The Mathematical Foundations of Bond Graphs—I. Algebraic Theory

by S. H. BIRKETT *and* P. H. ROE

Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada

ABSTRACT: *Elementary mathematical concepts from linear algebra are used to develop an independent theory for non-directed bond graphs. The definition of a bond graph is given and its structure as a combinatorial object is studied. This is accomplished by constructing a vector space, called the bond space of the bond graph. Topological information is encoded in the bond graph by defining a subspace of the bond space, the* s-*space.*

The internal bond information is eliminated by studying subspaces of the s-*space, the cycle and internal space. A precise meaning is given to the association between bond graphs and linear graphs by comparing their respective cycle spaces. A technique, called the cut and paste method, is described which produces a bond graph associated with a given graph via its planar representation. Several problems associated with the inverse procedure are discussed.*

I. Introduction

Over the past several decades two apparently different techniques have emerged for the modelling of physical systems. In the electrical domain, graph theory has been used since Kirchhoff's contributions to network theory in 1845 and 1847, but it was Trent (**1**) who first recognized the generality of the graph theory approach in all physical domains. The early work was extended by Koenig and Blackwell (**2**) and by Koenig, Tokad and Kesavan (**3**). There exists a substantial mathematical basis for this modelling technique, namely, the theory of graphs. A linear graph becomes a system graph when it is used in a system model.

A second technique is also used to model physical systems, parallel to the methods which use linear graphs. This technique was invented by Paynter (**4**), who first introduced the concept of bond graph. Information concerning the topology of interconnection of the system components is coded into symbols which appear in the bond graph. Other symbols are used to denote the physical components. The part of a bond graph model which contains the topological information is known as a *junction structure*, although, as used by some authors, this term can also include some physical parts of the model (such as *TF* or *GY* components). Several texts have appeared describing the fundamentals of bond graph modelling, for instance, Rosenberg and Karnopp (**5, 6**).

Bond graphs have always been used in an *ad hoc* manner. By this, we mean that there is no combinatorial theory of bond graphs, analogous to the mathematical theory of graphs which is used as the foundation for system graph modelling. In

this series of papers on the mathematical foundations of bond graphs, we hope to provide such a theory for bond graph modelling. If this is to be accomplished we must analyse the properties and capabilities of the bond graph notation as a pictorial representation of combinatorial information.

The current approach to modelling with bond graphs is similar to the use of the Heavyside methods for the solution of linear differential equations. These methods worked very well, as do bond graph techniques now, however, their *ad hoc* nature made them suspect and they were not universally accepted. The use of Laplace transforms provided the mathematical theory which explains why the Heavyside techniques work so well (**7**).

It is quite commonly understood that bond graphs are applicable mainly in practically oriented problems (such as industrial simulations), whereas linear graphs are used mainly in theoretical modelling and have little applicability in a practical problem. Both of these statements are incorrect. Any bond graph model has an equally useful parallel linear graph model, regardless of how "practical" it is. There are no physical components used in bond graph modelling which could not be used in a linear graph model. On the other hand, any linear graph model, no matter how "theoretical", has a parallel bond graph model which is equally useful.

The above statements are consequences of the theory we develop in this series of papers. In the engineering community the belief is that we are currently faced with a division of modelling problems into two types: those for which either linear graphs or bond graphs are most suitable. This could not be further from the truth.

A few authors have investigated the structure of the bond graph (**8**, **9**, **10**). However, most of this work does not analyse the inherent topological and algebraic structure of the bond graph in isolation from its use in modelling physical systems. In fact, Ort and Martens (**10**) feel that it is not possible to develop a mathematical theory of bond graphs. They suggest that "all procedures... seem to work just the right way." On the contrary, the mathematical theory which we describe here considers a bond graph as a combinatorial object, independent from any application to physical system modelling. We call the diagram which contains the combinatorial information a (*combinatorial*) *bond graph*. When physical information is added to a bond graph the resulting system model will be called a *system bond graph*. This process exactly parallels the formation of a system graph from a linear graph.

Perelson and Oster (**8**) attempt to provide a theoretical basis for bond graph modelling. As they point out, "the lack of a rigorous mathematical theory for bond graphs has probably contributed to their rather slow acceptance by the engineering community" (1976). These authors develop a theory in which bond graphs appear to be a special type of graph which they call an interconnection graph. They do not develop an independent mathematical theory of bond graphs, although they note that it might be possible to do this.

We can use the theory of bond graphs to show that, when applied to physical system models, the two techniques are equivalent. This is a much stronger statement than saying that the techniques give the same results when applied on common ground (i.e. state variable formulation), which has been clear since the invention

of bond graphs. Linear graphs and bond graphs are simply different pictorial diagrams of the same combinatorial object which records the interconnection topology of the system. To use an analogy, expressing a model in one or the other picture is no more difficult than translating text from one language to another. In theory, any formulation method used in one context may equally well be used in the other. There are clearly many possible benefits to be gained by practitioners of either method.

In addition to the practical benefits of cross-fertilization of techniques for system modelling, a thorough understanding of the concept of bond graphs has other advantages. Armed with a rigorous theory, we are in a position to prove exactly what it is possible to do with a bond graph. The authors feel that the full capabilities and power of bond graph modelling are revealed by careful consideration of the mathematical theory of bond graphs.

Another advantage to be gained by considering bond graphs as mathematical structures, is that many of the problems associated with the bond graph notation are of a mathematical nature. For instance, the problems with assignment of causality and orientation, especially in the presence of loops [see, for instance, Perelson (**9**, **11**)], are entirely mathematical. These problems will be confronted in detail in subsequent papers. It is natural to analyse such mathematical problems in a mathematical setting.

We shall show that all results concerning *junction structure matrices* [for instance, Ort and Martens (**10**)] express properties of matrix representations of the algebraic structures used in our theory. However, much of the transparency of the results is lost when matrices are used to express them. It is also interesting to note that bond graphs may be useful to mathematicians as combinatorial diagrams, especially due to their close connection with matroid theory. In spite of establishing many connections between bond graphs and linear graphs, the theory we describe in these papers for bond graphs is parallel to the theory of graphs. We should note that bond graphs can exist whether or not linear graphs exist.

In Part I, we use elementary mathematical concepts from linear algebra and graph theory to develop the basics of an independent theory of *non-directed bond graphs* (meaning no power half-arrows). We use the graph theory as a convenience; the entire theory of bond graphs can be developed without using graph theory in any way. Once we have established the theory it is then possible to prove rigorously, as theorems, the procedures which are commonly applied to system bond graphs. For instance, the "diamond equality" is well-known (**5**), however, without a mathematical theory, it is not even possible to give a precise meaning to the "equivalence" of two bond graphs. The connection with conventional bond graph techniques will not be immediately obvious. Many of these results will appear in subsequent papers in this series. Note that we have used s- and p-junctions rather than the customary 1- and 0-junctions. It should be quite clear why we have used this notation. The use of numerical subscripts, in the algebraic context of our theory, would be very confusing. Also, in a combinatorial setting the s- and p-notation takes on a non-ambiguous meaning.

In Part II (**12**), we expand the theory to include the concept of duality, which appears to be extraordinarily natural in the context of bond graphs. In (**13**) and

Vol. 326, No. 3, pp. 329–350, 1989
Printed in Great Britain

(**14**), we explore the combinatorial structure of bond graphs in terms of matroid theory. The procedure of selecting causality is most transparent when introduced in this setting and we discuss it in great detail in (**14**). We complete the program of the mathematical theory with a paper dealing with orientation for *directed bond graphs* (**15**), which is, of course, essential for the application of bond graphs to physical system modelling.

We begin here by giving a precise definition of a combinatorial bond graph and some other preliminaries in Section II. In Section III, we describe a construction which associates a bond graph with a planar representation of a graph. The algebraic structure of bond graphs in terms of their vector spaces is discussed in Section IV, and in Section V we describe an important class of bond graphs corresponding to those which represent physical systems. In Section VI we present conclusions and a summary of the main results of the paper.

Listed, as an appendix, are definitions of all the elementary concepts we use from graph theory. This should serve as a reference to our use of the graph theoretical terminology, since the definitions are not entirely standard.

II. Definitions and Notation

Definition. A (*combinatorial*) *bond graph*, B, consists of:

(1) a graph, (V, E_i), called the *underlying graph* of B, which has the following properties:
   (i) there are no self-loops,
   (ii) the vertex set, V, has a specified partition into two disjoint sets, P and S.
(2) a set, E_e, and
(3) a relation which associates precisely one element of V with each element of E_e.

Definition. The vertices of the underlying graph of a bond graph, B, are called the *junctions* of B. Vertices in P are called p-junctions and vertices in S are called s-junctions. The set

$$E = E_i \cup E_e$$

is called the *bond set* of B and its elements are called the *bonds* of B. The elements of E_e are called the *external bonds* of B and the elements of E_i are called the *internal bonds* of B. If an external bond is associated with a vertex, v, we say that it is *incident on* v.

A bond graph can be visualized by using a planar diagram of the underlying graph, indicating the vertices by s or p instead of a dot, and attaching extra lines to some of the vertices to show the external bonds. The external bonds are like edges but they have only one vertex. This is the usual picture of a bond graph junction structure (**9**) without bond direction. *We shall always use such a diagram to represent a bond graph.* For simplicity, we call the diagram a *bond graph*. However, we emphasize that this is simply a convenience. The definition of a bond graph is stated in terms of incidence relations and does not rely on any such geometric representation.

Remark 1. The concept of a graph together with extra lines is similar to the definition of a Feynman graph used in quantum field theory (**16**), although there is no connection between the bond graph defined above and a Feynman graph, except for this superficial similarity.

The number of p-junctions of a bond graph B will be denoted by $p(B)$, or simply p if the context is clear. Similarly $s(B)$ or s will denote the number of s-junctions of B. The number of external bonds of a bond graph B will be denoted by $e(B)$ or e and the number of internal bonds will be denoted by $i(B)$ or i. The total number of all the bonds of B will be $n(B) = e(B) + i(B)$ or n.

The bonds of a bond graph will be identified by numbers. We use the integers $1, 2, 3, \ldots, e$ for external bonds and internal bonds will be identified by the numbers $o_1, o_2, \ldots, o_i$.

Definition. Consider a single junction, J, of B. The number of external bonds incident on J is called the *external degree* of J and will be denoted by $\deg_e(J)$. Similarly the *internal degree*, $\deg_i(J)$, is the number of internal bonds incident on J. The *degree* of the junction J is $\deg(J) = \deg_e(J) + \deg_i(J)$. The *internal junctions* of a bond graph B are those junctions J for which $\deg_e(J) = 0$.

Definition. A *simple* bond graph is one which satisfies the following two conditions:

(i) the underlying graph has no parallel edges, and
(ii) there is no isolated junction with one single incident bond.

Junctions of the type referred to in (ii) are called *degenerate junctions*. In this paper and the others in the series we shall consider only simple bond graphs, although we shall indicate the significance of degenerate junctions.

Definition. If the partition of the junctions of B is a bi-partition we shall say that B is a *proper* bond graph. We shall see that it is necessary to consider only proper bond graphs in the topological analysis of non-directed bond graphs.

Definition. If the underlying graph of B contains no cycles and is connected (non-connected) we say that B is a *tree* bond graph (*forest* bond graph). Note that a tree bond graph is necessarily simple but not necessarily proper.

III. Bond Graphs Associated with a Graph

Let G be a planar graph. We describe now a geometric procedure which can be used to construct a bond graph associated with G_p, some planar representation of G. For each bounded face of G_p the collection of its boundary edges is a cycle of G_p. If there are at least three edges in this cycle then we call the corresponding face an s-face of G_p. Otherwise there must be two edges in the cycle and we call the face a p-face of G_p. A p-face corresponds to a pair of parallel edges of G. Note that we are not including the unbounded face in this discussion. By definition an s-face or p-face is one of the bounded faces of G_p.

Let b be an edge of G_p which lies in the boundary of two bounded faces of G_p. Now add a minimal number of extra edges to G_p, parallel to b, so that b no longer lies on the boundary of an s-face. If the two faces are both s-faces this will require

two new edges (the new edges 'enclose' b). If the two faces are an s-face and a p-face we only need one new edge to isolate b from the s-face. If the two faces are both p-faces then we need not add any edge to G_p. When we have considered all such edges b, we will have constructed a planar representation which we denote by G_{pa}.

It is clear that G_{pa} is unique and has the property that the boundary of any s-face of G_{pa} consists only of:

(i) edges which are not edges of G_p, or
(ii) edges on the boundary of G_p.

This is the point behind the construction of G_{pa}. Also note that G_{pa} is a planar representation of a graph which has G as a subgraph.

Definition. The graph for which G_{pa} is a planar representation will be called the *augmented graph* of G with respect to G_p. For simplicity, we shall refer to G_{pa} as the augmented graph of G_p.

It will be useful to define two numbers associated with a planar representation G_p. We shall denote by $s(G_p)$ the number of s-faces of G_p. The number of maximal sets of mutually parallel edges of G_p will be denoted by $p(G_p)$. For the augmented graph of G_p we have $s(G_{pa}) = s(G_p)$, since the new edges were only added parallel to edges of G_p and so cannot introduce new s-faces. However, we must have $p(G_{pa}) \geqslant p(G_p)$, since, in general, the new edges produce new p-faces.

The graph G_{pa} will now be decomposed into $s(G_{pa})+p(G_{pa})$ subgraphs. An s-subgraph of G_{pa} is a subgraph consisting of those edges of G_{pa} which form the boundary of exactly one s-face of G_{pa}. A p-subgraph of G_{pa} is a subgraph consisting of exactly one maximal set of mutually parallel edges of G_{pa}. Clearly, there will be $s(G_{pa})$ of the first type and $p(G_{pa})$ of the second type of subgraph.

For each of the s- and p-subgraphs of G_{pa} we define a single-junction bond graph whose bonds are in one to one correspondence with the edges of the subgraph. For a p-subgraph we define a single p-junction and add one bond for each parallel edge in the subgraph. For an s-subgraph we define an s-junction and add one bond for each edge in the subgraph. This procedure defines a collection of single-junction bond graphs. Finally we piece together these single-junction bond graphs by identifying any bonds which have the same label. The resulting bond graph will be denoted by $B(G_p)$.

Definition. The procedure described above, used to construct $B(G_p)$, will be called the *cut and paste method.* $B(G_p)$ will be called the bond graph *associated with* the planar representation G_p.

Remark 2. An internal s-junction of $B(G_p)$ corresponds to an s-face of G_{pa} whose boundary has no edge on the boundary of G_p.

Remark 3. The cut and paste method shows the origin of the letters used to label the junctions of a bond graph. An s-junction corresponds to a collection of edges arranged in *series* to form a cycle of G_{pa}. A p-junction corresponds to a collection of edges of G_{pa}, all of which are in *parallel.*

Theorem I

Let G_p be a planar representation of a graph G. The bond graph $B(G_p)$ constructed using the cut and paste method is a proper bond graph. There are $s(G_{pa})$

s-junctions and $p(G_{pa})$ p-junctions in $B(G_p)$. The internal bonds of $B(G_p)$ are precisely the new edges which were added to G_p to produce the augmented graph G_{pa}. The external bonds of $B(G_p)$ are the edges of G_p. (These last two statements are to be interpreted in the sense of a one to one correspondence between bonds and edges identified by their numbers.)

Proof: Because of the way in which the extra edges are added to G_p the only edges which can be common to two subgraphs are the added edges. In particular each added edge is common to precisely two subgraphs. Hence each added edge corresponds to a label which appears on precisely two single-junction bond graphs. When bonds with the same label are identified a bond will join two junctions if it corresponds to an added edge—that is, each added edge corresponds to an internal bond of $B(G_p)$. Conversely, edges of G_p can be in precisely one subgraph. Hence these edges will correspond to bonds which are incident on only one junction in $B(G_p)$—that is, they will be external bonds.

It is obvious from the construction that no two adjacent faces of G_{pa} are of the same type. Hence adjacent junctions in the associated bond graph must have different letters, which means that $B(G_p)$ is a proper bond graph.

Theorem II

Let G_p be a planar representation of a graph G. The number of parts of the underlying graph of $B(G_p)$ is precisely the number of components of G. In particular, the underlying graph of $B(G_p)$ is connected if and only if G has precisely one component (i.e. G is connected and has no cut-vertex).

Proof: if the cut and paste method is applied to a separable graph, G, each component of G will correspond to a different part of the underlying graph of $B(G_p)$. By definition, the underlying graph of $B(G_p)$ is connected if it has one part. This occurs if and only if G has exactly one component. Equivalently, G must be connected and have no cut-vertex.

Example 1. In Fig. 1(a), we show a planar representation, G_p, which consists of edges numbered 1 to 7. We illustrate the cut and paste method by constructing the augmented graph, G_{pa}, in Fig. 1(b); the s- and p-subgraphs in Fig. 1(c); and the bond graph, $B(G_p)$, in Fig. 1(d).

The cut and paste method provides an explicit interpretation of the internal bonds in terms of the planar representation of the graph. In particular, these internal bonds have the function of coupling together the component subgraphs of G_{pa}. This observation is the motivation for the subsequent algebraic analysis of bond graphs. If we use a different planar representation of the graph the "face structure" of the representation will be different, which will necessitate a different coupling. In general, the bond graph will be different for each planar representation of G. We illustrate these comments in the following example.

Example 2. In Fig. 2(a), we show two topologically distinct planar representations, G_p and G'_p, of the same graph. We construct the augmented graphs for both G_p and G'_p in Fig. 2(b) and the associated bond graphs in Fig. 2(c).

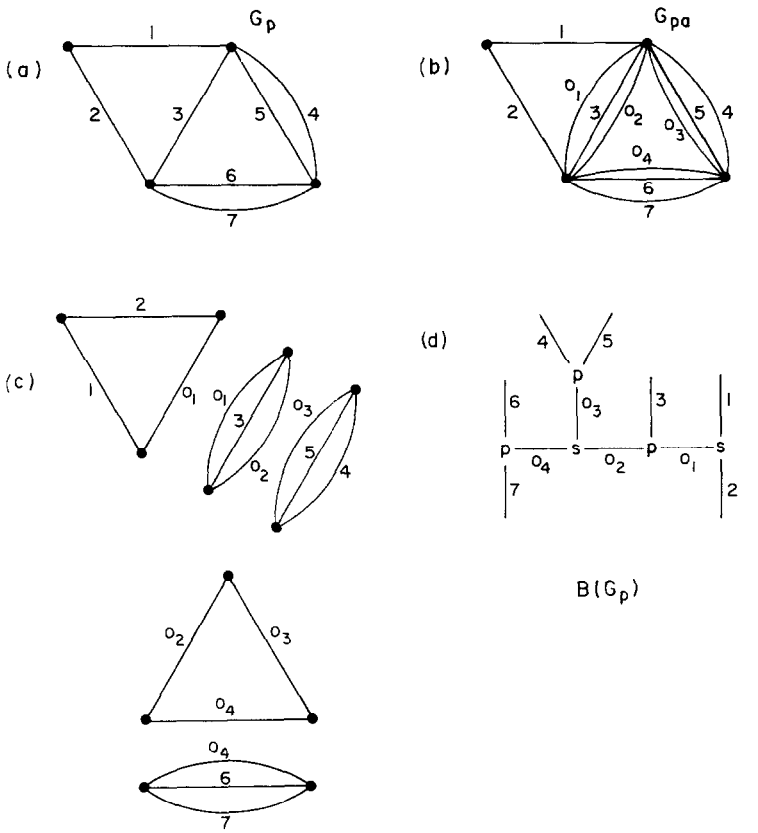

FIG. 1. Cut and paste method. (a) G_p: planar representation of a graph; (b) G_{pa}: augmented graph of G_p; (c) s- and p-subgraphs from G_{pa}; (d) $B(G_p)$: bond graph associated with G_p.

The two bond graphs constructed in Example 2 are quite different even though they are associated with planar representations of the same graph. In general, each topologically distinct planar representation of G will produce a different bond graph. Hence if we are to establish a correspondence between bond graphs and graphs, it is necessary to define an equivalence relation on the class of all bond graphs. Then we can associate an equivalence class of bond graphs with each graph. We shall see in the next section that the correct equivalence relation is defined using algebraic invariants.

The cut and paste method can only be applied to associate bond graphs with a planar graph, via its planar representations. Therefore, we have yet to show how to associate a bond graph with a non-planar graph, for which the cut and paste method does not apply. This can be accomplished using the same algebraic structures used to solve the equivalence problem mentioned above.

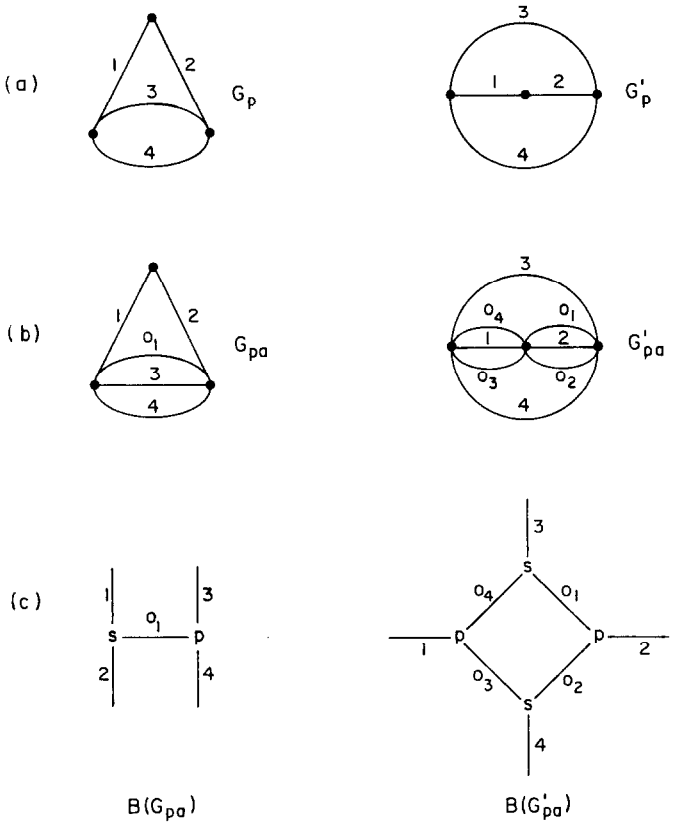

FIG. 2. Cut and paste method. (a) G_p and G'_p: planar representations of a graph; (b) G_{pa} and G'_{pa}: augmented graphs of G_p and G'_p; (c) $B(G_p)$ and $B(G'_p)$: bond graphs associated with G_p and G'_p.

IV. The s*-space of a Bond Graph*

We begin the analysis of the algebraic structure of a bond graph by defining a vector space associated with it. Let B be a bond graph. Consider the collection of all the subsets of E, the bond set of B, including the empty set which we shall denote by $\varnothing$. We shall denote this collection by $W(B)$. $W(B)$ can be given a vector space structure over the field $GF(2)$, the field of integers *modulo* 2. For any two subsets E_1 and E_2 of E we define the sum, E_1+E_2, by

$$E_1+E_2 = (E_1 \cup E_2)-(E_1 \cap E_2),$$

the *symmetric difference* of the two sets. The additive inverse of a subset E_1 of E is E_1, since $E_1+E_1 = (E_1 \cup E_1)-(E_1 \cap E_1) = \varnothing$, which is the additive identity. We also define $1 \cdot E_1$ to be E_1 and $0 \cdot E_1$ to be $\varnothing$. Using these definitions it is elementary to check that $W(B)$ is a vector space over $GF(2)$. In fact, this is the usual definition

of a vector space of the collection of subsets of a given set. We summarize the standard results, applied to our structures, in the next theorem.

Theorem III

For a bond graph B with n bonds, $W(B)$ is an n-dimensional vector space over $GF(2)$ with the sum and scalar multiplication defined above. If $E = \{b_1, b_2, \ldots, b_n\}$ then a basis for $W(B)$ is

$$\{\{b_1\}, \{b_2\}, \ldots, \{b_n\}\}.$$

$W(B)$ is a finite vector space with 2^n vectors.

Proof: The results follow from the standard procedure for defining a vector space on the collection of subsets of an arbitrary set. See, for instance, Swamy and Thulasiraman (**17**). The vectors in $W(B)$ are simply the subsets of E, since the field has only 0 and 1. There are 2^n of these, including the empty set, $\varnothing$.

Definition. We call the vector space $W(B)$ the *bond space* of the bond graph B.

It should be emphasized that all the bonds of B are included in the definition of $W(B)$, both internal and external bonds.

We shall adopt a simplified notation for writing sets of vectors in $W(B)$, since it is typographically awkward to notate sets of sets. A set of bonds (i.e. a subset of E) will be denoted by juxtaposition. Thus the set $\{b_1, b_2, \ldots, b_k\}$ will be denoted by $b_1 b_2 \ldots b_k$. Following our practice of labelling the external bonds of B with integers, this means, in particular, that a subset of external bonds, $\{1, 2, 3, \ldots, k\}$, will be denoted by $123 \ldots k$. Also, note that $\{b\}$, a singleton set with the bond b as its only element, will be denoted by b. It should be quite clear from the context whether 'b' means a bond b or a set $\{b\}$.

The juxtaposition notation is very convenient and should cause no confusion. In this notation, the vector space sum may be calculated quite simply by juxtaposing the two vectors and simplifying using the rules:

$$b_1 b_2 = b_2 b_1 \quad \text{and} \quad b_1 b_1 = \varnothing.$$

For example, the sum $12457 + 1268$ may be calculated by

$$\begin{aligned} 12457 + 1268 &= 124571268 \\ &= 112245678 \\ &= 45678. \end{aligned}$$

Here '+' is the vector space sum (symmetric difference). It is easily checked that the new notation gives results identical to the customary notation: $\{1, 2, 4, 5, 7\} + \{1, 2, 6, 8\} = \{4, 5, 6, 7, 8\}$. We shall usually use the juxtaposition notation, because the set notation is so cumbersome [see, for instance Gould (**18**)].

Definition. Consider the collection of all vectors in $W(B)$ which consist of either:

(i) all the bonds incident on one s-junction of B, or
(ii) any two bonds incident on the same p-junction of B.

A vector which is of either of these two types will be called an *elementary junction vector*. An elementary junction vector of type (i) will be called an s-type vector.

An elementary junction vector of type (ii) will be called a p-type vector. The linear span of all the elementary junction vectors of $W(B)$ is a subspace of $W(B)$, called the s-space of B, and denoted by $W_s(B)$. We shall often use the notation W and W_s, provided the bond graph B is clear from the context.

In general, bond graphs associated with different planar representations of a graph do not have isomorphic s-spaces, as the following example illustrates.

Example 3. Let us denote the bond graphs of Example 2, $B(G_p)$ and $B(G_p')$, by B_1 and B_2 respectively (see Fig. 2). The bond spaces $W(B_1)$ and $W(B_2)$ have dimension 5 and 8 respectively. The s-space $W_s(B_1)$ is spanned by the elementary junction vectors:

$$12o_1, 3o_1, 34.$$

The s-space $W_s(B_2)$ is spanned by the elementary junction vectors:

$$1o_3, o_3o_4, 2o_1, o_1o_2, 3o_1o_4, 4o_2o_3.$$

In fact, it is easy to show that these are each linearly independent sets of elementary junction vectors and therefore they are bases for $W_s(B_1)$ and $W_s(B_2)$ respectively. Thus $W_s(B_1)$ has dimension 3 and $W_s(B_2)$ has dimension 6. Plainly, the two spaces cannot be isomorphic since they do not even have the same dimension.

Remark 4. In general, the set of all elementary junction vectors of B will be linearly dependent. This linear dependence can occur due to the presence of a cycle in the underlying graph of B or simply because some of the vectors on a p-junction are redundant. To obtain a basis for $W_s(B)$, we must select a linearly independent set of elementary junction vectors of B. Usually it is a simple procedure to make such a choice by inspection of the bond graph.

Suppose that a bond graph B is associated with one of the planar representations of a graph G. The s-space of B is "too large", in its entirety, to isolate the structure of the graph G. As demonstrated in Examples 2 and 3, the s-space is intimately connected with the particular planar representation, G_p, which is chosen for G. This space includes information about the internal edges which were required to construct B from G_p. Nevertheless, we can consider a decomposition of the s-space which eliminates this extra information. This decomposition will provide us with the required algebraic invariant which does not depend on the planar representation chosen.

Definition. The following sets of vectors are closed under addition and are therefore subspaces of $W_s(B)$:

$$W_{cy}(B) = \{v \in W_s(B) \mid v \subset E_e\}$$

$$W_i(B) = \{v \in W_s(B) \mid v \subset E_i\}.$$

The subspace $W_{cy}(B)$ is called the *cycle space* and the subspace $W_i(B)$ is called the *internal space* of B. We shall often omit the reference to B in the notation, using for instance W_{cy} instead of $W_{cy}(B)$ if the B intended is clear.

Now, we summarize a few results which we shall use from linear algebra. If V is a vector space and T and U are subspaces of V, the *sum* of T and U, denoted by $T+U$, is the set of all vectors in V of type $t+u$, where $t \in T$ and $u \in U$. The sum of

two subspaces is a subspace and if T and U have only the zero vector in common, the sum is called the *direct sum* of T and U. In this case, we use the notation $T \oplus U$. The dimension of $T \oplus U$ is the sum of the dimensions of T and U. If $V = T \oplus U$ we say that T and U are complementary subspaces of V. In this case, any vector in V, can be expressed *uniquely* in the form $t+u$, where $t \in T$ and $u \in U$. The reader who is unfamiliar with this aspect of linear algebra may consult, say, Hoffman and Kunze (**19**).

The cycle space contains all vectors in W_s which consist only of external bonds and the internal space contains all the vectors in W_s which consist only of internal bonds. Thus, by definition, the only vector common to both the cycle and internal spaces is $\varnothing$, the empty set [zero vector in $W(B)$]. Hence the sum of W_{cy} and W_i is a direct sum of subspaces of W_s.

Definition. Let W_c be any subspace of W_s which is complementary to $W_{cy} \oplus W_i$. This means that W_s has the direct sum decomposition:

$$W_s = W_{cy} \oplus W_i \oplus W_c.$$

The space W_c is called a *coupling space* for the bond graph B. Note that, in general, a coupling space will not be unique. Any subspace of W_s with the above property will be called a coupling space.

Definition. We say that the bond graphs B_1 and B_2 are s-equivalent if $W_{cy}(B_1)$ is isomorphic to $W_{cy}(B_2)$. We shall write $B_1 \triangleq B_2$ if B_1 is s-equivalent to B_2.

It is sometimes convenient for calculations to be able to use n-tuples to represent vectors in $W(B)$. In order to do this we choose an ordered basis for $W(B)$:

$$\{1, 2, \ldots, e, o_1, o_2, \ldots, o_i\}.$$

Note that this basis is a set of $n = e+i$ singleton sets of bonds, since in the juxtaposition notation the symbol '1' for instance, represents the set $\{1\}$. With this fixed choice of basis we can represent vectors in $W(B)$ by n-tuples of vectors with 0 and 1 entries. Whenever we use n-tuples to represent vectors in $W(B)$ we shall assume that this is done with respect to the ordered basis above. We shall call this basis the *standard basis* of $W(B)$.

Example 4. We continue the analysis of the bond graphs B_1 and B_2 used in Examples 2 and 3. The s-space of B_1 has the following decomposition:

$$W_{cy}(B_1) = \{123, 124, 34, \varnothing\}$$
$$W_i(B_1) = \{\varnothing\}$$
$$W_c(B_1) = \{3o_1, \varnothing\}.$$

The cycle span of B_1 has dimension 2, the internal space has dimension 0 and so the coupling space necessarily has dimension 1. As noted above the coupling space is not unique. We could have chosen, say, $\{4o_1, \varnothing\}$ instead.

Considering bond graph B_2, the s-space has the decomposition:

$$W_{cy}(B_2) = \{123, 124, 34, \varnothing\}$$
$$W_i(B_2) = \{o_1o_2, o_3o_4, o_1o_2o_3o_4, \varnothing\}$$

$$W_c(B_2) = \{o_2 2, o_3 1, 12 o_2 o_3, \varnothing\}.$$

Any vector in W_s can be written uniquely as a linear combination of three vectors, one from each of the subspaces above. For instance the vector $x = o_2 o_4 4$ is in $W_s(B_2)$ since it is a linear combination of elementary junction vectors:

$$x = 4 o_2 o_3 + o_3 o_4.$$

The decomposition of x illustrating the direct sum decomposition is shown below:

$$x = 124 + o_3 o_4 + 12 o_2 o_3.$$

The dimension of $W_s(B_2)$ is 6. W_{cy} and W_i each have dimension 2 and so W_c must also have dimension 2. A basis for any one of the subspaces consists of the first two vectors in the list. Note that the choice we make for the coupling space is one of several possibilities.

Note that $W(B_2)$ has dimension 8 and so there are vectors in W which are not in W_s, for instance the vector 134.

The s- and p-type elementary junction vectors span W_s but these are not necessarily in any one of the subspaces above. However, it is always possible to express these vectors uniquely as a linear combination of three vectors, one in each of the subspaces. For example, we show below the appropriate decompositions for some elementary junction vectors:

$$\begin{aligned} 3 o_1 o_4 &= 123 + o_1 o_2 o_3 o_4 + 12 o_2 o_3 \\ o_3 o_4 &= \varnothing + o_3 o_4 + \varnothing \\ 4 o_2 o_3 &= 124 + \varnothing + 12 o_2 o_3. \end{aligned}$$

It is evident from the decompositions above that the two cycle spaces are identical and as we shall see it is this information which isolates the graph G from its various different planar representations G_p and their associated bond graphs $B(G_p)$.

If we use the standard basis and represent vectors with n-tuples of 0 and 1 entries we can arrange basis vectors for W_s as the rows of a matrix. Then we can show the decomposition by blocks in the matrix. Of course, we have to choose appropriate basis vectors for W_s so that the decomposition is apparent. We show first a marix whose rows are basis vectors for $W_s(B_1)$. The column in the matrix below correspond to the bonds 1, 2, 3, 4, o_1:

$$\begin{array}{c} W_{cy} \\ \\ W_c \end{array} \left[\begin{array}{cccc|c} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 1 \end{array}\right].$$

The matrix below is the corresponding matrix for $W_s(B_2)$. Here the columns correspond to the bonds, 1, 2, 3, 4, o_1, o_2, o_3, o_4:

$$
\begin{array}{c}
W_{cy} \\
\\
\hline
W_i \\
\\
\hline
W_c \\
\\
\end{array}
\left[\begin{array}{cccc|cccc}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right].
$$

In either of the s-space representations the cycle space has the first two rows of the matrix as a basis. The two cycle spaces are evidently isomorphic.

We can see that the decomposition above splits G_{pa} into two graphs, an external and an internal graph, corresponding to the cycle and internal spaces. The coupling space provides the information required to assemble the two graphs into G_{pa}. This is the reason that $W_s(B_1)$ has a null internal space, since the corresponding G_{pa} has only one internal bond and hence there can be no cycle involving internal edges alone. These coupled external and internal graphs are discussed further at the end of this section.

Example 4 illustrates a general result which we state in the next theorem. Before we state the theorem we describe how to define a vector space of a graph, called the *cycle space*. It is via this vector space that we obtain an algebraic association between a graph and a bond graph, which does not rely on any geometric representation.

Consider the collection of all the subsets of the edge set of a graph G. We can define a vector space structure on this collection of subsets in precisely the same way we define $W(B)$ for a bond graph B. This vector space over $GF(2)$ will be denoted by $W(G)$ and we shall use the juxtaposition notation for subsets of edges of G, just as we do for the sets of bonds. The collection of all cycles and edge-disjoint unions of cycles of G, denoted by $W_{cy}(G)$, is a subspace of $W(G)$, called the *cycle space* of G. These definitions are standard in graph theory [see, for example, Swamy and Thulasiraman (**17**), but note that these authors call the cycle space the *circuit space* of G].

Theorem IV

Let G_p be a planar representation of a graph G and let $B = B(G_p)$, the bond graph associated with G_p. The cycle space of B is isomorphic to the cycle space of G.

Proof: If we identify bonds in B with edges in G_{pa}, then we have an isomorphism between $W_{cy}(G_{pa})$ and $W_s(B)$. This follows by comparison of the definitions of the two spaces. Now restriction to the external edges in $W_{cy}(G_{pa})$ or equivalently to the external bonds in $W_s(B)$, will produce isomorphic subspaces. The first will be $W_{cy}(G)$ and the second will be $W_{cy}(B)$. Hence the two subspaces are isomorphic.

Corollary. All bond graphs associated with planar representations of a graph G are s-equivalent.

Definition. The relation of s-equivalence on the class of bond graphs is an equivalence relation. We shall denote by $\mathscr{B}(G)$ the equivalence class of all bond

graphs whose cycle space is isomorphic to the cycle space of the graph G. If a bond graph B is in $\mathscr{B}(G)$ then we say that the bond graph B and the graph G are *associated.*

With each graph G we can now associate a class of s-equivalent bond graphs $\mathscr{B}(G)$. Theorem IV establishes that if G is a planar graph then, for each planar representation G_p of G, the bond graph obtained using the cut and paste method, $B(G_p)$, is in $\mathscr{B}(G)$. However not every bond graph in $\mathscr{B}(G)$ can be obtained in this manner. We shall discuss this in the next section.

Remark 5. Suppose that B is a bond graph in $\mathscr{B}(G)$. The information which is encoded in B enables us to construct two separate graphs. The first, the *external graph* G_e, is the graph whose cycle space is isomorphic to $W_{cy}(B)$. The second, the *internal graph* G_i, is the graph whose cycle space is isomorphic to $W_i(B)$. In addition we can construct a coupling graph, G_c, using one of the coupling spaces of B. The coupling space provides the information required to associate the vertices of the external and internal graphs and hence couple them together to produce an augmented graph. It is the external graph which we are associating with the bond graph. For some bond graphs it may be combinatorially impossible to construct one of the graphs above and/or couple them together: see the next section, in particular Example 6, and also comment (v) regarding the inverse cut and paste method.

Example 5. Consider the bond graph B shown in Fig. 3(a). A basis for the s-space of B, has the following decomposition:

$$W_{cy}: \ 123, 15, 26, 34$$

$$W_i: \ o_7o_8o_9, o_5o_7, o_6o_7, o_1o_8, o_2o_8, o_3o_9, o_4o_9$$

$$W_c: \ 1o_5, 2o_1.$$

The vectors given above are bases for the subspaces and so the cycle space is 4-dimensional, the internal space is 7-dimensional and the coupling space is 2-dimensional. This agrees with the dimension of W_s which is seen to be 13, since there are 13 independent elementary junction vectors which span W_s. In Fig. 3(b), we show the external graph (which, of course, is also the graph with which B is associated) and the internal graph, together with a coupling graph. When these are assembled by identifying common vertices we construct the augmented graph G_{pa} which is shown in Fig. 3(c).

Remark 6. A degenerate s-junction is associated with the graph which consists of a single vertex with one self-loop. A degenerate p-junction is associated with the graph which consists of a single edge joining two vertices.

V. Graphic Bond Graphs

In this section, we investigate the general problem of associating graphs with bond graphs. This concept is central to the investigation of the applications of bond graphs to physical system modelling and the comparison with the linear graph approach.

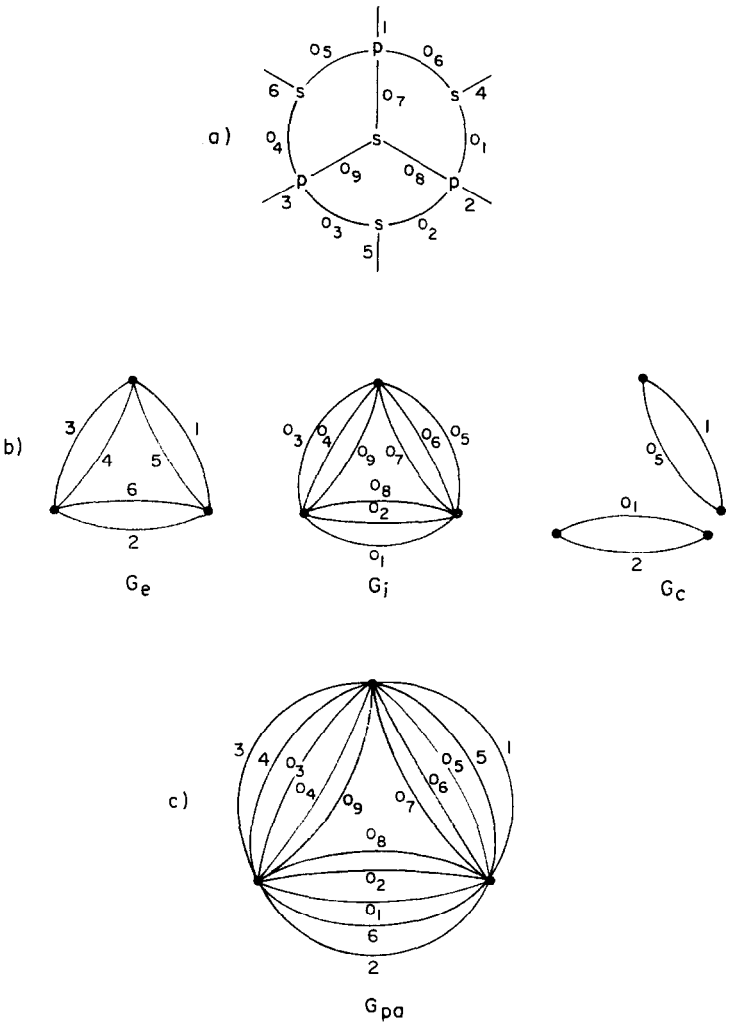

FIG. 3. Coupling of external and internal graphs. (a) Bond graph; (b) external, internal and coupling graphs; (c) augmented graph.

Definition. A bond graph B is *graphic* if there exists a graph G so that $B \in \mathscr{B}(G)$. Equivalently, a bond graph B is graphic if there exists a graph G which is associated with B.

As we have seen in Section III, the cut and paste method is completely satisfactory when we wish to produce a bond graph associated with a planar graph. However, if we try to reverse the procedure by producing a graph asssociated with a given bond graph, we find the technique is somewhat more difficult to apply and in some cases it is, in fact, impossible to construct any graph G associated with B (see Example 6, below, for a non-graphic bond graph).

Before describing the method, we give a useful definition of graph theory. A

graph G is said to be a *restriction* of a graph H if G can be obtained from H by deleting edges. If G is a restriction of H we also say that H is an *augmented graph* of G. We use the notation G^a for an augmented graph of a graph G. If G_p is planar representation of a graph G, then the augmented graph, G_{pa}, constructed when applying the cut and paste method to G_p, is a planar representation of an augmented graph G^a in the sense defined above.

Consider a bond graph B. For each junction of B we form a single junction bond graph with the same label and with the corresponding bonds attached. For each s-junction we form an s-subgraph whose edges correspond to the bonds on the junction and are arranged to form a cycle. For each p-junction we form a p-subgraph whose edges correspond to the bonds on the junction and are arranged as a mutually parallel collection of edges. These subgraphs are to be drawn as planar diagrams. If it is possible to assemble the subgraphs by identifying edges with the same label, the resulting diagram will be a planar representation of a graph, G^a. Denote by G the graph obtained from G^a by deleting all edges corresponding to internal bonds. Thus G^a is an augmented graph of G, in the sense defined above.

Definition. We shall call the procedure described above, the *inverse* cut and paste method.

Remark 7. It is necessary to distinguish between the method and the inverse method, since the method is always successful (for planar graphs), but, as we shall see, the inverse method is not.

Remark 8. In general, an augmented graph, G^a, obtained from a bond graph using the inverse cut and paste method, will not be unique. If a cycle has three or more edges, a re-arrangement of them will produce a graph with the same cycle but different vertex connections, provided the rearrangement is still consistent with the s- and p-subgraph information. The two graphs will have the same cycle space, but different vertex connections. There is also a non-uniqueness associated with separable graphs. This is discussed after the following theorem.

Theorem V

Let G^a be an augmented graph obtained from a (graphic) bond graph B, using the inverse cut and paste method. Let G be the restriction of G^a to edges corresponding to external bonds of B. Then G and B are associated.

Proof: It is clear from the construction that $W_s(B)$ is isomorphic to $W_{cy}(G^a)$. Under this isomorphism, restriction to G corresponds to the subspaces $W_{cy}(B)$ and $W_{cy}(G)$, which are therefore isomorphic. By definition the graph G and the bond graph B are associated.

The question of separable graphs is interesting, especially for the application of this theory to physical system modelling. When applying the inverse cut and paste method to a bond graph it is required to construct the graph G^a, having a given collection of cycles. Now if the graph G^a has two components, it is equally correct to join these two as one part with a cut-vertex or to separate the components as separate parts of the graph. Either of these two graphs would be consistent with the cycle information. Thus, the bond graph cannot distinguish between a cut-vertex and a separation into two disjoint parts of G^a.

Now, provided there are sufficient internal bonds, it is quite possible for the augmented graph to have one component, while the external graph, G_e, has more than one component. In this case, the external graph has a definite cut-vertex which is precisely determined by the construction of G^a. It is clear that the external graph, G_e, associated with a bond graph, has a precisely determined cut-vertex, v_c, if and only if the internal graph, G_i, has a cycle which includes the vertex v_c. Thus, it is possible for a graph with a definite cut-vertex to be associated with a bond graph. Note, however, that any other graph, G, with the same cycle space as G_e is also associated with the bond graph. In particular, G may be the same graph as G_e but with the cut-vertex disconnected. There is no requirement that an associated graph be constructed by any specific method.

To be quite definite, we shall adopt the convention that, when we construct a graph associated with a bond graph, we shall always separate distinct components into separate parts. In particular, with this convention, if we obtain an augmented graph using the inverse cut and paste method and this has two components, then the associated graph will also have two parts rather than a cut-vertex.

Remark 9. In systems modelling a topological disconnection between two parts of a system appears in a system graph as a separation into two parts. The system bond graph which is associated with this system graph, will have two topologically separate parts, meaning that they are coupled only via a physical coupling such as a *TF*. Such a system bond graph is topologically disconnected, just as is its associated system graph. In the theory which we provide here, we discuss the topological structure of bond graphs. The only system bond graph multiports which provide topological coupling are the *s*- and *p*-junctions. Any other elements, such as *TF* and *GY*, provide physical coupling which has no place in the topological theory. The distinction between these two types of coupling is not clear in the literature [see, for instance, the discussion of weighted junction structures in Perelson (**9**)].

Remark 10. A procedure similar to the inverse cut and paste method has been discussed in Ort and Martens (**10**), although in the context of physical systems theory. Our theory, developed herein, formalizes the technique for non-directed bond-graphs. In the context of physical systems modelling, orientation is required, and this is included in Ort and Martens. However, by including orientation the essential topological procedure is obscured. We prefer to consider the orientation of bond graphs separately and, as noted in the introduction, we develop this in its entirety in the final paper in this series. Ort and Martens give an example of a bond graph for which the technique fails to produce a graph. It is interesting to note that the technique fails due to an orientation problem and not for topological reasons. We give an example below of a bond graph for which the inverse method fails, regardless of any orientation (or causality) problems.

Example 6. Consider the bond graph shown in Fig. 4. Application of the inverse cut and paste method to this bond graph produces an inconsistent collection of *s*- and *p*-subgraphs. It is impossible to construct an augmented graph from these subgraphs, by identifying internal edges. This is impossible regardless of any re-

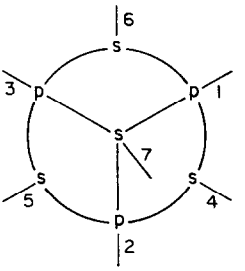

FIG. 4. A non-graphic bond graph.

arrangement of the edges in the subgraphs. Thus the inverse cut and paste method fails to produce a graph associated with this bond graph.

The problem of constructing a graph associated with this bond graph is actually deeper than the failure of the inverse cut and paste method. We can show that this bond graph is non-graphic by considering its cycle space. The cycle space is 4-dimensional and a basis is {1237, 136, 124, 235}. This can be checked by writing an elementary junction vector for each junction and finding a maximal linearly independent set of vectors consisting only of external bonds. It is well-known in graph theory that no graph with 7 edges exists with these four cycles. Thus we see that B is non-graphic.

As a practical method for producing a graph associated with a given bond graph, the inverse cut and paste method has some limitations:

(i) It is not always clear whether it is possible to assemble the subgraphs to form a planar representation of G^a and thus construct the required planar diagram of G;
(ii) If the subgraphs chosen cannot be assembled to form an augmented graph, G^a, it may be possible to rearrange the edges in the s-subgraphs with three or more edges and produce a collection of subgraphs which can be assembled (see Remark 8);
(iii) The method explicitly uses planar representations and so it fails to produce any non-planar graph associated with B;
(iv) The method does not provide an *explicit* indication when a particular bond graph is non-graphic;
(v) It is possible for the method to fail even in the situation where a bond graph is graphic and there is a planar graph associated with it. This can occur when the internal and external graphs both exist, but they cannot be coupled together using the coupling space. Thus the method fails,even though the external graph is clearly a graph associated with the bond graph. We provide an example of such a case in (**13**).

In spite of these difficulties the inverse cut and paste method is often quite useful.

Remark 11. The problem of determining whether or not a given bond graph is graphic is a highly non-trivial combinatorial problem. It is clear that the bond graph concept is much more general than simply a modelling device for physical

systems. The combinatorial bond graph seems to be an extremely interesting mathematical object. Perelson and Oster (**8**) remark that bond graphs might have some connection with matroid theory. We explore the connections with the theory of matroids in (**13**) and (**14**).

Remark 12. The bond graphs used in applications to physical system modelling are graphic bond graphs, since otherwise they would correspond to a physical system which had no graphical representation. In the bond graph literature, the physical system is invariably the starting point and hence the relevant bond graph will be graphic. (These remarks occasionally depend on the choice of physical analogy, but a complete discussion of this point is deferred to (**12**), since it requires the concept of duality.) Nevertheless, the problem of determining whether or not a given bond graph is graphic might have applications to the synthesis of a physical system with a given bond graph structure.

VI. Conclusions

We have outlined several reasons why it is useful to have an abstract theory of bond graphs on which to base system bond graph modelling techniques. Combinatorial bond graphs can be used as bond graph junction structures in forming system bond graphs, analogous to the formation of system graphs from linear graphs, in system graph modelling.

The definition of a bond graph is given and its structure as a combinatorial object is studied. This is accomplished by constructing a vector space, called the bond space of the bond graph. Topological information is encoded in the bond graph by defining a subspace of the bond space, the s-space. Finally, we eliminate the internal bond information, by studying subspaces of the s-space, the cycle and internal space.

We have given a precise meaning to the association between bond graphs and linear graphs by comparing their respective cycle spaces. A technique, called the cut and paste method, has been described which produces a bond graph associated with a given graph, via its planar representations. Some problems associated with the inverse procedure have been studied.

References

(**1**) H. M. Trent, "Isomorphisms between oriented linear graphs and lumped parameter physical systems", *J. Acoust. Soc. Amer.*, Vol. 27, pp. 500–527, 1955.

(**2**) H. E. Koenig and W. A. Blackwell, "Electromechanical System Theory", McGraw-Hill, New York, 1961.

(**3**) H. E. Koenig, Y. Tokad and H. K. Kesavan, "Analysis of Discrete Physical Systems", McGraw-Hill, New York, 1967.

(**4**) H. M. Paynter, "Analysis and Design of Engineering Systems", M.I.T. Press, Cambridge, MA, 1961.

(**5**) R. C. Rosenberg and D. C. Karnopp, "Analysis and Simulation of Multiport Systems—The Bond Graph Approach to Physical System Dynamics", M.I.T. Press, Cambridge, MA, 1968.

(**6**) R. C. Rosenberg and D. C. Karnopp, "Introduction to Physical System Dynamics", McGraw-Hill, New York, 1983.

(7) M. R. Spiegel, "Applied Differential Equations", Prentice-Hall, Englewood Cliffs, NJ, 1981.
(8) A. S. Perelson and G. F. Oster, "Bond graphs and linear graphs", *J. Franklin Inst.*, Vol. 302, pp. 159–185, 1976.
(9) A. S. Perelson, "Bond graph junction structures", *Trans. ASME J. Dynamic Syst. Meas. Cont.*, Vol. 97, pp. 189–195, 1975.
(10) J. R. Ort and H. R. Martens, "A topological procedure for converting a bond graph to a linear graph", *Trans. ASME J. Dyn. Syst. Meas. Cont.*, Vol. 96, pp. 307–314, 1974.
(11) A. S. Perelson, "Bond graph sign conventions", *Trans. ASME J. Dyn. Syst. Meas. Cont.*, Vol. 97, pp. 184–188, 1975.
(12) S. H. Birkett and P. H. Roe, "The mathematical foundation of bond graphs II—Duality", *J. Franklin Inst.*, in press. 1988.
(13) S. H. Birkett and P. H. Roe, "The mathematical foundations of bond graphs III—Matroid theory".
(14) S. H. Birkett and P. H. Roe, "The mathematical foundations of bond graphs IV—Matrix representations and causality".
(15) S. H. Birkett and P. H. Roe, "The mathematical foundations of bond graphs V—Orientation", forthcoming.
(16) J. C. Taylor, "Gauge Theories of Weak Interactions", Cambridge University Press, New York, 1976.
(17) M. Swamy and K. Thulasiraman, "Graphs, Network and Algorithms", Wiley-Interscience, New York, 1981.
(18) R. Gould, "Graphs and vector spaces", *J. Math. Phys.*, Vol. 38, pp. 193–214, 1958.
(19) K. Hoffman and R. Kunze, "Linear Algebra", Prentice Hall, Englewood Cliffs, NJ, 1971.
(20) W. T. Tutte, "Graph Theory", Encyclopedia of Mathematics and its Applications, Vol. 21, Addison-Wesley, Reading, MA, 1984.

Appendix: Definitions from Graph Theory

We have used some concepts from graph theory in the construction of the theory of bond graphs. For completeness and to serve as a reference we have included this appendix summarizing the definitions of the terminology. The reader who is unfamiliar with graph theory may consult, say, Tutte (**20**) for expansion of the relevant theory.

A *graph* $G(V, E)$ is a pair of sets V and E, together with a relation which associates two (not necessarily distinct) elements of V with each element of E. V is called the *vertex set* and its elements are called the *vertices* of G. E is called the *edge set* and its elements are called the *edges* of G. The two vertices associated with a given edge are called the *endpoints* of the edge and the edge is said to be *incident* on its endpoints. Two vertices of G are said to be *adjacent* if they are the endpoints of some edge of G. A *self-loop* is an edge whose endpoints are the same vertex. Edges with both endpoints in common are called *parallel* edges, provided they are not self-loops. A *simple* graph is one which has no parallel edges nor self-loops.

The *degree* of a vertex v, $\deg(v)$, is the number of distinct edges which are incident on v. If V is the disjoint union of two sets V_1 and V_2 then we say the two sets form a *partition* of the vertex set. If the partition has the property that each edge in E has one endpoint in V_1 and one endpoint in V_2 then the graph is said to be *bi-partite* and the partition of V is called

a *bi-partition* of the graph. A bi-partition of a graph has the property that adjacent vertices are in different sets of the partition.

A *subgraph* of a graph G is a graph obtained from G by deleting edges from the edge set. Thus, a subgraph has an edge set which is a subset of the edge set of G. Two subgraphs are *edge-disjoint* if they have no edge in common. The *union* and *intersection* of two subgraphs are subgraphs obtained by taking the union and intersection (respectively) of the two edge and vertex sets. We use the usual set notation for these: $\cup$ and $\cap$.

Suppose that we have two vertices, v_1 and v_2, and a sequence of distinct edges, $e_1, e_2, \ldots, e_m$, such that e_1 is incident on v_1, e_m is incident on v_2, and each edge e_k is incident on e_{k+1} and e_{k-1}, for $k = 2, \ldots, m-1$. If every vertex common to two edges of the subgraph has degree 2 (counted with respect to the subgraph), then the subgraph is called a *path* joining the *terminal* vertices v_1 and v_2. A path whose terminal vertices are the same vertex, is called a *cycle* or *circuit* of the graph.

Suppose there exist subgraphs, H and K, of a graph G, each with at least one edge, such that $H \cup K = G$ and $H \cap K$ is a single vertex graph. Then the single vertex in $H \cap K$ is called a *cut-vertex*.

A graph is *connected* if there is some path joining each pair of vertices.

A *part* of a graph is a maximal connected subgraph. In particular, a connected graph has one part. A *component* of a graph is a maximal connected subgraph that has no cut-vertex. A graph is *non-separable* if it has one part and no cut-vertex and otherwise it is called *separable*.

An edge of a graph which, when deleted increases the number of parts by one, is called a *bridge*.

A *co-cycle* or *cutset* of a graph G is a set of edges of G such that the removal of these edges increases the number of parts of G, and it is minimal with respect to this property (meaning no proper subset has this property).

A graph is *planar* if it can be represented by a diagram in a plane in which the vertices are points and the edges are plane curves which do not intersect, except possibly at their endpoints. Such a diagram is called a *planar representation* of the graph. Suppose G_p is a planar representation of a graph G. Let $R^2 - G_p$ denote the set of all points in the plane which are neither vertices nor points on an edge of G_p. Then $R^2 - G_p$ is the union of mutually disjoint open sets which are called the *faces* of G_p.

A point which is in the closure of a face F of a planar representation G_p, but not in F, will be called a *boundary point* of F. The set of all boundary points of a face F will be called the *boundary* of F. An edge which consists of boundary points for some face F will be said to *lie* on the boundary of that face. The boundary of the unbounded face of G_p will be called the boundary of G_p.

It is absolutely essential in developing the theory of bond graphs to make a clear distinction between a graph, which expresses a given incidence structure, and its planar representations (if it has any). Nevertheless, most readers will prefer to visualize a graph as a planar diagram. Also, even though we have been careful to state all of the above definitions in terms of incidence relations, it is obvious that the motivation for them can be seen by considering a planar diagram. If we adopt the convention that crossing lines do not intersect, except at their endpoints, then we can also draw a planar diagram which can be used to visualize the incidence structure of a non-planar graph.