Automated Bond Graph Modeling and Simplification to Support Design

by

James R. Rinderle, L. Balasubramaniam

EDRC 24-54-91

J.R. Rinderlo and L. Balasubramaniam, "Automated Bond Graph Modeling and **Simplification to Support Design,"** ***Journal of Dynamic Systems Measurement and Control,*** (in review), 1990. EDRC-JAA?*91v^^

Automated Bond Graph Modeling and Simplification to Support Design

James R. Rinderle
Associate Professor

L. Balasubramaniam,
Graduate Research Assistant

Department of Mechanical Engineering
Engineering Design Research Center
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

Designs evolve from a preliminary concept by a process of iterative evaluation and refinement. Every cycle of this process necessarily consists of assessing the extent to which functional specifications have been satisfied. It is hypothesized that providing a tool for automatically modeling and analyzing devices and relating individual component characteristics to device behavior would aid the conceptual designer by facilitating the consideration of more varied alternatives. In this paper we discuss issues pertaining to automated modeling to support design including issues of modeling relevance and model simplification. We propose a framework for automatic modeling based on the modular aggregation of bond graph fragments corresponding to components and connections. Although a component based modeling paradigm is convenient for the designer, it leads to bond graph models which are difficult to analyze and difficult to comprehend due to the presence of redundant, irrelevant and circuitous structure and due to a high degree of dependency among energy storage elements. Bond graph simplification methods have been developed that mitigate these problems. These methods identify and eliminate inert elements and redundant and constraining junction structures and replace dependent energy storage elements with functionally equivalent modifications to the remaining storage elements. The application of these methods makes it possible to apply standard equation formulation techniques and results in simple bond graph models which can more easily be interpreted by the designer.

Submitted for publication in the "Journal of Dynamic Systems, Measurement and Control"

Automated Bond Graph Modeling and Simplification to Support Design

James R. Rinderle
Associate Professor

L. Balasubramaniam,
Graduate Research Assistant

Department of Mechanical Engineering
Engineering Design Research Center
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

Designs evolve from a preliminary concept by a process of iterative evaluation and refinement. Every cycle of this process necessarily consists of assessing the extent to which functional specifications have been satisfied. It is hypothesized that providing a tool for automatically modeling and analyzing devices and relating individual component characteristics to device behavior would aid the conceptual designer by facilitating the consideration of more varied alternatives. In this paper we discuss issues pertaining to automated modeling to support design including issues of modeling relevance and model simplification. We propose a framework for automatic modeling based on the modular aggregation of bond graph fragments corresponding to components and connections. Although a component based modeling paradigm is convenient for the designer, it leads to bond graph models which are difficult to analyze and difficult to comprehend due to the presence of redundant, irrelevant and circuitous structure and due to a high degree of dependency among energy storage elements. Bond graph simplification methods have been developed that mitigate these problems. These methods identify and eliminate inert elements and redundant and constraining junction structures and replace dependent energy storage elements with functionally equivalent modifications to the remaining storage elements. The application of these methods makes it possible to apply standard equation formulation techniques and results in simple bond graph models which can more easily be interpreted by the designer.

Introduction

There has been increasing interest, in recent years, in developing computer tools that can **aid** designers in evaluating and comparing alternative solutions at the preliminary design stage. Analysis at this stage is primarily motivated by the need to determine behavioral trends rather than specific numerical values. The designer-analyst would, for instance, be interested in the relative importance of bicycle wheel weight and frame weight rather than the specific value of effective mass. Computer tools which formulate mathematical models from physical descriptions of systems free the designer from modeling details. This class of tools have been referred to as "self-formulating" in the literature. [Haug 89, Mclnnis 89, Paul 70, Sheth 72, Orlandea 77] discuss different methods to automatically analyze mechanical systems. [Haug 89], for example, describes the Dynamic Analysis and Design System which represents mechanical devices internally by a set of constrained second order differential equations. Each massive element is assigned three co-ordinates: X, Y and 6 to account for planar motion. Kinematic connections are expressed as algebraic constraints among the body co-ordinates. The "planar" bicycle of Figure 1 would be modeled in this system as an aggregate of four massive objects, the two wheels, the frame and the sprocket, joined by several kinematic connections. The mathematical formulation in DADS results in a set of twenty-three second-order coupled differential-algebraic equations. This is a convenient enough model for simulation but its size and complexity make it hard for the designer to appreciate the design trade-offs. Because analysis at the preliminary stages of design is invariably followed by design refinement, techniques which establish a close correspondence between behavior and individual component characteristics are very useful to the designer. In this paper we present some techniques that allow building and analyzing symbolic models of dynamic behavior given a description of the artifact in terms of its components and kinematic connections among them.

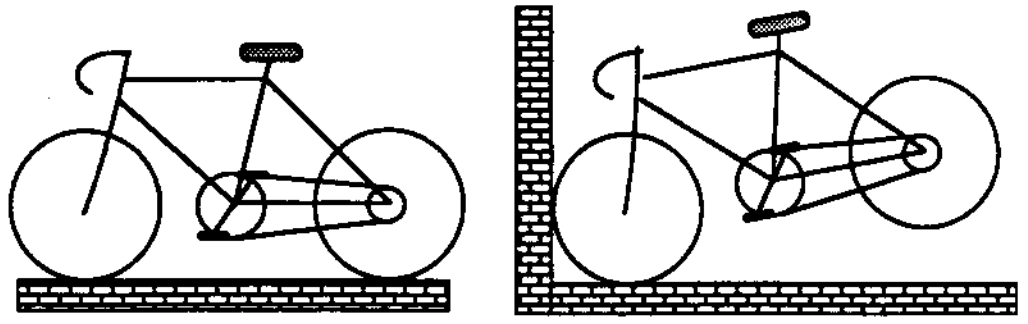


Figure 1: A bicycle with two different boundary conditions.

An Environment for Automated Modeling of Dynamic Systems

[Paz-Soldan 88, Balasubramaniam 91] discuss general requirements for an effective design-oriented modeling system. The essence of modeling of any sort is to include only the effects relevant to the question posed. Typically the designer or modeling expert determines on an ad hoc basis which characteristics must be included to conrectly predict device behavior Engineering judgment is required in choosing an appropriate model for any given context and the design tool should support the use of such skills. While modeling a bicycle as shown in Figure 1 for example, the bicycle frame can be considered as a rigid massive body or as a compliant massive structure. Modeling insight allows the engineer to concentrate on the relevant aspects of the problem, while ignoring dimensions where nothing interesting is happening. Making such simplifying assumptions involves in part reasoning about and deleting parts of the model which either correspond to constrained degrees of freedom or degrees of freedom which are not excited. In the case of the bicycle, for example, the rotational inertia of the frame may or may not be relevant depending on the boundary conditions on the problem as illustrated in Figure 1. Proficient modellers routinely delete irrelevant characteristics of components to simplify analysis.

Approach

Natural interface and modeling flexibility requirements of a design-oriented modeling system as discussed by [Paz-Soldan 88, Balasubramaniam 91] can be met through a novel use of bond graphs, a formal graph based representation used for physical system modeling [Paynter 61, Rosenberg 75]. A modular fragment of a bond graph is associated with each component and with each type of kinematic connection. The designer builds up devices by selecting components and specifying the kinematic connections among components. As the components are connected the bond graph fragments corresponding to the components are assembled into a device model. Bond graph theory provides a consistent basis for this process of aggregation, so that the kinematic constraints specified by the designer, are sufficient information for assembling the component bond graph fragments. This modular component based approach to constructing a device model is very convenient for a designer. By contrast the standard method of constructing a bond graph model as explained in [Rosenberg 83] is unsuitable because the designer has to be familiar with bond graphs and their construction rules. In addition the kinematic degrees of freedom of the device must be determined prior to the formulation of the dynamic model. Mclnnis and Elmaraghy [Mclnnis 89] propose a systematic method to automatically construct bond graph models which follows the direct formulation procedure in [Rosenberg 83]. While the user does not have to be familiar with bond graph construction procedures, the kinematic problem still has to be solved a priori. This places an additional burden on the designer but leads to fairly straight-forward models which can be analyzed

easily. By contrast the modularity of our approach, which allows the designer to remain oblivious to the modeling details, results in circuitous bond graphs that have to be simplified before analysis can proceed further. Fortunately this can be done algorithmically by using the methods explained in subsequent sections.

Several other researchers have used also bond graphs in design related research [Rosenberg 75, Finger 89, Ulrich 89, Hoover 89, Macfarlane 89, Prabhu 89, Hood 87]. While [Finger 89, Ulrich 89, Hoover 89, Prabhu 89] address issues of design synthesis and use bond graphs as the representational framework for their synthesis strategies, [Macfarlane 89, Hood 87] use it as a tool for analysis. [Bos 85] gives a bibliography of bond graph literature through 1985.

We summarize the issues that need to be addressed in implementing a design oriented analysis tool as follows:

- Develop models for the set of primitive components and kinematic connections.
- Develop simplification rules and transformations so that dominant behavior is clearly brought out and equations can be readily formulated in an explicit form.

Bond Graph Representation

The number of power connections that can be made to a primitive bond graph element is not the same as can be made to the physical component which it represents. For example, a conventional spring has two connections, however, the bond graph compliance element, C, which represents a spring is a one port element. A natural interface demands that the user continue to think of a spring as a two-port device. To overcome this problem we use bond graph fragments with topology similar to the physical components which they model. Figure 2 shows a "two-port-spring[11] model. The -0- is a power conserving, common force multi-port element. As such, the bond graph fragment shown in Figure 2 requires that the same force acts on both ends of the spring and that the spring velocity is actually the difference of the velocities at the two ends. This simple example illustrates the possibility as well as the utility of creating such models for more complicated components, for instance, an engine valve spring with significant inertia. Figure 3 shows the bond graph *mass* model for planar motion. The model has three external *connection sockets.* Each socket corresponds to a distinct location and includes three bond-graph ports to accommodate rotation as well as independent X and Y velocities. The forces and moments at these ports excite the rotational and translational energy storage modes of the mass. The degree to which a force at one of the ports causes rotation or translation depends on the location of interaction. The -TF- elements in the model account for this by transforming a force and moment at an arbitrary position to equivalent forces and moments about the center of mass. The moduli of the transformer (TF) elements depends only on the location of force/moment application relative to the center of mass. A mass may of course interact with other

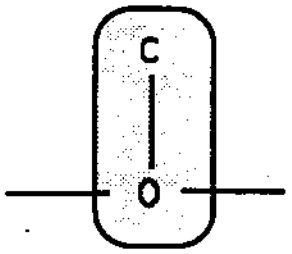


Figure 2: Internal model of a spring.

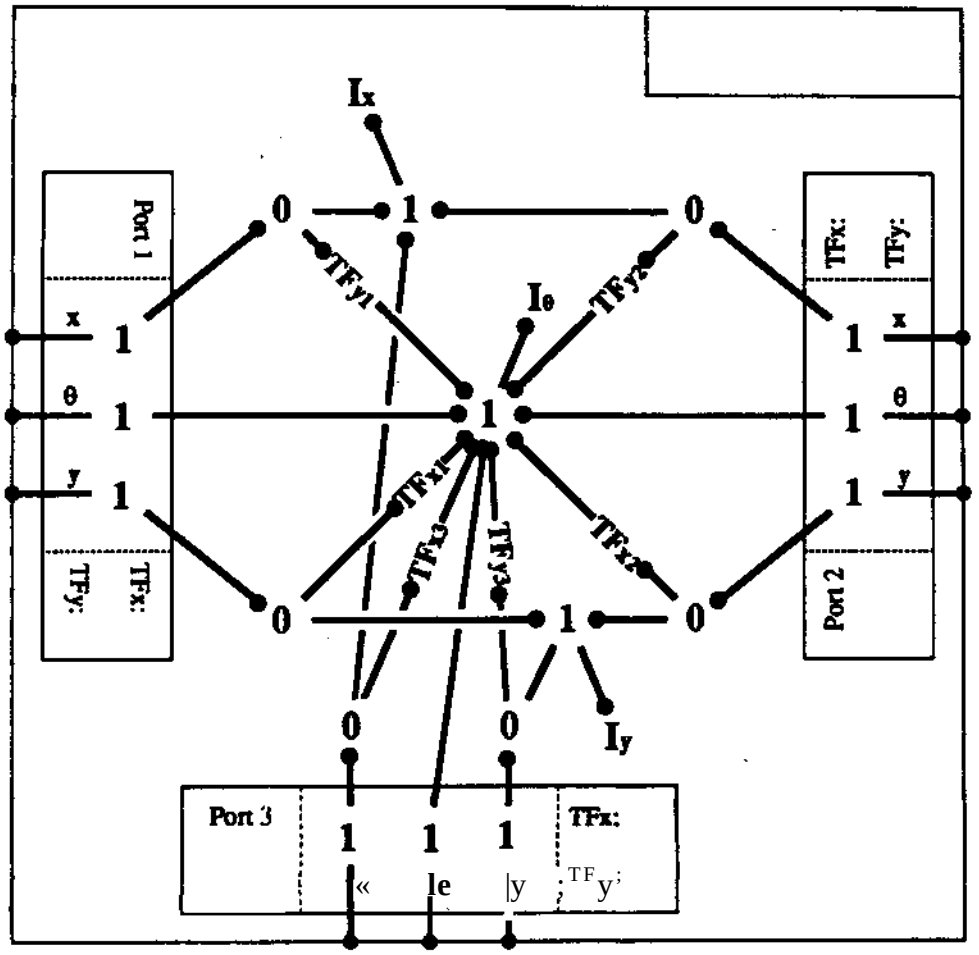


Figure 3: Bond graph model of a massive body.

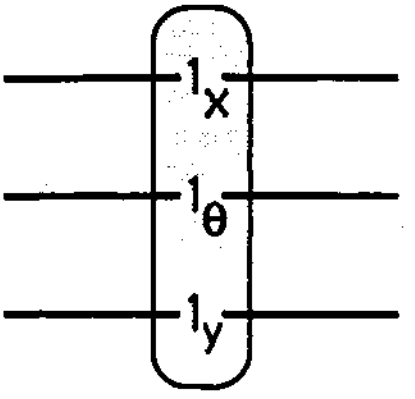


Figure 4: Bond graph model of a rigid connector.

components at an arbitrary number of points. We define a composite mass to accommodate this arbitrary degree of connectivity by establishing rigid connections among many mass elements and thereby preserve the natural designer interface. A consistent power-sense assignment, conventionally indicated by a "power half arrow[11] is shown by a hatched circle in the figure for graphical convenience.

Kinematic connections are also modeled as bond graph fragments. The model of a connector has a -1-, or common velocity, junction corresponding to every velocity, X, Y or rotational, that it constrains. A pinned connection includes two -1- junctions corresponding to common X and Y velocities. A rigid connection, as in Figure 4, includes an additional -1- junction because rotation of two rigidly connected masses is also identical.

Need for Simplification

Even a simple component such as a mass can exhibit a variety of different behaviors depending on the kinematic constraints imposed on it. One way to create models appropriate to the context is to have simple ad-hoc models for each class of behavior **and** then let the designer decide which physical effects are relevant and significant enough to be modeled. The designer then has to be aware of the implications of choosing a particular behavioral model for a component. Making a design modification in some component or altering the boundary conditions may require that models of other components in the device be changed. For instance, modeling the rotational inertia of the bicycle frame is not required when the bicycle-wheels remain on horizontal ground Qeft-hand side of Figure 1) but becomes necessary when this constraint is relaxed as in the right-hand side of

Figure 1. Alternatively very general component models which incorporate characteristics relevant to a broad range of behavior can be employed. Complex, general models of this sort free the designer from modeling details but result in overly complex device models that obfuscate basic characteristics and therefore hinder simple reasoning about device behavior. If general, complex models are to be used it is necessary that systematic methods are available to remove superfluous modeling elements or analysis results.

The advantages [Balasubramaniam 91] of the latter option in a design-oriented system justify the computational expense of simplifying the first-cut models produced. The models may contain constrained elements which will not be excited at all for the configuration under study. They may also contain a number of energy storage elements, masses and springs, connected so that they cannot all be independently assigned energy variables. The standard techniques of equation formulation from bond graphs are computationally difficult if dependent elements are present and fail completely for certain classes of bond graphs.[1] Because the first-cut graphs we produce often contain such intractable elements it is necessary to simplify the graph before we formulate the equations of motion.

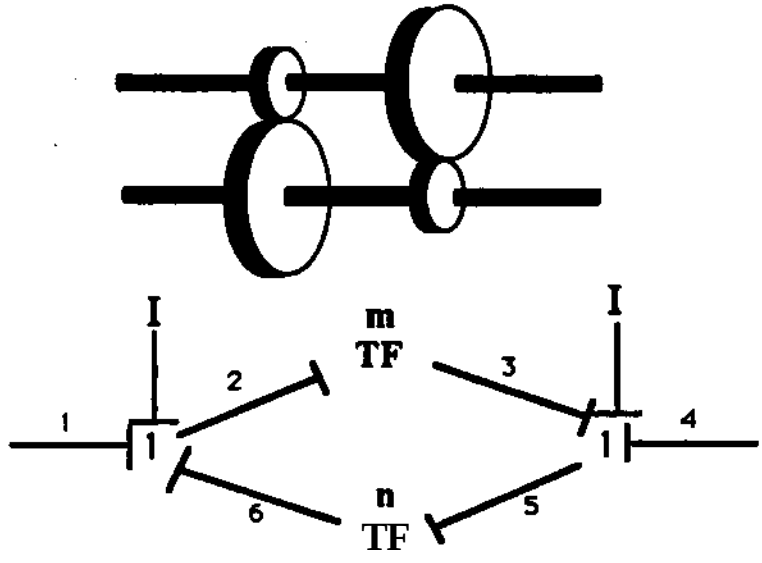
Techniques for Bond Graph Simplification

We have identified two classes of simplification procedures, both of which arise from studying the causal assignment in a bond graph.

Simplification: Inert bond graph segments

The first step of simplification is to remove bonds and junction structure elements which have not been connected. Because each of the ports on the model represents a point of mechanical interaction, the absence of a connection indicates that no force is applied at that point in the associated direction. An open port therefore behaves exactly as if a source of effort of zero magnitude was applied. The effects of these effort sources are propagated through the model. A zero effort source incident on a -0- junction has the effect of imposing a zero effort on all of the other power bonds emanating from that junction. The -0- and the original effort source can therefore be replaced by zero effort sources on the remaining bonds. The effects of these new zero effort sources are then propagated. An effort source of zero magnitude incident upon a -1- junction has no effect, because efforts at a -1- junction must sum to zero and because the variable associated with a -1- junction is a velocity and therefore does not affect any aspects of causality assignment. We can therefore simply delete zero effort sources incident on -1- junctions. Dual reasoning allows us to propagate and delete zero flow sources, *i.e.* ground connections. We also delete two-port -1- junctions and two-port -0- junctions (being careful to respect sign convention).

[1]**This class includes bond graphs, with constraining junction structures which we discuss later.**



$f_4 = \langle h, U \rangle$

$f_1 = \{ f_2, f_3, f_5, f_6 \}$

$$A_2 = \begin{bmatrix} 0 & m & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & n \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$f_1 = f_2$
$f_4 = f_5$
$f_2 = f_3\, m$
$f_3 = f_5$
$f_5 = f_6\, n$
$f_6 = f_2$

$$\text{Det}\,[\, I - A_2 \,] = 1 - m\,n$$

Figure 5: Two shafts locked by meshing gears.

Inert elements also arise when kinematic constraints on a multi-degree of freedom body interact to impose zero velocity or effort on elements which are not explicitly constrained. This would arise, for instance, if two gears keyed to a shaft are meshed with two other gears which are keyed to another shaft as illustrated in Figure 5. Neither shaft can rotate unless the two gear ratios are identical. The way the gears are coupled imposes zero velocity on them. In bond graph terms the connected set of -0-, -1- and TF junctions, henceforth referred to as a junction structure, constrains the attached elements. We seek to identify and delete bond graph elements and junction structures that represent such static parts.

The constitutive equations describing a junction structure consist of a set of effort equations and flow equations. In general the flow equations have the form:

$$\begin{bmatrix} \\ \\ 'I \end{bmatrix} = \begin{bmatrix} Ai & A_3 \\ A_2 & A_4 \end{bmatrix} \begin{bmatrix} 'I \\ f. \end{bmatrix} \qquad (1)$$

where $[f_+]$ is the column of flows (velocities) imposed by the junction structure on the elements connected to it, [fJ is the column of flows imposed on the junction structure, [f|] is the column of internal flows and Aj, A_2, A_3, A_4 arc matrices of constants. Equation 1 expresses flow variables in terms of *other* variables more fundamental in a causal sense and hence the terms on the diagonal of the partition A_2 arc all zero. Now, let there exist a junction structure in a bond graph with admissible causality such that all external bonds impose effort on the junction structure.[2] In this specific case, [fJ is a null column because none of the external bonds impose flow. Then Equation 1 for this junction structure must take the form:

$$\begin{bmatrix} f_+ \\ f_I \end{bmatrix} = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} [f_I] \qquad (2)$$

From Equation 2, it is possible to identify conditions for a junction structure to impose zero velocity on all elements connected to it These junction structures are not really n-ports because the behavior of internal variables is independent of what is externally imposed and some effort/flow variables cannot be expressed in terms of state variables. Such a junction structure is illustrated in Figure 5 which shows the bond graph and corresponding flow equations of two shafts locked by two pairs of meshed gears.

A junction structure cannot cause any element to have zero velocity if that element is imposing flow causality on the junction structure because this violates the definition of causality. Hence if the junction structure is to impose zero velocity on all attached elements then it is necessary that the elements impose only efforts on the junction structure. From the flow equation, which was derived for the above situation we have:

---

[2] [Karnopp 75a] states that such a junction structure cannot exist While that argument is valid if bond graphs are created using the standard rules prescribed in [Rosenberg 83], such junction structures can arise when bond graphs are assembled from sub-system models. Our contention is that a junction structure such as this represents kinematic constraints which although self-consistent within the bond graph frame-work represent either redundant constraints admitting motion or conflicting constraints precluding motion.

$$[f_1] = [A_2]\ [f,] \quad (3)$$

and hence $[\ I - A_j H f \wedge O \quad (4)$

If Det[I - A_2] is equal to zero then fj is arbitrary and f_+ is not necessarily equal to zero. If Det[I - A_2] is not equal to zero then [fj] = 0 and hence from the flow equation [f_+] = 0. From these arguments we conclude that the necessary and sufficient condition for a junction structure to impose zero velocity on all elements connected to it is that all external bonds impose effort on the junction structure and Det[I - A_2] * 0. For the example shown in Figure 5, we can sec that the determinant is zero only if the gear ratios are exactly the same for the two pairs. In all other cases the two shafts are completely locked.

To determine if a junction structure is constraining, we first check that valid causality can be assigned to the junction structure. We then strip away all the elements and check that efforts are imposed on all the junction structure ports. This determines if the junction structure could stall other elements attached to it. If flows are imposed on one or more of the ports, then these ports are deleted and the reduced junction structure is examined to determine if it is constraining. This iterative procedure will isolate a constraining junction structure if one exists. The final step is to evaluate Det[I - A_2] to check if the junction structure is redundant or constraining.

If on the other hand, the bond graph under consideration does not have such a structure then all the ports will be deleted. When there are no ports remaining we can conclude that a constraining junction structure did not exist in the graph.

Bond graph junction structures with causal or power loops and conditions under which they can be solved have been discussed in [Perelson 75, Rosenberg 79, Karnopp 75b, Ort 73]. All causal loops whose external bonds impose effort (or flow as the case may be) are constraining junction-structures. The conditions for a junction structure to be constraining can be alternatively derived from the theorems in [Rosenberg 79]. However, in the context of the modeling methodology presented here, these structures are not pathological. Rather, they model device behaviors, such as kinematic redundancy which are of interest to the designer.

Simplification: Dependent energy storage elements

The second issue, removing dependent energy storage elements, can be resolved because bond graph representation allows us to identify two primary classes of behavior preserving graph transformations to replace connected inertias and springs by their equivalences, and thus obtain a minimal representation of the system.

The first set of these transformations are well known in circuit theory: Energy storage elements connected in parallel or series are replaced by their equivalents, after moving them across transformers if necessary. Similarly elements connected

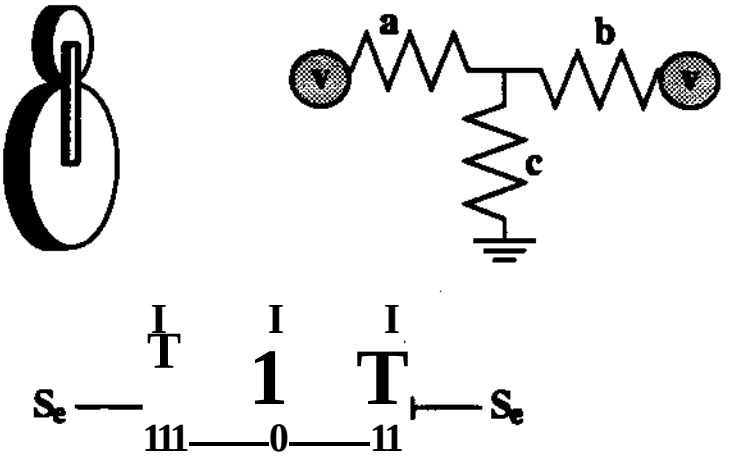


Figure 6: Mechanical, electrical and bond graph star.

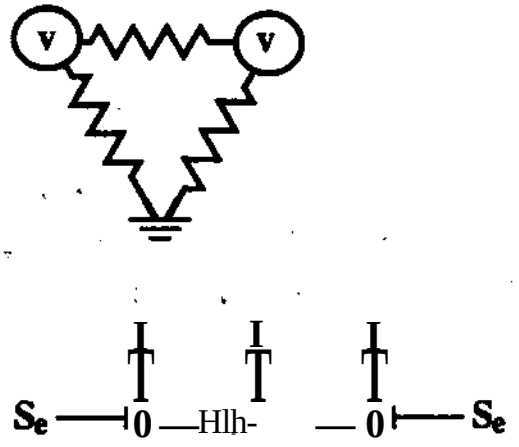


Figure 7: Equivalent delta.

in a star arrangement can be transformed into delta formation and vice-versa. An I clement is not independent *Le.* has derivative causality, when current or velocity is imposed on it. In a general star network of Fs, current or velocity will be imposed on one I. Figure 6 shows a mechanical, electrical and bond-graph version of the star connection. In the equivalent delta form, as in Figure 7, all of the I elements have integral causality.[3]

[3]**The independence of the state-variables in the delta form is, however, illusionaiy as we illustrate later.**

The second set of transformations come from the energy expressions. In a number of cases of derivative causality, the energy contribution from a dependent element can be expressed as a linear combination of the energy contributions from more than one other energy storage element In this case the dependent inertia cannot simply be added to a single inertia as with a parallel connection but must be distributed among several of them. If the parametric values of the independent elements are suitably changed and the dependent element removed, the energy expressions will not change. Since graph simplifications that do not change the expressions for kinetic and potential energy leave the dynamic equations unaltered, this method can also be used to eliminate derivative causality.

Consider a system with *N* inertias. Among the *N* let there be *M* inertias having derivative causality. Every inertia has a velocity associated with it. The velocities of the *M* dependent inertias can all be expressed as a linear combination of the *N-M* velocities of the independent inertias. Each of the *N* inertias results in a contribution to the Lagrangian proportional to the square of its associated velocity and of the form of a sum of quadratic expressions of the form v_i^2, v_j^2, v_t-$v_;$ where v_p *Vj e* (The set of velocities associated with the *N-M* independent inertias). If all of the inertias result in *P* such linearly independent terms, the coefficient matrix of the contributions from the *N* inertias can be written as a *(NxP)* rectangular matrix such that:

$$M \quad s \quad f \quad s \qquad \frac{(N-M)(N-M+1)}{2} \qquad (5)$$

The lower bound in Equation 5 arises because each of the *(N-M)* independent inertias contributes one and only one term to *P.* The upper bound is the number of possibly different quadratic combinations of *(N—M)* velocities.

We define a basis set as some set of *Q* inertias such that the rank of the *(QxP)* co-efficient matrix of the contributions from the *Q* basis inertias is the same as the rank of the original *(NxP* matrix. Such a basis will include all of the *P* linearly independent terms in the kinetic energy portion of the Lagrangian. If such a basis set can be identified then the effect of the remaining *(N-Q)* inertias can be distributed among the inertias comprising the basis set

When $N>Q_9$ then Gauss elimination followed by row interchanges will reduce the *(NxP)* rectangular matrix to a *(QxP)* matrix and *(N-Q)* rows with zeros in them. The values of the equivalent inertias for the system can be determined directly from the rows of the *(QxP)* portion of the transformed inertia coefficient matrix. Thus *(N-Q)* of the *M* derivative inertias can be eliminated.

Figure 8 shows a rack and pinion arrangement with the corresponding bond graph. The kinetic energy expression in the Lagrangian can be shown to be:

$$\frac{1}{2}\left[I_a V_a^2 + I_b V_b^2 + \frac{I_w (V_a - V_b)^2}{4r^2} + \frac{I_h (V_a + V_b)^2}{4}\right] \qquad (6)$$

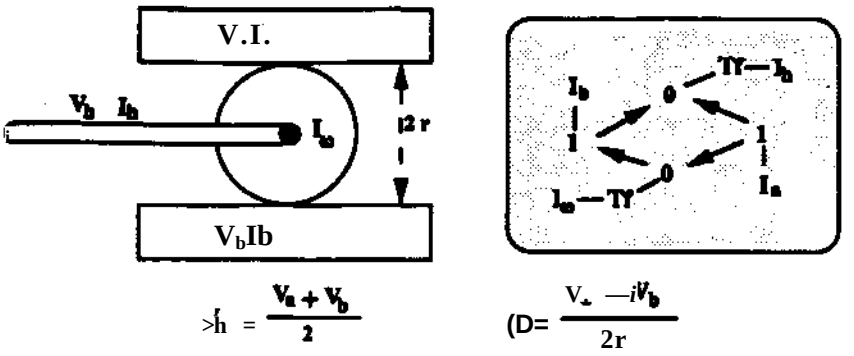


Figure 8: Rack and pinion arrangement.

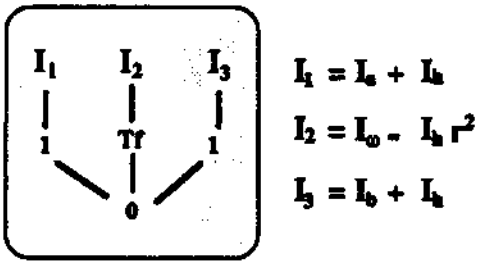


Figure 9: Simplified bond graph of a rack and pinion.

If we replace l_h by a source of zero effort in the bond graph and add the mass of I_h to I_a and l_b and subtract /^r^2 from I_w then expression 6 will not change. The resultant graph is shown in Figure 9. It can be shown that state-equations derived from the bond graphs in Figure 8 and Figure 9 are identical.

This class of transformations is especially useful because the energy expressions give a global view of the possible simplifications and thus avoid graph walking. In fact, the parallel-series simplifications are pre-compiled special instances of this method. **The** expression for kinetic energy also remains invariant in the star-delta transformation. Although the delta form has all integral causality, it actually has only as many independent state variables as the corresponding star form. The apparent additional state is actually a linear combination of the other states and the initial conditions but does not show up in the causal assignment. A similar situation arises when two inductances are connected in parallel across ^voltage source. Although causality will show that both inductances have integral causality it is clear that the states are related. In the presence of derivative causality, the formulation of state-space equations from bond graphs involves inverting a matrix.

While the star-delta transformation serves to mitigate this problem, the numerical integration routines have to solve larger systems of equations unless the independence of the equations is checked explicitly and "dependent states" eliminated The other two transformations do address the problem of dependent energy storage elements directly and make the bond graph more comprehensible and are expected to make symbolic equation-formulation computationally more tractable.

Domain Description

We have restricted our test domain to planar motion. Component models for spatial motion would be larger and more cumbersome but would not present any theoretical difficulties. Although our implementation handles only linear behavior, the representation and the simplification techniques are general enough to handle non-linear behavior as well. Non-linear behavior in such systems may arise from non-linear constitutive laws describing specific components. It may also arise from the nature of component connectivity, as in a four-bar mechanism, or from changes in component connectivity as in a Geneva mechanism. While the first two cases essentially fall within the modeling framework described above, the third will simplify to two or more different models - the active one being decided by the state of the system. For all three cases, if the non-linear parameters are determined by the state-variables then forward integration in time can be used to obtain solutions quite readily.

Example: Bicycle

A device such as the bicycle shown in Figure 1 consists of a number of rigid masses which interact with each other and impose kinematic constraints on each other. Consider the motion of the bicycle in a plane arising as a result of a force applied to the pedal when the pedal arm is horizontal. The designer/modeller first selects *mass* primitives to represent the crank assembly, the frame, and each of the two wheels and then establishes kinematic connections between components. The *pinned* connections between the wheels and the frame are represented by bonds which impose common translation^ velocities on the frame and each wheel. The no-slip wheel/ground *rolling* connection also imposes common translational velocity but in the case of rolling the point of common velocity is on the periphery of the wheel. At the contact point, therefore, the vertical and horizontal velocities of the tire are zero, assuming that the tire remains on the road. The rear wheel also interacts with the crank assembly through the chain. We have chosen to neglect the massive characteristics of the chain and to include the kinematic characteristics as a simple transformer element as indicated in Figure 10. Although we know from experience that there is no motion of the crank assembly relative to the rear wheel, these constraints are not imposed by the chain, and therefore are not included as

kinematic constraints between the crank and the rear wheel[4] The crank assembly does, however, share common translational velocities with the frame. Lastly, the crank interacts with the applied force at the pedal. The aggregate model comprised of the basic multi-port massive models representing the wheels, frame, and crank assembly is shown in Figure 10. Also shown in the figure are bonds representing the kinematic constraints among these elements. The figure therefore represents a bond graph suitable for determining the motion of the bicycle resulting from a force applied to the pedal.

It is clear, however, that this model is much more complex than necessary. Complexity is significant from the point of view of a designer or modeller trying to understand the behavior of the device, and also has significant ramifications on equation formulation techniques. A perusal of the figure shows that the overall bond graph is circuitous, redundant, exhibits a high degree of derivative causality, and includes elements and bonds which we know from our own experience have no relevance to the dynamic behavior of the bicycle. It is important therefore that we are able to simplify the bond graph representing the bicycle prior to the application of equation formulation techniques.

The first step of simplification is to remove bonds and junction structure elements which have not been connected after propagating their effect as explained in the section on deletion of inert elements. Next we consider the simplifications arising from the specific physical location of the interaction ports. The attachment locations appear as the transformer moduli in the multi-port massive body bond graph element. All of these transformers are needed in the general case, however, the specific geometry of the bicycle causes many of them to be degenerate. For example, the vertical force applied to the tire at the point of contact with the ground does not contribute a torque on the wheel because the point of contact is directly below the axle. In this specific case the parameter TF_{xl} is zero. Because the torque is zero, the bond emanating from TF_{xI} incident on the -1- junction has no effect since a zero effort applied to a -1- junction has no effect as discussed previously. Dual reasoning allows us to conclude that the deletion of the bond connecting TF_{xl} and the -0- junction will have no effect. All other zero modulus transformer elements and their connected bonds may be deleted for exactly the same reason. These simplifications are shown in Figure 11. The same figure also shows simplifications arising from the deletion of zero value flow source connections propagated through one junctions up to zero junctions as justified by reasoning dual to the justification for deleting zero value effort source elements and their connected bonds.

[4]Redundant constraints can be included, however, there are liabilities as well as benefits to doing so.

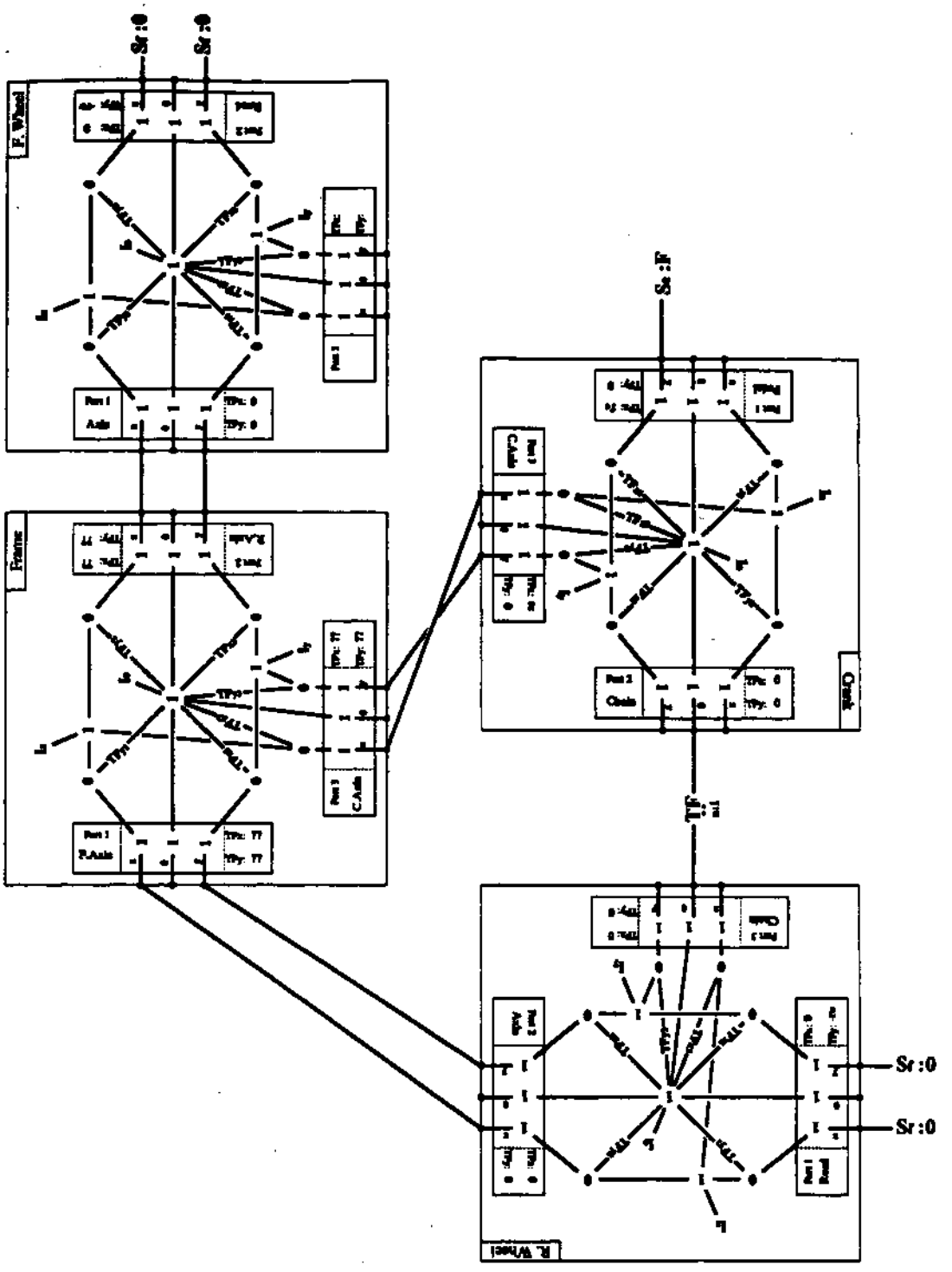


Figure 10: First Cut Mcxicl of the Bicycle

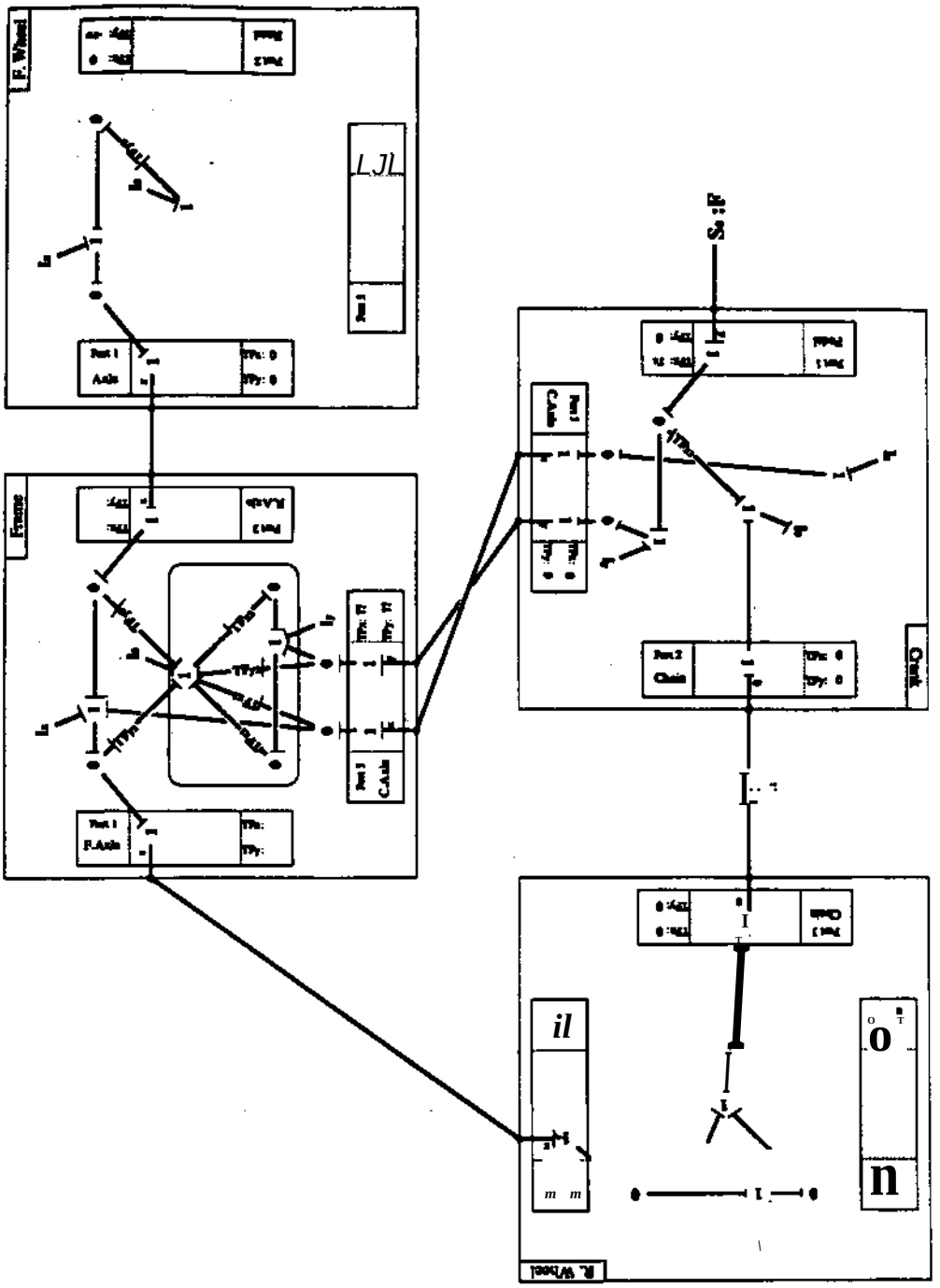


Figure 11: Partly simplified model.

The resulting bond graph is a much simplified version of the original but is still too complex, the greatest degree of remaining complexity is embodied in the frame element where three inertial elements and four bond graph circuits are present. Although we know intuitively that this is too complex (inertia in the vertical direction cannot, for example, be important for a bicycle moving horizontally in a plane) we need to identify unambiguously the source of these unnecessary complexities. The bond graph fragment highlighted in Figure 11 is a critical sub-structure.

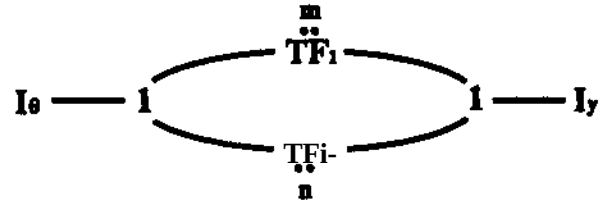


Figure 12: Simplified example of constraining junction structure.

Consider the slightly simplified version of the highlighted area shown in Figure 12 consisting of two -1- junctions, two transformer elements and two inertial elements. Assigning either of the inertial elements integral causality results in an inconsistency and therefore cannot be valid. Assigning both of the inertial elements derivative causality admits a consistent assignment of causality but one in which causality is circuitous and in which equation formulation techniques would require a simultaneous solution of algebraic equations. In this particular case those algebraic equations can be satisfied only if the two transformer moduli are reciprocals[5] of each other. In that case the graph represents a kinematic redundancy. If not, the velocity corresponding to each of the -1- junctions must be identically zero. This is an example of the constraining junction structure discussed previously. Going back to the bicycle problem, we can algorithmically isolate the sub-graph shown in Figure 13. The determinant of $[I - A_2]$ in this case is given by (1- the product of the moduli of TF_{xl} and TF^). The determinant is zero only if the horizontal position of the front axle and the rear axle are identical. This clearly cannot be the case for a stable bicycle. Hence the entire structure shown in Figure 13 can be deleted from Figure 11 to arrive at the simplified bond graph shown in Figure 14.

[5]Figurc 12 is really a bond graph model of the example in Figure 5. The transformer moduli in any bond graph are assigned such that they represent the ratio of flows in the causal direction through the -TF- junction. The -TF- ports in Figure 12 have opposing causal directions. Hence the corresponding moduli will be exact reciprocals if the gear ratios of the two pairs are identical

Although spread over most of the page, Figure 14 is really quite simple. It consists of an acyclic bond graph with inertial elements, transformer elements, a force source **and** the junction structure. If we view this structure as a tree emanating from the force source, we may simplify the tree by moving from the leaves up towards die root by determining equivalent inertias across transformers and by combining inertias. Alternatively we could have used the energy approach to determine the equivalent inertia. We arrive at the bond graph shown in Figure 15. This simple bond graph shows us that the bicycle behaves as an inertia. The effective inertia depends on the translational mass of all elements as well as the rotational inertia of the front wheel, the rear wheel and the crank. The relative importance of these inertias is determined by the transformer moduli which depend on the chain sprocket ratio, crank arm length, and the wheel diameters.

The simplification made it possible to consider not only motions in the horizontal direction, but also the resulting rotational velocities and a force in the vertical direction. Although direct formulation methods were employed, constraint forces did not need to be considered explicitly.

We can use the same model to analyze a situation such as shown in the right-hand side of Figure 1. Here the bicycle is shown tipping over after crashing into a wall. At the instant of crashing the boundary conditions on the bicycle change. The front wheel now makes contact at two points and the rear-wheel is not constrained to remain on the ground. The bicycle bond graph with revised boundary conditions is shown in Figure 16. The model can be reduced to two separate bond graphs as shown in Figure 17. These can be further simplified to two equivalent inertias. The equivalent inertia in this case depends on the rotational inertia of the bicycle frame which can no longer be deleted.

Summary and Conclusion

We have presented some issues and ideas underlying automated modeling in a design tool. In our design environment we define as primitives idealized physical components and kinematic connections in terms of bond graph fragments that describe their behavior. These bond graph fragments are not only accurate models of the physical objects but also satisfy the iequirements of natural interface. The user specifies the kinematic connections between different components of the design and the system translates this into a procedure for aggregating component-level models to form a device-level model. This model is then simplified to a much reduced problem. The simplification procedures propagate externally imposed boundary conditions to delete constrained parts of the model and also replace dependent energy storage elements by their equivalents. Simplifying the model facilitates drawing inferences about dominant behavior and makes analysis and simulation more tractable. Characterizing the resulting equations of motion symbolically will enable us to evaluate design trade-offs and determine high level form-function relationships for the designed device.

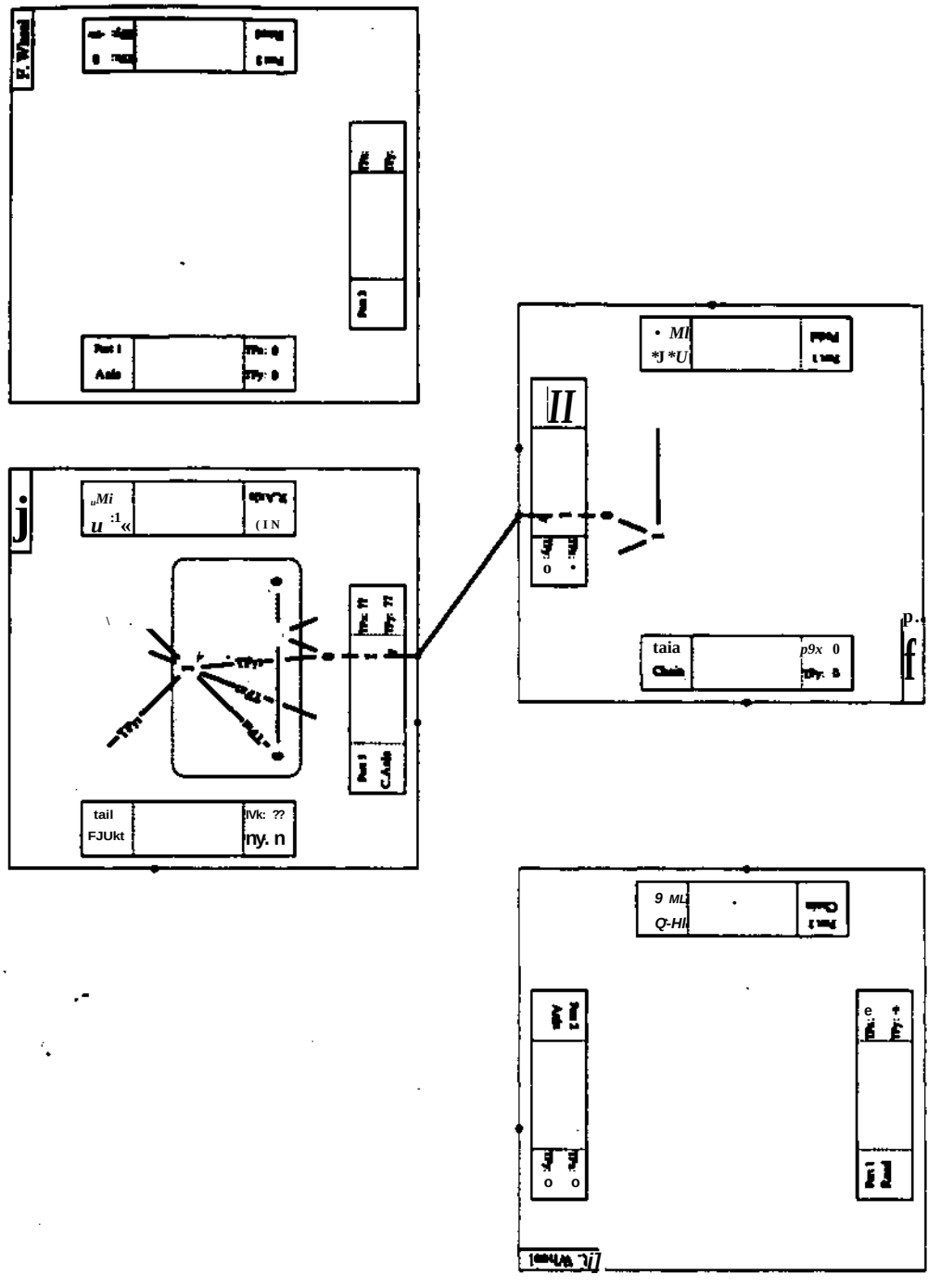


Figure 13: Constraining junction structure.

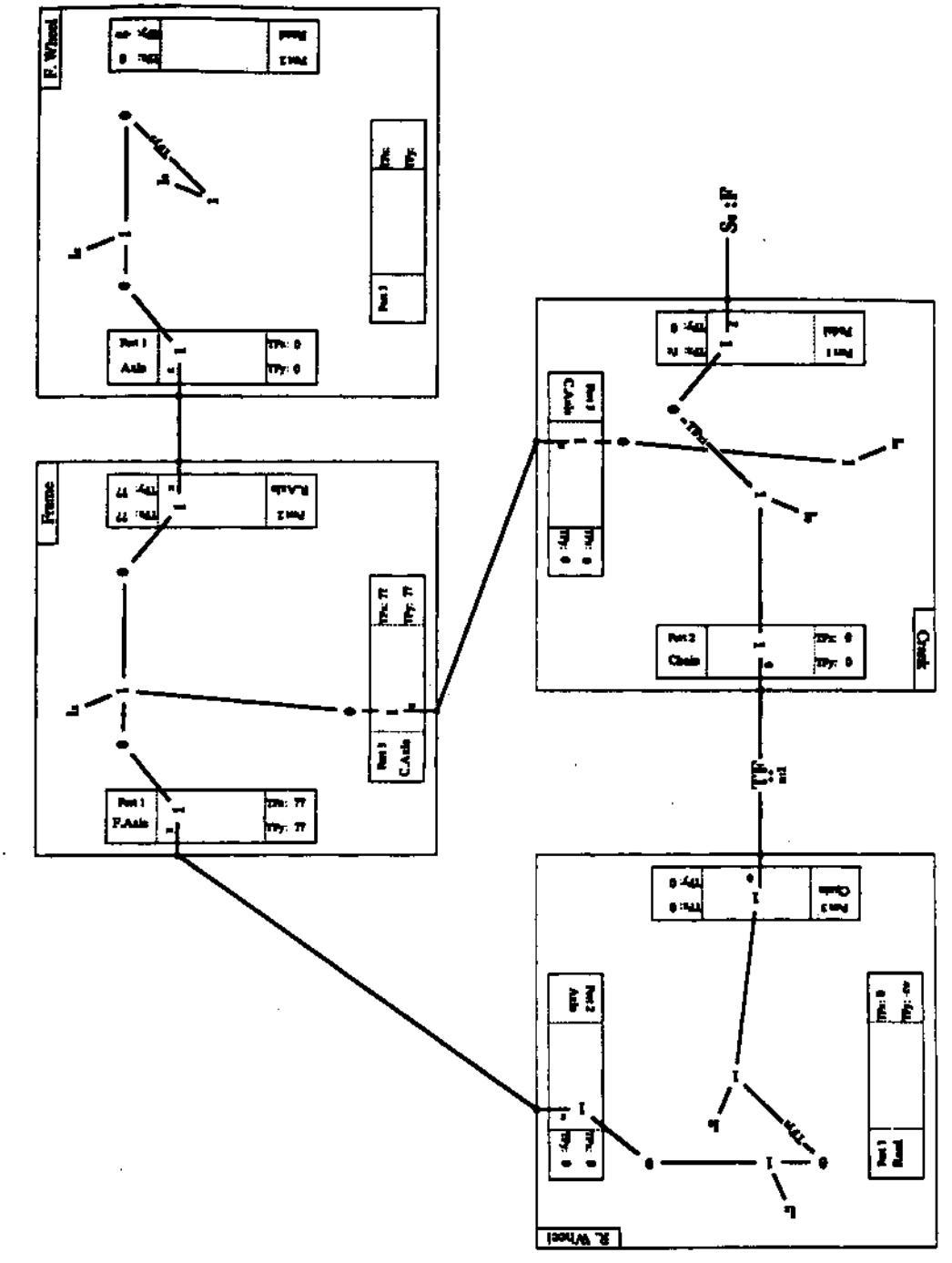


Figure 14: Model without statically constrained elements.

$$I_{eq} = \frac{1}{r_c^2}\left\{ I_{\theta C} + n^2 \left[I_{\theta RW} + r_{RW}^2 \left(I_{XRW} + I_{XF} + I_{XC} + I_{XFW} + \frac{I_{\theta FW}}{r_{FW}^2} \right) \right] \right\}$$

with	Ieq	2	Equivalent inertia
	rc	2	Pedal/crank radius
	Iec	2	Crank inertia
	n	2	Chain sprocket ratio
	IGRW	=	Rear wheel inertia
	rew	2	Rear wheel radius
	IXRW	2	Rear wheel mass
	IXF	2	Frame mass
	IXC	2	Crank mass
	IXFW	2	Front wheel mass
	rFW	2	Front wheel radius
	I6FW	2	Front wheel inertia

Figure 15: Final simplified bond graph model of the bicycle.

Acknowledgments

The authors are pleased to acknowledge the support of the Design Theory and Methodology Program of the National Science Foundation (NSF Grants DMC-84-51619 and DMC-88-14760) and the Engineering Design Research Center at Carnegie Mellon University (NSF Grant CDR-85-22616). The authors are also very grateful to Stephen Ray and Stephen P. Hoover. The former for his help in software implementation and the latter for his many thoughtful comments and, in particular, for pointing out that a junction structure can be thought of as a n-port with a complicated constitutive law. This lead to the algorithm for isolating inert junction structures.

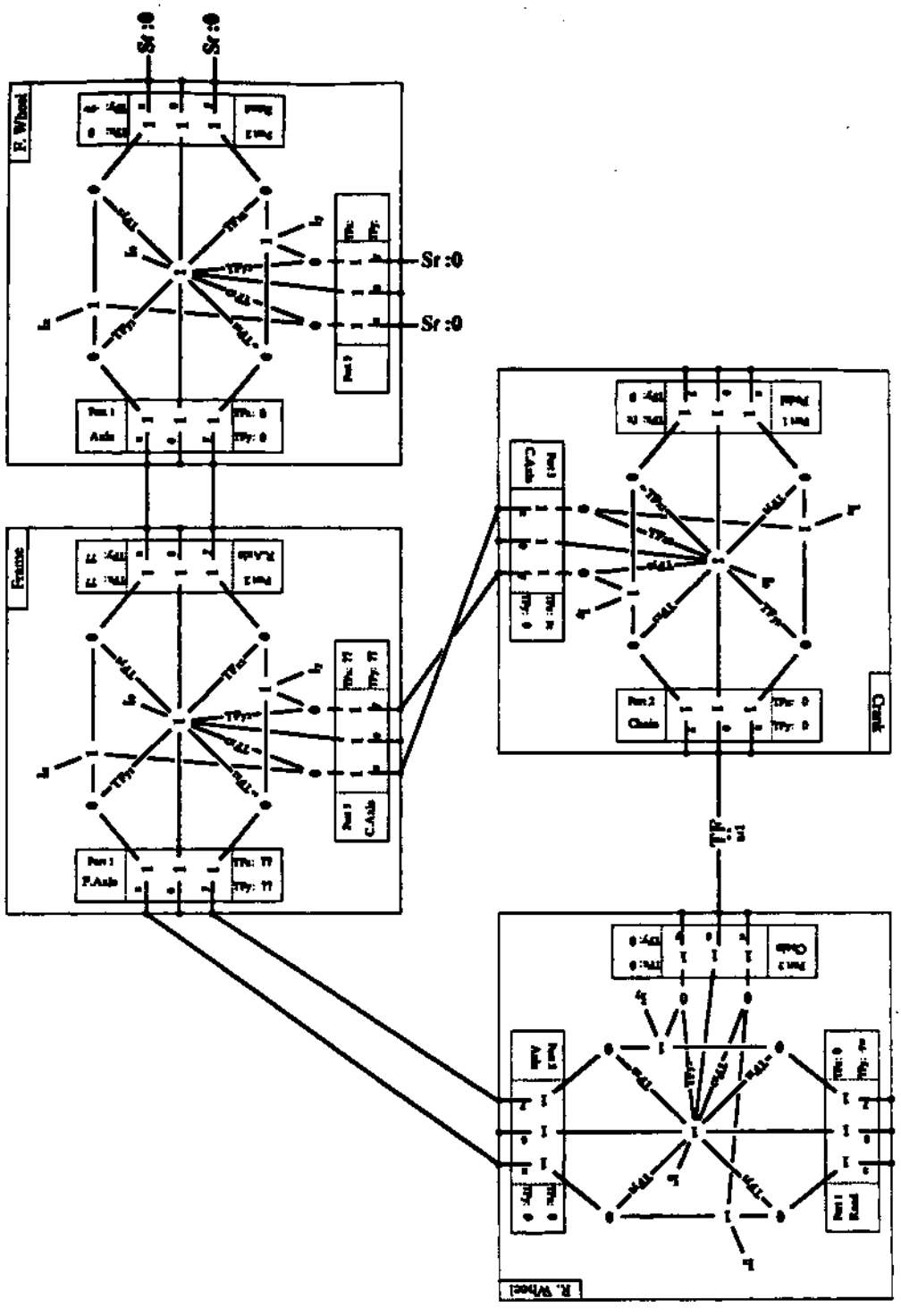


Figure 16: First-cut model of bicycle tipping.

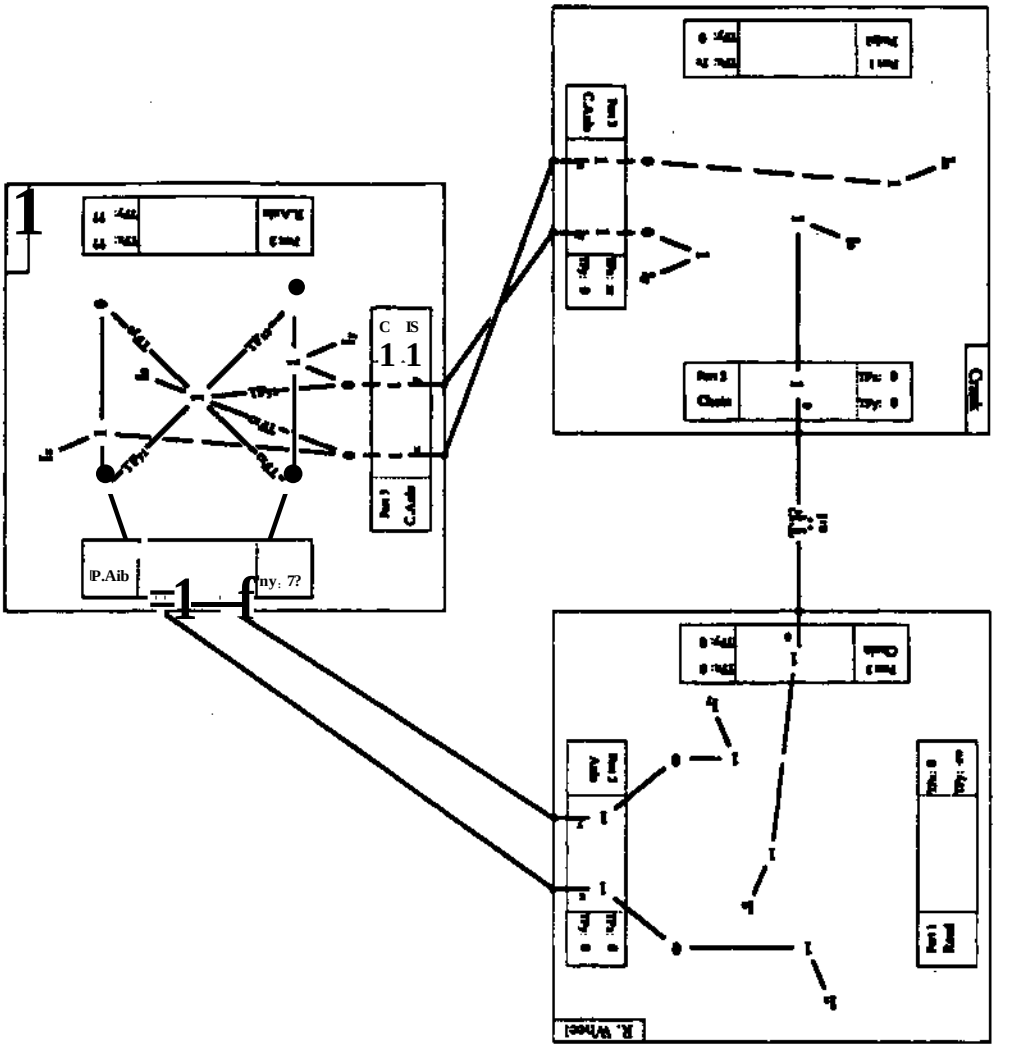


Figure 17: Simplified tipping model.

References

[Balasubramaniam 91]
Balasubramaniam, L., Paz-Soldan, J. P. and Rinderle, J. R., "Automated Modeling to Support Conceptual Design/ *Computer Aided Design,* 1991, pp. under review.

[Bos 85]
Bos, A. M. and Breedveld, P. C, "1985 Update of the Bond Graph Bibliography," *Journal of the Franklin Institute,* Vol. 319, No. 1/2, Feb 1985, pp. 269-286.

[Finger 89]
Finger, S. and Rinderle, J. R., "A Transformational Approach to Mechanical Design Using a Bond Graph Grammar," *Proceedings of the First ASME Design Theory and Methodology Conference,* September 1989.

[Haug 89]
Haug, E J, *Computer-Aided Kinematics and Dynamics of Mechanical Systems Volume I: Basic Methods,* Allyn and Bacon, Boston, 1989.

[Hood 87]
Hood, S. J. and Palmer E. R., "The Design Analysis for Reliability Tool," *Proceedings of the American Control Conference,* June 1987.

[Hoover 89]
Hoover, S. P. and Rinderle, J. R., "A Synthesis Strategy for Mechanical Devices," *Research in Engineering Design,* Vol. 1, No. 2,1989, pp. 87-103.

[Kamopp 75a]
Kamopp, D. and Rosenberg, R., *System Dynamics: A Unified Approach,* John Wiley & Sons, New York, 1975.

[Karnopp 75b]
Kamopp, D., "Some Bond Graph Identities Involving Junction Structures," *Journal of Dynamic Systems, Measurement and Control,* Dec 1975, pp. 439-441.

[Macfarlane 89]
Macfarlane, J. F., *Qualitative and Symbolic Analysis of Dynamic Physical Systems,* PhD dissertation, University of Minnesota, 1989.

[Mclnnis 89]
Mclnnis, J. B. and Elmaraghy, W. H., "Automated Bond Graph Construction and Analysis for MSBD," *Proceedings of ASME Computers in Engineering,* July 1989, San Francisco

[Orlandea 77]
Orlandea, N., Chace, M.A and Calahan, D.A, "A Sparsity-Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems," *Journal of Engineering for Industry,* Vol. 99,1977, pp. 773-784.

[Ort73]
On, J. R and Martens, H. R., "The Properties of Bond Graph Junction Structure Matrices," *Journal of Dynamic Systems, Measurement and Control,* Dec 1973, pp. 362-367.

[Paul 70]
PauU B. and Krajcinovic, D., "Computer Analysis of Machines with Planar Motion," *Journal of Applied Mechanics,* Vol. 37,1970, pp. 697-712.

[Paynter 61]
Paynter, H. M., *Analysis and Design of Engineering Systems,* MIT Press, Cambridge, 1961.

[Paz-Soldan88]
Paz-Soldan, JJP. and Rinderle, J.R., "Conceptual Design Environments: Requirements and Representation," Internal Lab Report CMU-MEDL-88-24, Department of Mechanical Engineering, Carnegie-Mellon University, 1988.

[Perelson 75]
Perelson, A. S., "Bond Graph Junction Structures," *Journal of Dynamic Systems, Measurement, and Control, Jut** 1975, pp. 189-195.

[Prabhu89]
Prabhu, D. R. and Taylor, D. L., "Synthesis of Systems from Specifications Containing Orientation and Positions Associated with Power Flow," *Design Automation Conference,* 1989.

[Rosenberg 75]
Rosenberg, R. C, "The Bond Graph as a Unified Data Base for Engineering System Design," *Journal of Engineering for Industry,* Vol. 97, No. 4, November 1975, pp. 1333-1337.

[Rosenberg 79]
Rosenberg, R. C. and Andry, A. N., "Solvability of Bond Graph Junction Structures with Loops," *IEEE Transactions on Circuits and Systems,* Vol. CAS-26, No. 2, Feb 1979, pp. 130-137.

[Rosenberg 83]
Kamopp, D. and Rosenberg, R., *Introduction to Physical System Dynamics,* McGraw-Hill, New York, 1983.

[Sheth 72]
Sheth, P. N. and Uicker, J. J., Jr., "IMP, A Computer Aided Design Analysis System for Mechanisms and Linkages," *Journal of Engineering for Industry,* Vol. 94,1972, pp. 454-464.

[Ulrich 89]
Ulrich, K. T. and Seering, W. P., "Synthesis of Schematic Descriptions in Mechanical Design," *Research in Engineering Design,* Vol. 1, No. 1,1989, pp. 01-16.