle can be found in a paper by K. Åström et al. [1]. The relevant parameter values for mass and geometry of this specific model are taken out of a paper by Schwab et al. [11]. The described bicycle is self-stabilizing for driving velocities between 4.3 ms^{-1} and 6.1 ms^{-1}.

Figure 13 depicts the multi-body diagram of the bicycle model. Although no closed loop is visible in the multi-body diagram, the model contains a closed kinematic loop, since both wheels are connected to the road. The Mechanics3D library offers an outer world3D model that is used in exactly the same fashion as the corresponding model of the standard MultiBody library.

The selected state variables are the cardan angles of the rear wheel and their derivatives. These three cardan angles represent the orientation on the plane, the lean of the rear frame, and the roll angle of the rear wheel. Additional state variables are the position of the rear wheel on the plane, the angle of the steering joint, and the angle of the front revolute joint. Each of the two wheels defines one holonomic constraint equation that prevents the bicycle from sinking into the road.

Figures 14 and 15 show the results of the simulation. The lean angle can be examined in a plot window, and the bicycle is nicely animated within Dymola.

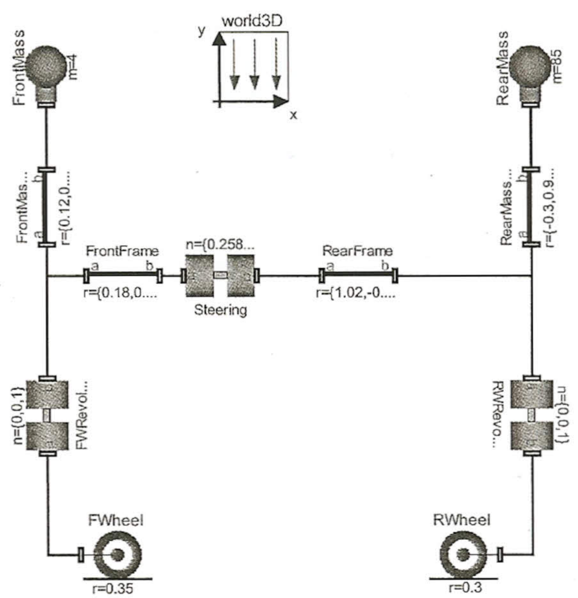

Figure 13: Multi-body diagram of a bicycle

[1] These are: defineRoot(), definePotentialRoot(), defineBranch(), isRoot(), and rooted().

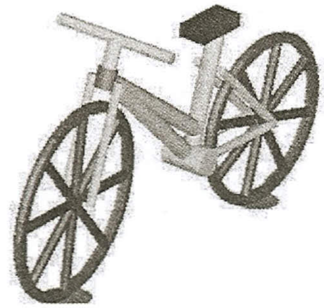

Figure 14: Animation model of the bicycle

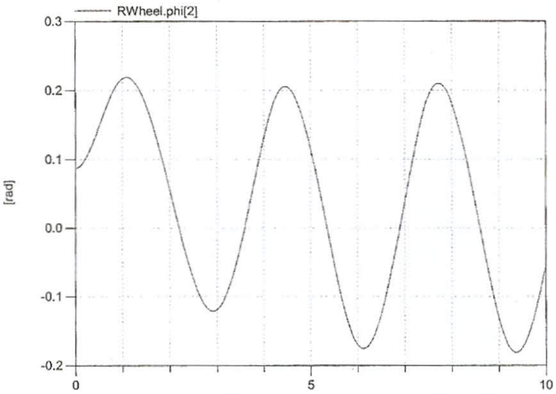

Figure 15: Plot of the lean angle

4.4 Run-time Efficiency

The selection of the state variables is of major importance for the efficiency of the resulting simulation. Usually this selection is automatically achieved by Dymola. Nevertheless variables can be suggested to the simulation program via the advanced Modelica attribute: "StateSelect". These variables are the states of a joint's relative position and velocity. Each joint is declaring state variables, unless there is a kinematic loop.

In the latter case, there are often many equivalent sets of state variables. Dymola uses the feature of dynamic state selection, where the state variables are chosen at run time. This offers the most flexible solution but leads to a slower simulation and to many additional equations that are often unnecessary. Hence it is strongly recommended to determine a static set of state variables manually in such a case whenever this is feasible.

All mechanical components that contain potential state variables have the option of a manual enforcement of their state variables via the parameter menu. To this end, the Boolean parameter "enforce-States" needs to be activated.

	MultiBody		Mechanics3D	
Experiment	Non-lin. eq.	Integr. steps	Non-lin. eq.	Integr. steps
Double Pendulum	0	549	0	549
Crane Crab	0	205	0	205
Gyroscopic Exp. with Quaternions	0	24438	0	25574
Planar Loop	2	372	2	372
Centrifugal	{2,2}	70	{2,2}	70
FourBar Loop*	5	446	5	625
Bicycle*	1	97	1	84

Table 3. Comparison of the two mechanical libraries

The Mechanics3D library and the standard MultiBody library contain elements for the same purpose that can be used in very similar ways. It is therefore easily possible to translate the model for a mechanical system from one library to the other. Both libraries are examined with respect to their run-time efficiency. The complexity of the resulting systems of equation and the computational effort of the simulation are compared for a given set of examples. The results of this examination are presented in Table 3. There is hardly any difference. In fact, the generated equations of both libraries are very similar. There is only one remarkable difference: In contrast to the Mechanics3D library, the translational velocity is not part of the connector variables of the standard MultiBody library. The elements are only connected by the translational position. The equations for the translational velocities are then derived (when necessary) by differentiation. In the bondgraphic models of the Mechanics3D library, these equations are explicitly stated.

Table 3 lists the sets of non-linear equations for a given set of experiments. The number of integration steps is counted for a simulation period of 10 seconds. The simulation method was Dassl with a tolerance of 10^{-4} A * indicates that the parameters of the experiment setup differ slightly between the two libraries.

5 Conclusions and Further Work

A Modelica library for convenient multi-bondgraphic modeling has been developed. It provides a general solution to modeling all kinds of multidimensional processes in continuous physical systems. Multi-bond graphs offer a good framework for

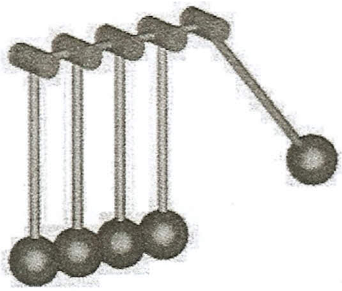

Figure 16: An implementation of Newton's cradle

modeling mechanical systems. The bondgraphic methodology proved to be powerful enough for modeling all important ideal subparts of mechanical systems accurately and efficiently.

The resulting libraries for mechanical systems PlanarMechanics and Mechanics3D provide an extensive set of component models. These domain-specific models have an intuitive appeal and are easy to use. They consist in wrapped bondgraphic models, and a closer look reveals a bondgraphic explanation of their behavior. Also quality and efficiency of the resulting solution are not impaired by the bondgraphic modeling technique. The selected state variables are chosen wisely, and the resulting systems of equations can be solved fast and accurately.

Furthermore, a third library for mechanics has been developed and included in the MultiBondLib. It is called "Mechanics3DwithImpulses" and represents an extension of the Mechanics3D library. The existing purely continuous mechanical models were extended by their corresponding impulse equations to hybrid models that contain an additional discrete event part.

This library was designed to model the behavior of mechanical systems in situations of hard ideal impacts. The type of mechanical system is thereby not limited at all. Impacts can be modeled between single objects, as well as between kinematic loops (e.g. a car suspension). The provided parameter set for the impact characteristics includes also specifications for elasticity and friction.

Sadly, the Modelica support for discrete event modeling [10] is not yet fully sufficient. Hence the resulting solution leads partially to models that are unnecessarily complicated in execution and design. Nevertheless the solution is fine and workable for small scale models. Figure 16 shows a simple example of a mechanical system modeled using the extended 3D library.

References

[1] Åström, K.J., R.E. Klein and A. Lennartsson, "Bicycle Dynamics and Control: Adapted Bicycles for Education and Research," *IEEE Control Systems Magazine*, **25**(4), pp.26-47, 2005.

[2] Breedveld, P.C., *Physical Systems Theory in Terms of Bond Graphs*, Ph.D. dissertation, University of Twente, The Netherlands, 1984.

[3] Cellier, F.E., *Continuous System Modeling*, Springer-Verlag, New York, 755p, 1991.

[4] Cellier, F.E. and A. Nebot, "The Modelica Bond Graph Library," *Proc. 4th International Modelica Conference*, Hamburg, Germany, Vol.1, pp.57-65, 2005.

[5] Dynasim AB, *Dymola Users' Manual*, Version 6.0, Lund, Sweden, 2006.

[6] Elmqvist, H., M. Otter and J. Díaz López, "Collision Handling for the Modelica MultiBody Library." *Proc. 4th International Modelica Conference,* Hamburg, Germany, Vol.1, pp.45-53, 2005.

[7] Gussn, N.M.N, *On the Bondgraphic Power Postulate and its Role in Interpreting the 1905 and 1915 Relativity Theories*, MS Thesis, Dept. of Electr. & Comp. Engr., University of Arizona, Tucson, AZ, 1994.

[8] Karnopp, D.C., D.L. Margolis and R.C. Rosenberg, *System Dynamics: Modeling and Simulation of Mechatronic Systems, 4th edition*, John Wiley& Sons, New York, 576p, 2006.

[9] Otter, M., H. Elmqvist and S.E. Mattsson, "The New Modelica MultiBody Library," *Proc. 3rd International Modelica Conference*, Linköping, Sweden, pp.311-330, 2003.

[10] Otter, M., H. Elmqvist and S.E. Mattsson, "Hybrid Modeling in Modelica Based on the Synchronous Data Flow Principle," *Proc. IEEE International Symposium on Computer Aided Control System Design*, Hawaii, pp.151-157, 1999.

[11] Schwab, A.L., J.P. Meijaard and J.M. Papadopoulos, "Benchmark Results on the Linearized Equation of Motion of an Uncontrolled Bicycle" *KSME Journal of Mechanical Science and Technology*, **19**(1), pp.292-304, 2005.

[12] Zimmer, D., *A Modelica Library for MultiBond Graphs and its Application in 3D-Mechanics*. MS Thesis, ETH Zurich, Switzerland, 2006

Corresponding author: Dirk Zimmer
Institute of Computational Science
ETH Zurich,
CH-8092 Zurich, Switzerland
dzimmer@inf.ethz.ch

Accepted Modelica 2006 Conference, June 2006
Received: December 6, 2006
Accepted: January 20, 2007