An algorithm for automatic identification of R-fields in bond graphs

by S. J. Hood
R. C. Rosenberg
D. H. Withers
T. Zhou

Bond graphs may be used to model the power flow in dynamic systems. They are especially attractive for modeling systems which function in coupled energy domains, for example, electromechanical systems. For such systems, bond graphs can be used to provide a natural subdivision into power/energy fields: storage, sources, transformers, and dissipation. In the case of nonlinear dissipative fields, implicit, nonlinear, coupled systems of algebraic equations may arise. Causality assignment on the bond graph provides a basis for detecting implicit formulations. This paper presents an algorithm for detection and solution of these forms within a model, thereby providing an opportunity for efficient numerical solution, and includes a brief introduction to bond graphs via an electromechanical system example.

Introduction

Bond graphs are playing an increasingly useful role in the modeling, analysis, and design of engineering systems. In this paper we introduce bond graphs, identify an important, practical computational problem, and present an algorithm to help provide better feedback to the modeler and improve the efficiency of computation.

The particular problem of interest is that of coupled nonlinear algebraic equations and their solution. We focus on the identification, formulation, and calculation of such a problem in the bond graph context. Our particular contribution is to show how to maximize the use of bond graph causality information in conjunction with data from the graph structure. This leads to increased efficiency in calculating solutions. In the process of presenting the algorithm, we answer a previously posed question regarding a minimum set of computing variables [1].

In the next section, bond graphs are introduced briefly and are illustrated for an electromechanical system. Following that, the general equation formulation procedure now in common use is described. The specific computational problem, that of R-fields, is elucidated, and the current state of the art is summarized. Next, a new algorithm is presented and applied to some examples. Finally, a summary is given.

A brief introduction to bond graphs

• *History*

The basic concept of bond graphs and many details of their representation were developed by H. M. Paynter in 1959 and presented in his classic text published in 1961 [2]. Further research and development focused on extending modeling applications and building up a set of automated processing procedures and algorithms [3]. Progress has continued to the point that a recent bibliography contained references to 290 papers and 10 books [4]. Interest in bond graph methods has

become international, as a review of that bibliography indicates.

Advantages

The major advantages of bond graph modeling are that in such modeling a topological structure is used to represent the power/energy characteristics of engineering systems, and that systems with diverse energy domains are treated in a unified manner. A topological representation, such as a bond graph, offers great advantage at the conceptual design level, since quantitative details are not required prematurely. In addition, graphical representations document complex models clearly and unambiguously. They often are the easiest way for a group of engineers to communicate the description of energy flows in dynamic systems.

Since a bond graph is an unambiguous representation of an energy system, it is possible for a computer program to automatically generate the equations for dynamic analysis of the system [5–8].

Because the bonds in bond graphs represent the power coupling, such models apply to mechanical translation and rotation, electrical circuits, thermal, hydraulic, magnetic, chemical, and other physical domains. They are especially useful in systems which function in coupled domains, such as electromechanical systems. We illustrate this aspect later.

Disadvantages

The major disadvantage of bond graphs is that the notation is new. Experienced modelers sometimes find it difficult to change from the methods of block diagrams and state equations to bond graphs.

Notation

The two graph elements of a bond graph are the node (or vertex) and the bond (or edge). The node denotes a multiport element, with associated energy laws. The bond denotes power flow between a pair of nodes. A node, or multiport, is usually represented by a letter or number. A bond is represented by a line with a half-arrowhead at the end, indicating the direction of positive power; an information signal is represented by a line with a full arrow at the end. See Figure 2 (discussed later). Associated with each bond is a pair of scalar variables, an effort (e) and a flow (f). Their instantaneous product is the power on the bond.

There are nine basic multiports used to model a wide variety of engineering systems. They are given in **Table 1**, together with their definitions. When the basic name is used as the initial part of a node label, it is typically assumed to indicate the type of multiport. For example, R3 is an R type of node.

In Table 1, the first two entries denote system inputs. E is an effort source; F is a flow source. These two nodes are always 1-ports (i.e., exactly one bond is incident). In the table and in general, t denotes time, the independent variable. The third entry is R, the generalized resistance effect (i.e., dissipation). Note that an R node can have any number of ports greater than zero. In the case of more than one port, e, f, and ϕ are vectors. In certain modeling domains, 2- and 3-port R nodes are not uncommon.

Table 1 The elements of bond graphs.

Multiport type	*Equation*	*Action*
E——	$e = \phi(t)$	Source of effort
F——	$f = \phi(t)$	Source of flow
R——	$\phi(e, f) = 0$	Dissipation
C——	$\phi(e, q) = 0$	"Potential" energy storage
I——	$\phi(f, p) = 0$	"Kinetic" energy storage
$\vert_2$ ——$_1$0——$_3$	$\Sigma f_i = 0$ $e_1 = e_2 = e_3$	Zero junction (Common effort)
$\vert_2$ ——$_1$1——$_3$	$\Sigma e_i = 0$ $f_1 = f_2 = f_3$	One junction (Common flow)
——$_1$T——$_2$	$e_2 = ne_1$ $f_1 = nf_2$	Transformer
——$_1$G——$_2$	$e_1 = rf_2$ $e_2 = rf_1$	Gyrator

To discuss the next two node types, C and I, we must first introduce two additional variable types, q and p, the generalized displacement and momentum, respectively:

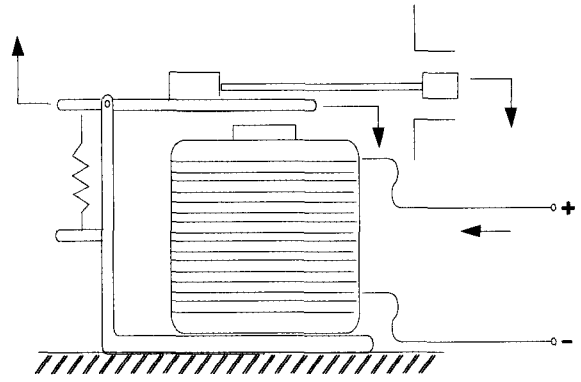
$$q(t) = q(t_0) + \int_{t_0}^{t} f(\tau)d\tau, \tag{1}$$

$$p(t) = p(t_0) + \int_{t_0}^{t} e(\tau)d\tau. \tag{2}$$

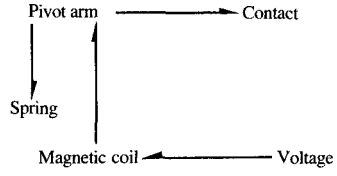
These equations are frequently used in the related form $dq/dt = f$ and $dp/dt = e$, respectively. The C and I nodes denote energy storage effects. Their constitutive equation forms are given in the table. Note that both C and I may have more than one port. In that case, the associated variables and functions become vectors.

Perhaps the most important node types in bond graph modeling are the ideal power junctions, 0 and 1. (Paynter thought that this was the case.) Each obeys a power conservation law: namely, the net power into the multiport at each instant is zero. In electrical circuit modeling, a 0-junction represents a parallel connection (common effort) and a 1-junction represents a series connection (common flow). In mechanics, a 1-junction is typically associated with a mass (common velocity), while a 0-junction is often associated with a spring (common force) or a damper.

Finally, the transformer T and the gyrator G are power-preserving node types. They are strictly 2-port nodes, and are very useful in modeling transducer and power-conversion effects. The moduli n and r do not have to be constant. When one is a function of other system variables, the



(a)



(b)

Figure 1

An electromechanical relay. (a) Sketch of relay. (b) Major components of relay.

transformer or gyrator is said to be modulated. Such a node is designated by the input of a signal bond denoted by a full arrow on the end.

• *An electromechanical example*

A sketch of an electromechanical relay and a diagram labeling the major components are shown in **Figure 1**. The voltage input energizes the magnetic coil. The magnetized coil attracts the pivot arm, pulling it down toward the coil. The spring is the return spring that holds the pivot arm open when the coil is not energized. The contact at the end of the pivot arm closes an electrical circuit (not shown) when the pivot arm closes.

The detailed bond graph for this relay model is shown in **Figure 2**. The input voltage is modeled by an effort source, E. The resistance in the windings around the magnetic coil is represented by the R element on bond 2. The gyrator element, G, relates the voltage input to the coil to the magnetic flux output. The C element models energy storage in both the iron in the coil and the air gap above the coil. Bonds 7, 8, and 9, signal 10, and the attached nodes form an impact model for the pivot arm hitting the top of the coil. This impact model is repeated (bonds 21, 22, and 23, signal 24, and attached nodes) in the contact model. The impact model includes a signal with a full arrow on the end. The signal indicates information flow with negligible energy flow. The information in this case is whether or not the pivot arm or contact is closed, making physical contact. There is energy dissipation, represented by the R element, only when there is physical contact. When the pivot arm or contact is open, there is no energy flow through bond 8 or bond 22. The pivot arm is represented by bonds 12, 13, 14, 16, 17, 18, and the attached nodes. The transformers convert translational velocities to rotational velocities and back again. The flow variable for bonds 12, 13, 14, and 16 is the rotational velocity at the pivot point. The I element models kinetic energy storage. The C element models potential energy storage due to the bending in the pivot arm. Finally, the I element attached to bond 20 represents the mass of the contact at the end of the pivot arm, and the remaining bonds and nodes form the impact model for the contact hitting the stop.

There are standard modeling procedures that help make the task of bond graph model generation routine [9].

The R-field problem

• *Simulation objectives*

Our major objective is to automate the process of formulating and solving the state equations associated with bond graph models of engineering systems. A secondary objective is to provide timely and insightful feedback to the designer. There are several sources of difficulty in accomplishing the major objective. The one we wish to focus on here is that of coupled nonlinear algebraic equations that arise when the R nodes are connected in particular ways in the model. The implicit equations are often difficult to solve, and they typically must be solved several times in each integration step. Consequently, it is helpful to be able to inform the modeler in detail of the existence of such coupling. Furthermore, increasing the efficiency with which such solutions are obtained can dramatically decrease the overall solution time.

Algebraic loops may be broken by introducing parasitic elements, but doing so results in stiff differential equations which are difficult to solve numerically.

• *Identifying R-fields*

R-fields arise when there are several coupled dissipative effects and very little associated energy storage.

A given bond graph can be thought of as having several fields, depending upon the types of basic nodes that are present. **Figure 3** shows a sorting of the nodes by power/energy features. We have included the T and G nodes with the 0 and 1 nodes, since they all conserve power strictly.

In preparation for formulating the system equations, we assign to the bond graph model a set of indicators called

"causality strokes." One of these is assigned to each bond, giving the bond a causal orientation. The purpose of causality is to define input/output relations at each of the nodes. Nodes such as E and F have a required causal orientation, while nodes such as 0, 1, T, and G must meet certain required conditions. For example, a 0-junction must have exactly one stroke next to it, indicating the independent (input) effort variable. Causal orientation is indicated by a short perpendicular stroke at the effort end of the bond. The causality assignment is summarized in the templates in **Figures 4(a)–(f)**.

When causality has been assigned to the bond graph according to the Sequential Causality Assignment Procedure (SCAP) [9], it is possible to identify each separate implicit R-field within the graph. As a result of assigning causality, the diagram of Figure 3 can be converted to a computing diagram based on a set of key vectors for the various fields in the graph. This diagram is shown in **Figure 5**. For convenience we have assumed that T and G nodes have constant moduli; they can then be incorporated into the junction structure component (JS), creating a weighted junction structure (WJS).

The system equations implied by the diagram of Figure 5 are

$$Z = \phi_s(X) \quad \ldots \text{storage field} \quad (3a)$$

$$D_0 = \phi_L(D_i) \quad \ldots \text{dissipation field} \quad (3b)$$

$$U = \phi_u(t) \quad \ldots \text{source field} \quad (3c)$$

$$dX/dt = S_{11}Z + S_{12}D_0 + S_{13}U \quad \ldots \text{junction structure} \quad (3d)$$

$$D_i = S_{21}Z + S_{22}D_0 + S_{23}U \quad \ldots \text{junction structure} \quad (3e)$$

$$V = S_{31}Z + S_{32}D_0 + S_{33}U \quad \ldots \text{junction structure} \quad (3f)$$

Proper use of causality leads to considerable insight about the nature of the equation structure. For example, one can tell at the causal graph level whether dependent C and/or I ports exist.

• *Current status*

Once causality is assigned using the SCAP, it is known whether or not S_{22} is zero [10]. If S_{22} is zero, then no implicit R-fields exist, and a straightforward procedure for integrating the system equations can be employed. If the bond graph contains some implicit R-fields (IRFs), they can be identified and isolated by the proper use of causality data [11]. The D_0 and D_i vectors can be sorted into explicit and implicit subsets. Then the implicit D_0 and D_i can be further grouped by their fields, leading to a block form of equations. These features have been implemented in the ENPORT-6 program, a nonlinear bond graph processor [5].

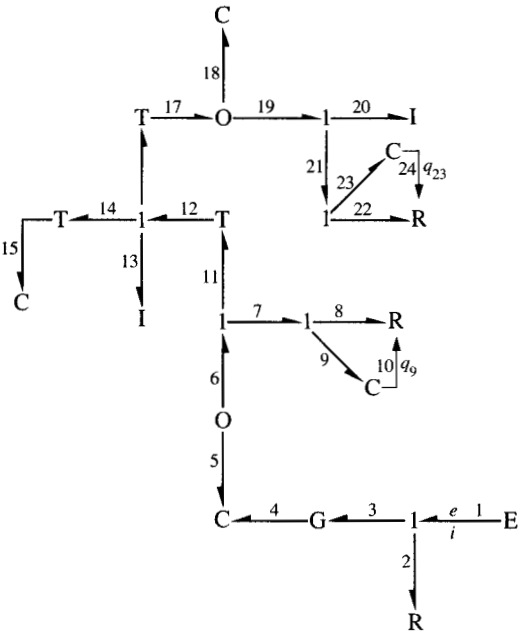


Figure 2

Bond graph model of the relay; q is displacement.

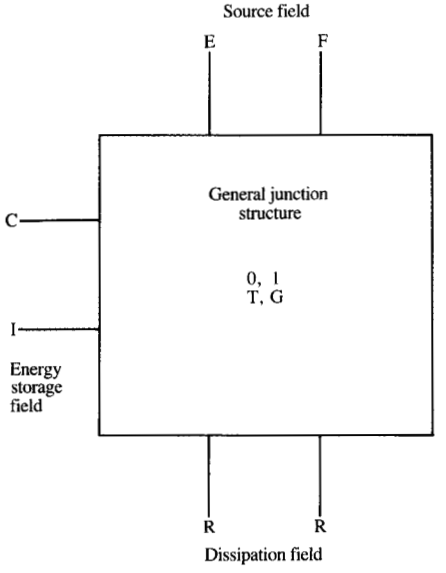


Figure 3

Fields in a bond graph.

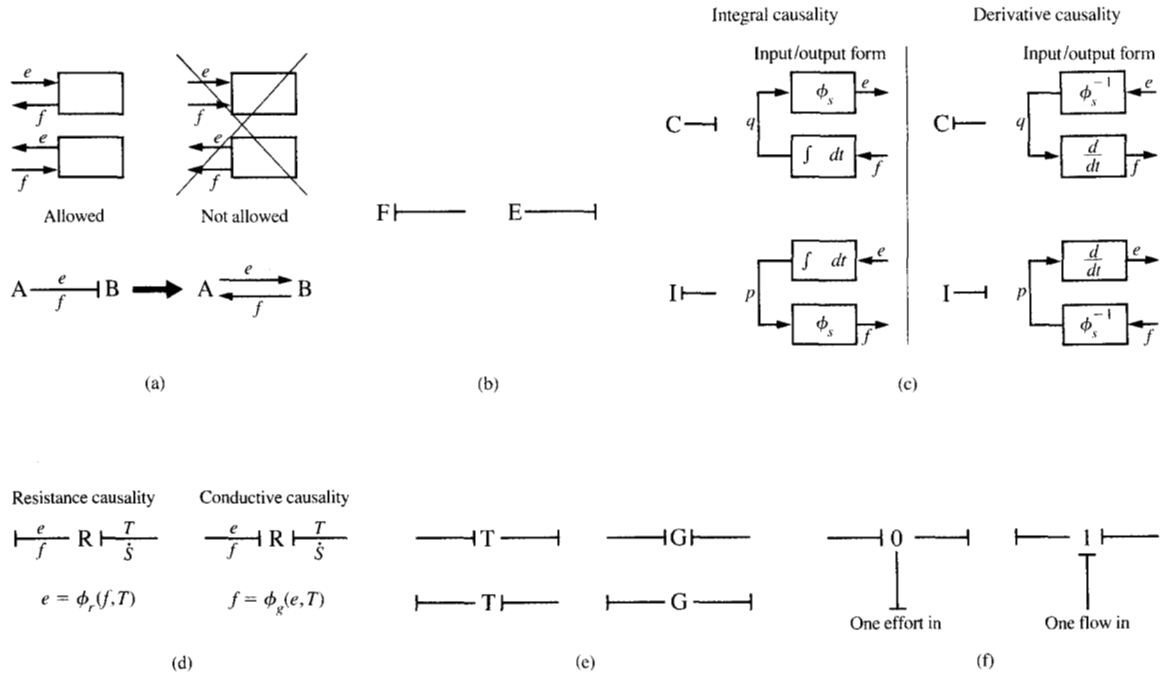


Figure 4

Causality. (a) Input/output relationships defining causality. (b) Causality for flow source F and effort source E. (c) Causality for potential energy storage element C and kinetic energy storage element I. (d) Causality for energy dissipation element R. (e) Causality for energy transformation elements: transformer T and gyrator G. (f) Causality for energy-conserving junctions: the 0-junction and the 1-junction.

Since iterative computation costs of IRFs appear to vary approximately as the square of the size of the iteration vector, the savings over several IRFs can be substantial. Basically, we have a cost of

$$c \sum_{1}^{N_f} m_i^2$$

rather than

$$c \left(\sum_{1}^{N_f} m_i \right)^2$$

by partitioning effectively, where N_f is the number of IRFs and m_i is the number of R ports in field i.

In a series of tests with practical R-fields, savings of up to 70% were realized. Clearly the nature of the problem at hand determines the typical savings to be expected.

A number of workers have addressed the issue of R-field computation either directly [12] or in terms of concepts derived from block diagram algebra or signal flow graphs [1, 13]. A related method based on dynamic augmentation of the R-field is discussed in [11]. This is a bond graph equivalent of introducing parasitic dynamics to interrupt an algebraic loop. Lorenz and Wolper [1] also suggest that a computing diagram derived from a causally oriented bond graph might be helpful in identifying a set of rules for finding an "optimal" solution.

The algorithm presented below substantially improves on prior results with respect to efficiency. It builds on several existing results, including use of causality data assigned by the SCAP and properties of junction structures. Because the algorithm operates on bond graph equations derived in field form, it is compatible with much of the current software for bond graph processing. Since it relies primarily on classification and sorting to achieve its increased efficiency, it is easy to implement. The algorithm can be used by hand to aid in the organization of R-field equations, although it is unlikely that anything practical can be done with relatively large, nonlinear implicit problems by hand.

A computational algorithm for a class of implicit R-fields

- *Definition of the problem*

The general problem can be stated as follows:

1. Identify the separate implicit R-fields.
2. For each implicit R-field,
   a. Find the minimum number of iteration variables;
   b. Find a suitable set of iteration variables that is minimum in number;
   c. Organize the equations in a form leading to efficient iterative solution.

Referring to the field structuring of systems equations of the previous section, the particular subset of equations with which we are concerned is

$$D_0 = \phi_L(D_i), \tag{4}$$

$$D_i = S_{21}Z + S_{22}D_0 + S_{23}U. \tag{5}$$

We seek an efficient solution to these equations at each time step, given values for the Z and U vectors. A more succinct form for the equations is

$$D_0 = \phi_L(D_i), \tag{6}$$

$$D_i = S_{22}D_0 + C, \tag{7}$$

where C is a constant vector. Clearly, it would be possible to iterate on the D_i vector to obtain a solution to the problem. It should be noted that formulating the problem in this manner implies that either all T and G nodes have constant moduli, or they are all evaluated at each time step and treated as constant over the integration interval. Under these conditions, their effects can be captured in the S_{ij} matrices.

We impose two restrictions on the R-field problem, which still leaves us with the most common practical subclass of the general problem:

1. All R nodes in implicit fields are 1-ports.
2. The bond graph does not contain T and G nodes in the implicit fields.

The resulting problem is practically important, since typical dynamic models of electromechanical and electrical systems fall within the subclass. Extension to the more general problem is an open research issue.

• *Some preliminaries*

The first level of organization of the system R-fields derives from applying the SCAP to the source (E, F) and storage (C, I) nodes. If causality is completely assigned by these steps, then no implicit R-fields exist. Assume that this is not the case; then some bonds are acausal. We now attend to the acausal part of the bond graph.

By using simple reach relation calculations on the subgraph composed of all nodes with at least one acausal incident bond, the separate implicit R-fields can be identified. Acausal fields that do not have at least one R node will not concern us further here: They are junction structure (JS) complexes. The bonds in the IRFs can be ordered by field, thereby grouping the equations into blocks.

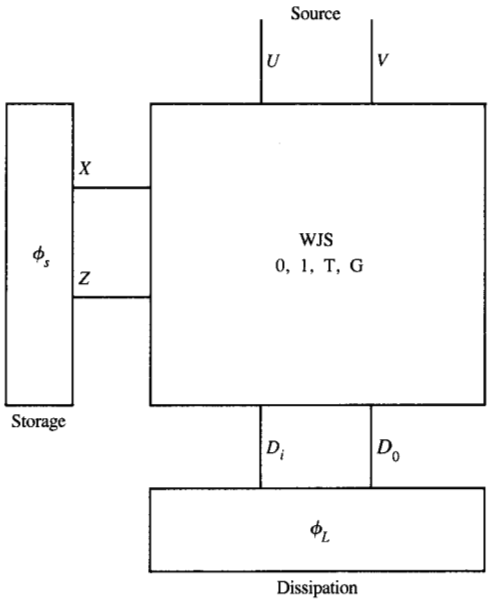


Figure 5

Key vectors in the field structure.

Now we arrive at the basic questions for a given IRF. Subject to the restrictions imposed earlier, each obeys local equations of the form of Equations (6) and (7). We ask, "What is the smallest number of iteration variables?" "How can such a set be found?" "How should equation set (6), (7) be used for best computing efficiency?"

• *Implicit R-field equation formulation*

The bonds of a given IRF can be sorted into one of three mutually exclusive sets:

1. The external bonds, connecting the IRF to the rest of the graph (these bonds are causal).
2. The bonds incident to R nodes, with which are associated the (local) D_i and D_0 vectors.
3. The remaining bonds, which are internal to the (local) junction structure (JS).

First we focus on the JS. Earlier work [14] has shown that there are two critical numbers associated with a JS. These indicate the number of effort (E) and flow (F) inputs required at the JS ports in order to determine all internal variables and the outputs. For completeness we state the rule here for calculating the numbers:

$$E = N_B + N_0 - N_1 - B_0, \tag{8}$$

$$F = N_B + N_1 - N_0 - B_1, \tag{9}$$

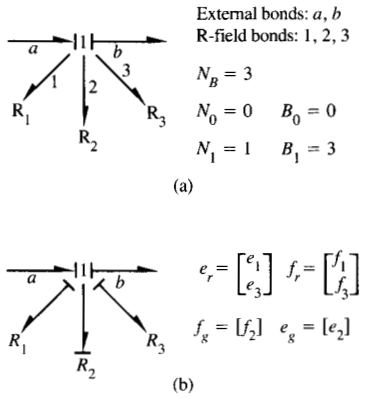


Figure 6

An example of an IRF. (a) Bond graph of the R-field and the values needed for the calculation of Equations (8) and (9). (b) The vectors from Equations (10) and (11).

where

N_B is the number of bonds of the JS,
N_0 is the number of 0-junctions,
N_1 is the number of 1-junctions,
B_0 is the number of bonds incident to the 0-junctions,
B_1 is the number of bonds incident to the 1-junctions.

Next, we observe that a JS composed only of 0- and 1-junctions has a pair of separate but related transformations associated with it. Namely, input efforts determine output efforts, and input flows determine output flows. Furthermore, if all powers at the JS ports are oriented out (or all in), the associated matrix is skew-symmetric, subject to proper ordering of the port variables [10, 15].

Now assume that causality assignment to the IRF has been completed and is consistent. The preceding observations allow us to organize Equations (6) and (7) in detail as follows.

Sort the D_i and D_0 vectors into a resistance set (r) and a conductance set (g). The r set has flow inputs to the R nodes and effort outputs; the g set has effort inputs to the R nodes and flow outputs. Write Equation (6) as

$$e_r = \phi_r(f_r), \tag{10}$$

$$f_g = \phi_g(e_g), \tag{11}$$

where e_r and f_r are associated with the r bond set, and f_g and e_g are associated with the g bond set. Write Equation (7) as

$$f_r = S_{22rr}e_r + S_{22rg}f_g + C_r, \tag{12}$$

$$e_g = S_{22gr}e_r + S_{22gg}f_g + C_g. \tag{13}$$

Since the (0, 1) JS transforms efforts to efforts and flows to flows, then S_{22rr} and S_{22gg} must be zero. Consequently, we have

$$f_r = S_{22rg}f_g + C_r, \tag{14}$$

$$e_g = S_{22gr}e_r + C_g. \tag{15}$$

Furthermore, we note that the combined set (f_r, e_g) contains the JS outputs, while the combined set (f_g, e_r) contains the JS inputs. We are now prepared to state the computational algorithm.

- *Algorithm*

A computational algorithm for solving implicit R-fields by local iteration methods is stated below:

1. Assign causality to the source and storage nodes, using the SCAP.
2. Identify each implicit R-field within the (partially) causal bond graph.
3. For each implicit R-field:
   a. Calculate E and F.
      - If either E or F is less than one, stop. (There is no guarantee that there are unique outputs from the inputs for this JS.)
   b. Obtain a complete, consistent causal orientation for the IRF. (It will obey the E, F numbers.)
   c. Order the R bonds by resistance (r), then conductance (g) causality. Define the vectors e_r, f_r, f_g, and e_g.
   d. Assume that E is less than or equal to F. Use f_r as the iteration vector. Make an initial guess f_{ri} for f_r.
      - Use Equation (10) to find e_r.
      - Use Equation (15) to find e_g.
      - Use Equation (11) to find f_g.
      - Use Equation (14) to find f_r.

      Compare f_{ri} to f_r. If the error is within tolerance, stop. Else return to Equation (10) and repeat sequence with the new guess for f_r.
      - Note: If E is greater than F, use e_g as the iteration vector. The equation order is then (11), (14), (10), (15).

- *Observations about the algorithm*

We observe that the minimum iteration set that we seek has the size min(E, F). This is always less than or equal to one half the number of bonds on the IRF R nodes. Reducing the dimension of the iteration vector has a major positive influence on computing efficiency, as noted previously.

The restrictions placed on the problem structure can be relaxed to a certain extent without changing the algorithm as stated above. A given IRF can contain R nodes with more than 1 port, provided each such R node is a pure r, or a pure g, type. See Equations (10) and (11) in this regard. In addition, the R-field JS can contain T nodes, since they do not alter the structure of the effort-to-effort, flow-to-flow

transformation properties. See Equations (12) and (13). As mentioned previously, if the T nodes are modulated, we assume them to be constant over a single integration interval. Their effects are combined into S_{22}.

Some examples

An example of an implicit R-field with three 1-port R nodes is given in **Figure 6(a)**. The inputs to the IRF are e_a and e_b. The goal is to calculate all R-field variables. We first find E and F from the data given:

$$E = 3 + 0 - 1 - 0 = 2,$$

$$F = 3 + 1 - 0 - 3 = 1.$$

A solution does exist, since both E and F are greater than zero. We obtain a complete, consistent causal orientation, as shown in **Figure 6(b)**. The ordered r bond vectors are defined, based on the causality, as indicated in the figure. The equations can be written as

$$e_1 = \phi_1(f_1), \tag{16}$$

$$e_3 = \phi_3(f_3), \tag{17}$$

$$f_2 = \phi_2(e_2), \tag{18}$$

$$f_1 = f_2, \tag{19}$$

$$f_3 = f_2, \tag{20}$$

$$e_2 = -e_1 - e_3 + (e_a - e_b). \tag{21}$$

A suitable iteration vector is e_2, since E is greater than F. The iterative solution pattern is (18), (19), (20), (16), (17), (21).

Another example is shown in **Figure 7(a)**. On the basis of the R-field structure, we get

$$E = 6 + 1 - 3 - 3 = 1,$$

$$F = 6 + 3 - 1 - 6 = 2.$$

A satisfactory causal orientation is shown in **Figure 7(b)**. The ordered r bond variables are indicated:

$$e_1 = \phi_1(f_1), \tag{22}$$

$$f_2 = \phi_2(e_2), \tag{23}$$

$$f_3 = \phi_3(e_3), \tag{24}$$

$$f_1 = f_2 + f_3, \tag{25}$$

$$e_2 = -e_1 + (e_a + e_b), \tag{26}$$

$$e_3 = -e_1 + (e_a + e_c). \tag{27}$$

Since E is less than F, use f_1 as the iteration variable. The equation sequence is (22), (26), (27), (23), (24), (25).

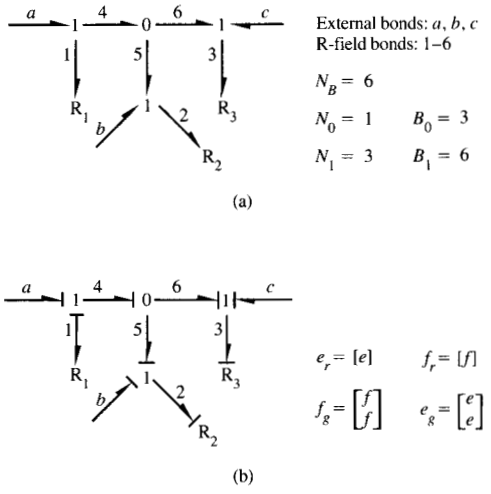


Figure 7

Another example of an IRF. (a) Bond graph of the R-field and the values needed for the calculation of Equations (8) and (9). (b) The vectors from Equations (10) and (11).

Summary

In this paper we have introduced the bond graph representation for physical (i.e., energy-based) systems and have illustrated its application to an electromechanical device, a relay. We have stated a major objective of completely automating the generation and solution of system equations, given a bond graph model. A major problem was discussed, namely the calculation of implicit R-fields, which generally leads to coupled nonlinear algebraic equations. An algorithm was presented to treat a major subclass of the general problem. Its implementation will lead to increased efficiency in the iterative solution of implicit equations.

At least three questions remain to be answered:

1. Can the given algorithm be extended to treat the more general R-field problem? More specifically, can the restriction on multiport R nodes within an implicit R-field having mixed causality be relaxed? And can gyrators be included in the field junction structure?
2. Within the framework of the given algorithm, what is the numerically most robust set of iteration variables to use? No attention has been given to that important issue in this paper.
3. Should systems with modulated transformer and gyrator nodes be treated differently from current practice?

We look forward to continued progress in obtaining maximum efficiency from bond graph computing methods, as a step toward the goal of completely automated solution.

References

1. F. Lorenz and J. Wolper, "Assigning Causality in the Case of Algebraic Loops," *J. Franklin Inst.* **319**, No. 1/2, 237–241 (1985).
2. H. M. Paynter, *Analysis and Design of Engineering Systems*, MIT Press, Cambridge, MA, 1961.
3. D. C. Karnopp and R. C. Rosenberg, *Analysis and Simulation of Multiport Systems*, MIT Press, Cambridge, MA, 1968.
4. A. M. Box and P. C. Breedveld, "1985 Update of the Bond Graph Bibliography," *J. Franklin Inst.* **319**, No. 1/2, 269–286 (1985).
5. *The ENPORT-6 User's Manual*, Rosencode Associates, Inc., Lansing, MI, 1986.
6. R. Prakash, E. Palmer, and D. Withers, "Design Assessment: Electro-Mechanical Systems," *Research Report RC-10993*, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 1985.
7. Jose J. Granda, "Computer Aided Design of Dynamic Systems," *Proceedings of the 1984 Summer Simulation Conference*, Society for Computer Simulation, La Jolla, CA, pp. 1189–1194.
8. J. J. Van Dixhoorn, "Simulation of Bond Graphs on Minicomputers," *J. Dynam. Syst., Meas. & Control* **99**, 9–14 (1977).
9. R. C. Rosenberg and D. C. Karnopp, *Introduction to Physical System Dynamics*, McGraw-Hill Book Co., Inc., New York, 1983.
10. R. C. Rosenberg, "State-Space Formulation for Bond Graph Models of Multiport Systems," *Trans. ASME* **93**, No. 1, 35–40 (March 1971).
11. T. Zhou, "A Parallel Computation Method for Dynamic Systems with Coupled Nonlinear Dissipation," M.S. thesis, Department of Mechanical Engineering, Michigan State University, E. Lansing, MI, 1984.
12. J. Barreto and J. Lefevre, "R-fields in the Solution of Implicit Equations," *J. Franklin Inst.* **319**, No. 1/2, 227–236 (1985).
13. F. T. Brown, "Direct Application of the Loop Rule to Bond Graphs," *Trans. ASME, Series G* **94**, No. 3, 253–261 (1972).
14. R. C. Rosenberg and B. Moultrie, "Basis Order for Bond Graph Junction Structures," *IEEE Trans. Circuits & Syst.* **CAS-27**, No. 10, 909–920 (October 1980).
15. D. C. Karnopp, "Power-Conserving Transformations: Physical Interpretations and Applications Using Bond Graphs," *J. Franklin Inst.* **288**, No. 3, 175–201 (1969).

Received October 6, 1986; accepted for publication December 11, 1986

Sarah Jean Hood *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Dr. Hood received a Ph.D. in mechanical engineering from the University of California at Davis in 1985. Previously, she had received a B.S. in environmental engineering from the University of Florida, Gainesville, and an M.S. in mechanical engineering from the University of California at Davis. Since 1986, she has been a Research Staff Member in the systems and application design group in the Manufacturing Research Department. Her research involves differential equation modeling of electromechanical systems.

Ronald C. Rosenberg *Department of Mechanical Engineering, Michigan State University, East Lansing, Michigan 48824.* Professor Rosenberg has been a member of the Mechanical Engineering Department since 1974. He received an S.M., S.B. from the Massachusetts Institute of Technology, Cambridge, in 1960; a D.I.C. from Imperial College, London, England, in 1962; and a Ph.D. from the Massachusetts Institute of Technology in 1965. He is currently a consultant for General Motors, IBM, and Alcoa, as well as a reviewer for the National Science Foundation and several technical journals/publishers. Among his current research projects are "Improvement of Design Assessment Techniques for Reliability and Yield by Bond Graph Technology" in collaboration with the IBM Thomas J. Watson Research Center, a study of the development of improved methods for assessing device performance and application to electromechanical devices such as disk drives; and "Vehicle Power System Analysis: A Unified Approach," with General Motors Research Laboratories, a study of the development of bond graph models of automotive vehicle power flows, and implementation of software to simulate such models. Professor Rosenberg is a member of the American Society for Engineering Education, the American Society of Mechanical Engineers, and the Institute of Electrical and Electronics Engineers.

David H. Withers *IBM Information Systems, P.O. Box 2150, Atlanta, Georgia 30055.* Mr. Withers is a senior planner with the IBM Information Systems Group at the Atlanta Software Development Laboratory. He began his current assignment in 1987 after managing the product and process analysis group at the IBM Thomas J. Watson Research Center. Mr. Withers received a B.S. in engineering from the U.S. Coast Guard Academy, New London, Connecticut, and M.S. degrees in mathematics and computer science from Rensselaer Polytechnic Institute, Troy, New York. His work has emphasized the development of mathematical models of physical systems, especially those relating to product reliability and service delivery. He joined IBM in 1969 and has had both technical and management assignments at Burlington, Vermont; Lexington, Kentucky; Franklin Lakes, New Jersey; and Yorktown, New York. He has received an IBM Outstanding Contribution Award for mathematical simulation of photolithographic defects for yield prediction and an IBM Information Products Division Award for element exchange evaluation for typewriters. His current interests are requirements specifications for computer-integrated manufacturing systems. He is a member of the National Research Council Study Committee on Enabling Technologies for Unified Life Cycle Engineering of Structural Components, the Association for Computing Machinery, the ACM Special Interest Group on Numerical Mathematics, the Operations Research Society of America, the Institute of Management Sciences, and the TIMS College on Simulation and Gaming.

Tong Zhou *Department of Mechanical Engineering, Michigan State University, East Lansing, Michigan 48824.* Mr. Zhou is currently a graduate research assistant working on his Ph.D. in mechanical engineering. His initial studies were at Zhegiang University in China, where he studied mechanical engineering from 1962 to 1968. In 1984, he received an M.S. in mechanical engineering from Michigan State University. Mr. Zhou began his professional career in 1968 as a manufacturing engineer. In 1975 he became a project design engineer engaged in the research and design of coal mining machinery and other machine tools. Mr. Zhou published *Nomographs on Machine Design* in Peking, China, in 1983.