AN ALGORITHM FOR CALCULATING THE GAIN OF A SIGNAL FLOWGRAPH *

William R. Dunn, Jr.
Senior Engineer
Lockheed Missiles and Space Company
Sunnyvale, California

and

S. Park Chan
Associate Professor of Electrical Engineering
University of Santa Clara
Santa Clara, California

ABSTRACT

An algorithm is presented which permits automatic computation of transfer functions given a Mason signal flowgraph description of a system. Specifically, the path sets and loop sets corresponding to the numerator and denominator terms of Mason's gain formula are calculated from the non-oriented incidence matrix A_m and the fundamental circuit matrix B_f of the flowgraph.

First order loops of the flowgraph can be found by performing row sum operations on the rows of B_{mf} and retaining only the non-redundant rows in which all non-zero elements have the same sign. The first-order loop matrix L_1, the k^{th}-order loop matrix L_k, and the loop adjacency matrix $\mathcal{L}_k$ are defined. It is related that $\mathcal{L}_k$ can be found from L_k through the equation $\mathcal{L}_k = L_k \cdot \xi$, where $\xi = A_m^T \cdot A_m \cdot L_1$, all matrix operations being performed in Boolean algebra.

From the foregoing relationships matrices $L_1, L_2, \ldots, L_n$, which represent loop sets corresponding to all denominator terms of Mason's gain formula, can be systematically generated. Terms corresponding to the numerator expression can be found by applying the same procedure to a modified flowgraph. In the algorithm only the modified flowgraph is employed since it develops that the procedure for calculating the numerator terms from it automatically provides the denominator terms as well.

In the summary, the problem is obtained, basic definitions and the algorithm are stated, and an example of its use is presented. Finally, the efficiency of the algorithm in terms of arithmetic and memory requirements for digital computation is discussed.

*This work was supported in part by the National Aeronautics and Space Administration, Grant No. NGR-05-017-012.

AN ALGORITHM FOR CALCULATING THE GAIN OF A SIGNAL FLOWGRAPH+

William R. Dunn, Jr.
Senior Engineer
Lockheed Missiles and Space Company
Sunnyvale, California

and

S. Park Chan
Associate Professor of Electrical Engineering
University of Santa Clara
Santa Clara, California

SUMMARY

1. INTRODUCTION

Mason's flowgraph technique,[1][2] as applied to the derivation of system transfer functions,[3] has in the past depended upon inspection of the flowgraph for solutions. This paper presents an algorithm which enables use of this technique by digital computer.

Given the flowgraph description of a system, the technique provides an overall gain G where,[1]

$$G = \frac{\sum_k G_k \Delta_k}{\Delta};$$

G_k = gain of the k^{th} distinct forward path* from source to sink;

$$\Delta = 1 - \sum_m p_{m1} + \sum_m p_{m2} - \sum_m p_{m3} + \ldots + (-1)^j \sum_m p_{mj};$$

p_{m1} = loop* gain (product of all the branch gains around a loop);

p_{m2} = product of the loop gains of the m^{th} set of two non-touching loops;

p_{mj} = product of the loop gains of the m^{th} set of j non-touching loops;

Δ_k = the value of Δ for that part of the graph not touching the k^{th} forward path.

Since the calculation of the loop- and path-gains is straightforward once the loops and paths are identified (as edge sets), the paper is concerned principally with the topological aspect of solving the flowgraph gain, that is, finding the loops and paths themselves.

---

*Use of terms "loop" and "path" implies that both are unidirectional.

+ This work was supported in part by the National Aeronautics and Space Administration, Grant No. NGR-05-017-012.

2. DEFINITIONS

It is assumed that a flowgraph Γ is given of e edges (i.e. branches) and v vertices (i.e. nodes or "variables").

Definition 1. A k^{th}-order loop set is a set of k non-touching loops. Each loop within the set is a k^{th}-order loop.

Definition 2. The loop matrix L_1 of Γ has e columns each corresponding to a distinct edge in Γ and rows each corresponding to a distinct loop in Γ; the (i,j)-element of L_1 is a "1" if the i^{th} loop includes the j^{th} edge and is "0" otherwise.

Definition 3. A k^{th}-order loop matrix L_k of Γ has e columns each corresponding to a distinct edge in Γ and rows each corresponding to a k^{th}-order loop set in Γ ; the (i,j)-element of L_k is a "1" if the i^{th} loop set includes the j^{th} edge and is "0" otherwise.

Definition 4. The k^{th} loop adjacency matrix $\mathcal{L}_k$ is a matrix each row of which corresponds to a k^{th}-order loop set of Γ and each column of which corresponds to a first-order loop of Γ . The (i,j)-element of $\mathcal{L}_k$ is "1" when the i^{th} loop set of order k touches the j^{th} first-order loop and is "0" otherwise.

Definition 5.[4] The non-oriented incidence matrix A_m of Γ has rows corresponding to the vertices of Γ and columns corresponding to the edges of Γ . The (i,j)-element of A_m is "1" if the j^{th} edge is incident to (i.e. "touches") in the i^{th} vertex and is "0" otherwise.

Definition 6.[4] The oriented f-circuit matrix B_{mf} of Γ has rows corresponding to the fundamental (f-) circuits of Γ and columns corresponding to the edges of Γ . The orientation of each f-circuit is chosen in the clockwise sense. The (i,j)-element is "+1" if edge j is in f-circuit i and in the same direction as the circuit; is "-1" if in the i^{th} f-circuit but in the opposite direction, and is "0" otherwise.

3. ALGORITHM FOR FINDING THE LOOPS AND PATHS IN THE MASON SIGNAL FLOWGRAPH

The following Theorem and Lemma which are stated without proof[5] comprise the algorithm of this paper.

THEOREM. Given a connected Mason flowgraph G_m of e edges and v vertices, the loop and path sets corresponding to Mason's gain formula for the gain $G_{i\to j}$ from vertex i to vertex j can be determined through application of the following steps:

Step 1 - Form the graph G'_m by adding to G_m the edge e' (of unity gain) directed from vertex j to vertex i.

Step 2 - Write the non-oriented incidence matrix A_m and the oriented f-circuit matrix B_{mf} for the graph G'_m.

Step 3 - Form L_1 using the steps outlined in the Lemma (below).

Step 4 - Form the (Boolean) product $\xi = A_m^T \cdot A_m \cdot L_1^T$.

Step 5 - Starting with $k = 1$, calculate the (Boolean) product

$$\mathcal{L}_k = L_k \cdot \xi = [\ell_{ij}] .$$

Evaluate all ℓ_{ij}, and where $\ell_{ij} = 0$ add the i^{th} row of L_k to the j^{th} row of L_1 and let the sum form a row of a new matrix L'_{k+1}. Remove all redundant rows of L'_{k+1} and let the remaining rows form L_{k+1}. Continue this process until a matrix $\mathcal{L}_n$ is calculated which contains no "0" elements.

Step 6 - List (or store) matrices $L_1, L_2, \ldots, L_n$ found in step 5. For each row of each matrix examine the column representing e'. If the element value is "1", the other "1" elements in the corresponding row denote a path and loop set for the numerator term of Mason's gain formula for G_m. If the element value is "0", the "1" element values in the corresponding row denote a loop set in the denominator term of the gain formula. Such examination of all rows (of all matrices) provide all the numerator and denominator terms.

LEMMA. L_1 is obtainable from B_{mf} through application of the following steps:

Step 1 - Systematically perform all possible row-sums on rows of B_{mf} retaining only those sums containing either all "+1"s or all "-1"s. Let these form rows of a new matrix L'_1.

Step 2 - Remove signs prefixing elements in L'_1 thereby forming L_1''.

Step 3 - Remove redundant and edge disjoint rows of L_1'' thereby forming L_1.

---

*A row-sum of two rows r_1 and r_2 of B_{mf} is the sum of $\pm r_1$ and r_2 such that the resultant row contains entries that are "+1", "-1" or "0". Thus it is sometimes necessary to multiply a row of B_{mf} by -1 before adding it to another row to form the row-sum, since entries with value "2" are not permissible.

4. EXAMPLE

The following example serves to illustrate the algorithm.

Example: Given G_m as shown in Fig. 1. We wish to find terms for $G_{1\to 3}$.

Step 1 - Add e' to form G'_m. (Fig. 2)

Step 2 - Form A_m and B_{mf}:

$$A_m = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & e' \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \end{array} \qquad B_{mf} = \begin{array}{c} \begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & e' \end{array} \\ \begin{bmatrix} 0 & +1 & +1 & 0 & 0 & 0 & 0 & 0 & 0 \\ +1 & 0 & +1 & 0 & 0 & 0 & 0 & 0 & +1 \\ +1 & 0 & +1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & +1 & -1 & 0 & +1 & 0 & 0 \\ 0 & 0 & 0 & 0 & +1 & +1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Step 3 - Form L_1 (Lemma)

$$L_1 = \begin{array}{c} \\ \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \\ \ell_5 \\ \ell_6 \end{array} \begin{array}{c} \begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & e' \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \end{array}$$

Step 4 - Form $\xi = A_m^T \cdot A_m \cdot L_1^T$ (Boolean)

Step 5 - For k = 1 calculate $\mathcal{L}_1$ and L_2:

$$\mathcal{L}_1 = L_1 \cdot \xi = \begin{array}{c} \\ \ell_1 \\ \ell_2 \\ \ell_3 \\ \ell_4 \\ \ell_5 \\ \ell_6 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \end{array} \qquad L_2 = \begin{array}{c} \ell'_1 \\ \ell'_2 \\ \ell'_3 \end{array} \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

For k=2,

$$\mathcal{L}_2 = L_2 \cdot \xi = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

The process terminates since $\mathcal{L}_2$ has no "0" entries.

Loop sets (obtained from the rows in the above matrices) for the denominator and numerator terms in Mason's gain expression $G_{1\to 3}$ are shown in Figs. 4 and 5 respectively. The reader can verify that these are all the loops sets for the graph.

5. CONCLUSIONS

The algorithm given by the Theorem and the Lemma provides a machine programmable means for determining the paths and loop sets that are necessary to implement Mason's gain formula. The reader can easily verify that the numerical and/or symbolic expressions of the gain formula can be obtained at once from matrices $L_1, L_2, \ldots, L_n$ and a listing of the branch gain values for the flowgraph.

In terms of computer arithmetic and memory requirements the algorithm is very efficient: (1) except for the formation of L_1 all arithmetic operations are in binary arithmetic (as contrasted to symbolic methods empolying Boolean algebra); (2) storage requirements are minimal. For example, B_f can be eliminated after forming L_1 ; A_m can be eliminated after forming ξ . Similarly, each $\mathcal{L}_k$ can be eliminated after forming L_{k+1}. Consequently, considering $L_1, L_2, \ldots, L_n$ as being the solution, working memory is conveniently confined to ξ and the current value of $\mathcal{L}_k$.

REFERENCES

1. Mason, S.J., "Feedback Theory--Some Properties of Signal Flow Graphs," Proc. IRE 41, 1144-1156, Sept. 1953.
2. Mason, S.J., "Feedback Theory--Further Properties of Signal Flow Graphs," Proc. IRE 44, 920-926, July 1956.
3. Happ, W.W., "Flowgraph Techniques for Closed Systems," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-2, No. 3, pp 252-263, May 1966.
4. Chan, S.P., <u>Introductory Topological Analysis of Electrical Networks</u>, to be published by Holt, Rinehart and Winston, 1968.
5. W.R. Dunn, Jr. and S.P. Chan, "An Algorithm for Finding the Loops and Paths in a Mason Signal Flowgraph," to be published.

FIGURES

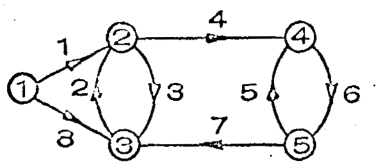

FIG. 1. THE MASON GRAPH G_m

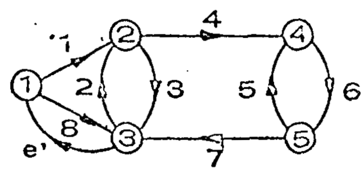

FIG. 2. GRAPH G'_m

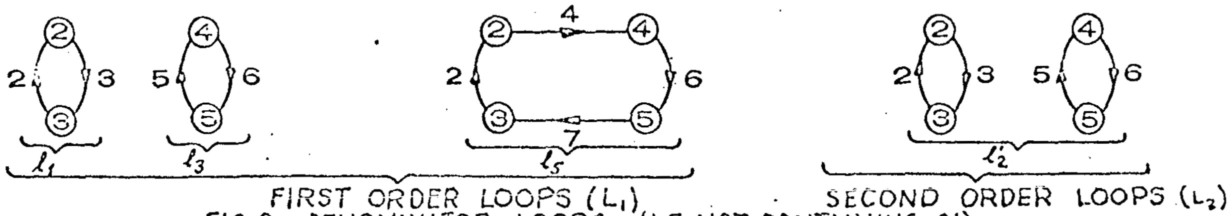

FIG. 3. DENOMINATOR LOOPS (I.E. NOT CONTAINING e')

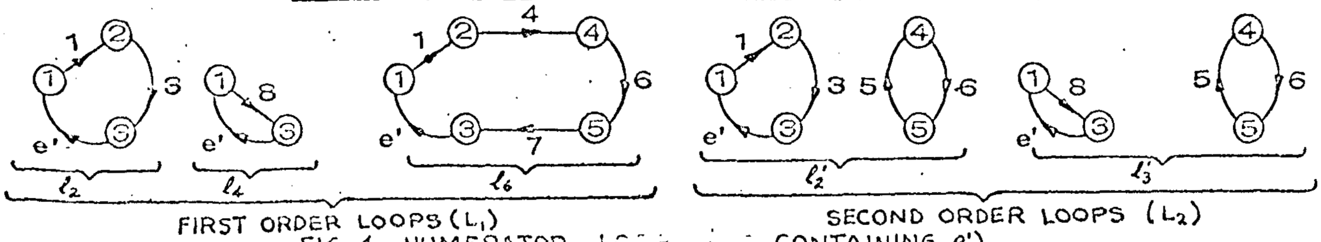

FIG. 4. NUMERATOR LOOPS (I.E. CONTAINING e')