#5

FEEDBACK THEORY

I. SOME PROPERTIES OF SIGNAL FLOW GRAPHS

SAMUEL J. MASON

LOAN COPY only

TECHNICAL REPORT NO. 153

FEBRUARY 2, 1953

RESEARCH LABORATORY OF ELECTRONICS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS

The research reported in this document was made possible through support extended the Massachusetts Institute of Technology, Research Laboratory of Electronics, jointly by the Army Signal Corps, the Navy Department (Office of Naval Research) and the Air Force (Air Materiel Command), under Signal Corps Contract No. DA36-039 sc-100, Project No. 8-102B-0; Department of the Army Project No. 3-99-10-022.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS

Technical Report No. 153 February 2, 1953

FEEDBACK THEORY

I. Some Properties of Signal Flow Graphs

Samuel J. Mason

This report is based on a doctoral thesis in the Department of Electrical Engineering, M.I.T. 1951.

Abstract

The equations characterizing a systems problem may be expressed as a network of directed branches. (The block diagram of a servomechanism is a familiar example.) A study of the topological properties of such graphs leads to techniques which have proven useful, both for the discussion of the general theory of feedback and for the solution of practical analysis problems.

SOME PROPERTIES OF SIGNAL FLOW GRAPHS

1. Introduction

A signal flow graph is a network of directed branches which connect at nodes. Branch jk originates at node j and terminates upon node k; its direction is indicated by an arrowhead. A simple flow graph is shown in Fig. 1(a). This particular graph contains nodes 1, 2, 3, and branches 12, 13, 23, 32, and 33. The flow graph may be interpreted as a signal transmission system in which each node is a tiny repeater station. The station receives signals via the incoming branches, combines the information in some manner, and then transmits the result along each outgoing branch. If the resulting signal at node j is called x_j, the flow graph of Fig. 1(a) implies the existence of a set of explicit relationships

$$
\begin{aligned}
x_1 &= \text{a specified quantity or a parameter} \\
x_2 &= f_2(x_1, x_3) \\
x_3 &= f_3(x_1, x_2, x_3).
\end{aligned}
\tag{1}
$$

The first equation alone would be represented as a single isolated node; whereas the second and third equations, each taken by itself, have the graphs shown in Fig. 1(b) and Fig. 1(c). The second equation, for example, states that signal x_2 is directly influenced by signals x_1 and x_3, as indicated by the presence of branches 12 and 32 in the graph.

In this report we shall be concerned with flow graph topology, which exposes the structure (Gestalt) of the associated functional relationships, and with the manipulative techniques by which flow graphs may be transformed or reduced, thereby solving or programming the solution of the accompanying equations. Specialization to linear flow graphs yields results which are useful for the discussion of the general theory of feedback in linear systems, as well as for the solution of practical linear analysis problems. Subsequent reports will deal with the formal matrix theory of flow graphs, with sensitivity and stablity considerations, and with more detailed applications to practical problems. Our purpose here is to present the fundamentals, together with simple illustrative examples of their use.

2. The Topology of Flow Graphs

Topology has to do with the form and structure of a geometrical entity but not with its precise shape or size. The topology of electrical networks, for example, is concerned with the interconnection pattern of the circuit elements but not with the characteristics of the elements themselves. Flow graphs differ from electrical network graphs in that their branches are directed. In accounting for branch directions we shall need to take an entirely different line of approach from that adopted in electrical network topology.

2.1 Classification of paths, branches, and nodes

As a signal travels through some portion of a flow graph, traversing a number of successive branches in their indicated directions, it traces out a path. In Fig. 2, the sequences 1245, 2324, and 23445 constitute paths, as do many other combinations. In general, there may be many different paths originating at a designated node j and terminating upon node k, or there may be only one, or none. For example, no path from node 4 to node 2 appears in Fig. 2. If the nodes of a flow graph are numbered in a chosen order from 1 to n, then we may speak of a forward path as any path along which the sequence of node numbers is increasing, and a backward path as one along which the numbers decrease. An open path is one along which the same node is not encountered more than once. Forward and backward paths are evidently open.

Any path which returns to its starting node is said to be closed. Feedback now enters directly into our discussion for the first time with the definition of a feedback loop as any set of branches which forms a closed path. The flow graph of Fig. 2 has closed paths 232 (or 323) and 44. Multiple encirclements such as 23232 or 444 also constitute closed paths but these are topologically trivial. Notice that some paths, such as 12324, are neither open nor closed.

We may now classify the branches of a flow graph as either feedback or cascade branches. A feedback branch is one which appears in a feedback loop. All others are called cascade branches. Returning to Fig. 2, we see that 23, 32, and 44 are the only feedback branches present. If each branch in a flow graph is imagined to be a one-way street, then a lost automobilist who obeys the law may drive through Feedback Street any number of times but he can traverse Cascade Boulevard only once as he wanders about in the graph.

The nodes in a flow graph are evidently susceptible to the same classification as branches; that is, a feedback node is one which enters a feedback loop. Two nodes or branches are said to be coupled if they lie in a common feedback loop. Any node not in a feedback loop is called a cascade node. Two special types of cascade nodes are of interest. These are sources and sinks. A source is a node from which one or more branches radiate but upon which no branches terminate. A sink is just the opposite, a node having incoming branches but no outgoing branches. Figure 2 exhibits feedback nodes 2, 3, 4, a source 1, and a sink 5. It is possible, of course, for a cascade node to be neither a source nor a sink. The intermediate nodes in a simple chain of branches are examples.

2.2 Cascade graphs

A cascade graph is a flow graph containing only cascade branches. It is always possible to number the nodes of a cascade graph in a chosen sequence, called the order of

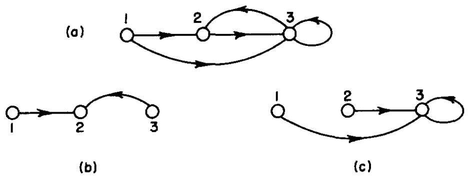

Fig. 1
Flow graphs.

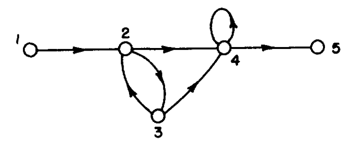

Fig. 2
A flow graph with three feedback branches
and four cascade branches.

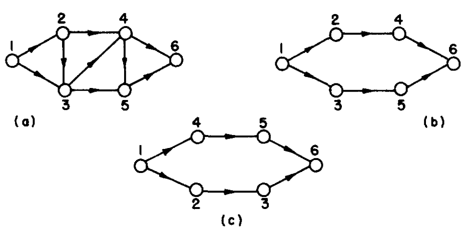

Fig. 3
Cascade graphs.

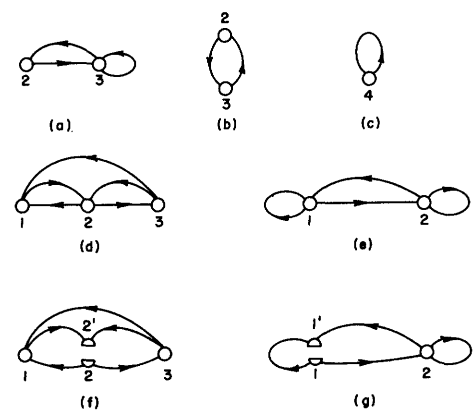

Fig. 4
Feedback units.

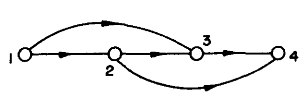

Fig. 5
A cascade graph.

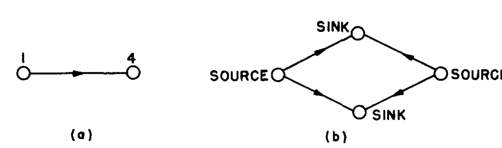

Fig. 6
Residual forms of a cascade graph.

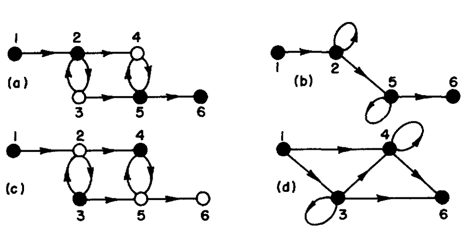

Fig. 7
Feedback graphs and the index-residues.

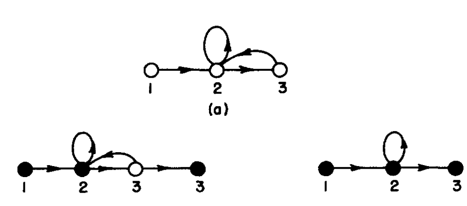

Fig. 8
Retention of a desired node as a sink.

flow, such that no backward paths exist. For a proof of this we first observe that a cascade graph must have at least one source node. Let us choose a source, number it one, and then remove it, together with all its radiating branches. This removal leaves a new cascade graph having, itself, at least one source. We again choose a source, number it two, and continue the process until only isolated nodes remain. These remaining nodes are the sinks of the original graph and they are numbered last. It is evident that this procedure establishes an order of flow.

Figure 3 shows two simple cascade graphs whose nodes have been numbered in flow order. The numbering of graph 3(a) is unique, whereas other possibilities exist for graph 3(b); the scheme shown in graph 3(c) offers one example.

2.3 Feedback graphs

A feedback graph is a flow graph containing one or more feedback nodes. A feedback unit is defined as a flow graph in which every pair of nodes is coupled. It follows that a feedback unit contains only feedback nodes and branches. If all cascade branches are removed from a feedback graph, the remaining feedback branches form one or more separate feedback units which are said to be imbedded or contained in the original flow graph. The graph of Fig. 1, for example, contains the single unit shown in Fig. 4(a), whereas the two units shown in Fig. 4(b) and Fig. 4(c) are imbedded in the graph of Fig. 2.

The units shown in Fig. 4(d) and Fig. 4(e) each possess three principal feedback loops. The number of loops, however, is not of great moment. A more important characteristic is a number called the index. Preparatory to its definition, let us introduce the operation of node-splitting, which separates a given node into a source and a sink. All branch tails appearing at the given node must, of course, go with the source and all branch noses with the sink. The result of splitting node 2 in Fig. 4(d) is shown in Fig. 4(f). Similarly, Fig. 4(g) shows node 1 of Fig. 4(e) in split form. We shall retain the original node number for both parts of the split node, indicating the sink by a prime. Splitting effectively interrupts all paths passing through a given node and makes cascade branches of all branches connected to that node.

We can now conveniently define the index of a feedback unit as the minimum number of node-splittings required to interrupt all feedback loops in the unit. For the determination of index, splitting a node is equivalent to removing that node, together with all its connecting branches.

The index of the graph in Fig. 4(d) is unity, since all feedback loops pass through node 2. Graph 4(e), on the other hand, is of index two.

2.4 The residue of a graph

A cascade graph represents a set of equations which may be solved by explicit operations alone. Figure 5, for example, has the associated equation set

$$x_2 = f_2(x_1)$$

$$x_3 = f_3(x_1, x_2)$$

$$x_4 = f_4(x_2, x_3). \tag{2}$$

Given the value of the source x_1, we obtain the value of x_4 by direct substitution

$$x_4 = f_4\left\{f_2(x_1), f_3\left[x_1, f_2(x_1)\right]\right\} = F_4(x_1). \tag{3}$$

In general, there may be s different sources. Once an order of flow is established, a knowledge of the source variables $x_1, x_2, \ldots, x_s$ fixes the value of x_{s+1}, since no backward paths from later nodes to x_{s+1} can exist. Similarly, with $x_1, x_2, \ldots, x_{s+1}$ known, x_{s+2} is determined explicitly, and so on to the last node x_n. A cascade graph is immediately reducible, therefore, to a residual form in which only sources and sinks appear. The residual form of Fig. 5 is the single branch shown in Fig. 6(a), which represents Eq. 3. Had two sources and two sinks appeared in the original graph, the residual graph would have contained, at most, four branches, as indicated by Fig. 6(b).

Unlike those associated with a cascade graph, the equations of a feedback graph are not soluble by explicit operations. Consider the simple example shown in Fig. 1. An attempt to express x_3 as an explicit function of x_1 fails because of the closed chain of dependency between x_2 and x_3. Elimination of x_2 from Eq. 1 by substitution yields

$$x_3 = f_3\left[x_1, f_2(x_1, x_3), x_3\right] = F_3(x_1, x_3). \tag{4}$$

Although a feedback graph cannot be reduced to sources and sinks by explicit means, certain superfluous nodes may be eliminated, leaving a minimum number of essential implicit relationships exposed.

In any contemplated process of graph reduction, the nodes to be retained in the new graph are called residual nodes. It is convenient to define a residual path as one which runs from a residual node to itself or to another residual node, without passing through any residual nodes. The residual graph, or residue, has a branch jk if, and only if, the original graph has one or more residual paths from j to k. This completely defines the residue of any flow graph for a specified set of residual nodes.

We are interested here in a reduction which can be accomplished by explicit operations alone. The definition of index implies the existence of a set of index nodes, equal in number to the index of a graph, whose splitting interrupts all feedback loops in the graph. The set is not necessarily unique. Once a set of index nodes has been chosen, however, all other nodes except sources and sinks may be eliminated by direct substitution, leaving a residual graph in which only sources, sinks, and index nodes appear. We shall call such a graph the index-residue of the original graph.

Figure 7 shows a flow graph (a) and its index-residue (b). Residual nodes are blackened. Branch 25 in (b) accounts for the presence of residual paths 245 and 235 in (a). All paths from 2 to 6 in (a) pass through residual node 5. Hence graph 7(a) has no residual paths from 2 to 6, since a residual path, by definition, may not pass through a residual node. Accordingly, graph 7(b) has no branch 26. Figure 7(c) illustrates an alternate choice of index nodes and Fig. 7(d) shows the resulting index-residue. Choice (a) is apparently advantageous in that it leads to a simpler residue.

A minor dilemma arises in the reduction process if we desire, for some reason, to preserve a node which is neither an index node nor a sink. In Fig. 8(a), for example, suppose that an eventual solution for x_3 in terms of x_1 is required. A node corresponding to variable x_3 must be retained in the residual graph. Apparently, no further reduction is possible. The simple device shown in Fig. 8(b) may be employed, however, to obtain the residue (c). The trick is to connect node 3 to a sink through a branch representing the equation $x_3 = x_3$. The original node 3 then disappears in the reduction, leaving the desired value of x_3 available at the sink. This trick is simple but topologically nontrivial.

2.5 The condensation of a graph

The concept of an order of flow may be applied, in modified form, to a feedback graph as well as to a cascade graph. Consider the feedback graph in Fig. 9(a), which contains two feedback units. If each imbedded feedback unit is encircled and treated as a single supernode, then the graph condenses to the form shown in Fig. 9(b), where supernodes are indicated by squares. Since the condensation is a cascade structure, an order of flow prevails. Within each supernode the order is arbitrary, but we shall agree to number the internal nodes consecutively.

The index-residue of a flow graph shows the minimum number of essential variables which cannot be eliminated from the associated equations by explicit operations. The condensation of the residue programs the solution for these variables. In Fig. 9(b), for example, the condensation directs us to specify the value of x_1, to solve a pair of simultaneous equations for x_2 and x_3, to solve a single equation for x_4, and to compute x_5 explicitly. The complexity of the solution, without regard for the specific character of the mathematical operations involved, is indicated by the number of feedback units and the index of each, since the index of a feedback unit is the minimum number of simultaneous equations determining the variables in that unit.

Carrying the condensation one step further, we may indicate the basic structural character of a given flow graph by a simple listing of its nodes in the order of condensed signal flow, with residual nodes underlined and feedback units overlined. The sequence

$$\underline{1}\ \ \underline{2}\ \ \overline{3\ \ \underline{4}\ \ \underline{5}\ \ 6}\ \ 7\ \ \overline{\underline{8}\ \ 9\ \ 10}\ \ 11\ \ \underline{12}$$

for example, states that nodes 1 and 2 are sources, 7 and 11 are cascade nodes, and 12 is a sink. Also, nodes 3, 4, 5, 6 lie in a feedback unit of index two, having index nodes 4 and 5. Finally, nodes 8, 9, 10 comprise a later feedback unit of index one, 8 being the index node.

2.6 The inversion of a path

A single constraint or relationship among a number of variables appears topologically as a cascade graph containing one sink and one or more sources. Figure 10(a) is an elementary example. At least in principle, nothing prevents us from solving the equation in Fig. 10(a) for one of the independent variables, say x_1, to obtain the form shown in Fig. 10(b). In terms of the flow graph, we say that branch 14 has been inverted.

By definition, the inversion of a branch is accomplished by interchanging the nose and tail of that branch and, in moving the nose, carrying along all other branch noses which touch it. The tails of other branches are left undisturbed. The inversion of a path is effected by inverting each of its branches.

Figure 11 shows (a) a flow graph, (b) the inversion of an open path 1234, and (c) the inversion of a feedback loop 343. To obtain (c) from (a), for example, we first change the directions of branches 34 and 43. Then we grasp branch p by its nose and move the nose to node 4, leaving the tail where it is. Finally, the nose of branch q is shifted to node 3. Branches 12 and 32 are unchanged since they have properly minded their own business and kept their noses out of the path inversion. Topologically, the two parallel branches running from 4 to 3 are redundant. One such branch is sufficient to indicate the dependency of x_3 upon x_4.

The inversion of an open path is significant only if that path starts from a source. Otherwise, two expressions are obtained for the same variable and two nodes with the same number would be needed in the graph. In addition, inversion is not applicable to a feedback loop which intersects itself. The reason is that two of the path branches would terminate upon a common node. Hence the inversion of one would move the other, thereby destroying the path to be inverted. Such paths as 234 and 23432 in Fig. 11(a), therefore, are not candiates for inversion.

The process of inversion, as might be expected, influences the topological properties of a flow graph. Of greatest interest here is the effect upon the index. Graphs (a), (b), and (c) of Fig. 11 have indices of two, zero, and one, respectively. In general, paths parallel to a given path contribute to the formation of feedback loops when the given path is inverted, and conversely. Hence, if we wish to accomplish a reduction of index we should choose for inversion a forward path having many attached backward paths but few parallel forward paths.

3. The Algebra of Linear Flow Graphs

A linear flow graph is one whose associated equations are linear. The basic linear flow graph is shown in Fig. 12. Quantities a and b are called the branch transmissions, or branch gains. Thinking of the flow graph as a signal transmission system, we may associate each branch with a unilateral amplifier or link. In traversing any branch the signal is multiplied, of course, by the gain of that branch. Each node acts

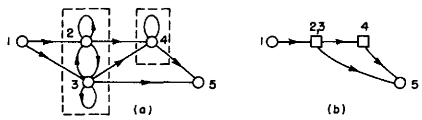

Fig. 9

The condensation of a flow graph.

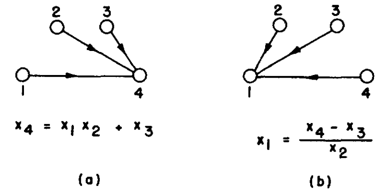

Fig. 10

Inversion of a branch.

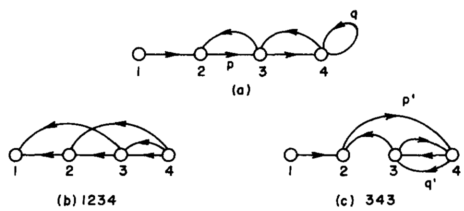

Fig. 11

Path inversions.

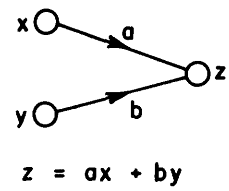

Fig. 12

The basic linear flow graph.

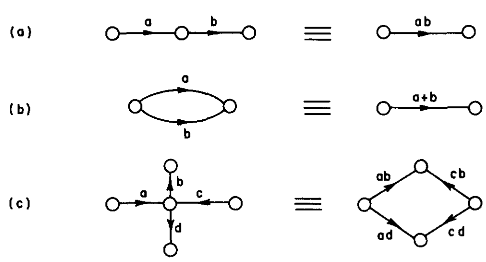

Fig. 13

Elementary transformations.

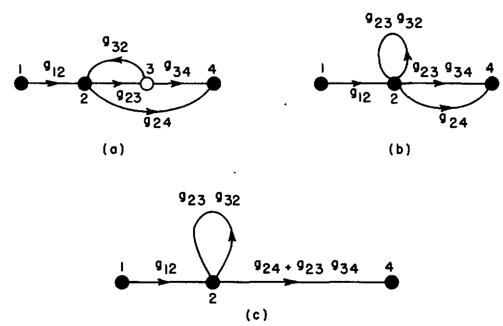

Fig. 14

Reduction to an index-residue by elementary transformations.

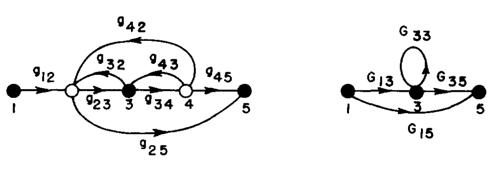

Fig. 15

Reduction to an index-residue by inspection.

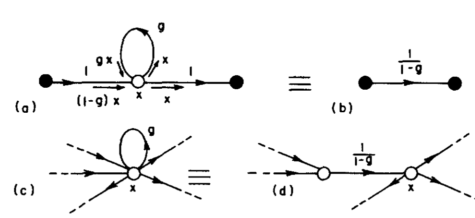

Fig. 16

Replacement of a self-loop by a branch.

as an adder and ideal repeater which sums the incoming signals algebraically and then transmits the resulting signal along each outgoing branch.

3.1 Elementary transformations

Figure 13 illustrates certain elementary transformations or equivalences. The cascade transformation (a) eliminates a node, as does the star-to-mesh transformation (c), of which (a) is actually a special case. The parallel or multipath transformation (b) reduces the number of branches. These basic equivalences permit reduction to an index-residue and give us, as a result of the process, the values of branch gains appearing in the residual graph. Figure 14 offers an illustration. The residual nodes are the source 1, the sink 4, and the index node 2. Node 3 could be chosen instead of node 2, but this would lead to a more complicated residue. The star-to-mesh equivalence eliminates node 3 in graph 14(a) to give graph 14(b). The multipath transformation then yields the residue (c).

For more complicated structures the repeated use of many successive elementary transformations is tedious. Fortunately, it is possible under certain conditions to recognize the branch gains of a residue by direct inspection of the original diagram. In order to provide a sound basis for the more direct process, we shall define a path gain as the product of the branch gains along that path. In addition, the residual gain G_{jk} is defined as the algebraic sum of the gains of all different residual paths from j to k. As defined previously, a residual path must not pass through any of the residual nodes which are to be retained in the new graph. It follows that each branch gain of the residue is equal to the corresponding residual gain G_{jk} of the original graph. Moreover, if the residual graph is an index-residue, then each G_{jk} is the gain of a cascade structure and contains only sums of products of the original branch gains. For index-residues, therefore, the gains G_{jk} are relatively easy to evaluate by inspection.

The feedback graph of Fig. 15(a), for example, has an index-residue (b) containing four branches. By inspection of the original graph, the residual gains are found to be

$$G_{13} = g_{12}g_{23}$$

$$G_{15} = g_{12}g_{25}$$

$$G_{33} = g_{32}g_{23} + g_{34}g_{42}g_{23} + g_{34}g_{43}$$

$$G_{35} = g_{34}g_{45} + g_{32}g_{25} + g_{34}g_{42}g_{25}. \tag{5}$$

Notice that there are three different residual paths from node 3 to itself and also from 3 to 5. We must be very careful to account for all of them. There is only one residual path from 1 to 5, however, and this is 125. Path 12345, which we might be tempted to include in G_{15}, is not residual, since it passes through node 3.

3.2 The effect of a self-loop

When a feedback graph is simplified to a residue containing only sources, sinks, and index nodes, one or more self-loops appear. The effect of a self-loop at any node upon the signal passing through that node may be studied in terms of Fig. 16(a). The signal existing at the central node is transmitted along the outgoing paths as indicated by the detached arrows. The signal returning via the self-loop is gx, where g is the branch gain of the self-loop. Since signals entering the node must add algebraically to give x, it follows that the external signal entering from the left must be $(1-g)x$. The node and self-loop, therefore, may be replaced by a single branch (b) whose gain is the reciprocal of $(1-g)$. When several branches connect at the node, as in Fig. 16(c), it is easy to see that the proper replacement is that shown in Fig. 16(d). Quantity g is usually referred to as the loop gain and $1-g$ is called the loop difference.

Approaching the self-loop effect from another viewpoint, we may treat Fig. 16(b) as the residual form of Fig. 16(a). This is not, of course, an index-residue. The gain G of (b) is the sum of the gains of all residual paths from the source to the sink in (a). One path passes directly through the node, the second path traverses the loop once before leaving, the third path circles the loop twice, and so on. Hence the residual gain is given by the infinite geometrical series

$$G = 1 + g + g^2 + g^3 + \dots = \frac{1}{1-g} \tag{6}$$

which sums to the familiar result. The convergence of this series, for $|g|<1$, poses no dilemma in view of the validity of analytic continuation. The result holds for all values of g except the singular point $g = 1$, near which the transmission G becomes arbitrarily large.

The self-loop-to-branch transformation places in evidence the basic effect of feedback as a contribution to the denominator of an expression for the gain of a graph in terms of branch gains. In our algebra, feedback is associated with division or, more generally, with the inversion of a matrix whose determinant is not identically equal to unity.

3.3 The general index-residue of index one

If we restrict attention to a single source and a single sink, then the most general index-residue of index one, or first-index-residue, is that shown in Fig. 17(a). Other sources or sinks in the system may be considered separately, without loss of generality, since the system is linear and superposition applies. A knowledge of the self-loop-to-branch transformation enables us to write the (source to sink) gain of graph 17(a) by inspection. The gain is

$$G = d + \frac{bc}{1-a}. \tag{7}$$

When the total index of the graph is greater than one, as in Fig. 17(b), it is still a simple matter to find the gain, provided each imbedded feedback unit is only of first index. For graph 17(b)

$$G = g + \frac{ef}{1 - d} + \frac{bcf}{(1-a)(1-d)}. \tag{8}$$

With practice, the gain of a graph such as that of Fig. 15(a) can be written at a glance, without bothering to make an actual sketch of the residue. The principal source of error lies in the possibility of overlooking a residual path.

Of special interest is the theorem that if each feedback unit in a graph is a simple ring of branches, the gain of that graph is equal to the sum of the gains of all open paths from source to sink, each divided by the loop differences of feedback loops encountered by that path. For illustration, we shall apply this theorem to the graph shown in Fig. 18. There are nine different open paths from the source to the sink and each one makes contact with the feedback loop. The resulting gain is

$$G = \frac{ah + bdh + cgdh + aei + bdei + cgdei + aefj + bdefj + cj}{1 - defg}. \tag{9}$$

3.4 The general index-residue of index two

Again taking one source and one sink at a time, we shall study the most general second-index-residue shown in Fig. 19.

Suppose that the self-loops are temporarily removed, leaving the simple imbedded ring shown in (b). Graph (b) exhibits five open paths from source to sink, namely i, ab, cd, afd, ceb; and the last four of these encounter the feedback loop ef. Hence the gain of graph (b) is

$$G = i + \frac{ab + cd + afd + ceb}{1 - ef}. \tag{10}$$

Now, in order to account for the self-loops g and h in graph 19(a), we need only divide each path gain appearing in expression 10 by the loop difference (1-g) if that path passes through the upper node, and by (1-h) if it passes through the lower node. Paths afd, ceb, and ef, of course, pass through both nodes, and their gains must be divided by both loop differences. The resulting modification of formula 10 yields the gain of the general second-index-residue

$$G = i + \frac{\frac{ab}{1-g} + \frac{cd}{1-h} + \frac{afd + ceb}{(1-g)(1-h)}}{1 - \frac{ef}{(1-g)(1-h)}}. \tag{11}$$

The derivation of this formula is important only as a demonstration of the power of the method. To find the (source-to-sink) gain of any graph whose feedback units are no worse than second index, we reduce to an index-residue; temporarily remove the self-loops; express the gain as the sum of open path gains, each divided by the loop differences of feedback loops touching that path; and modify the result to account for

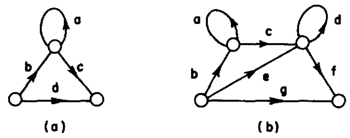

Fig. 17

Residues having first-index feedback units.

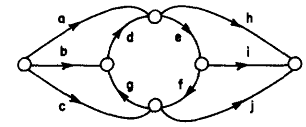

Fig. 18

A simple ring imbedded in a graph.

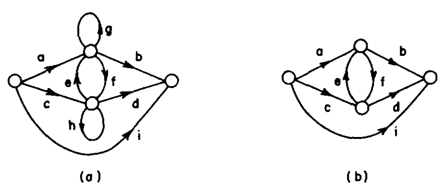

Fig. 19

The general second-index-residue with and without self-loops.

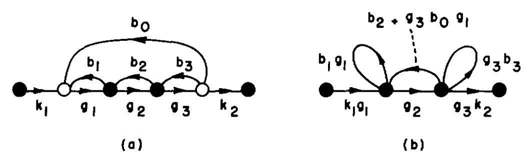

Fig. 20

A three-stage feedback amplifier diagram.

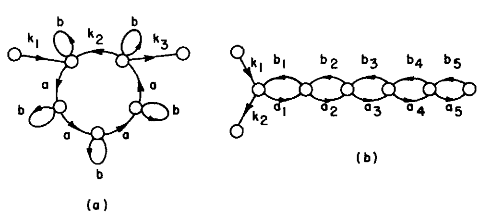

Fig. 21

Simple high-index structures.

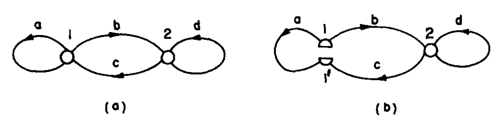

Fig. 22

The loop gain of a node.

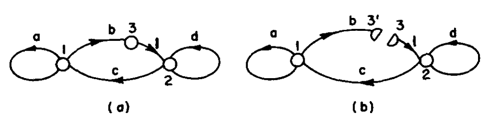

Fig. 23

The loop gain of a branch.

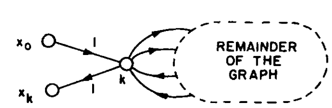

Fig. 24

The injection gain at node k.

the original self-loops.

The importance of the method justifies a final example. Figure 20(a) shows the feedback diagram of a three-stage amplifier having local feedback around each stage and external feedback around the entire amplifier. With the self-loops temporarily removed, the gain of the residue (b) is

$$G = \frac{k_1 g_1 g_2 g_3 k_2}{1 - g_2(b_2 + g_3 b_0 g_1)}. \tag{12}$$

Since all paths appearing in expression 12 touch both index nodes, the actual gain of the amplifier is

$$G = \frac{\dfrac{k_1 k_2 g_1 g_2 g_3}{(1 - b_1 g_1)(1 - b_3 g_3)}}{1 - \dfrac{g_2(b_2 + b_0 g_1 g_3)}{(1 - b_1 g_1)(1 - b_3 g_3)}} = \frac{k_1 k_2 g_1 g_2 g_3}{(1 - b_1 g_1)(1 - b_3 g_3) - g_2(b_2 + b_0 g_1 g_3)}. \tag{13}$$

3.5 Graphs of higher index

The formal reduction process for an arbitrary feedback graph involves a cycle of two steps. First, reduction to an index-residue; and second, replacement of any one of the self-loops by its equivalent branch. Exactly n such cycles are required for reduction to cascade form, where n is the total index of the original graph. Transformation of more than one self-loop at a time is often convenient, even though this may increase the total number of self-loop transformations required in later steps. In practice, of course, the formal procedure should be modified to take advantage of the peculiarities of the structure being reduced. The process effectively ends when the index has been reduced to two, since the evaluation of gain by inspection of the index-residue then becomes tractable.

Figure 21 shows two graphs containing high-index feedback units. With the self-loops removed from the circular structure (a), the gain is equal to that of the single open forward path $k_1 a^4 k_3$ divided by the loop difference of the closed path $k_2 a^4$, and we have

$$G = \frac{k_1 a^4 k_3}{1 - k_2 a^4}. \tag{14}$$

Since both paths pass through every index node, the reintroduction of the self-loops yields

$$G = \frac{\dfrac{k_1 a^4 k_3}{(1-b)^5}}{1 - \dfrac{k_2 a^4}{(1-b)^5}} = \frac{k_1 a^4 k_3}{(1-b)^5 - k_2 a^4}. \tag{15}$$

The feedback chain shown in Fig. 21(b) is of third index. Instead of reducing it to an index-residue, we shall take advantage of the simplicity of the chain structure to write the gain by a more direct method. First, with the last four loops of the chain removed, the gain is

$$G = \frac{k_1 k_2}{1 - a_1 b_1}. \tag{16}$$

Now, the addition of loop $a_2 b_2$ modifies the path gain $a_1 b_1$ to give

$$G = \frac{k_1 k_2}{1 - \dfrac{a_1 b_1}{1 - a_2 b_2}}. \tag{17}$$

Addition of the remaining elements leads to the continued fraction

$$G = \cfrac{k_1 k_2}{1 - \cfrac{a_1 b_1}{1 - \cfrac{a_2 b_2}{1 - \cfrac{a_3 b_3}{1 - \cfrac{a_4 b_4}{1 - a_5 b_5}}}}}. \tag{18}$$

3.6 Loop gain and loop difference

Thus far we have spoken of loop gain only in connection with feedback units of the simple ring type. A more general concept of loop gain will now be introduced. We shall define the loop gain of a node as the gain between the source and sink created by splitting that node. In terms of signal flow, the loop gain of a node is just the signal returned to that node per unit signal transmitted by that node. The loop difference of a node is by definition equal to one minus the loop gain of that node. We shall use the symbol T for loop gains and D for loop differences. In the graph of Fig. 22(a), for example, the loop gain of node 1 is equal to the gain from 1 to 1' in graph (b), which shows node 1 split into a source 1 and a sink 1'. By inspection

$$T_1 = a + \frac{bc}{1 - d}. \quad D_1 = 1 - a - \frac{bc}{1 - d}. \tag{19}$$

Another quantity of interest is the loop gain of a branch. Preparatory to its definition, let us replace the branch in question by an equivalent cascade of two branches, whose path gain is the same as the original branch gain. This creates a new node, called an interior node of the branch. The loop gain of a branch may now be defined as the loop gain of an interior node of that branch. To find the loop gain of branch b in Fig. 22(a), for instance, we first introduce an interior node 3 as shown in Fig. 23(a). The loop gain of branch b is the gain from 3 to 3' in (b),

$$T_{12} \text{ (or } T_b) = \frac{bc}{(1-a)(1-d)}. \tag{20}$$

The loop gain of a branch can be designated by either a single or double subscript, whichever is a more convenient specification of the branch. The double subscript is usually preferable, since it avoids confusion with the loop gain of a node. The loop gain of a given node (or branch) evidently involves only the gains of branches which are coupled to that node (or branch). Hence, in computing T, we need to consider only the feedback unit containing the node (or branch) of interest.

Having defined the loop gain of a node, we may extend the simple self-loop equivalence to a more general form which may be stated as follows. If an external signal x_o is injected into node k of a flow graph, as shown in Fig. 24, the injection gain from the external source to node k is

$$G_k = \frac{x_k}{x_o} = \frac{1}{1 - T_k} = \frac{1}{D_k}. \tag{21}$$

The very nature of the reduction process for an arbitrary (finite) graph implies that the gain is a rational function of the branch gains. In other words, the gain can always be expressed as a fraction whose numerator and denominator are each algebraic sums of various branch gain products. Moreover, the gain G is a linear rational function of any one of the branch gains g. Thus

$$G = \frac{ag + b}{cg + d} \tag{22}$$

where quantities a, b, c, d are made up of other branch gains. To prove this we may insert two interior nodes into the specified branch g, as shown in Fig. 25(a) and (b), and then consider the residue (c), which contains only the source, the sink, and the two interior nodes. The gain of this residue evidently can be expressed as a linear rational function of g. It is also apparent that if branch g is directly connected to either the source or the sink, or to both, then the source-to-sink gain G is a linear function of the branch gain g, that is,

$$G = ag + b \tag{23}$$

where a and b depend upon other branch gains.

The foregoing results apply equally well to loop gains and loop differences, since T and D, by their definitions, have the character of gains. Any loop difference D_k is a rational function of the branch gains, a linear rational function of any single branch gain, and a linear function of the gain of any branch connected directly to node k.

We shall now derive an important fundamental property of loop differences which is of general interest. Consider an arbitrary graph containing nodes 1, 2, 3, ..., n, and let nodes m+1, m+2, ..., n-1, n be removed, together with their connecting branches, so that only nodes 1, 2, 3, ..., m remain. Now suppose that the graph is reduced to a residue showing only nodes m-1, and m, as in Fig. 26. Branches a, b, c, d account for all coupling among nodes 1, 2, 3, ..., m of the original graph. Sources and sinks may be ignored, of course, since only feedback branches are of interest in loop

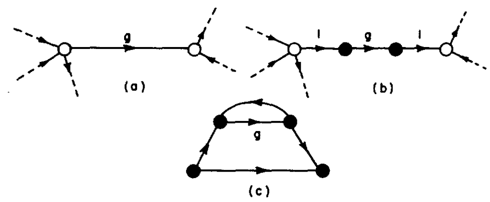

Fig. 25
The graph gain as a function of a particular branch gain.

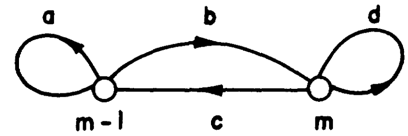

Fig. 26
A residue showing nodes m - 1 and m.

difference calculations. Let us define the partial loop difference D'_k as the loop difference of node k with only the first k nodes taken into account. By inspection of Fig. 26

$$D'_m = 1 - d - \frac{bc}{1 - a} \tag{24}$$

$$D'_{m-1} = 1 - a \tag{25}$$

and

$$D'_{m-1}D'_m = (1-a)(1-d) - bc. \tag{26}$$

If the numbers of nodes m-1 and m are interchanged in Fig. 26, then

$$D'_m = 1 - a - \frac{bc}{1 - d} \tag{27}$$

$$D'_{m-1} = 1 - d \tag{28}$$

and the product given in Eq. 26 is unaltered. Since this result holds for any value of m, and since a sequence may be transformed into any other sequence by repeated adjacent interchanges (1234 can become 4321, for example, by the adjacent interchanges 1243, 2143, 2413, 4213, 4231, 4321), it follows that the product

$$\Delta'_m = D'_1D'_2D'_3 \ldots D'_{m-1}D'_m \tag{29}$$

is independent of the order in which the first m nodes are numbered. With all n nodes present, we have $D'_n = D_n$ and

$$\Delta = D'_1D'_2D'_3 \ldots D'_{n-1}D_n. \tag{30}$$

Quantity Δ, which we shall call the determinant of the graph, is invariant for any order of node numbering. Equation 30 shows that the determinant of any graph is the product of the determinants of its imbedded feedback units, and that the determinant of a cascade graph is unity.

The dependence of Δ upon the branch gains may be deduced as follows. Let g be any branch directly connected to node n, whence it follows that D_n is a linear function of branch gain g and that the partial loop differences D'_k are independent of g. Hence

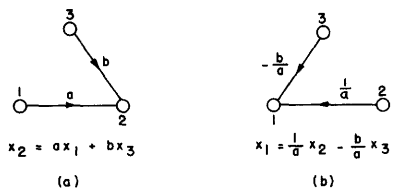

Fig. 27
Branch inversion in a linear graph.

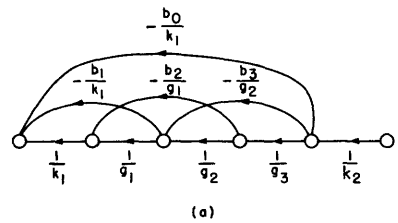

Fig. 28
The result of path inversion in Fig. 20(a).

Δ is a linear function of g. Since the numbering of nodes is arbitrary, Δ must be a linear function of any given branch gain in the graph. The determinant Δ, therefore, is composed of an algebraic sum of products of branch gains, with no branch gain appearing more than once in a single product.

From Eq. 29 and Eq. 30 we see that D_n is the ratio of Δ to Δ'_{n-1}. Since the node number is arbitrary, we may write

$$D_k = \frac{\Delta}{\Delta_k} \tag{31}$$

where Δ_k is to be computed with node k removed. Once Δ is expressed in terms of branch gains, Δ_k may be found by nullifying the gains of branches connected to node k.

The introduction of an interior node into any branch leaves the value of Δ unaltered. To prove this we may number the new node zero, whence $D'_o = 1$ and the other partial loop differences are unchanged. It follows directly that the loop difference of any branch jk is given by

$$D_{jk} = \frac{\Delta}{\Delta_{jk}} \tag{32}$$

where Δ_{jk} is to be computed with branch jk removed, that is, with $g_{jk} = 0$.

Incidentally, if we write the linear equations associated with the flow graph and then evaluate the injection gain G_k by Kramer's rule (that is, by inverting the matrix of the equations), we find from Eq. 21 and Eq. 31 that Δ is just the value of the determinant of these equations.

3.7 Inverse gains

We have already seen how the form of a flow graph is altered by the inversion of a path. For linear graphs it is profitable to continue with an inquiry into the quantitative effects of inversion. Figure 27(a) shows two branches which may be imagined to form part of a larger graph. The signal entering node 2 via branch b is bx_3. The contribution arriving from branch a, then, must be $x_2 - bx_3$, since the sum of these two contributions is equal to x_2. Hence, given x_2 and x_3, the required value of x_1 is that indicated in graph (b).

The general scheme is readily apparent and may be stated as follows. The inversion of any branch jk is accomplished by reversing that branch and inverting its gain, and shifting any other branch ik having the same nose location k to the new position ij and dividing its gain by the negative of the original branch gain g_{jk}.

For gain calculations, the usefulness of inversion lies in the fact that the inversion of a source-to-sink path yields a new graph whose source-to-sink gain is the inverse of the original source-to-sink gain. Since inversion may accomplish a reduction of index, the inverse gain may be much easier to find by inspection. For illustration, we shall invert path $k_1g_1g_2g_3k_2$ in Fig. 20(a) to obtain the graph shown in Fig. 28. The new graph is a cascade structure of zero index. By inspection of the new graph, the inverse gain of the original graph is

$$\frac{1}{G} = \frac{1}{k_2}\left[\left(\frac{1}{g_3g_2} - \frac{b_3}{g_2}\right)\left(\frac{1}{g_1k_1} - \frac{b_1}{k_1}\right) - \frac{b_2}{g_3g_1k_1} - \frac{b_0}{k_1}\right]. \tag{33}$$

Simplification yields

$$\frac{1}{G} = \frac{1}{k_1k_2}\left[\frac{1}{g_2}\left(\frac{1}{g_1} - b_1\right)\left(\frac{1}{g_3} - b_3\right) - \frac{b_2}{g_1g_3} - b_0\right] \tag{34}$$

which proves to be identical with Eq. 13.

A simpler example is offered by Fig. 21(a). Inversion of the open source-to-sink path gives the structure shown in Fig. 29. By inspection of the new graph, we find

$$\frac{1}{G} = \frac{1}{k_3}\left[\left(\frac{1}{a} - \frac{b}{a}\right)^4\left(\frac{1}{k_1} - \frac{b}{k_1}\right) - \frac{k_2}{k_1}\right] = \frac{(1-b)^5}{k_1k_3a^4} - \frac{k_2}{k_1k_3} \tag{35}$$

which checks Eq. 15.

3.8 Normalization

In the general analysis of an electrical network it is often convenient to alter the impedance level or the frequency scale by a suitable transformation of element values. A similar normalization sometimes proves useful for linear flow graph analysis. The self-evident normalization rule may be stated as follows. If each branch gain g_{jk} is multiplied by a scale factor f_{jk}, with the scale factors so chosen that the gains of all closed paths are unaltered, then the gain of the graph is multiplied by $f_{12}f_{23}\cdots f_{mn}$, where 1, 2, 3, ..., m, n is any path from the source 1 to the sink n.

Figure 30 illustrates a typical normalization. Graph (a) might represent a two-stage amplifier with isolation between the two stages, local feedback around each stage, and external feedback around both stages. The normalization shown in (b) brings out very clearly the fact that certain branch gains may be taken as unity without loss of generality.

4. Illustrative Applications of Flow Graph Techniques

The usefulness of flow graph techniques for the solution of practical analysis

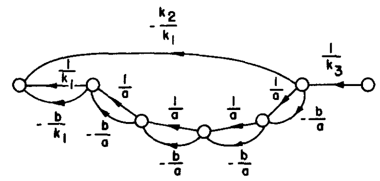

Fig. 29
The result of path inversion in Fig. 21(a).

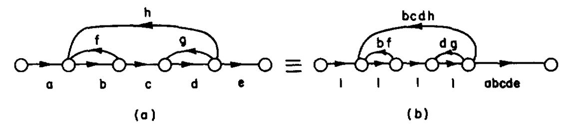

Fig. 30
Normalization.

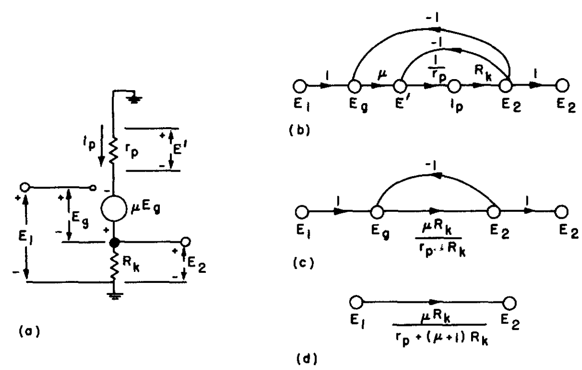

Fig. 31
Flow graphs for a cathode follower.

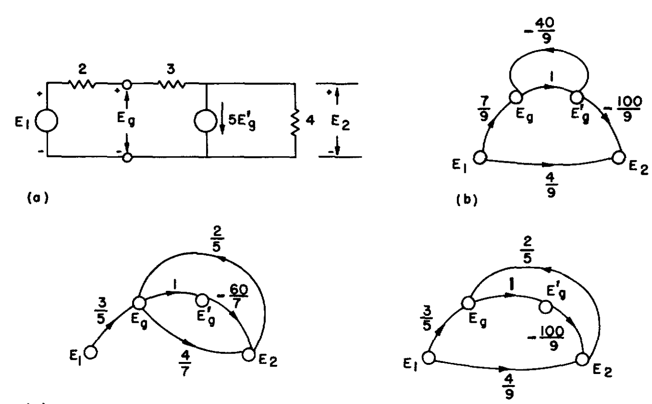

Fig. 32
An amplifier with grid-to-plate impedance.

problems is limited by two factors: our ability to represent the physical problem in the form of a suitable graph, and our facility in manipulating the graph. The first factor has not yet been considered. We turn to it now with the necessary background material at hand.

The process of constructing a graph is one of tracing a succession of causes and effects through the physical system. One variable is expressed as an explicit effect due to certain causes; they, in turn, are recognized as effects due to still other causes. In order to be associated with a single node, each variable must play a dependent role only once. A link in the chain of dependency is limited in extent only by our perception of the problem. The formulation may be executed in a few complicated steps or it may be subdivided into a larger number of simple ones, depending upon our judgment and knowledge of the particular system under consideration. No specific rules can be given for the best approach to an analysis problem. Therein lies the challenge and the possibility of an elegant solution. Whatever the approach, flow graphs offer a structural visualization of the interrelations among the chosen variables. It is quite possible, of course, to construct an incorrect graph, just as it is entirely possible to write a set of equations which do not properly represent the physical problem. The direct formulation of a flow graph from a physical problem, without actually writing the chosen equations, requires some practice before confidence is gained. It is hoped that the following examples, taken mostly from electronic circuit analysis, will be suggestive.

4.1 Voltage gain calculations

Figure 31(a) shows the low-frequency linear incremental equivalent circuit of a cathode follower. Suppose that we want to find the gain E_2/E_1 in terms of the circuit constants. Proceeding very cautiously in small steps, we might construct the graph shown in Fig. 31(b). This graph states that $E_g = E_1 - E_2$, $E' = \mu E_g - E_2$, $I_p = E'/r_p$, and $E_2 = R_k I_p$. Alternatively, were we able to recognize at the outset the direct dependence of E_2 upon E_g, then graph 31(c) could have been sketched by inspection of the circuit. The more extensive our powers of perception, the simpler the formulation. Powerful perception (or a familiarity with the cathode follower) would permit us to construct graph 31(d) directly from the network shown in Fig. 31(a). The reader is invited to evaluate the gains of graphs 31(b) and 31(c) by inspection and to compare them with 31(d).

Another example is offered by the amplifier of Fig. 32(a). For convenience of illustration, the impedances and the transconductance have been given numerical values. In this circuit the grid voltage influences the output voltage both by transconductance action and by direct coupling through the grid-to-plate impedance. To avoid confusion between the actual voltage E_g and the factor E_g appearing in the transconductance current, it is very helpful to designate one of them with a prime while we are setting up the graph. This distinction splits node E_g. It is a simple matter to complete the graph with a unity-gain branch representing the equation $E_g' = E_g$, which effectively rejoins the node.

The direct application of superposition, with voltage E_1 and current $5E'_g$ treated as independent electrical sources, each influencing the dependent quantities E_g and E_2, leads to graph (b) of Fig. 32. The gain from E'_g to E_g, for example, is the product of a transconductance 5, a current division ratio 4/9, and an impedance 2, as measured with $E_1 = 0$.

An alternative approach, actually equivalent to classical network formulation on the electrical-node-pair-voltage basis, gives graph 32(c). Here E_2 is expressed as a function of E_g and E'_g. In accordance with superposition, the gain from E'_g to E_2 must be computed with $E_g = 0$ (rather than $E_1 = 0$, as in the previous graph). Hence, in this particular calculation, the impedance presented to the current source does not include element 2. The other independent electrical-node-pair voltage E_g is expressed in terms of E_1 and E_2, as shown.

Graph 32(d), a third possibility, is actually the simplest and most elegant of the three. Responding to a certain physical appeal, we express E_2 in terms of the two electrical sources, as in graph 32(b). Taking advantage of the fact that E_2 and $5E'_g$ are across the same electrical node-pair, we formulate E_g in terms of E_1 and E_2 as in graph 32(c). This has topological appeal, since the resulting feedback loop touches both open paths from E_1 to E_2. As a result, the graph gain is a simple fractional function of the branch gains. The verification of graphs (b), (c), and (d) of Fig. 32 and the evaluation of their gains is suggested as an exercise for the reader. The answer is -8/7. If symbols are substituted for the numerical element values in the circuit, the suitability of the structure of Fig. 32(d) for this particular problem becomes more apparent.

4.2 The impedance formula

Suppose that the input or output impedance Z of an electronic circuit is influenced by a certain tube transconductance in such a manner that the effect is not immediately obvious. To find Z we must introduce a set of variables and write the equations relating them. Let us choose the terminal current and voltage, I and E = IZ, together with the grid voltage E_g of the offending tube, as shown in Fig. 33(a). The graphical structure which naturally suggests itself, perhaps, is that of the previous problem, Fig. 32(b),

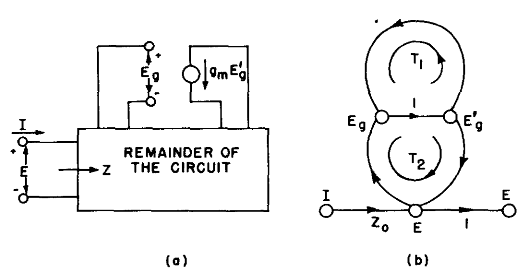

Fig. 33
The circuit and graph for terminal impedance formulation.

with a source I and a sink E. Since E and I are located at the same pair of terminals, however, it is just as easy to express E_g in terms of E_g' and E, rather than E_g' and I. This choice gives graph (b) of Fig. 33, which is particularly convenient for our present purpose. Notice that the structure of Fig. 33(b) is obtainable directly from that of Fig. 32(b) by inversion of the source-to-sink branch.

The three gains of interest in Fig. 33(b) are

$$Z_o = \left(\frac{E}{I}\right)_{E_g' = 0} = \text{the impedance without feedback} \tag{36}$$

$$T_g^{sc} = \left(\frac{E_g}{E_g'}\right)_{E = 0} = \text{the short-circuit loop gain} = T_1 \tag{37}$$

$$T_g^{oc} = \left(\frac{E_g}{E_g'}\right)_{I = 0} = \text{the open-circuit loop gain} = T_1 + T_2 . \tag{38}$$

The terminal impedance is given by the graph gain

$$Z = \frac{Z_o}{1 - \frac{T_2}{1 - T_1}} = Z_o\left(\frac{1 - T_1}{1 - T_1 - T_2}\right) \tag{39}$$

which may be identified as the well-known feedback formula

$$Z = Z_o\left(\frac{1 - T_g^{sc}}{1 - T_g^{oc}}\right). \tag{40}$$

Our conclusion is that flow graph methods provide a relatively uncluttered derivation of this classical result.

Flow graph representation also brings out the similarities between feedback formulas for electronic circuits and compensation theorems for passive networks. Consider, for comparison, the determination of the input impedance of the circuit shown in Fig. 34(a).

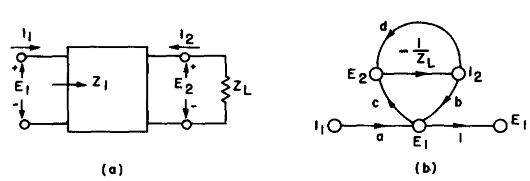

Fig. 34
The effect of load impedance upon input impedance.

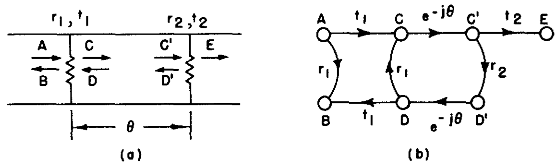

Fig. 35
Two discontinuities on a transmission line.

Superposition tells us that the branch gains of the accompanying graph, Fig. 34(b), have the physical interpretations

$$Z_1^{oc} = \left(\frac{E_1}{I_1}\right)_{I_2 = 0} = \text{open-circuit input impedance} = a \tag{41}$$

$$Z_2^{oc} = \left(\frac{E_2}{I_2}\right)_{I_1 = 0} = \text{open-circuit output impedance} = bc + d \tag{42}$$

$$Z_2^{sc} = \left(\frac{E_2}{I_2}\right)_{E_1 = 0} = \text{short-circuit output impedance} = d. \tag{43}$$

By analogy with the previous problem

$$Z_1 = Z_1^{oc} \frac{1 + \dfrac{Z_2^{sc}}{Z_L}}{1 + \dfrac{Z_2^{oc}}{Z_L}} = Z_1^{oc}\left(\frac{Z_L + Z_2^{sc}}{Z_L + Z_2^{oc}}\right). \tag{44}$$

4.3 A wave reflection problem

The transmission line shown in Fig. 35(a) has two shunt discontinuities spaced θ electrical radians apart. A voltage wave of complex amplitude A is incident upon the first discontinuity from the left. We desire to find the resulting reflection B and the transmitted wave E. Let C, D, C', D' be the waves traveling in opposite directions just to the right of the first obstacle and just to the left of the second. In addition, let r and t denote the per unit reflection or transmission of a single discontinuity.

The accompanying graph 35(b) is self-explanatory. The only feedback loop present is the simple ring CC'D'DC. By inspection of this graph, the over-all reflection and transmission coefficients are

$$\frac{B}{A} = r_1 + \frac{t_1^2 r_2 e^{-j2\theta}}{1 - r_1 r_2 e^{-j2\theta}} \tag{45}$$

$$\frac{E}{A} = \frac{t_1 t_2 e^{-j\theta}}{1 - r_1 r_2 e^{-j2\theta}}. \tag{46}$$

4.4 A limiter design problem

Figure 36(a) shows a vacuum-tube circuit commonly employed as a two-way limiter or level selector. The static transfer curve shown in Fig. 36(b) exhibits a high-gain central region limited on each side by cutoff. In the neighborhood of point p, where both tubes are conducting, the linear incremental circuit of Fig. 36(c) applies. If we design the incremental circuit for infinite gain, then the transfer curve becomes vertical at point p, and the switching interval is made desirably small.

Assume for simplicity that the voltage divider feeding the second grid has a resistance much greater than R_1 (or let R_1 denote the combined parallel resistance). Now let us attempt to formulate E_1 in terms of E_o and E_k by superposition. With $E_k = 0$, the ratio E_1/E_o is simply the gain of a grounded-cathode stage. Similarly, with $E_o = 0$, the first tube becomes a grounded-grid stage driven by E_k. This gives us branches 01 and k1 in the flow graph shown in Fig. 36(d). Branches 12 and k2 follow the same pattern for the second tube. We must now formulate E_k in a convenient manner. One possibility is the computation of the two tube currents $-E_1/R_1$ and $-E_2/R_2$, whose sum may be multiplied by R_k to obtain E_k, as shown.

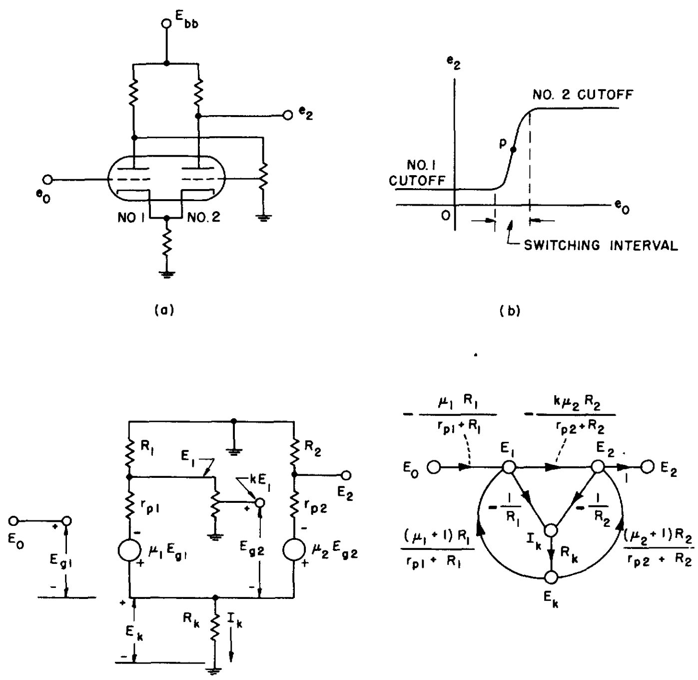

Fig. 36

A cathode-coupled limiter.

The resulting graph is of index one, and either E_k or I_k may be taken as the index node. The index-residue would have the familiar form shown in Fig. 17(a). For infinite gain we need only specify that the loop gain of node E_k (or node I_k, or branch R_k) must be unity. By inspection of the graph, the three paths entering T_k are k12k, k1k, and k2k. Hence

$$T_k = R_k\left[\frac{k(\mu_1 + 1)\mu_2 R_1}{(r_{p1} + R_1)(r_{p2} + R_2)} - \frac{\mu_1 + 1}{r_{p1} + R_1} - \frac{\mu_2 + 1}{r_{p2} + R_2}\right] = 1. \qquad (47)$$

It is a simple matter to solve this equation for the desired value of the voltage divider parameter k.

5. Concluding Remarks

The flow graph offers a visual structure, a universal graphical language, a common ground upon which causal relationships among a number of variables may be laid out and compared. From this viewpoint the similarity between two physical problems arises not from the arrangement of physical elements or the dimensions of the variables but rather from the structure of the set of relationships which we care to write.

The organization of the problem comes from within our minds and feedback is present only if we perceive a closed chain of dependency. The challenge facing us at the start of an analysis problem is to express the pertinent relationships as a meaningful and elegant flow graph. The topological properties of the graph may then be exploited in the manipulations and reductions leading to a solution.

Acknowledgement

The influence of H. W. Bode is clear and present in this paper, as will be obvious to anyone familiar with his work. The writer is particularly indebted to E. A. Guillemin and G. T. Coate, and also to J. B. Wiesner, W. K. Linvill, and W. H. Huggins, for many helpful ideas, criticisms, and encouragements.