

1 Introduction

The role of computer simulation in engineering design continues to increase as companies strive to gain competitive advantage by reducing the time required to move from concept to final product. As model complexity increases in step with advances in computer software and hardware, the engineer remains well served to use “proper models”—simulation models with sufficient predictive ability but minimal complexity [1].

Proper modeling can be defined as the systematic determination of the model of minimal complexity that (a) satisfies the modeling objectives and (b) retains physically meaningful design parameters and variables. Methodologies compatible with the proper modeling philosophy should be systematic and algorithmic, minimizing the need for a domain expert to override the algorithms and leverage his or her experience and intuition to generate the optimally deduced or reduced model. The methodologies are to be applicable to multi-domain models; i.e., models comprised of electrical, hydraulic, thermal, and multi-body mechanical components. To ensure that underlying assumptions remain valid throughout the process, the required complexity of the model should be reevaluated as the system parameters and environment change.

Proper modeling techniques exist for deducing the required complexity of linear system models in the frequency domain [1], and for reducing nonlinear system models using time domain methods [2]. The applicability of the time-based reduction techniques to more complex models has led to their use in large simulation-based design problems [3,4]. While initial overhead is required to construct a high-fidelity “full” model for subsequent reduction, the effort is justified for design processes in which an optimization algorithm may execute the model thousands of times. In an enterprise with considerable modeling resources, the full model may be created by a team of experts, then archived, and used as the starting point for developing a library of models spanning a spectrum of required complexity depending on the design objective. Reduced-complexity models can be effectively employed by users who may appreciate their ease of use and lack of unnecessary detail.

In addition to eliminating unnecessary complexity from a full model, the modeler may wish to systematically determine if boundaries or partitions exist in the model that allow creation and use of two or more simpler submodels. A priori assumptions of one-way coupling within a system are often made to achieve model simplification. As an example, consider a high-fidelity pitch-plane vehicle model subjected to small road undulations. For a suitably smooth road, the pitch motions will not affect the prediction of the longitudinal velocity and the vehicle can be approximated without including the pitch degree of freedom. The longitudinal dynamic outputs may then be used to drive a pitch model that predicts the time response of the sprung mass rotational inertia and suspension components [5,6]. This reduction becomes questionable when rougher road inputs excite pitch motions more vigorously. The intuition of the analyst may allow him or her to recognize extreme conditions that rule out a decoupled model; however, even the assumptions of experienced modelers may break down as the system parameters or environment change during the design cycle. A quantitative measure is necessary to monitor the validity of underlying modeling assumptions.

This paper describes a method for approaching an arbitrary lumped parameter model, systematically and quantitatively searching for local one-way coupling among individual elements, and determining if the local decoupling sites allow the model to be partitioned into “driving” and “driven” subsystems. The driving subsystem outputs excite the driven subsystem, but the driven dynamic responses do not affect those of the driving system. The algorithm does not rely on assumptions of one-way coupling or an assumed form of the reduced model, and can quantitatively monitor the validity of such assumptions as the design variables and inputs vary. It is proposed that decoupling or partitioning can lead
to significant model reduction and the possibility of parallel simulation of the driving and driven dynamics. Partitioned and reduced models retain a subset of the full model parameters, thus allowing physical interpretation of the resulting analysis.

The next section reviews prior proper modeling and model partitioning research, introduces the general decoupling search method, and reviews the bond graph formalism. The proposed partitioning method is facilitated by, but not restricted to, a bond graph representation of a dynamic system. Section 3 describes the methodology in detail, and in Sec. 4 a lever mechanism is presented as an illustrative example. Discussion, summary, and conclusions follow in Secs. 5 and 6.

2 Background

2.1 Literature Review. Early proper modeling research led to algorithms for deducing a proper model by incrementing model complexity until frequency domain accuracy criteria were met (for example, Stein and Wilson [1], Ferris et al. [7], and Walker et al. [8]). While the deduction techniques were frequency based and therefore effective for linear systems, the need to construct proper models from the large, generally nonlinear models relevant to engineering design led to temporal reduction techniques. Some of these, such as Rosenberg and Zhou [9] and Louca et al. [2] concentrate on finding reduced models whose parameters map to dominant system dynamics, and are effective at finding and removing low-power elements. For example, the model order reduction algorithm [Louca et al. [2]] ranks energy storage and dissipative elements according to an aggregate power flow measure called activity (defined in a later section of this paper), and eliminates elements with activities below a user-defined threshold. The effectiveness of these methods is limited when the analyst is interested in outputs that have low energy or power relative to other system variables. Preliminary work by Ye and Youcef-Toumi [10] that investigated output-specific proper modeling using an energy-based sensitivity approach did not resolve this limitation.

A further limitation of these nonlinear system reduction algorithms is that they do not assess the location or intensity of coupling among elements, nor leverage decoupling to partition a model and facilitate computation. A partitioned model allows existing reduction methods to be applied to smaller submodels, each of which has its own set of dominant and nondominant dynamic elements. Misleading comparisons of the relative contributions of decoupled elements can be prevented (Rideout et al. [6]). Identifying decoupling within a system and partitioning the model can improve the efficiency of the simulation-based design process even if the aforementioned model reduction techniques are not applied a posteriori.

Partitioning methods, of course, do exist in the optimal design community, but are not oriented towards generating proper models that increase insight into intra-system interactions. Partitioning can be done in several ways, such as object, aspect (or discipline), and model-based [11]. Object decomposition divides a system into physical components, and is limited by the fact that defining boundaries comprehensively for large systems is highly subjective [12].Aspect decomposition divides a system according to the specialties of those involved in its modeling, rather than along component lines. Object and aspect partitioning are “natural” partitions and typically large companies employ both types of partitions simultaneously. For example, an automotive manufacturer partitions its organization into powertrain, body, chassis, or electronics divisions (object), but also has dedicated groups for durability, packaging, dynamics, safety, or noise-vibration-harshness (aspect) [13]. These partitioning methods create models of manageable size, but are arrived at in an ad hoc manner, and are not systematically generated in a manner that assesses or minimizes coupling to determine if certain partitions can be eliminated.

Optimal model-based decomposition (OMBD) [11] divides an all-at-once problem into equally weighted subproblems for easier optimization or utilization of parallel computing resources. However, OMBD partitions are mathematical optimal design constructs rather than physically meaningful partitions, and thus are not helpful for generating proper models.

To indicate the strength of couplings among parts of a system model, partial sensitivity derivatives [14] are a possible approach; however, costly symbolic or numerical evaluation of derivatives is required for systems with large numbers of variables.

This paper proposes a power-based metric, evaluated numerically in the time domain, to identify decoupling by predicting the relative importance of individual constraint equation terms. The required calculation of power flow, the need to keep track of relationships among constraint equations, and the desire to partition models remain models motivated the use of bond graph formalism; however, the partitioning algorithm can be applied to any formulation so long as the requisite power variables can be extracted.

2.2 Review of Bond Graphs. In the bond graph formalism, energy is conserved, power flow paths can be identified, and power flow associated with elements and their connections can be readily determined. Generalized inertias \( I \) and capacitances \( C \) store energy as a function of the system state variables, which are generalized momenta and displacements. The time derivatives of generalized momentum \( p \) and displacement \( q \) are generalized effort \( e \) and flow \( f \), the product of which is power. Generalized resistors \( R \) remove energy from the system, and have a constitutive law relating generalized effort to generalized flow. Sources of effort and flow (\( Se \) and \( SF \) ) represent ports through which the system interacts with its environment.

Energy is transported among source, storage and dissipative elements through power-conserving “junction structure” elements. Such elements include power-continuous generalized transformers (TF) and gyrators (GY) that algebraically relate elements of the effort and flow vectors into and out of the element. The constitutive laws of “modulated” transformers and gyrators (MTF and MGY) are functions of external variables, for example coordinate transformations that are functions of generalized coordinates. Kirchhoff’s loop and node laws are modeled by power-conserving 1- and 0-junctions, respectively. Elements bonded to a 1-junction have common flow, and their efforts algebraically sum to zero. Elements bonded to a 0-junction have common effort, and their flows algebraically sum to zero.

The power bonds contain a half-arrow that indicates the direction of algebraically positive power flow, and a causal stroke normal to the bond that indicates whether the effort or flow variable is the input or output from the constitutive law of the connected element. Full arrows are reserved for modulating signals that represent powerless information flow, such as orientation angles for coordinate transformation matrices.

Table 1 defines the symbols and constitutive laws for energy storage and dissipative elements (“energetic” elements), sources, and power-conserving elements. The constitutive laws are written in an input-output form consistent with the placement of the causal strokes. The reader is referred to Karnopp et al. [15] for a more thorough development of bond graphs.

3 Hypothesis and Methodology

The power-conserving constraint equations represented by the junction structure of a bond graph model link the constitutive law variables of energetic elements, and thus can give insight into the location of weak coupling. It is proposed to define and examine the “relative activity” of bonds at each 0- or 1-junction as a means of unearthing negligible constraint equation terms. One-way coupling in the model manifests itself as groups of energetic elements demarcated by negligible constraint terms. Here the insight provided by bond graphs is extremely valuable.

Activity \( A \) [2] of an element or bond over the time interval \([t_1,t_2]\) is defined as

\[ A = \frac{\int_{t_1}^{t_2} (e \cdot f) dt}{\int_{t_1}^{t_2} (e \cdot e) dt} \]
A = \int_{t_1}^{t_2} |P| \, dt \quad (1)

where \( P \) is instantaneous power (product of effort and flow) of the element or bond. Activity is always positive, and either constant or monotonically increasing, over a given time interval.

3.1 Proposition for Constraint Term Assessment Based on Local Activity Comparison. The following method is proposed for assessing constraint equation term contributions.

**Article 1.** The activity of a bond (constraint equation term) attached to a 0- or 1-junction (constraint equation), compared to the activities of the other bonds at that junction, is a measure of the relative importance of the term compared to the other terms in the equation. Relative activity \( R_{Ai} \) of bond \( i \) at a junction is defined as the ratio of bond activity \( A_i \) to maximum bond activity \( A_{max} \) at the junction:

\[
R_{Ai} = \frac{A_i}{A_{max}} \quad (2)
\]

Low relative activity of bond \( i \) at a junction implies that:

(i) for a 0-junction with \( n \) bonds, the flow \( f_j \) can be neglected in the flow constraint equation

\[
\sum_{j=1}^{n} f_j = 0 \quad (3)
\]

(ii) for a 1-junction with \( n \) bonds, the effort \( e_i \) can be neglected in the effort equation

\[
\sum_{j=1}^{n} e_j = 0 \quad (4)
\]

The effort or flow common to the junction in item (i) or (ii) may be negligible in relation to other efforts or flows within the system. This is immaterial given that the bond activity is compared only to the other activities at that single junction.

**Article 2.** A power bond with low relative activity at a 0-junction can be “conditioned,” or converted to a modulated effort flow source. The modulating signal is the effort flow from the junction. The effort flow is applied to the node at the other end of the bond; i.e., the end that is not adjacent to the 0-junction at which the bond is inactive. The half-arrow direction of algebraically positive power flow of the modulated source is the same as that of the original low-activity bond. Table 2 illustrates the conversion.

**Article 3.** The activity of an “internal” junction structure bond between two junction structure elements must be compared to the activities of the other bonds connected to both elements, as shown in scenario (i) of Table 2. If both cases A and B apply, then the original bond can be eliminated. If the local activity of an “external” junction structure bond between a junction and an energy storage or dissipative element is negligible, a trivial driven partition results as shown in scenarios (ii) and (iii) of Table 2. The generalized impedance \( Z \) representing an \( I, C, \) or \( R \) element can then be eliminated from the model.

Note that in the table, the locally inactive bonds are causally weak; i.e., they do not provide the flow input to a 1-junction nor the effort input to a 0-junction. The issue of causal strength is addressed in Sec. 3.4.

Other possible internal bond connections are shown in Table 3 where the internal bond connects a 1- or 0-junction to a (M)TF or (M)GY element. If both bonds are locally inactive compared to their respective junctions, then the transformer or gyrator can be eliminated. If one bond is locally inactive, a modulated source in sequence with an (M)TF or (M)GY results. The transformer or...
gyrator can be incorporated into the source, resulting in the
equivalent conditioned junction structure shown in Table 3.

Article 4. Replacing a negligible constraint term with a modu-
lated source is equivalent to replacing a two-way power flow path
with a one-way signal. Modulating signal locations and directions
define subgraphs and partitions according to the following
definitions.

DEFINITION 1. A conditioned model or bond graph is a bond
graph with all locally negligible power bonds converted to modu-
lated sources or eliminated.

DEFINITION 2. Given a conditioned bond graph, a
subgraph $S_i$ is a set of elements from that bond graph (including sources, ener-
geric elements, and junction structure) that has no power bonds
connected to any bond graph element outside the set. The sub-
graph may be connected to the rest of the bond graph by modu-
lating signals, or unconnected. Figure 1(a) is a schematic of a
general bond graph with two sets of elements connected by two
power bonds and one modulating signal. The modulating signal is
shown as a dashed line to distinguish it from signals arising from
conditioned bonds. No subgraphs exist. Figure 1(b) depicts the
subgraphs that might result from conditioning the power bonds.

DEFINITION 3. Driving and driven subgraphs: When all new
modulating signals (due to bond conditioning) between two sub-
graphs are directed from one subgraph to another subgraph, the
subgraph from which the signals originate is the driving subgraph
$S^-$, and the other is the driven subgraph $S^+$ as shown in Fig.
1(b).

DEFINITION 4. A subgraph loop $S_L$ is a set of subgraphs in
which each $S \in S_L$ is connected to the previous by modulating
signals directed into $S$, and to the next by modulating signals
directed out of $S$. The two subgraphs of Fig. 1(b) form a subgraph
loop.

DEFINITION 5. A driving partition is a subgraph $P^-$ that is
connected to at least one other subgraph strictly by modulating
signals directed outwards from $P^-$ to that subgraph. See Fig. 2
for examples.

DEFINITION 6. A driven partition is a subgraph $P^-$ that is not
an element of a subgraph loop, that is connected to at least one
other subgraph strictly by modulating signals directed inwards
towards $P^-$ from that other subgraph (Fig. 2). A subgraph loop
can be a driving or driven partition.

The distinction between driving (or driven) subgraphs and par-
titions is drawn because modulating signals may exist, prior to
conditioning, which create subgraph loops and prevent partition-
ing, as in Fig. 1(b).

3.2 Conditioning Algorithm. The activity comparison and
interpretation described in the previous section are incorporated
into the algorithm shown schematically in Figs. 3 and 4. The first step is to construct a model that predicts the outputs of interest acceptably, and assign causality (input-output relations). The outputs of interest of the full model are recorded for later comparison (Plot outputs of interest) with the reduced or partitioned model.

Activities must then be calculated. The integration limits in Eq. (1), which give the time window over which activity is calculated, are chosen by the analyst based on the modeling objectives. The modeler may wish to simulate the full model once, prior to calculating activities, to check for periods of transient or steady state behavior that suggest natural time windows. For example, a truck model hitting a pothole will show a pitch angle transient. A reduced model to predict the pitch response can be generated by taking a time window around the transient produced model to predict the pitch response can be generated by

1. Calculating activity
2. Choosing threshold
3. Comparing local activity of I junction
4. Locally inactive bond set

When all junctions have been checked, the search stops and the output, the driving model structure can be eliminated after the simulation of both partitions is required, then each comprises a smaller submodel that may be run separately (Simulate driving partition only). To predict a driven partition output, the driving model structure can be eliminated after the necessary modulating signals are generated and stored in an input file (Generate modulating signals). The driving partition can be excited by arbitrary inputs as long as the resulting outputs are not severe enough to compromise the original one-way coupling. If simulation of both partitions is required, then each comprises a smaller submodel that may be run separately (often with reduced computation time) on different processors. Partitioning also suggests parallel processing, with the driven and driving partition outputs generated nearly simultaneously on different computers with information flowing from driving to driven as necessary.

Note that the driving elements are unaffected by driven dynam-
ics in either the conditioned or partitioned model, and the driven element state trajectories will be the same for both models unless the required outputs from the driving subgraph are sampled too slowly. Therefore, agreement of the conditioned and full models assures agreement of the partitioned and full models.

Regarding the set of candidate elements for elimination from Fig. 4, the user may choose to retain these terminal elements and condition only the inactive “internal” junction structure bonds. The retention of terminal elements will not affect the existence of subgraphs, and the resulting partitioned model will retain all state variables from the original model. Reduction techniques such as the model order reduction algorithm [2] can then be applied to each individual partition if desired. Alternately, conditioning all inactive bonds, whether external and internal, allows the algorithms to be used to eliminate state variables as well as find partitions.

3.4 Parasitic Constraints. Note that in Fig. 4, a bond conditioning exception exists when causally strong, locally inactive bonds connect the junction structure to constraining or “parasitic” elements (Fig. 3, Identify parasitic elements). To create ordinary differential equations in multibody systems where rigid body moments are constrained, Karnopp and Margolis [17] propose the use of joint models with stiff “parasitic” springs rather than ideal velocity constraints. In bond graph terms, a C element, bonded to a 0-junction, is placed between the 1-junctions representing the velocities to be constrained. If a parasitic damper is used in parallel with the spring (to hasten settling of high-frequency transients), then an internal bond will lead from the 0-junction to a 1-junction at which the parasitic C and R elements are bonded.

The velocity associated with the bond leading to the parasitic elements is the constraint violation velocity, which must be insignificant for an approximated rigid constraint. For bonds to parasitic elements, local inactivity of the bond does not indicate decoupling within the system, but rather that the constraint approximation is adequate. The bond must not be conditioned lest the parasitic elements be lost. The algorithm can therefore be used to tune parasitic elements as described in Rideout and Stein [18]. The modeler has control over the placement of these parasitic elements, and is aware of their location. After calculating and comparing activities, he or she can then choose not to condition parasitic element bonds. If the algorithms were fully automated, a parasitic C or R element would have to be tagged as such, with local activity used to confirm its efficacy rather than to decouple or remove it.

3.5 Causality Considerations in Conditioning. As per Fig. 4, the causality of a locally inactive bond affects the decision to convert it to a modulated source. A fundamental difficulty arises in removing the causally strong input to a junction as shown in Fig. 6 below. A bond provides a “causally strong” input to a junction if it defines the flow input to a 1-junction, or the effort input to a 0-junction.

The physical interpretation of the low activity of bond 1 at the 0-junction is that the flow \( f_1 \approx f_2, \ldots, f_m \). At the 1-junction where \( A_1 \) is on the order of the other activities, we then assume \( e_1 \approx e_{m+1}, \ldots, e_m \). Simply replacing bond 1 with a modulated flow source into the 1-junction is nonsensical given that \( f_1 \) cannot be extracted from the 0-junction. The appropriate modulating signal can only be generated after causality reassignment. Reassigning causality is similarly necessary when eliminating energetic elements through bond conditioning (see Fig. 7). Reassignment of causality allows us to excite the locally inactive mass in the figure with a modulated flow source, which is correct given that the inertial effort, and not the common flow, is the negligible quantity at the junction.

4 Illustrative Example

A primary advantage of the algorithms of Sec. 3 is that the partition boundaries need not be assumed, however intuitive they may appear, and no a priori assumptions about the form of the partitioned model are required. The conditioning and partitioning algorithms are demonstrated for the example system shown in Fig. 8. The exact partition boundaries are not obvious, even for this simple system, and will change as a function of the system parameters.
parameters. The example system is linear, but note that no changes to the algorithms are required for application to nonlinear system models.

The two masses and three parallel spring-damper arrangements are connected through a lever. Inertia of the lever is neglected. The lever is assumed to be long enough so that the endpoints approximately translate in the $v_1$ and $v_2$ directions. A step input force $F_1$ is applied to the mass $m_1$.

For this system, increasing the ratio $a/b$ will increasingly attenuate the force $F_1$ transmitted through the lever to $k_2-c_2$. Thus, depending on the energetic element parameters, the response of $m_1$ may not be affected by the dynamics of $m_2$. The velocity $v_3$, which would be significantly lower than $v_3$, for large $a/b$, would still be necessary to define the spring velocity $v_3$ and set $m_2$ into motion.

Note that neglecting the lever inertia creates an algebraic loop, i.e., an algebraic constraint equation between the forces $F_{c2}$ and $F_{c3}$ in dampers $c_2$ and $c_3$. With state variables defined as the mass velocities and spring displacements, the differential-algebraic equations of motion are given below:

\[
\begin{align*}
\dot{v}_1 & = -c_1 v_1 - k_1 x_1 - c_3 x_3 + F_{c2} \\
\dot{v}_2 & = b a m_2 - k_1 x_1 + F_{c2} \\
\dot{x}_{c1} & = 0 \\
\dot{x}_{c2} & = b c_3 x_3 + F_{c3} \\
\dot{x}_{c3} & = b a m_1 - b k_3 x_1 + F_{c3} \\
\end{align*}
\]

\[
\begin{bmatrix}
\dot{v}_1 \\
\dot{v}_2 \\
\dot{x}_{c1} \\
\dot{x}_{c2} \\
\dot{x}_{c3} \\
\end{bmatrix} =
\begin{bmatrix}
-c_1 & 0 & 0 & -k_1 & -b k_3 & a m_1 \\
b c_3 & -b & 0 & 0 & k_1 & m_2 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & -b & 0 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
x_{c1} \\
x_{c2} \\
x_{c3} \\
\end{bmatrix} +
\begin{bmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\begin{bmatrix}
F_{c2} \\
F_{c3} \\
\end{bmatrix}
\]

(5)

Figure 9 shows the system bond graph, with activity values (corresponding to $a/b=1$) adjacent to each bond, and internal junction structure bonds shown in bold. The lever is modeled as a TF element with modulus $a/b$. Note that both bonds of the transformer have the same activity given that the lever is power conserving. The 0-junctions $F_a$ and $F_b$ represent the combined spring and damper forces at each end of the lever, and the 1-junctions $v_{c2}$ and $v_{c3}$ define the relative velocities of the endpoints of springs 2 and 3. Bonds 1 and 2 in the figure have relative activities of 0.14% and 2.4% with respect to the $v_1$ and $F_2$ junctions, respectively. For lever ratios ranging from 0.1 to 1, bonds 1 and 2 are consistently the internal junction structure bonds with the lowest relative activities. These relative activities are plotted against the lever ratio $a/b$ in Fig. 10.

A sufficiently high lever ratio will render one or both bond activities negligible compared to the junction maximum. Conditioning bond 1 will create driving and driven partitions. The ability to condition bonds 1 and 2 also allows the spring and damper $k_1-c_1$ to be eliminated within the driven partition. Eliminating $c_2$ breaks the algebraic loop, thus facilitating equation formulation and computation.

Consider a 3% activity threshold. A system with a lever ratio of 0.16 or greater can be partitioned by conditioning bond 1. The physical interpretation is that the effect of elements $k_2$, $c_2$, $k_3$, $c_3$, $m_3$, and the lever on mass $m_1$ can be neglected. The output $x_1$ can be predicted with a 5.8% error on the peak value, using only the bond graph structure to the left of the $F_2$ 0-junction. The remaining structure, along with an input signal $v_1$, can be used to generate predictions of $x_2$. Figure 11 shows the conditioned model, with bond 1 from Fig. 9 replaced by a signal and modulated flow source (MSF). Figure 12 shows the driving and driven partitions.

The outputs are plotted in Fig. 13.

For lever ratios greater than 0.89, bond 2 in Fig. 9 can be conditioned as well, resulting in the conditioned model and driven...
partition of Figs. 14 and 15, and the output time series shown in Fig. 16. The trivial driven subgraph containing \( k_2-c_2 \) is discarded in Fig. 15 as it does not affect the outputs of interest. For the partitioned system of Fig. 15, the equations of motion below illustrate the partitioned state variable vector and the presence of the input \( v_1(t) \) to the driven partition:

\[
\begin{align*}
\dot{v}_1 &= -c_1 - \frac{k_1}{m_1} v_1 + \frac{1}{0} \{F_2(t)\} \\
\dot{x}_{k2} &= \left( -c_1 - \frac{k_1}{m_1} \right) v_1 + \frac{1}{0} \{F_2(t)\} \\
\dot{u}_2 &= \left( -c_3 - \frac{k_3}{m_2} \right) u_2 + \frac{b}{a} v_1 \\
\dot{x}_{k3} &= \left( -c_3 - \frac{k_3}{m_2} \right) u_2 + \frac{b}{a} v_1 \\
\end{align*}
\]

(6)

For the lever ratio of 0.89, model sizes, simulation times, and number of computation steps for the fully coupled, conditioned, and partitioned models are compared in Table 4. Simulations were performed using the 20SIM [19] Runge-Kutta-Fehlberg integrator with absolute, relative, and algebraic constraint tolerances of \( 10^{-6} \). The fully coupled and conditioned model computation times are based on running the model with both partitions present. The driving partition time is the time required to simulate the driving partition only, while writing to a data file the signals necessary to excite the driven partition. The driven partition time is that required to simulate the driven partition, excited by the driving input \( v_1(t) \), without the driving partition junction structure and energetic elements present. Simulation times are compared in Fig. 17.

Simply conditioning the model eliminates the algebraic loop and allows prediction of both outputs more efficiently than with the fully coupled model, even though no states are eliminated and the model structure of both partitions is present. Simulation of the driving or driven partition (or both in parallel or in sequence) requires fewer steps and less time.

The computation steps and time reductions are more significant for larger models with more conditioned bonds and more balanced partitions. In the example problem presented the fully coupled model runs so quickly that factors such as overhead to graphically display results on the monitor comprise a significant portion of the total time. In Rideout et al. [20], a 279-state three-dimensional engine model was partitioned, with sequential driving-driven partition simulation reducing the computation time by 54%. Significant computational savings are also reported for a decoupled vehicle dynamics model in Rideout et al. [6]. For optimization exercises requiring large numbers of model executions, the aggregate savings can far outweigh the initial overhead required to develop the fully coupled model.

5 Discussion

The method proposed and demonstrated in this paper allows the modeler to

- systematically find decoupling within an arbitrary lumped-parameter model of a nonlinear system, rather than using a priori decoupling assumptions;
- monitor the decoupling strength as the system parameters or environment change;
- validate a priori decoupling assumptions and subsequent model reductions;
- eliminate model structure and simulate partitions individually.

Decoupling can be found by, and benefits can accrue to, an engineer without a large amount of specific domain expertise or intuition. Even a comparative expert, who may not realize when an initially conservative assumption about a complex system has become risky, can benefit from the algorithms.
Note that even in the simple illustrative system, partitions are found without making intuitive (and potentially risky) assumptions about where the exact partition boundaries lie within the equation structure. Bond conditioning sites dictate the existence of partitions. When parameter or input changes (such as lowering of the lever ratio) compromise decoupling, then previously conditioned bonds must revert to their original form as conduits for two-way power flow. Reinstating power bonds as their local activities rise above the threshold automatically updates the model to an appropriate level of complexity. Parameter sweeps with local activity checking will automatically create a spectrum of models from partitioned, through intermediate complexity (possibly with fewer partitions or partitions with shifted boundaries), to fully coupled. Initially conservative assumptions of decoupling based on intuition and experience can be monitored and the model updated as necessary as the design evolves and the environment changes. Retention of physical parameters maximizes insight into the coupling among dynamic system elements.

As with other proper modeling methods (for both linear and nonlinear systems), the partitioned and/or reduced model is valid in the neighborhood of the design point and inputs used in calculating the reduction metrics. Total computation time for a design exercise can be significantly reduced if partitioned and reduced models are used, even if the full model must be simulated periodically to recalculate the activities and confirm that appropriate bonds have been conditioned. Investigation of the range of validity is a topic of future research; however, the useful design space of the partitioned example system can be demonstrated through a parameter sweep. Figure 18 shows the relative activity of bond 1 (into the $v_1$ -junction in Fig. 9) as a function of forcing frequency, if the 100 N step input is replaced by a sinusoid with 100 N amplitude. For a threshold of 3% and a lever ratio of 0.16 (the lowest value for which bond 1 can be conditioned), partitions exist for forcing frequencies above approximately 6 rad/s; and for a lever ratio of 0.96 (slightly greater than the minimum for conditioning bonds 1 and 2), “partitionability” is insensitive to the forcing frequency over the tested bandwidth.

Figure 19 shows the results for a parameter sweep of each individual spring stiffness, using the original step input. As expected, given that 0.16 is the lowest possible lever ratio for conditioning of bond 1, design with the partitioned model can proceed if $k_1$ is raised and $k_3$ is lowered. The 0.96 lever ratio shows similar trends but allows much larger variation of all three stiffnesses. Although a detailed multivariable parameter sweep would not be desirable for large nonlinear systems, occasional numerical evaluation of relative activity sensitivity to parameter variation is a possible means of predicting the extents of the partitioned model design space.

**Table 4 Computation effort comparison**

<table>
<thead>
<tr>
<th>Model</th>
<th>Equations</th>
<th>Variables</th>
<th>Indep. states</th>
<th>Algebraic Loops</th>
<th>Comp. steps</th>
<th>Comp. Time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully coupled</td>
<td>40</td>
<td>51</td>
<td>24</td>
<td>1</td>
<td>20,153</td>
<td>0.305</td>
</tr>
<tr>
<td>Conditioned (bonds 1, 2)</td>
<td>39</td>
<td>47</td>
<td>22</td>
<td>0</td>
<td>3888</td>
<td>0.198</td>
</tr>
<tr>
<td>Driving partition</td>
<td>11</td>
<td>13</td>
<td>7</td>
<td>0</td>
<td>2709</td>
<td>0.102</td>
</tr>
<tr>
<td>Driven partition</td>
<td>26</td>
<td>31</td>
<td>13</td>
<td>0</td>
<td>2804</td>
<td>0.081</td>
</tr>
</tbody>
</table>

Figure 17 shows the computation time comparison. Figure 18 shows the relative activity (bond 1) versus forcing frequency.
Future improvements to the algorithm include more rigorous selection of the local activity threshold. Threshold-setting for the conditioning algorithm is complicated by the fact that, given system nonlinearity and the presence of an absolute value operator in the activity definition, the closed-form relationship between the activity threshold and system response is unknown. Future work will move the threshold assignment away from an arbitrary model-wide activity ratio, towards tolerances on points of engineering relevance such as overshoot and rise time that can be identified from the full model output plots. By combining the conditioning algorithm with a quantitative model accuracy algorithm, bonds can be converted in ascending order of activity ratio beginning with the most locally inactive bond in the model. When conversion of the next bond causes the model’s performance index [21] to become negative, conversion can stop.

Despite their acceptance in the proper modeling community and successful application to a range of engineering case studies, activity-based methods for model partitioning and reduction lack the rigorous mathematical underpinnings of linear system reduction techniques. However, activity-based proper modeling can be directly applied to nonlinear systems, is not restricted to any particular type of input, and maximizes physical insight by retaining physically meaningful parameters. Fathy and Stein [22] have shown that for linear system models with certain initial conditions and inputs, the reduced model attained through balanced truncation will be identical to a bond graph model in which the lowest-activity elements are eliminated. While general equivalence between balanced truncation models and models reduced using activity has not yet been proven, the theoretical justification of activity as a reduction metric for linear systems expressed in a particular form increases confidence in the activity approach. The work of Fathy and Stein is relevant inasmuch as activity is calculated for elements of a set of constraint terms, and elimination of negligible-activity terms is held to have no significant effect on the dynamic response of the remaining elements of the set. Further extensions in this area are required before the conditioning and partitioning algorithms can be subject to formal proof.

An additional subject of future research is a formal procedure for breaking subgraph loops, as described in Sec. 3.1. In addition to modulating signals created by bond conditioning, other signals may be present due to modulated bond graph elements in the original model. Examples include dependent voltage or current sources modeled as $R$ elements with resistance that varies as a function of valve position; and modulated transformers representing coordinate transformations. Modulated element signals that carry information from a driven to a driving subgraph must be assessed to quantify whether or not the subgraph loop can be broken. The contribution of such signals can be evaluated if the modulated energetic element can be replicated into modulated and nonmodulated components, the total effect of which is the same as the original element. If the modulated elements can be shown to be locally inactive, then they can be eliminated and the loop broken at that site. Possible tools for element replication include Taylor series expansion, analytical separation of constitutive laws, or separating the modulating signal into nominal and varying components.

Finally, the algorithms will be more fully automated. Mature bond graph simulation environments such as 20SIM and CAMP/G [23] exist that allow potential implementation of structured local activity calculation and comparison, causality reassignment when necessary, calculation of performance indices, and the identification of subgraphs and partitions. Other platforms could be used given that graphical modeling formalisms facilitate, but are not a theoretical prerequisite for, algorithm development and implementation.

6 Summary and Conclusions

In response to the ongoing need to systematically determine accurate models of minimal complexity for simulation-based engineering design, algorithms have been developed that allow the engineer to approach an arbitrary model consisting of discrete dynamic system elements and their connections, and perform the following operations:

1. search each constraint equation for negligible terms;
2. convert the model to a “conditioned” model in which the negligible terms are eliminated from the mathematical equations;
3. identify “partitions”—collections of dynamic elements between which one-way coupling exists;
4. simulate “driving” and “driven” partitions separately or in parallel.

Driven partition outputs are excited by, but do not significantly affect, the driving dynamics. Partitions are lower-order subsets of the original model that present smaller design vectors to an optimization algorithm, and can typically predict system outputs of interest with significant computational savings.

The method was demonstrated for a lever mechanism in which decoupling and partition boundaries varied with the lever ratio. The partitioned model was able to remove algebraic constraints, generate outputs in each partition accurately, and reduce computation time. The decoupling search and partitioning algorithms thus provide a new automated modeling tool (in addition to existing techniques such as model reduction and accuracy assessment) to facilitate simulation-based design of dynamic systems with maximal accuracy and efficiency.

References


