DRIVING-POINT IMPEDANCE SYNTHESIS
WITH TOPOLOGICAL SPECIFICATIONS

By

LOUIS CHARLES THOMASON

Bachelor of Science
Oklahoma State University
Stillwater, Oklahoma
1961

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1961

Submitted to the faculty of the Graduate College of
the Oklahoma State University
in partial fulfillment of the requirements
for the degree of
DOCTOR OF PHILOSOPHY
May, 1966

DRIVING-POINT IMPEDANCE SYNTHESIS

WITH TOPOLOGICAL SPECIFICATIONS

Richard L. Cummins
Thesis Adviser

Wm L Hughes

Harold Fristoe

William J. Lewis

E. K. McLachlan

J H Boggs
Dean of the Graduate College

ACKNOWLEDGEMENTS

The general subject of this thesis was suggested by Dr. R. L. Cummins, my Thesis Adviser. I wish to express my appreciation to him for his counsel and guidance during its preparation.

The interest and encouragement of Dr. W. L. Hughes, Head of the School of Electrical Engineering and Chairman of my advisory committee is gratefully acknowledged.

For their advice and the challenge they have provided in the classroom, I wish to thank the other members of my advisory committee, Drs. H. T. Fristoe, E. K. McLachlan, and W. J. Leivo.

The contribution of my wife and two daughters to my graduate studies was immeasurable. Particular recognition is due my wife, Karen, for typing the thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

<u>1.1 General Discussion of the Problem.</u> A large volume of information about the synthesis of electrical networks has been developed during the last thirty years. However, it is now clearly recognized that classical techniques have ignored the topological properties of the design. Seshu (1) in an early paper discussing topology and synthesis stated that there are "... several methods of synthesizing driving-point functions known at the present time and they are practically satisfactory. Esthetically, however, they are unsatisfactory in that one of the most important characteristics of a network, namely its topology, has been neglected."

Interest in this aspect of circuit synthesis is a natural result of the advanced state of classical procedures and the development of topological methods of analysis. Since the time of Seshu's first paper in 1955, the requirements of circuit designers have changed. There is, for example, a need to control the topology when designing a network to be constructed by integrated circuit techniques. As topological studies continue, it appears that other important synthesis problems, such as specifying several functions to be realized by a single network, may be solved. Thus the study seems to have far-reaching practical, as well as esthetic, value.

The circuit designer is directly concerned with three attributes of the network: the specified function, the topology, and the element values. Classical synthesis procedures begin with a specified network function and, by one of several different techniques, derive both the topology and the element values. In certain cases, especially the two-element-kind network, there is a limited choice of topology, i. e. parallel, series, or ladder circuits. Such a choice falls far short of being topological design. Thus the general problem at hand is to develop circuit synthesis procedures which allow the circuit designer to control the topology.

1.2 Review of Literature. Seshu's paper, quoted previously, derived the fundamental circuit matrix and the incidence matrix of the network from the specified driving-point function and certain elementary functions. These represent the resistors, capacitors, and inductors of the network and their values. Seshu did not present a procedure to obtain the elementary functions from the specified network function but declared that such a procedure would be necessary before the method could be practical.

In 1960, Onodera (2) developed a method for the topological synthesis of networks from the transfer admittance matrix. His procedure derived the incidence matrix and the branch impedance matrix. This paper differs from our present objective, however, as the specified function is a matrix function of the network. No general procedure is known to obtain the matrix from a specified network function.

Iterative methods for finding the element values of a network with a specified function and topology are proposed by Bellert (3) and Calahan (4). Bellert's algorithm for topological synthesis generates a sequence of networks, and he suggests that each one be tested by iterative numerical methods to see whether it will realize the specified function or not. Calahan has prepared a computer program to determine the element values of a network from a specified function and topology. If the iteration process does not converge to a solution, Calahan's program automatically adds an element to the network and attempts to solve it again. The difficulty is that the user must make an 'appropriate' choice of starting values for the iteration. If it fails to converge, either the network cannot be realized in the specified topology or a bad starting point has been chosen. Experience indicates that this situation occurs frequently in all but simple examples.

The work of Seshu presents quite a contrast to that of Bellert and Calahan. The former attempts to derive the network topology, while the latter are suggesting that it be tested. Thus it is reasonable to question whether or not one should look for a procedure to derive a topology from a specified function. An uncountable number of networks can realize a specified function (if it can be realized at all). Thus if a topology is to be derived, it appears necessary to make additional specifications. For example, Seshu specified the element values while Onodera specified a system matrix. In accordance with our general objective, any additional specifications should relate specifically to the topology. Now the original question is rephrased. What properties of the topology are determined by the specified function, and how are these related to the element values? Once an answer is found, one may look for a topological synthesis procedure.

1.3 Delineation of the Problem. The present objective is to study the driving-point impedance function of passive linear networks. Mutual inductance is excluded to further limit the scope of the problem. It is assumed that this function is realizable, that is, it is positive real and hence satisfies the classical test for realizability. Principles developed for the impedance function can be applied in a similar manner to driving-point admittance functions. Perhaps such an investigation might be extended to other network functions, such as the transfer impedance function.

The two questions proposed in Section 1.2 are applied specifically in this thesis to the driving-point impedance. Topics for consideration are:

(a) What is the nature of the relationship between the form of the driving-point impedance and the network topology?

(b) Can a given network topology realize a specified driving-point impedance?

(c) What are the network element values?

In addition, a classification of networks is suggested as the basis of a topological synthesis procedure.

1.4 Organization of the Thesis. The topics involving only topology are discussed in Chapter II. Relations between the form of the driving-point impedance function and the network topology are derived and presented in a table, and network classification is discussed.

Questions (b) and (c) are, as in some classical synthesis methods, allied. A general discussion will be presented in Chapter III relating the driving-point impedance to the network topology by sums of tree admittance products. Chapter IV describes a method for solving these sums to determine the element values and the realizability of the network topology. Several examples are presented.

CHAPTER II

RELATIONS BETWEEN THE DRIVING-POINT IMPEDANCE FORM AND THE NETWORK TOPOLOGY

2.1 Introduction and Objective. The objective of this chapter is to make explicit the relations between the form of the driving-point impedance function (to be abbreviated ZDP) and the network topology. Several papers are of special interest in this discussion because they deal with the ZDP form. Hakimi and Mayeda (5) have shown that a necessary and sufficient condition for a network function polynomial to be even or odd is that the number of resistors in all trees of the network be constant. Brown and Reed (6) have developed detailed conditions on the ZDP form based on classical positive real conditions.

Networks with two kinds of elements have been studied by Hakimi (7), and their topological properties are related to the number of poles and zeros of network functions. His work is also extended to include networks with three kinds of elements. Similar results have been obtained independently by others, Seshu and Reed (8).

2.2 Driving-point Impedance Forms. Two forms of driving-point impedances are of interest. The determinant or improper form representing the most general function is defined by Equation 2.2.1.

$$Z_{dp} = \frac{c_k s^k + c_{k-1} s^{k-1} + \dots c_0 + \dots c_{-1} s^{-1}}{d_p s^p + d_{p-1} s^{p-1} + \dots d_0 + \dots d_{-q} s^{-q}} \qquad 2.2.1$$

This form represents the arbitrary fashion in which a ZDP specification may be prepared. It is assumed that

$$c_k, c_{-1}, d_p, d_{-q} \neq 0$$

and that the zero coefficients are specified. The numerical values of the non-zero coefficients remain unspecified.

As an aid to generalization, a normal or proper form is defined by Equation 2.2.2.

$$Z_{dp} = \frac{a_m s^m + \dots a_1 s + a_0}{b_n s^n + \dots b_1 s + b_0} \qquad 2.2.2$$

It is obtained from the determinant form by multiplying both numerator and denominator by s^x, where $x = \max(1, q)$. Note that a_m and $b_n \neq 0$; however, either a_0 or b_0 may be zero. The zero coefficients are also specified here, while the value of the non-zero coefficients is unspecified.

A normal form has six attributes. They are:

(a) the value of m

(b) a_0 specification (zero or non-zero)

(c) alternating numerator (even or odd polynomial)

(d) the value of n

(e) b_0 specification (zero or non-zero)

(f) alternating denominator (even or odd polynomial)

Extensive use of these attributes will be made in the following sections. It is clear that an uncountable number of determinant forms reduce to a single normal form.

Relations between several of the attributes are presented in various texts on classical synthesis; the reader is referred to Weinberg (9) for one such development. The integer values m and n can be shown, for example, to differ by not more than one. These conditions are deduced from the positive real test for realizability and are only implicitly involved in the development presented here.

2.3 Graph-Theoretic Principles. The following is a unified graph-theoretic presentation of definitions and theorems required for this discussion of topological synthesis. A comprehensive treatment of linear graph theory and the analysis of electrical networks can be found in the standard text by Seshu and Reed (8).

It is necessary to differentiate between the linear graph describing the topology of a network and the network electrical components; the basic definitions reflect this distinction.

Definition 2.3.1 (Edge): An edge is a line segment with distinct end points.

Definition 2.3.2 (Element): A network element is an edge identified with a resistor, capacitor, or inductor.

An edge is later identified with a color. However, this is not to be thought of as a property of the edge but simply an aid to visualizing a classification. The number of edges or elements will be denoted by E with an appropriate subscript if necessary.

Definition 2.3.3 (Vertex): The end points of an edge or element are called vertices.

The term node is frequently used in the literature as another name for vertex. V will designate the number of vertices.

Definition 2.3.4 (Graph): A graph is a set of edges coinciding only at vertices.

Definition 2.3.5 (Network): A network is a set of elements coinciding only at vertices.

Here again there is a distinction between the topological arrangement and the network. The terms subgraph and subnetwork will be used to denote graphs and networks containing subsets of edges and elements respectively. Unless stated otherwise, the theorems will remain valid when the graph edges are identified with electrical components. In certain cases it will be necessary to discuss graphs having one or more isolated vertices, i. e. vertices not touched by an edge.

Several properties of a graph are now considered.

Definition 2.3.6 (Nonseparable): A graph G is nonseparable if every subgraph of G has at least two vertices in common with its complement. All other graphs are separable.[1]

In order to designate clearly the separable subgraphs of a graph, a related term is defined below.

[1]A definition given by Seshu and Reed (8), this conveys precisely the concept of interest here.

Definition 2.3.7 (Component): A maximal nonseparable subgraph of a graph G is a component of G .

Special emphasis is placed on another class of subgraphs distinguished by the following property.

Definition 2.3.8 (Connected): A graph is said to be connected if there exists a path or sequence of edges between any two vertices.

Definition 2.3.9 (Part): A part of G is a maximal connected subgraph of G .

The number of parts of a graph will be denoted by P . It is clear that a part of G will always be a component or perhaps more than one component. On the other hand, if G is connected and nonseparable, it will be regarded as a component of itself.

Definition 2.3.10 (Rank): The rank of G is $R = V - P$.

This term will be used frequently in the discussion.

The tree concept defined below is the vehicle for developing the relations between the ZDP form and the network topology. To utilize this important graphical concept to the full extent, the definition here differs from that commonly used in electrical engineering.

Definition 2.3.11 (Tree Graph): A tree graph of a graph G of rank R is a subgraph of rank R having R edges.

This definition may be shown to be equivalent to the conventional one if the graph is connected.

Definition 2.3.12 (Tree): A set of edges which form a tree graph is called a tree.

This second definition is provided to distinguish the tree graph from the set of symbols corresponding to the edges of the tree graph. It will frequently be necessary to refer to the tree t_i and its edge set,

$$t_i = \left\{ e_{i_1}, e_{i_2}, \ldots e_{i_R} \right\}$$

To represent conveniently the complete set of trees of a graph, the tree set

$$T = \left\{ t_1, t_2, \ldots t_N \right\}$$

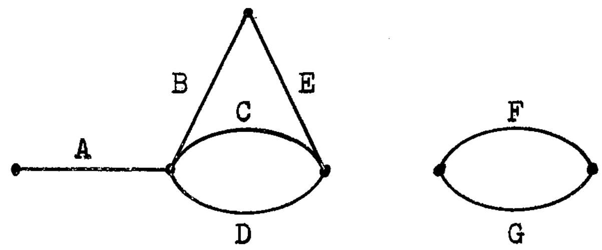
is defined.

Definition 2.3.13 (Tree set): The tree set of G is the set of all trees of G .

These definitions are now illustrated by referring to the graph in Figure 2.3.1. This graph is composed of three components denoted by the edge sets $\{A\}$, $\{B, C, D, E\}$ and $\{F, G\}$. The subgraphs $\{A, B, C, D, E\}$ and $\{F, G\}$ are by definition parts of the graph. The tree set is also shown in the figure.

The theorems to be developed can be conveniently stated and proved in terms of operations on a graph.

Operation 2.3.1 (Edge deleting): A specified edge is removed from G . If an isolated vertex is created by an edge-deleting operation, it is also removed.



Tree Set

ABCF ABDF ABEF ACEF ADEF
ABCG ABDG ABEG ACEG ADEG

Figure 2.3.1. Example for Graph Definitions

The edge deleting operation is extended to the subgraph.

Operation 2.3.2 (Subgraph deleting): An edge-deleting operation is performed on each edge of a specified subgraph.

Operation 2.3.3 (Vertex shorting): Two specified vertices v_i and v_j are superimposed. Edges with both endpoints on the new vertex, self-loops, are deleted.

As most vertex-shorting operations are designated by an edge, it is convenient to make the following definition.

Operation 2.3.4 (Edge shorting): A vertex-shorting operation is performed on the vertices denoting the endpoints of a designated edge.

Note that self-loops are again removed. This operation is also extended to subgraphs.

Operation 2.3.5 (Subgraph shorting): An edge-shorting operation is performed on each edge of the specified subgraph.

The theorems that follow develop the necessary and sufficient conditions relating the form of the ZDP and the network. They are intended to be constructive in nature, that is, they are stated and proved in a fashion suitable for use in an algorithmic process. For an example of such an application the reader is referred to (10).

The first five theorems prescribe the effect of the corresponding five operations on the tree list of a graph.

Theorem 2.3.1 (Deleted edge theorem): A graph G' is formed by an edge-deleting operation on e_d.

(a) If the rank of G' is equal to the rank of G,

  (1) every tree of G' is a tree of G, and

  (2) every tree of G which does not contain e_d is in the tree set of G'.

(b) If the rank of G' is not equal to the rank of G,

  (1) $R' = R - 1$, where the ranks of G' and G are R' and R respectively, and

  (2) every tree of G' will be a tree of G if e_d is added.

Proof (a): Since the rank has not changed and G' is a subgraph of G, its trees are trees of G. In addition, a tree graph of G which does not contain e_d is a subgraph of G'. This subgraph of G' has appropriate rank and number of edges to be a tree of G'.

Proof (b): Deleting an edge does not change the number of vertices, but may increase the number of maximal connected subgraphs by one. Thus if the rank of G' is not equal to the rank of G,

$$R' = (V - P) - 1 = R - 1$$

Note that a tree graph of G' will be a subgraph of G of rank $R-1$ with $R-1$ edges. Since deleting e_d in G divided a maximal connected subgraph into two parts, adding e_d to a tree graph of G' will connect the corresponding parts of the tree graph. This newly formed graph will be a subgraph of G having rank R and R edges; thus it is a tree.

Theorem 2.3.2 (Shorted vertex theorem): A graph G' of rank R' is formed from graph G of rank R by a vertex-shorting operation on v_i and v_j.

(a) If and only if v_i and v_j are in separate parts of G,

(1) $R = R'$, and

(2) the tree set corresponding to G' is identical to that of G.

(b) If and only if v_i and v_j are in the same part of the graph G, $R' = R - 1$.

Proof (a): The number of vertices and parts of G are both reduced by one; thus the rank is unchanged. Every tree of G is by definition a tree graph of G' after a vertex reduction is performed on v_i and v_j. In a similar way separating the reduced vertex of any tree graph of G', while maintaining the vertex-edge incidence relation of G, will produce a tree graph of G. Thus the tree sets are identical.

Proof (b): In this case the number of parts of G and G' is the same, while the number of vertices of G' is one less than that of G. Thus the rank of G' is one less than G.

Theorem 2.3.3 (Shorted edge theorem): One edge of a graph G is selected and designated e_s. A graph G' is formed by an edge-shorting operation on e_s. The tree set of G' is designated

$$T' = \left\{t'_1, t'_2, \ldots t'_n\right\} .$$

A second set of edge sets

$$T = \left\{t_1, t_2, \ldots t_n\right\} ,$$

is formed by adding e_s to each t'_i to form t_i.

(a) Every set of edges t_i is a tree of G.

(b) Every tree of G that contains e_s is included in the tree set T.

Proof (a): Each t' is a subgraph of G' of rank R-1 having R-1 edges. The vertices shorted by the edge-shorting operation are separated while maintaining the vertex-edge incidence relation of G. When the edge e_s is added between the separated vertices, a subgraph of G with rank R and R edges is formed.

Proof (b): Considering any tree graph of G which contains e_s, an edge-shorting operation on e_s produces a subgraph of G' of rank R-1 with R-1 edges. This subgraph is by definition a tree of G'.

Theorem 2.3.4 (Shorted graph theorem): A set of edges shorted one by one (no edge is removed as a result of shorting a previous edge) reduces a graph G to zero rank if and only if the edge set is a tree of G.[1]

Proof: Each edge of the set of N shorted edges is designated e_i and the graph formed, G_i. The tree of G_N is the null set of edges. According to the shorted edge theorem (2.3.3), e_N is a tree of G_{N-1}. Continuing to apply this reasoning, $\{e_{N-1}, e_N\}$ is a tree of G_{N-2}, etc. Finally, $\{e_1, e_2, \ldots e_N\}$ is a tree of G.

Assuming that shorting a set of edges corresponding to a tree in G did not reduce the graph to rank zero, then an additional set of edges could be shorted to reduce the rank to zero. As described above, the union of this second set of edges with the tree set would be a tree. This contradicts the assumption, since the new tree would have more edges than the assumed tree. Thus shorting the edges of a tree reduces G to rank zero.

A discussion of the interrelation of the tree and the ZDP is presented in Section 2.4. The following theorems provide the desired association between the properties of the graph and the tree set. Several of them correspond to theorems by Hakimi (7). However, the present discussion is entirely graph-theoretic and unified. In addition, the theorems here stated are in a form suitable to apply to graphs of more than one part.

Colors will be used to denote classes of edges. The subscripts w and b will denote subgraphs of white and black elements. The black subgraph is the complement of the white, and vice versa. The s subscript designates a graph that is formed by shorting the sub-

[1] This theorem corresponds to part of an algorithm for listing all of the trees of a graph described by Minty (11).

graph which is the complement of the designated edge set. For example, G_{ws} is derived by shorting the non-white edges. In a similar fashion d indicates that the graph is produced by a subgraph-deleting operation on the complement of the indicated edge set. Thus G_{bd} is obtained by deleting all non-black edges.

In some applications of the theorems below the graph will be composed of three classes of edges. When this occurs, two of the classes are treated as a single class. For example, the edges of a graph are classified as red, blue, or green. If the red edges are of particular interest, they are assigned to the white class, while the blue and green are assigned to the black class.

The following theorem presents the basic relation between the derived graphs G_{ws} and G_{bd} and a tree.

Theorem 2.3.5 (Composite tree theorem): If

(a) G is a graph with derived graphs G_{ws} and G_{bd} ,

(b) t_w is a tree of G_{ws} ,

(c) t_b is a tree of G_{bd} ,

then $t = \left\{t_w \cup t_b\right\}$ is a tree of G .

Proof: To prove the theorem the edges of the tree t_b are shorted in G_{bd} and G . Since, according to the shorted graph theorem (2.3.4), G_{bd} is now reduced to zero rank, all of its black edges have been removed. Thus all black edges in G have also been removed and G_{ws} remains. Shorting the tree t_w in G_{ws} reduces G to rank zero. The shorted graph theorem is again applied to identify the set of edges $\left\{t_w \cup t_b\right\}$ that reduce G to rank zero as a tree of G .

As will be seen later in this chapter, the maximum and minimum numbers of edges of a particular class in any tree are of primary importance. They are determined by the ranks of derived graphs.

Theorem 2.3.6 (Minimal tree theorem): If the graph G_{bs} derived from G has rank R_{bs}, every tree of G contains at least R_{bs} black edges.
Proof: By the composite tree theorem (2.3.5), a tree with R_{bs} black edges exists. Assuming that a tree with fewer than R_{bs} black edges exists, then shorting the non-black edges of this tree in G would produce a derived graph G' of rank R' less than R_{bs}, which must contain a black edge tree. This is a contradiction of the hypothesis since the rank of a graph formed by shorting any subset of non-black edges must be greater than or equal to R_{bs}. Shorting the remaining non-black edges can only reduce the rank.

Theorem 2.3.7 (Maximal tree theorem): If the graph G_{wd} derived from G has rank R_{wd},

(a) at least one tree of G contains R_{wd} white edges, and

(b) no tree contains a greater number of white edges.

Proof (a): The non-white elements of G are identified as black elements. The ranks of G_{bs} and G_{wd} are R_{bs} and R_{wd} respectively. If t_b is a tree of G_{bs} and t_w is a tree of G_{wd}, then, according to the composite tree theorem (2.3.6), $t = \{t_b \cup t_w\}$ is a tree of G. Note that t_w contains R_{wd} white edges; thus t contains R_{wd} white edges.

Proof (b): Assuming a tree graph exists with $R' > R_{wd}$ white edges, this tree must be a subgraph of G. In addition, the R' white edges must be a subgraph of G_{wd}. However, the G_{wd} graph has rank R_{wd} and no subgraph can have greater rank. This is verified by considering that G_{wd} has V vertices and P_{wd} parts. Any subgraph G' of G_{wd} with V' vertices and P' parts must have

$$V' \leqslant V$$

and

$$P' \geqslant P_{wd} .$$

Thus

$$V' - P' = R' \leqslant R'_{wd} = V - P_{wd} .$$

Having thus specified the extreme characteristics of the tree set, the next requirement of the development deals with the properties of the set between these extremes.

Theorem 2.3.8 (Tree sequence theorem): The tree set of a graph contains trees with K white edges for all K such that $R_{ws} \leqslant K \leqslant R_{wd}$.

Proof: The minimal tree theorem (2.3.6) states that at least one tree contains R_{ws} edges, while the maximal tree theorem (2.3.7) verifies that a tree contains R_{wd} edges. A graph G corresponding to the tree set is formed. Each node corresponds to a tree of the set and each edge to an elementary tree transformation. This transformation involves replacing one edge of a tree with another edge forming a different tree of the set. R. L. Cummins (12) has shown that a Hamilton circuit exists in a graph of trees. "... the set of trees of a network (graph) can be ordered in such a manner that successive trees are related by elementary tree transformations." Note that an elementary tree transformation can remove at most one white element from tree t_i . Thus there exists a tree with K white elements for all K such that $R_{ws} \leqslant K \leqslant R_{wd}$. The maximum and minimum values of K are given by Theorems 2.3.6 and 2.3.7.

In certain special cases the maximum and minimum are identical. The component graph property is associated with such a condition.

Theorem 2.3.9 (Component graph theorem): A necessary and sufficient condition for the white elements of G to form a component or set of components of G is that $R_{ws} = R_{wd}$.

Proof: Note that the G_{ws} graph can be obtained from G_{wd} by shorting the vertices which were connected by non-white edges in G. If the white edges of G form a component or set of components, the process of deriving G_{ws} from G_{wd} does not change the rank. The rank is now assumed to be changed by a shorting operation. Thus, according to the shorted vertex theorem (2.3.2), the two vertices must be in the same part of the graph. Since the only connected subgraph of two or more vertices in G_{wd} is a white edge subgraph, a non-white edge exists between two vertices in a white subgraph of G. Such a white subgraph has at least two vertices in common with its complement and by definition is nonseparable, i. e. not a component. This contradicts the hypothesis; thus the rank is unchanged in the process of deriving G_{ws} from G_{wd}.

Now it is assumed that G_{ws} is obtained from G_{wd} and the rank is not changed in the process. From the shorted vertex theorem (2.3.2) it is known that the vertices which were shorted must have been in separate parts of the graph. Extending this reasoning, no sequence of shorting operations used to obtain G_{ws} from G_{wd} will involve shorting two vertices which are common to a one-part subgraph. Thus, there exists no white one-part subgraph in G which has two or more vertices in common with a non-white subgraph; hence by definition, all white subgraphs are components.

An additional short theorem will prove useful.

Theorem 2.3.10 (Component tree theorem): Every tree in the tree set of a graph G has the same number of white edges if and only if the white edges form a component or set of components of G.

Proof: This theorem follows directly from the tree sequence theorem (2.3.8) and the component graph theorem (2.3.9). The number of white edges K in each tree of G is bounded by $R_{ws} \leq K \leq R_{wd}$. But $R_{ws} = R_{wd}$, and the number of white edges in each tree is the same.

2.4 Network Forms. Figure 2.4.1 represents a one-port network with the two input terminals identified.

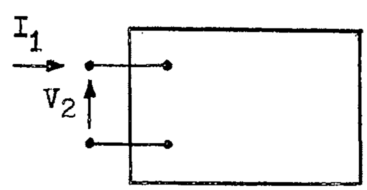


Figure 2.4.1. One-Port Network

The driving-point impedance is defined to be

$$Z_{dp} = \frac{V_1}{I_1}$$

and can be written as

$$z_{dp} = \frac{\Delta_{11}}{\Delta}$$

where Δ and Δ_{11} are the determinant and y_{11} cofactor respectively of the node-admittance matrix.

Symbols associated with the graph and topological properties of the one-port network will be denoted by a 0 subscript. A derived graph, formed by shorting the input terminals of G_0, is of special importance and its symbols will be designated by the subscript 1.

Percival (13) formalized the early work of Maxwell (14) to develop equations giving the determinants above in terms of the trees of G_0 and G_1. These are stated without proof; for further detail the reader is referred to Seshu and Reed (8).

<u>Definition 2.4.1 (Tree-admittance product):</u> The tree-admittance product t_i is the product of the admittances of the elements corresponding to the edges in t_i.

<u>Theorem 2.4.1:</u> If G_0 is a graph corresponding to a connected passive network without mutual inductance, the node-admittance matrix determinant of the network is

$$\Delta = \sum_{\substack{\text{tree} \\ \text{set} \\ \text{of } G_0}} (\text{tree-admittance product of } t_i) \qquad 2.4.1$$

<u>Theorem 2.4.2:</u> If G_1 is a graph derived from a G_0 satisfying the hypothesis of Theorem 2.4.1, the cofactor Δ_{11} of the node-admittance matrix is

$$\triangle_{11} = \sum_{\substack{\text{tree}\\ \text{set}\\ \text{of } G_1}} (\text{tree-admittance product of } t_i) \qquad 2.4.2$$

As the cofactor $\triangle_{11}$ is the determinant of a network derived from the original by shorting the input, it will be convenient to refer to both $\triangle$ and $\triangle_{11}$ as determinants. Representing each element of the network by its transform admittance, the determinants become polynomials in s.

Theorem 2.4.3 (Alternating-term theorem): A network determinant has alternating terms if and only if the R element subgraph is a component or set of components.

Proof: A network determinant has alternating terms if and only if each tree has the same number of resistors.

If each tree has the same number of resistors, every elementary tree transformation in a Hamilton circuit through the graph of trees will change the exponent of s by 0 or 2. Thus only alternating terms exist in the polynomial.

It is now assumed that an alternating polynomial can be formed by trees not having the same number of resistors. Then from the tree sequence theorem (2.3.8) it follows that there must be a tree with an even number of resistors and one with an odd number, a tree with an even number of reactive elements and one with an odd number. This contradicts the hypothesis since such a polynomial would not be alternating. Hence, the alternating polynomial trees must have the same number of resistors. The preceding corresponds to the hypothesis of the component tree theorem (2.3.10), and by its conclusion the R element subgraph must be a component or a set of components.

Theorem 2.4.3 specifies the ZDP-network relation involving two of the six attributes of the normal ZDP form. The remaining four are associated with the exponents of the determinant form k, l, p, q. For example, p is the exponent of s associated with the tree-admittance products for trees containing the maximum number of capacitors and the minimum number of inductors in the network. According to the maximal tree theorem (2.3.7), R_{Cd0} is the maximum number of capacitors in such a tree. If the resistors and inductors in this tree are denoted by X, the G_{Xs0} graph has rank R_{Xs0}. There are R_{Ls0} inductors in the trees of G_{Xso} that minimize this number (Theorem 2.3.6). This is the minimum number of inductors in any tree, and p becomes

$$p = R_{Cd0} - R_{Ls0} \qquad 2.4.3$$

In a similar manner q is determined. Here, however, the roles of capacitor and inductor are reversed as the minimal exponent of s is required. The result is

$$q = R_{Ld0} - R_{Cs0} \qquad 2.4.4$$

Applying this same reasoning to the shorted network with graph G_1, equations for k and l are obtained.

$$k = R_{Cd1} - R_{Ls1} \qquad 2.4.5$$

$$l = R_{Ld1} - R_{Cs1} \qquad 2.4.6$$

Finally, the exponents k, l, p and q determine the value of the exponents m and n by the reduction process described in Section 2.2. This yields

$$m = k + \max(1, q) \qquad 2.4.7$$

$$n = p + \max(1, q) \qquad 2.4.8$$

2.5 Nature of the Relationship Between the Network Form and the ZDP Form. The necessary and sufficient conditions for a network form to realize a specified driving-point impedance form are now derived. The first step is to classify the ZDP form in terms of its six normal form attributes listed on page 7. The major classification shown in Table 2.5.1 is based on the value (zero or non-zero) of the a_0, b_0 attribute. Note that the corresponding determinant condition is listed in the adjacent column. The fact that q and l differ by at most one can be shown by considering the rank relation in the shorted vertex theorem (2.3.2) and Equations 2.4.4 and 2.4.6. Thus since

$$q - l = (R_{Ld0} - R_{Ld1}) - (R_{Cs0} - R_{Cs1})$$

and from the theorem

$$R_{Ld0} - R_{Ld1} = \begin{cases} 0 \\ 1 \end{cases}$$

$$R_{Cs0} - R_{Cs1} = \begin{cases} 0 \\ 1 \end{cases}$$

the difference becomes

$$q - l = \begin{cases} +1 \\ 0 \\ -1 \end{cases}$$

For each class, the first equation in column three of Table 2.5.1 is obtained by substituting Equations 2.4.4 and 2.4.6 into the equation of column two. This result is the necessary and sufficient condition for a network form to correspond to the function class. The q and l

TABLE 2.5.1

DRIVING-POINT IMPEDANCE CLASSES

Class	Normal Form Attribute	Determinant Form Attribute	Necessary and Sufficient Graph Conditions
1	$a_0 \neq 0$ $b_0 \neq 0$	$q = l$	$R_{Ld0} - R_{Cs0} = R_{Ld1} - R_{Cs1}$ $m = R_{Cd1} - R_{Ls1} + R_{Ld1} - R_{Cs1}$ $n = R_{Cd0} - R_{Ls0} + R_{Ld0} - R_{Cs0}$
2	$a_0 = 0$ $b_0 \neq 0$	$q = l + 1$	$R_{Ld0} - R_{Cs0} = R_{Ld1} - R_{Cs1} + 1$ $m = R_{Cd1} - R_{Ls1} + R_{Ld0} - R_{Cs0}$ $n = R_{Cd0} - R_{Ls0} + R_{Ld0} - R_{Cs0}$
3	$a_0 \neq 0$ $b_0 = 0$	$q = l - 1$	$R_{Ld0} - R_{Cs0} = R_{Ld1} - R_{Cs1} - 1$ $m = R_{Cd1} - R_{Ls1} + R_{Ld1} - R_{Cs1}$ $n = R_{Cd0} - R_{Ls0} + R_{Ld1} - R_{Cs1}$

TABLE 2.5.2

DRIVING-POINT IMPEDANCE SUBCLASSES

Class	Attribute	Necessary and Sufficient Graph Conditions
0.0	no alternating polynomials	$R_{Rd0} \neq R_{Rs0}$ $R_{Rd1} \neq R_{Rs1}$
0.1	denominator alternating polynomial	$R_{Rd0} = R_{Rs0}$ $R_{Rd1} \neq R_{Rs1}$
0.2	numerator alternating polynomial	$R_{Rd0} \neq R_{Rs0}$ $R_{Rd1} = R_{Rs1}$
0.3	denominator and numerator alternating polynomial	$R_{Rd0} = R_{Rs0}$ $R_{Rd1} = R_{Rs1}$

attributes determine the maximum required in Equations 2.4.7 and 2.4.8. Again substituting for k, l, p, q, these equations are the necessary and sufficient conditions for the network form to realize specific values of m and n.

Four subclasses shown in Table 2.5.2 are defined by the alternating polynomial attributes. This property of the polynomial is associated with the R element subgraph as explained in the alternating-term theorem (2.4.3). The rank equalities and inequalities are convenient tests for the component property of the R subgraph.

2.6 Classification of Network Forms. Example 3 in Chapter IV illustrates a network form realizing a specified function form. However, it is shown that a positive real function having the required ZDP form cannot be realized by this network form. In general, then, the form of the ZDP determines a set of network forms, at least one of which will satisfy a ZDP function. The number of network forms in this set is uncountable. However, as the objective here is to synthesize a topology with specific properties, only certain forms of this set are of interest. Hence, a classification is presented, the objective being to assist the designer determine network forms, at least one of which is realizable, satisfying the specified topological properties. Such a classification is not unique and, in fact, a different one may be desirable in some cases.

Table 2.6.1 shows the hierarchy of classes. That is, each row in the table defines the classification of subsets of each of the preceding sets. The symbols DA and $\overline{\text{DA}}$ have been used to denote an alternating and non-alternating polynomial in the denominator.

TABLE 2.6.1

CLASSIFICATION OF NETWORK FORMS

Classification	Definition of Class	Parameters Determined
vertex class	$C_V = \{V \mid V \geq 1 + \max(m, n)\}$	R_0
q class	$C_q = \{q \mid q \text{ integer}\}$	l, k, p
Ld0 class	$C_{Ld0} = \{R_{Ld0} \mid \max(q, 0) \leq R_{Ld0} \leq R_0\}$	R_{Cs0}
Ld1 class	$C_{Ld1} = \{R_{Ld1} \mid R_{Ld1} = [R_{Ld0} \text{ or } R_{Ld0} - 1] \text{ and } R_{Cs1} \geq 0\}$	R_{Cs1}
Cd0 class	$C_{Cd0} = \{R_{Cd0} \mid R_{Cd0} = R_{Ld0} \text{ if } DA,$ $\max(R_{Cs0}, P) \leq R_{Cd0} \leq R_0 \text{ if } \overline{DA}\}$	R_{Ls0}
Cd1 class	$C_{Cd1} = \{R_{Cd1} \mid R_{Cd1} = R_{Ld1} \text{ if } NA,$ $R_{Cd1} = [R_{Cd0} \text{ or } R_{Cd0} - 1] \text{ and } R_{Ls1} \geq 0 \text{ if } \overline{DA}\}$	R_{Ls1}

TABLE 2.6.1 (Continued)

LCd0 class	$C_{LCd0} = \{R_{LCd0} \mid R_{LCd0}$ rank of G_{LCd0} formed from G_{Ld0} and $G_{Cd0}\}$	R_{Rs0}
LCd1 class	$C_{LCd1} = \{R_{LCd1} \mid R_{LCd1} = R_{LCd0}$ if NA and DA, $R_{LCd1} = [R_{LCd0}$ or $R_{LCd0} - 1]$ if $\overline{NA}$ or $\overline{DA}\}$	R_{Rs1}
LCs0 class	$C_{LCs0} = \{R_{LCs0} \mid R_{LCs0} = R_{LCd0}$ if DA, R_{LCs0} rank of G_{LCs0} formed from G_{Ls0} and G_{Cs0} if $\overline{DA}\}$	R_{Rd0}
LCs1 class	$C_{LCs1} = \{R_{LCs1} \mid R_{LCs1} = R_{LCd1}$ if NA, $R_{LCs1} = [R_{LCs0}$ or $R_{LCs0} - 1]$ if $\overline{NA}\}$	R_{Rd1}

Similarly, NA and $\overline{NA}$ represent the numerator conditions.

The first classification of sets is based on the number of vertices. The minimum number is determined by the minimum rank of G_0.

$$V \geqslant 1 + \max(m, n)$$

There is, of course, no theoretical upper limit to the number V. Within each vertex class a division of function forms based on the integer q is made. This q corresponds to the exponent in the determinant form of the ZDP and can have any positive or negative value. The parameters l, k, p are determined by the q class, that is, they are the same for all networks in one set of the q class.

The Ld0 class designates the rank of G_{Ld0}. It can never be zero and by Equation 2.4.4, not less than q. Since G_{Ld0} is formed by deleting edges from G_0, the rank R_{Ld0} is not greater than R_0, previously determined by the vertex class. The rank R_{Cs0} is determined from Equation 2.4.4. The Ld1 and capacitor classes are developed in a similar fashion.

The Ld0 class is formulated to designate the sets resulting from all possible combinations of the G_{Ld0} and G_{Cd0} graphs having the same rank. As these may contain differing numbers of vertices, there are in general many possible G_{LCd0} graphs. This classification procedure is similarly carried out for the other LC graphs. In each case the rank of a resistor graph is determined.

The subsets are further divided by the numbers of elements. Several relations are involved here. For example, if N_L, N_C, N_R represent the number of inductors, capacitors, and resistors, the definition of rank (2.3.10) yields

$$N_L \geqslant R_{Ld0}$$

$$N_C \geqslant R_{Cd0}$$

$$N_R \geqslant R_{Rd0}$$

Other parameters may be chosen to distinguish the network classes. Hence, a network designer could develop a network classification based on the topological properties of particular interest to him.

CHAPTER III

REALIZATION OF A SPECIFIED DRIVING-POINT IMPEDANCE

3.1 Relation of the Network Form and Element Values. The term realization refers to the process of determining the element values of a network to satisfy a prescribed driving-point impedance. According to the discussion in Section 1.3, the first concern of the network designer is this. Can the network form realize a specified driving-point impedance? It is assumed here that the network topology satisfies the conditions specified in the tables in Chapter II. Now the actual coefficient values of the ZDP are considered and the synthesis procedure must test the network form for realizability.

If the network form will realize the ZDP, the next step is to determine the element values. When the realization test fails, another network form satisfying the specified topological properties is sought. More than one set of element values may realize the specified ZDP. The ideal synthesis procedure would make all such values available to the designer.

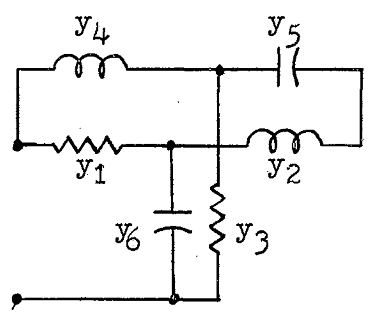
As was previously stated, Bellert (3) and Calahan (4) suggest iteration methods to realize the ZDP. Thus, the realizability test is combined with the determination of element values. Calahan's computer program utilizes the Newton-Raphson method of iteration.

There are several difficulties with this solution technique. It is not actually a test for realization since, if the iteration fails to converge, the designer is not assured that the network will not satisfy the ZDP specified. Initial estimates of the element values must be entered; thus an a priori knowledge of the approximate solution is required. Only one solution set is produced and a second set, if one is known to exist, may require a new iteration on a second initial estimate of the element values. The author's experience with such an iteration method indicates that the process frequently fails to converge. This is especially true of the non-standard network forms, i. e. those not composed of ladder, series, or parallel elements.

The author has also investigated realization by a transformation technique. In this approach, the desired network form is transformed to one of the standard or canonical forms, which is then realized by the standard techniques. The inverse transformation would then be applied to obtain the realization in the desired topology. Guillemin (15) has obtained a method for transforming networks with two kinds of elements, and in which the values are known, to an equivalent canonical form, the Foster networks. In general, however, the transformations are not known. Such a transformation and its inverse are not unique and thus probably are difficult to obtain.

A method of direct solution for the element values involving neither iteration nor transformation is presented in Chapter IV.

3.2 Solution of the Tree-sum Equations. The network form and the ZDP are explicitly related by the sum of tree admittance products defined by Equations 2.4.1 and 2.4.2. To illustrate this, the network shown in Figure 3.2.1 is considered.



$$y_1 = \frac{1}{R_1} = x_1$$

$$y_2 = \frac{1}{L_1}s^{-1} = x_2 s^{-1}$$

$$y_3 = \frac{1}{R_2} = x_3$$

$$y_4 = \frac{1}{L_2}s^{-1} = x_4 s^{-1}$$

$$y_5 = C_1 s = x_5 s$$

$$y_6 = C_2 s = x_6 s$$

Figure 3.2.1. Network Illustrating Tree-sum Functions

A computer program for listing all of the trees of such a network has been developed (10). The trees of G_0 and G_1 are listed in Tables 3.2.1 and 3.2.2.

TABLE 3.2.1

TREES OF G_0 IN FIGURE 3.2.1

$y_1\ y_2\ y_3\ y_4$	$y_1\ y_2\ y_5\ y_6$	$y_2\ y_3\ y_4\ y_6$
$y_1\ y_2\ y_3\ y_5$	$y_1\ y_3\ y_4\ y_5$	$y_1\ y_4\ y_5\ y_6$
$y_1\ y_2\ y_3\ y_6$	$y_1\ y_3\ y_5\ y_6$	$y_2\ y_4\ y_5\ y_6$
$y_1\ y_2\ y_4\ y_6$	$y_2\ y_3\ y_4\ y_5$	$y_3\ y_4\ y_5\ y_6$

TABLE 3.2.2

TREES OF G_1 IN FIGURE 3.2.1

$y_1\ y_2\ y_3$	$y_2\ y_3\ y_5$	$y_2\ y_4\ y_6$
$y_1\ y_2\ y_4$	$y_2\ y_3\ y_6$	$y_2\ y_5\ y_6$
$y_1\ y_2\ y_5$	$y_1\ y_4\ y_5$	$y_3\ y_5\ y_6$
$y_1\ y_3\ y_5$	$y_2\ y_4\ y_5$	$y_4\ y_5\ y_6$

Substituting the admittance of each element, the determinants are

$$\triangle = \left(\frac{1}{R_1R_2L_1L_2}\right)s^{-2} + \left(\frac{C_2}{R_1L_1L_2} + \frac{C_1}{R_2L_1L_2} + \frac{C_2}{R_2L_1L_2}\right)s^{-1}$$

$$+ \frac{C_1}{R_1R_2L_1} + \frac{C_2}{R_1R_2L_1} + \frac{C_1}{R_1R_2L_2} + \frac{C_1C_2}{L_1L_2}$$

$$+ \left(\frac{C_1C_2}{R_1L_2} + \frac{C_1C_2}{R_2L_2}\right)s + \left(\frac{C_1C_2}{R_1R_2}\right)s^2$$

$$\triangle_{11} = \left(\frac{C_1C_2}{R_2}\right)s^2 + \left(\frac{C_1C_2}{L_1} + \frac{C_1C_2}{L_2}\right) s$$

$$+ \frac{C_1}{R_1L_1} + \frac{C_1}{R_2L_1} + \frac{C_2}{R_2L_1} + \frac{C_1}{R_1L_2}$$

$$+ \left(\frac{1}{R_1R_2L_1} + \frac{C_1}{L_1L_2} + \frac{C_2}{L_1L_2}\right) s^{-1} + \left(\frac{1}{R_1L_1L_2}\right)s^{-2}$$

Thus the driving-point impedance has the form

$$Z_{dp} = \frac{a_4s^2 + a_3s + a_2 + a_1s^{-1} + a_0s^{-2}}{b_4s^2 + b_3s + b_2 + b_1s^{-1} + b_0s^{-2}}$$

Now equating corresponding coefficients in the ZDP and determinant expressions and substituting the x's for the 'element value' part of the admittance, the following set of equations is obtained.

$$a_4 = x_3\, x_5\, x_6$$
$$a_3 = x_1\, x_3\, x_5 + x_2\, x_5\, x_6 + x_4\, x_5\, x_6$$
$$a_2 = x_1\, x_2\, x_5 + x_2\, x_3\, x_5 + x_2\, x_3\, x_6 + x_1\, x_4\, x_5$$
$$a_1 = x_1\, x_2\, x_3 + x_2\, x_4\, x_5 + x_2\, x_4\, x_6$$
$$a_0 = x_1\, x_2\, x_4$$
$$b_4 = x_1\, x_3\, x_5\, x_6$$
$$b_3 = x_1\, x_4\, x_5\, x_6 + x_3\, x_4\, x_5\, x_6 + x_1\, x_2\, x_5\, x_6$$
$$b_2 = x_1\, x_2\, x_3\, x_5 + x_1\, x_2\, x_3\, x_6 + x_1\, x_3\, x_4\, x_5 + x_2\, x_4\, x_5\, x_6$$
$$b_1 = x_1\, x_2\, x_4\, x_6 + x_2\, x_3\, x_4\, x_5 + x_2\, x_3\, x_4\, x_6$$
$$b_0 = x_1\, x_2\, x_3\, x_4$$

The realization and element value problem is solved by finding a solution to this set of equations. If the solution is composed of real positive values, the ZDP is realizable in the specified network form.

These equations will be called the tree-sum equations and some of their properties are considered below. The admittance of element i will be denoted by y_i, while the 'element value' part of the admittance is designated x_i. For example, the 'element value' part of y_5 is C_1 and is denoted by x_5. Having removed the complex frequency variable s, x_i must be real and positive for a network of passive elements.

The tree-sum equations may be arranged in the form of a first degree polynomial in any one variable. Each equation is linear in each of the variables. Hence, the term multilinear is sometimes used to designate equations of this type.

The tree-sum portion (right-hand side in the example) of each equation is a homogeneous function. In particular, the a_i functions are homogeneous of degree $V-2$ while the b_i functions are homogeneous of degree $V-1$, where V is the number of vertices in the network. These functions are continuous and have partial derivatives of all orders.

The number M of tree-sum equations associated with a network is determined by the six attributes of the normal ZDP form. Table 3.2.3 illustrates this relation in terms of the ZDP class and the attributes m and n. N_n and N_d denote the number of numerator and denominator equations, respectively, while M is the total. They are determined by counting the number of coefficients in the corresponding normal form of the ZDP.

The number E of variables in the tree-sum equations (the number of elements in the network) is a more difficult subject. A general

TABLE 3.2.3

NUMBER OF TREE-SUM EQUATIONS

Subclass	Class 1	Class 2	Class 3
0.0	$N_n = m + 1$ $N_d = n + 1$ $M = m + n + 2$	$N_n = m$ $N_d = n + 1$ $M = m + n + 1$	$N_n = m + 1$ $N_d = n$ $M = m + n + 1$
0.1	$N_n = m + 1$ $N_d = \frac{n}{2} + 1$ $M = m + \frac{n}{2} + 2$	$N_n = m$ $N_d = \frac{n}{2} + 1$ $M = m + \frac{n}{2} + 1$	$N_n = m$ $N_d = \frac{n + 1}{2}$ $M = m + \frac{n + 1}{2}$
0.2	$N_n = \frac{m}{2} + 1$ $N_d = n + 1$ $M = \frac{m}{2} + n + 2$	$N_n = \frac{m + 1}{2}$ $N_d = n + 1$ $M = \frac{m + 1}{2} + n + 1$	$N_n = \frac{m}{2} + 1$ $N_d = n$ $M = \frac{m}{2} + n + 1$
0.3	$N_n = \frac{m}{2} + 1$ $N_d = \frac{n}{2} + 1$ $M = \frac{m}{2} + \frac{n}{2} + 2$	$N_n = \frac{m + 1}{2}$ $N_d = \frac{n}{2} + 1$ $M = \frac{m + 1}{2} + \frac{n}{2} + 1$	$N_n = \frac{m}{2} + 1$ $N_d = \frac{n + 1}{2}$ $M = \frac{m}{2} + \frac{n + 1}{2} + 1$

statement can be made about the minimum number. As there must be at least one tree in a connected network and this tree consists of $V-1$ elements, it is known that

$$E \geq V - 1$$

The relation between M and E is of special interest here, as in linear algebra. The four cases to be discussed are:

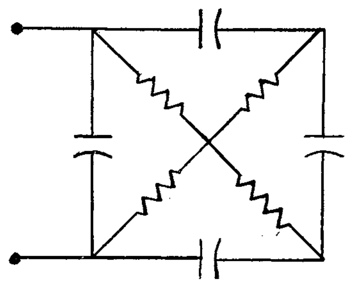
(a) $M < E$

(b) $M = E$

(c) $M = E + 1$

(d) $M > E + 1$

Figure 3.2.2 shows a network and its driving-point impedance form for case (a).



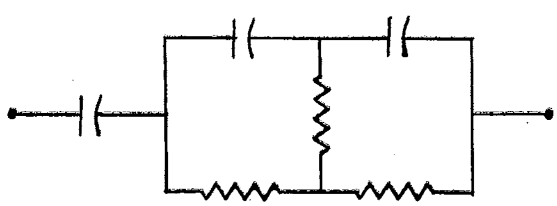
$E = 8$

$$Z_{dp} = \frac{a_2 s^2 + a_1 s + a_0}{b_3 s^3 + b_2 s^2 + b_1 s + b_0} \qquad M = 7$$

Figure 3.2.2. Network Illustrating Case (a)

It should be pointed out that there is a large number of trivial examples of this case, since any number of elements of the same type could be connected in series or parallel, each one being treated as a separate element.

An example of case (b) is given in Figure 3.2.3. This will later be called the definite coefficient case, as the coefficients of the ZDP are either realized exactly as specified or not at all.



E = 6

$$Z_{dp} = \frac{a_2 s^2 + a_1 s + a_0}{b_3 s^3 + b_2 s^2 + b_1 s}$$

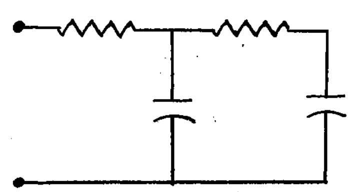
M = 6

Figure 3.2.3. Network Illustrating Case (b)

Case (c) is given special attention here because of its importance in classical synthesis techniques. An example is shown in Figure 3.2.4. This network is an R-C ladder which would be obtained by a continued fraction expansion synthesis procedure. The element values for such a network having been obtained by a classical method, tree-sum equations would not in general be satisfied. That is, the coefficients of the ZDP are not precisely realized. Rather, each coefficient in the set is multiplied by a constant. If the constant is moved to the right hand side of the equations and represented by x_0 , the coefficients become

$$a_0 = x_0\, x_3\, x_4$$
$$a_1 = x_0\, x_1\, x_4 + x_0\, x_2\, x_3 + x_0\, x_2\, x_4$$
$$a_2 = x_0\, x_1\, x_2$$
$$b_1 = x_0\, x_1\, x_3\, x_4 + x_0\, x_2\, x_3\, x_4$$
$$b_2 = x_0\, x_1\, x_2\, x_3$$

By introducing the auxiliary multiplier the set now contains five equations in five variables and corresponds to case (b). It is shown in classical synthesis texts (9) that the minimum number of elements required to realize a ZDP with a two-element-kind network is one less than the number of coefficients and that canonical networks always have the minimum number of elements. Thus case (c) includes a large class of problems.



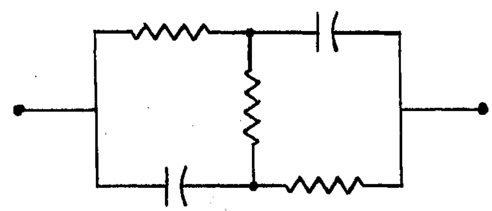
$E = 4$

$$Z_{dp} = \frac{a_2 s^2 + a_1 s + a_0}{b_2 s^2 + b_1 s}$$

$M = 5$

Figure 3.2.4. Network Illustrating Case (c)

Not all networks in case (c) are canonical, however. One such example is shown in Figure 3.2.5.



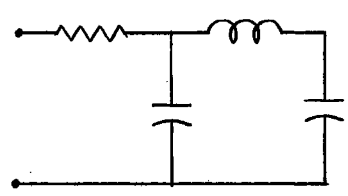
E = 5

$$Z_{dp} = \frac{a_2 s^2 + a_1 s + a_0}{b_2 s^2 + b_1 s + b_0}$$

M = 6

Figure 3.2.5. A Second Example of Case (c)

An example of the last condition, case (d), is Figure 3.2.6. Here, as in case (c), an arbitrary multiplier could also be introduced.



E = 4

$$Z_{dp} = \frac{a_3 s^3 + a_2 s^2 + a_1 s + a_0}{b_3 s^3 + b_1 s}$$

M = 6

Figure 3.2.6. Network Illustrating Case (d)

It will be shown in Chapter IV that a solution for the tree-sum equations can be obtained for each case above. Examples will be presented.

<u>3.3 Non-unique Solutions of the Tree-sum Equations.</u> While the elimination procedure to be presented will yield all of the solution sets, some special cases are now considered. These are applications of the theory of substitutions as described by Netto (16). The form of an equation is usually altered by an interchange of the variables. The process of changing the variables is known as substitution, a subject of mathematical interest since the early 1700's. There are some cases, however, in which a substitution leaves the equation invariant or unchanged. These are of particular interest here. The network shown in Figure 3.3.1 is used as an example.

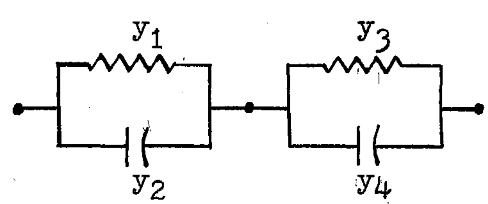


Figure 3.3.1. Network Illustrating Substitution

The tree-sum equations for this network are:

$$a_0 = x_1 + x_3$$
$$a_1 = x_2 + x_4$$
$$b_0 = x_1 x_3$$
$$b_1 = x_1 x_4 + x_2 x_3$$
$$b_2 = x_2 x_4$$

They are invariant with respect to the substitution of x_3 for x_1 and x_4 for x_2. Such a substitution is clearly a result of re-labeling the elements in the network. More complicated equations, however, do not yield to inspection. An algorithm for finding all substitutions that leave the tree-sum equations invariant could be programmed for the computer.

CHAPTER IV

SOLUTION OF TREE-SUM EQUATIONS BY ELIMINATION

4.1 Background. Electrical engineers have determined the general solution to a number of circuit design problems. For example, the equations and the procedure for the design of a cathode follower amplifier are well known. To obtain such a design technique, the engineer writes the equations or relations between the variables of the problem and then manipulates them by trial and error until the elements to be determined are explicit in terms of the specified quantities. In the driving-point impedance synthesis problem discussed here the tree-sum equations relate the specified ZDP coefficients to the elements of the network, and the elimination procedure to be described is a formal method for solving these equations.

The procedure is illustrated for the elementary circuit in Figure 4.1.1, and the concept of elimination is introduced. The ZDP is represented by the form

$$Z_{dp} = \frac{a_1 s + a_0}{b_1 s + b_0} \qquad 4.1.1$$

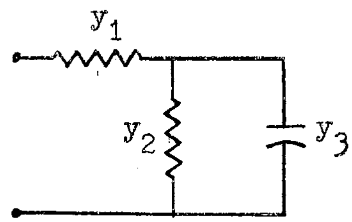


Figure 4.1.1. Elementary Circuit to Illustrate Elimination

and the tree-sum equations are

$$a_0 = x_0 x_1 + x_0 x_2 \tag{4.1.2}$$

$$a_1 = x_0 x_3 \tag{4.1.3}$$

$$b_0 = x_0 x_1 x_2 \tag{4.1.4}$$

$$b_1 = x_0 x_1 x_3 \tag{4.1.5}$$

The auxiliary variable x_0 is eliminated by solving 4.1.3 and substituting into the other equations. The reduced set of three equations in three unknowns is

$$- a_0 x_3 + a_1 x_1 + a_1 x_2 = 0 \tag{4.1.6}$$

$$- b_0 x_3 + a_1 x_1 x_2 = 0 \tag{4.1.7}$$

$$- b_1 + x_1 a_1 = 0 \tag{4.1.8}$$

Equation 4.1.8 is now solved for x_1 and this variable is eliminated from the remaining two equations.

$$- a_0 x_3 + b_1 + a_1 x_2 = 0 \tag{4.1.9}$$

$$- b_0 x_3 + b_1 x_2 = 0 \tag{4.1.10}$$

Upon substituting x_2 from Equation 4.1.10 into 4.1.9, an equation in the variable x_3 is obtained.

$$- b_1^2 + a_0 b_1 x_3 - a_1 b_0 x_3 = 0 \qquad 4.1.11$$

By eliminating one variable in each step, four sets of equations result. Note that the last equation contains a single variable x_3, while the next to the last set contains x_2 and x_3, etc. Since Equation 4.1.11 was obtained by solving 4.1.10 for x_2 and substituting into 4.1.9, the same value of x_2 will satisfy both equations. A corresponding statement is true for x_1 and x_0.

If the ZDP to be realized is specified as

$$Z_{dp} = \frac{3 s + 7}{s + 2}$$

Equation 4.1.11 becomes

$$1 = 7 x_3 - 6 x_3$$

and

$$x_3 = 1.0$$

Now substituting into the other equations, the other element values are found.

$$x_2 = 2.0$$

$$x_1 = 0.333$$

$$x_0 = 3.0$$

This procedure is quite general. Each equation in the reduced set is called an eliminant of the previous set.

4.2 The Eliminant. The eliminant was studied by mathematicians in the early 1700's. Euler first described the eliminant in terms of symmetric functions in his Berlin Memoirs in 1748. This is foundation for the discussion to follow. Both Bezout and Euler developed easier methods for determining the eliminant and some of its properties. Salmon (17) has published a review of this early work which can be consulted for more historic detail.

Definition 4.2.1 (Eliminant): The eliminant of two equations is a function F such that if F = 0 , the two equations have at least one common root.

The term resultant is used to mean the same thing as eliminant. It is always possible to obtain the eliminant of two polynomials. This is shown by considering the following two equations.

$$G(x) = a_n x^n + a_{n-1} x^{n-1} + \dots a_0 = 0 \qquad 4.2.1$$

$$H(x) = b_m x^m + b_{m-1} x^{m-1} + \dots b_0 = 0 \qquad 4.2.2$$

It will be assumed throughout this discussion that the coefficients may be functions of other variables. G(x) is identified as the tool equation and its n roots denoted by x_1 , x_2 , ... x_n . If at least one of these roots, for example x_i , solves H(x) ,

$$H(x_i) = 0$$

The product

$$F = H(x_1)\, H(x_2) \dots H(x_n) = 0 \qquad 4.2.3$$

must also be zero. In fact, by definition, the product is zero only if at least one x_i is a root of $H(x)$.

This product is a symmetric polynomial of the variables x_1, x_2, ... x_n since the form of the equation is not changed by interchanging any two variables (Theorem A.1.1). A brief discussion of symmetric polynomials is presented in Appendix A. Note that the function F is a polynomial in the coefficients of $H(x)$ and the roots. According to the fundamental theorem of symmetric polynomials (A.3.1), this equation can be expressed as a polynomial in the <u>elementary symmetric functions</u>. However, these are ratios of coefficients of the polynomial with roots x_1, x_2, ... x_n. Thus F is a polynomial of the coefficients of $G(x)$ and $H(x)$ and is by definition the eliminant.

The eliminant can be computed for two equations using the principles outlined above and in the appendix. If m and n are greater than 2, however, the computation is very tedious. Sylvester has described a method to obtain the eliminant from a determinant. The author has implemented this procedure on a digital computer and used it to solve the examples to follow.

<u>4.3 Solving a System of Polynomials by Elimination.</u> A tree-sum equation is a special type of polynomial equation in n variables. This discussion will deal with a system of polynomial equations.

<u>Definition 4.3.1 (Polynomial):</u> A polynomial in n variables is defined to be an equation of the form

$$P(x_1, x_2, \ldots x_n) = \sum_{j=1}^{N} c_j \phi_j$$

where C_j is a constant (real or complex) and ϕ_j is the product

$$\phi_j = \prod_{k=1}^{M_j} x_{\beta_k}^{\alpha_k}$$

The elimination procedure is now applied to a system of m equations in n unknowns to determine a reduced system of equations. A tool equation is selected from the m equations. The equation of lowest non-zero degree in the variable to be eliminated, called the 'object' variable, should be chosen to allow the calculation to proceed with minimum effort. m-1 eliminants are now formed between the tool equation and each of the other m-1 equations. The eliminants are called the reduced systems.

Theorem 4.3.1: A reduced system of equations in n-1 variables is satisfied if and only if an 'object' variable solution to the system (if one exists) is a root of the tool equation.

Proof: By definition the eliminant is zero if and only if there is a common root between the tool equation and one of the remaining m-1 equations. Thus if a value of the 'object' variable is a solution to each of the m equations, it must be a root of the tool equation.

The theorem does not guarantee that any or all of the roots of the tool equations are solutions; but if a solution exists, it is a root of the tool equation.

Theorem 4.3.2: If the reduced system is satisfied and the tool equation is of first degree in the 'object' variable, it has one root and this root is a solution for the system.

Proof: Since the $m-1$ eliminants are all satisfied, a common root exists between the tool equation and each of the other equations in the original set. But the tool equation has only one root. Thus it is a root of each equation in the system.

If the tool equation is of degree greater than one, each root is a possible solution and is checked by substituting into each of the other equations of the set. If none of the roots satisfy all of the equations, the original set is inconsistent. That is, no value of the "object" variable solves all of the equations. More than one root may satisfy them all. Then the solution of the system is not unique. This condition is illustrated in a later section.

The foregoing discussion is now applied to find the complete solution to a set of equations. The case in which n equations in n variables are to be solved is considered first.

$$P_{1,1}(x_1, x_2, \ldots x_n) = 0$$

$$P_{1,2}(x_1, x_2, \ldots x_n) = 0$$

$$\vdots$$

$$P_{1,n}(x_1, x_2, \ldots x_n) = 0$$

The variable x_1 is to be eliminated. A tool equation is selected and the reduced system containing $n-1$ equations in $n-1$ variables is formed.

$$P_{2,1}(x_2, x_3, \ldots x_n) = 0$$

$$P_{2,2}(x_2, x_3, \ldots x_n) = 0$$

$$\vdots$$

$$P_{2,n-1}(x_2, x_3, \ldots x_n) = 0$$

A reduced system is again determined and so on until a system of two equations in two unknowns is obtained.

$$P_{n-1,1}(x_{n-1}, x_n) = 0$$

$$P_{n-1,2}(x_{n-1}, x_n) = 0$$

The last reduced system is a single polynomial in the unknown x_n.

$$P_{n,1}(x_n) = 0$$

The roots of this equation are determined. Each one is a possible solution for the system. If the degree of the equation is k, there are potentially k or more solutions to the system. Each of the roots is substituted into the $(n-1)$th system. Thus, each equation becomes a polynomial in the single variable x_{n-1}. The roots of the tool equation are calculated and checked in the other equations of the system. Those which satisfy it are paired with the corresponding x_n values to form a partial solution. The substitution now continues to the $(n-2)$th system and so on until the x_1 values from the first tool equation are obtained. This is a complete set of solutions, as can be reasoned from Theorem 4.3.1. In each case the m values of x_i which satisfy the i-th system are joined with the values

$(x_{i+1}, x_{i+2}, \ldots x_n)$ to form a partial solution. Note that there may be more than one set, $(x_{i+1}, x_{i+2}, \ldots x_n)$, and the number of partial solutions may be multiplied by the degree of the i-th tool equation at each step. The Gauss reduction method for solving linear systems of equations is a special case of this procedure.

If the number of variables n is greater than the number of equations m, the solution is not unique. This is due to the fact that the last reduced system (one equation) contains $n - m + 1$ variables. For example, a system of five equations in eight variables would have the variable and equation count shown in Table 4.3.1.

TABLE 4.3.1

EQUATION SETS FOR 8 VARIABLES

AND 5 EQUATIONS

Equation set	Number of equations	Variables
1	5	8
2	4	7
3	3	6
4	2	5
5	1	4

Clearly $n-m$ of these variables can be assigned arbitrary values and the remaining variable determined from the roots of the last polynomial. The remaining $m-1$ variables are determined by substituting back into the reduced sets of equations as in the previous case.

The remaining case to be considered is $m > n$. A reduction process proceeds as in the previous cases until all n variables are eliminated. The last reduced set contains $m-n$ equations which must be identically zero if the system is consistent (Theorem 4.3.1). If a solution exists, it is obtained by solving for the variable x_n in the $(m - n + 1)$th set of equations and substituting into successive sets as previously described.

During the process of eliminating one variable from a system of equations, one or more other variables may be eliminated. This implies that an arbitrary value may be assigned to these when solving for the 'object' variable in the tool equation.

This process of solving sets of polynomial equations is superior to iteration methods in that no estimate of the solution is required to start the procedure, and that all solutions are obtained or, in the case of $m < n$, are placed in evidence. A test for consistency is an automatic part of the process.

4.4 Example. Several examples are now presented to illustrate the elimination method for solving the tree-sum equations of a network. The sets of equations associated with the solution are presented in Appendix B. They are written in a form suitable for computer processing. Each equation is understood to be a polynomial and thus equal to zero. Since the equations involve a number of multiplications, the product operator is not printed. It is understood to be present between two operands not otherwise connected by an operator. The asterisk denotes exponentiation. In each equation terms of like powers of the object variable are collected, enclosed by parentheses, and printed on the line immediately following the variable and its power. Each polynomial is delimited by the words BEGIN and END.

An example illustrating case (a) of Section 3.2 is shown in Figure 4.4.1. This is a non-canonical form of six elements, having a driving-point impedance of the form

$$Z_{dp} = \frac{a_1 s + a_0}{b_2 s^2 + b_1 s + b_0}$$

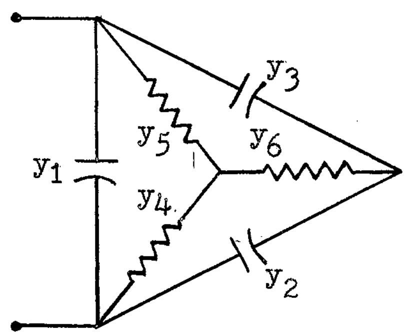
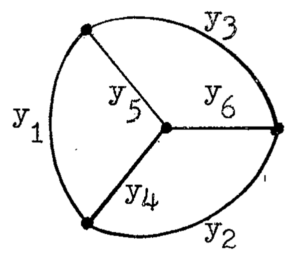
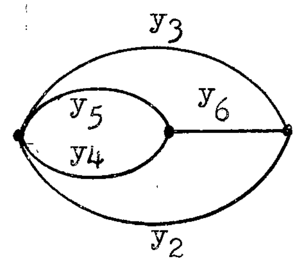


Figure 4.4.1. Network Form for Example 1

The G_0 and G_1 graphs of this network and their tree lists are shown in Figure 4.4.2.



Graph G_0



Graph G_1

Graph G_0	Graph G_1
$y_1\ y_2\ y_4$	$y_2\ y_4$
$y_1\ y_2\ y_5$	$y_2\ y_5$
$y_1\ y_2\ y_6$	$y_2\ y_6$
$y_1\ y_3\ y_4$	$y_3\ y_4$
$y_1\ y_3\ y_5$	$y_3\ y_5$
$y_1\ y_3\ y_6$	$y_3\ y_6$
$y_2\ y_3\ y_4$	$y_4\ y_6$
$y_2\ y_3\ y_5$	$y_5\ y_6$
$y_2\ y_3\ y_6$	
$y_1\ y_4\ y_6$	
$y_1\ y_5\ y_6$	
$y_2\ y_4\ y_5$	
$y_3\ y_4\ y_5$	
$y_3\ y_4\ y_6$	
$y_4\ y_5\ y_6$	
$y_2\ y_5\ y_6$	

Figure 4.4.2. Graphs and Tree List for Example 1

Thus, representing the value of the i-th element by x_i, the tree-sum equations are

$$a_0 = x_4 x_6 + x_5 x_6$$

$$a_1 = x_2 x_4 + x_2 x_5 + x_2 x_6 + x_3 x_4 + x_3 x_5 + x_3 x_6$$

$$b_0 = x_4 x_5 x_6$$

$$b_1 = x_1 x_4 x_6 + x_1 x_5 x_6 + x_2 x_4 x_5 + x_2 x_5 x_6 + x_3 x_4 x_5 + x_3 x_4 x_6$$

$$b_2 = x_1 x_2 x_4 + x_1 x_2 x_5 + x_1 x_2 x_6 + x_1 x_3 x_4 + x_1 x_3 x_5 + x_1 x_3 x_6 + x_2 x_3 x_4 + x_2 x_3 x_5 + x_2 x_3 x_6$$

It is convenient to multiply each coefficient by the variable x_0 as in the canonical case discussed in Section 3.2. Table B.1.1 shows the five tree-sum equations factored in the first object variable x_0. Equation 01.01 is chosen as the tool equation and the reduced system in Table B.1.2 obtained. Note that two of these do not involve the second object variable x_1. Thus 02.03 is used as the tool equation to derive 03.03 from 02.04, while 02.01 and 02.02 are factored in the third object variable x_2 to obtain 03.01 and 03.02. In the next reduction 03.01 does not contain x_2 and is thus shifted to the reduced set without elimination. 03.02 is used as the tool and x_2 eliminated in Equation 03.03. x_4 is the object variable in the reduced set shown in Table B.1.4, and Equation 04.01 is the tool since it is a first degree equation in x_4. Either x_4 or x_5 must be eliminated in this step since no other variable appears in both equations. The single equation shown in Table B.1.5 contains three variables x_3, x_5, and x_6 and terminates the reduction sequence.

The driving-point impedance to be realized is chosen as

$$Z_{dp} = \frac{s + 171}{0.0433\ s^2 + 8.42\ s + 163}$$

Choosing 1.2 and 5.8 for the element value part of x_5 and x_6, respectively, Equation 05.01 in Table B.1.5 is

$$0.2024\ 10^5\ x_3^2 - 0.1416\ 10^3\ x_3 + 0.1219 = 0$$

Its roots are

0.00100

0.00599

Since both are positive real numbers, there are potentially two realizable solutions for the network form. The tool equations are now solved for the remaining x's . The solutions are shown in Table 4.4.1.

TABLE 4.4.1

SOLUTIONS FOR EXAMPLE 1

Variable	Solution 1	Solution 2
x_0	0.19791	0.19791
x_1	0.03941	0.04235
x_2	0.01101	0.01601
x_3	0.00599	0.00100
x_4	4.63507	4.63507
x_5	1.20000	1.20000
x_6	5.80000	5.80000

Thus two networks with the specified form realize the ZDP. The element values are shown in Table 4.4.2. A different choice of x_5 and x_6 could yield a non-realizable set of elements. Hence, proving that a specified ZDP cannot be realized by a particular network is more difficult when $M < E$.

TABLE 4.4.2

ELEMENT VALUES FOR EXAMPLE 1

Element	Value 1	Value 2
C_1	0.03941 f	0.04235 f
C_2	0.01101 f	0.01601 f
C_3	0.00599 f	0.00100 f
R_4	0.21600 Ω	0.21600 Ω
R_5	0.83333 Ω	0.83333 Ω
R_6	0.17250 Ω	0.17250 Ω

As explained in Chapter III the same solution technique applies in both cases (b) and (c). This is illustrated by realization of the network form shown in Figure 4.4.3.

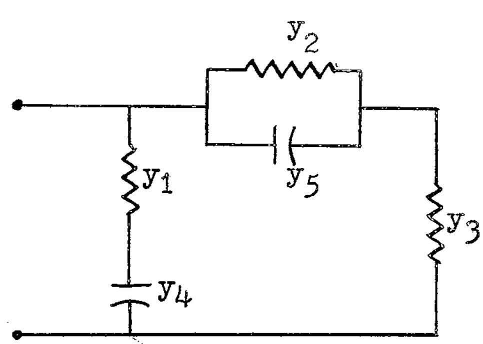
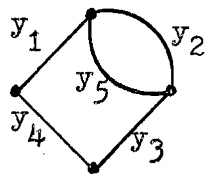
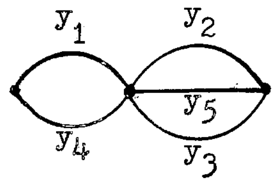


Figure 4.4.3. Network Form for Example 2

The graphs and tree lists for this network are presented in Figure 4.4.4.





Graph G_0	Graph G_1
$y_1\ y_2\ y_3$	$y_1\ y_2$
$y_1\ y_2\ y_4$	$y_1\ y_3$
$y_1\ y_3\ y_4$	$y_1\ y_5$
$y_1\ y_3\ y_5$	$y_2\ y_4$
$y_1\ y_4\ y_5$	$y_3\ y_4$
$y_2\ y_3\ y_4$	$y_4\ y_5$
$y_3\ y_4\ y_5$	

Figure 4.4.4. Graphs and Tree List for Example 2

The tree-sum equations are

$$a_0 = x_1 x_2 + x_1 x_3$$

$$a_1 = x_1 x_5 + x_2 x_4 + x_3 x_4$$

$$a_2 = x_4 x_5$$

$$b_0 = x_1 x_2 x_3$$

$$b_1 = x_1 x_2 x_4 + x_1 x_3 x_4 + x_1 x_3 x_5 + x_2 x_3 x_4$$

$$b_2 = x_1 x_4 x_5 + x_3 x_4 x_5$$

An arbitrary specification of the coefficients could produce an inconsistent system. Thus in accordance with the discussion in Section 3.3, each coefficient is multiplied by x_0. The system of equations factored in the first 'object' variable x_0 is shown in Table B.2.1. The next variable to be eliminated is x_1 ; Equation 02.03 is the tool equation. Since x_4 is the variable of lowest degree in the system of equations in Table B.2.3, it is the 'object' variable in this set. x_2 and x_5 are eliminated in the next two sets of equations.

The ZDP to be realized is chosen as

$$Z_{dp} = \frac{2\ s^2 + 9.55\ s + 6.2}{2.2\ s^2 + 2.5\ s + 0.6}$$

Substituting the coefficient values, Equation 06.01 becomes

$$- 0.7623\ 10^6 x_3^6 + 0.1207\ 10^7 x_3^5 - 0.4796\ 10^6 x_3^4$$

$$+ 0.8639\ 10^5 x_3^3 - 0.8072\ 10^4 x_3^2$$

$$+ 0.3831\ 10^3 x_3 - 7.3327 = 0$$

The roots of this equation are

0.095 - j 0.0066

0.095 + j 0.0066

0.103 - j 0.0043

0.103 + j 0.0043

0.100

1.097

All are possible solutions but only the last two are positive real numbers and thus realizable by passive elements. Each of these latter is substituted into Equations 05.02 and 05.01 to determine x_5. Both roots of Equation 05.01 are checked in 05.02 and only one is found to satisfy the set. Thus for $x_3 = 0.1$ and 1.97 the solution for x_5 is 4.0 and 0.3, respectively. The other tool equations are all first degree and the solution straight-forward. The two solutions are shown in Table 4.4.3.

TABLE 4.4.3

SOLUTIONS FOR EXAMPLE 2

Variable	Solution 1	Solution 2
x_1	1.00	0.003
x_2	3.00	0.106
x_3	0.10	1.097
x_4	0.25	0.004
x_5	4.00	0.300

The element values for the network are easily obtained from the solution.

A network illustrating case (d) is now considered. The ZDP form

$$Z_{dp} = \frac{a_3s^3 + a_2s^2 + a_1s + a_0}{b_3s^3 + b_2s^2 + b_1s + b_0}$$

is realized by the network form shown in Figure 4.4.5. The G_0 and G_1 graphs and associated tree lists are shown in Figure 4.4.6.

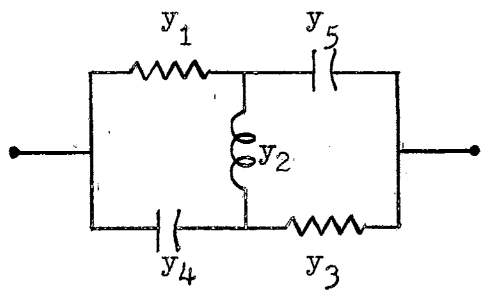
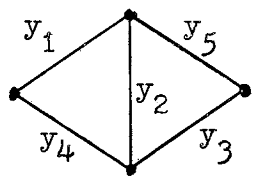
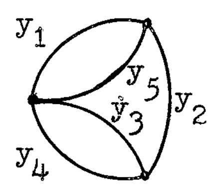


Figure 4.4.5. Network Form for Example 3





Graph G_0	Graph G_1
$y_1\ y_2\ y_3$	$y_1\ y_2$
$y_1\ y_2\ y_5$	$y_1\ y_3$
$y_1\ y_3\ y_4$	$y_1\ y_4$
$y_1\ y_3\ y_5$	$y_2\ y_3$
$y_1\ y_4\ y_5$	$y_2\ y_4$
$y_2\ y_3\ y_4$	$y_2\ y_5$
$y_2\ y_4\ y_5$	$y_3\ y_5$
$y_3\ y_4\ y_5$	$y_4\ y_5$

Figure 4.4.6. Graphs and Tree List for Example 3

The tree-sum equations are

$$a_0 = x_1 x_2 + x_2 x_3$$

$$a_1 = x_1 x_3 + x_2 x_4 + x_2 x_5$$

$$a_2 = x_1 x_4 + x_3 x_5$$

$$a_3 = x_4 x_5$$

$$b_0 = x_1 x_2 x_3$$

$$b_1 = x_1 x_2 x_5 + x_2 x_3 x_4$$

$$b_2 = x_1 x_3 x_4 + x_1 x_3 x_5 + x_2 x_4 x_5$$

$$b_3 = x_1 x_4 x_5 + x_3 x_4 x_5$$

In this case there are eight equations and five variables. As discussed in Section 4.3, eliminating the five variables will yield a set of three equations in the coefficients which must be satisfied identically if the system is consistent. This means that the eight coefficients of the ZDP form cannot be specified independently. In this example the four denominator coefficients and a_3 are specified and the others are left to be determined by the network form. By doing this, the possibility of specifying an inconsistent set of coefficients is avoided.

The five equations to be solved are shown in Table B.3.1 explicit in the first 'object' variable x_1. The elimination proceeds as in the previous examples.

The specified coefficients are now assigned values.

$$a_3 = 6.0$$

$$b_0 = 7.5$$

$$b_1 = 30.0$$

$$b_2 = 37.5$$

$$b_3 = 15.0$$

The first Equation 05.01 to be solved becomes

$$+ 0.91125\ 10^8 x_5^{10} - 0.18225\ 10^7 x_5^9 + 0.10320\ 10^8 x_5^8$$

$$+ 0.29046\ 10^8 x_5^7 - 0.50260\ 10^9 x_5^6 + 0.19259\ 10^{10} x_5^5$$

$$- 0.30156\ 10^{10} x_5^4 + 0.10456\ 10^{10} x_5^3 + 0.22291\ 10^{10} x_5^2$$

$$- 0.23620\ 10^{10} x_5 + 0.70859\ 10^9 = 0$$

The ten roots of this equation are

$$0.62815 - j\ 0.27444$$

$$0.62815 + j\ 0.27444$$

$$8.02083 - j\ 3.50436$$

$$8.02083 + j\ 3.50436$$

$$-0.93392$$

$$-6.42452$$

$$1.89635$$

$$2.00000$$

$$3.16394$$

$$3.00000$$

Only the last four values can be realized by passive elements. Thus the other possible solutions are ignored. Each root to be considered is substituted into Equation 04.02 and a corresponding value for x_4 determined. The second order Equation 03.02 was used as the tool in Table B.3.3. Its roots must be checked in Equation 03.01, and in each case, one of the roots fails the test. The remaining x's are determined by direct solution of the tool equations. The four realizable solutions are shown in Table 4.4.4. The unspecified coefficients are calculated from the appropriate tree-sum form and also displayed in this table.

This network form is now used to demonstrate the test for realizability. The ZDP shown below has a form which is realized by the network topology, and it is a positive real function.

$$Z_{dp} = \frac{s^3 + 2s^2 + s + 0.2}{s^3 + 5s^2 + 9s + 5}$$

To test the possibility of realizing the coefficients of this function by the network form, the four denominator coefficients and a_3 are substituted into Equation 05.01. All ten of the roots are found to be complex. Thus the specified ZDP cannot be realized by this network form.

TABLE 4.4.4

SOLUTIONS FOR EXAMPLE 3

x_5	x_4	x_3	x_2	x_1	a_0	a_1	a_2
1.896	3.164	1.170	(does not satisfy 03.01)				
		1.012	4.979	1.487	12.449	26.706	6.626
2.000	3.000	1.250	(does not satisfy 03.01)				
		1.000	5.000	1.500	12.500	26.500	6.500
3.164	1.896	1.487	4.979	1.012	12.449	26.705	6.626
		1.329	(does not satisfy 03.01)				
3.000	2.000	1.500	5.000	1.000	12.500	26.500	6.500
		1.250	(does not satisfy 03.01)				

CHAPTER V

SUMMARY AND CONCLUSIONS

5.1 Summary. The subject of this thesis is to make the topological properties of the network one of the specifications for a synthesis procedure. Experience has shown that there are realizability conditions on the topology just as there are on the coefficients of a network function. These conditions are known to be related to the form of the driving-point impedance function as well as the value of the coefficients. Thus two separate aspects of the topological synthesis of driving-point impedances are considered.

The relations between the form of the ZDP and the topology are examined by considering two questions:

(a) What are the conditions relating the form of the driving-point impedance and the network topology?

(b) How may these conditions be used in topological synthesis?

The network topology is represented by the linear graph. As the trees of the graph are known to determine the form of the ZDP, they are the foundation of the discussion. The ZDP form is characterized by six attributes, and the conditions relating the form of the ZDP and the network topology are specified in terms of these attributes. Table 2.5.1 presents the conditions by classifying the network in terms of its attributes.

The relations in Table 2.5.1 are used to develop a classification of the uncountable number of networks realizing a specified function. Using Table 2.6.1, the designer may determine network forms realizing a specified ZDP form. This classification is not unique and could be altered to include topological parameters of specific interest.

The relation between the value of the ZDP and the topology are studied by developing a procedure for answering the following questions:

(a) Can a specified network form realize a specified function value?

(b) What are the network element values?

The tree-sum equations contain the information sought.

A procedure for solving the tree-sum equations of any network is developed. It is shown that by a process of elimination the value of each element in the network can be determined for a specified ZDP value. Several classes are considered and examples presented to illustrate them.

5.2 Conclusions. A procedure for synthesis of a driving-point impedance with specific topological properties is discussed. The method involves testing for realizability the network forms having the desired properties, and thus is basically a 'cut and try' procedure. This appears to be characteristic of the problem. The author conjectures that any test for realization will involve solving the tree-sum equations, either directly or indirectly. Since the form of the network must be known before the equations can be determined, a topology must be assumed and tested.

Other network functions can be related to the network form by tree-sum equations. Thus the realization technique described here can be readily extended.

5.3 Suggestions for Further Study and Development. The use of the table classifying all networks realizing a given ZDP is a subject for future study. In particular, the known relations between the ranks R_{Ld}, R_{Ls}, R_{Cd}, R_{Cs}, R_{LCd} are unsatisfactory. An algorithm might be found for determining all of the three-element-kind networks having deleted and shorted graphs with specified rank; such an algorithm would be useful here, and would also be significant in the general theory of linear graphs.

The author has developed two computer programs for use in the elimination process. As these were to be used for solving the examples, they are elementary, but they have shown that such programs can be useful. A detailed study of the algorithm to determine efficient procedures and data structures would be challenging. The final step would be to implement a complete computer program for solving a system of polynomials. Such a program would be valuable in other fields of engineering and science. For example, equations of the type solved here occur in mechanical design problems.

The relation of the network forms and function forms for other types of network functions (transfer admittance, voltage gain, etc.) is a subject for investigation. As these functions involve both the addition and subtraction of tree admittance products, the necessary and sufficient conditions developed here for the ZDP will not apply. Equations similar to the tree-sum equations have been developed for active networks. Thus a logical extension would be to the synthesis of this class of networks.

Of particular interest is the possibility of realizing several specified functions by a single synthesis procedure. For example, using the elimination procedure described, equations relating the coefficients of the driving-point impedance, voltage gain, and output impedance of a network can be derived. Thus if a circuit designer wished to specify several functions to be realized, he could test a particular network form, then solve for the element values. Investigation of this topic, however, will require a sophisticated computer system to solve the equations.

The author is quite intrigued with the possibility of statistical circuit design using the realization techniques described here. If a circuit designer can specify the distributions of the coefficients of a network function or some property from which these can be obtained, it may be possible to determine the distribution of the element values by a Monte Carlo method.

A SELECTED BIBLIOGRAPHY

1. Seshu, S. "Topological Considerations in the Design of Driving-Point Functions." *I.R.E. Transactions on Circuit Theory*, Vol. CT-2, No. 4 (December, 1955), 356-367.

2. Onodera, R. "Topological Synthesis of Transfer-Admittance Matrices." *I.R.E. Transactions on Circuit Theory*, Vol. CT-7, No. 2 (June, 1960) 112-120.

3. Bellert, S. "Topological Analysis and Synthesis of Linear Systems." *Journal of the Franklin Institute*, Vol. 274, No. 6 (December, 1962), 225-243.

4. Calahan, D. A. "Computer Solution of the Network Realization Problem." *Proceedings of the Second Allerton Conference on Circuit and System Theory*, (1964), 175-193.

5. Hakimi, S. L. and W. Mayeda. "On Coefficients of Polynomials in Network Functions." *I.R.E. Transactions on Circuit Theory*, Vol. CT-7, No. 1 (March, 1960), 40-44.

6. Brown, D. P. and M. B. Reed. "On the 'Shape' of One-Port Driving Functions." *Journal of the Franklin Institute*, Vol. 277, No. 6 (June, 1964), 566-573.

7. Hakimi, S. L. "Graphs with Two Kinds of Elements." *Journal of the Franklin Institute*, Vol. 270, No. 12 (December, 1960), 451-467.

8. Seshu, S. and M. B. Reed. *Linear Graphs and Electrical Networks*. Reading, Massachusetts: Addison-Wesley, 1961.

9. Weinberg, L. *Network Analysis and Synthesis*. New York: McGraw-Hill, 1962.

10. Cummins, R. L. and L. C. Thomason. "An Efficient Tree-Listing Algorithm." presented at the Southwestern I.E.E.E. Conference, Dallas, Texas (April, 1964).

11. Minty, G. J. "A Simple Algorithm for Listing All the Trees of a Graph." *I.E.E.E. Transactions on Circuit Theory*, Vol. CT-12, No. 1 (March, 1965), 120.

12. Cummins, R. L. "Hamilton Circuits in Tree Graphs." I.E.E.E. Transactions on Circuit Theory, Vol. CT-13, No. 1 (March, 1966), 58-65.

13. Percival, W. S. "Solution of Passive Electrical Network by Means of Mathematical Trees." Journal of the Institute of Electrical Engineers (London), Vol. 100, Part III (1953), 143-150.

14. Maxwell, J. C. Electricity and Magnetism. Oxford: Clarendon Press, 1892.

15. Guillemin, E. A. Synthesis of Passive Networks. New York: John Wiley, 1957.

16. Netto, E. The Theory of Substitutions. New York: Chelsea, 1964.

17. Salmon, G. D. D. Modern Higher Algebra. New York: Chelsea, 1964.

18. Uspensky, J. V. Theory of Equations. New York: McGraw-Hill, 1948.

19. Bocher, M. Introduction to Higher Algebra. New York: Macmillan, 1938.

APPENDIX A

SYMMETRIC POLYNOMIALS

A.1 Definition of a Symmetric Polynomial. There are several excellent references on symmetric polynomials and symmetric functions, as this has been a subject of mathematical investigation since the 1700's. Texts by Uspensky (18) and Bocher (19) are modern reviews of this work. The pertinent parts of the theory are discussed below.

Definition A.1.1 (Symmetric Polynomials) A polynomial is said to be symmetric in the variables $x_1, x_2, \ldots x_n$ if it is unchanged by every substitution.

Thus the study of symmetric polynomials is a part of the general theory of substitutions. An easier test for a symmetric polynomial is desirable.

Theorem A.1.1: A polynomial is symmetric if an interchange of every pair of variables leaves its form unchanged.

Proof: All substitutions can be obtained from $(x_1, x_2, \ldots x_n)$ by a sequence of interchanges of two variables. Thus if every interchange leaves the form unaltered, any substitution could be made.

A.2 Special Symmetric Functions.

Definition A.2.1 (Sigma functions) The sigma function denoted by $\sum x_1^{\alpha_1} \; x_2^{\alpha_2} \; \ldots \; x_m^{\alpha_m}$ is the sum of the term $x_1^{\alpha_1} \; x_2^{\alpha_2} \; \ldots \; x_m^{\alpha_m}$ and all similar terms.

For example, if there are four variables,

$$\begin{aligned} \sum x_1 \, x_2^2 = {} & x_1 \, x_2^2 + x_1 \, x_3^2 + x_1 \, x_4^2 \\ & x_2 \, x_1^2 + x_2 \, x_3^2 + x_2 \, x_4^2 \\ & x_3 \, x_1^2 + x_3 \, x_2^2 + x_3 \, x_4^2 \\ & x_4 \, x_1^2 + x_4 \, x_2^2 + x_4 \, x_3^2 \end{aligned}$$

As background for the fundamental theorem, the sigma functions are now related to a symmetric polynomial.

Theorem A.2.1: A symmetric polynomial is a linear combination of sigma functions.

Proof: Any typical term of the polynomial is considered. As the polynomial is symmetric, all similar terms must be present and preceded by the same constant. These may be replaced by the sigma function of the term multiplied by the constant. Such a procedure is extended to every term.

A second symmetric function is defined and related to the symmetric polynomial by the sigma function.

Definition A.2.2 (Sum of Powers) The sum of powers function S_k is defined by the equation

$$S_k = \sum x_1^k$$

<u>Theorem A.2.2:</u> A symmetric polynomial can be expressed as a polynomial in the sum of powers function S.

<u>Proof:</u> It will be shown that the sigma functions can be expressed as a polynomial in S. This theorem is then proved utilizing its counterpart for sigma functions, Theorem A.2.1.

The two functions of n variables

$$\sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa} \qquad (k < n)$$

and

$$S_{\lambda} = \sum x_1^{\lambda}$$

are multiplied together. The result is the symmetric form

$$\begin{aligned} S_{\lambda}\left\{\sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa}\right\} &= C_1 \sum x_1^{\alpha+\lambda} x_2^{\beta} \cdots x_k^{\kappa} \\ &+ C_2 \sum x_1^{\alpha} x_2^{\beta+\lambda} \cdots x_k^{\kappa} \\ &+ \cdots C_k \sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa+\lambda} \\ &+ C_{k+1} \sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa} x_{k+1}^{\lambda} \end{aligned}$$

The constants result from sums of terms and are positive integers. Rearranging the equation, a recurrence formula for a sigma function of higher order is obtained.

$$\begin{aligned} \sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa} x_{k+1}^{\lambda} = \frac{1}{C_{k+1}} \Big\{ &\left(\sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa} \right) \cdot S_{\lambda} \\ &- C_1 \sum x_1^{\alpha+\lambda} x_2^{\beta} \cdots x_k^{\kappa} - C_2 \sum x_1^{\alpha} x_2^{\beta+\lambda} \cdots x_k^{\kappa} \\ &\cdots - C_k \sum x_1^{\alpha} x_2^{\beta} \cdots x_k^{\kappa+\lambda} \end{aligned}$$

Note that the lowest order sigma function, $\sum x_1^{\alpha}$, is by definition S_{α}. Thus by induction all sigma functions can be written as a polynomial in S.

The coefficients of a polynomial are symmetric functions of the roots, called elementary symmetric functions.

Definition A.2.3 (Elementary symmetric functions) The elementary symmetric function f_k of n variables is

$$f_k = \sum x_1 x_2 \ldots x_k$$

The nth order polynomial is represented by the product of its n factors.

$$a_n x^n + a_{n-1} x^{n-1} + \ldots a_0 = a_n(x - x_1)(x - x_2)\ldots(x - x_n)$$

When these factors are multiplied, the equation becomes

$$a_n x^n + a_{n-1} x^{n-1} + \ldots a_0 =$$

$$a_n \left\{x^n - (x_1 + x_2 + \ldots x_n)x^{n-1} + (x_1x_2 + x_1x_3 + \ldots)x^{n-2} + \ldots (-1)^n x_1x_2 \ldots x_n\right\}$$

and upon substituting the elementary functions

$$a_n x^n + a_{n-1} x^{n-1} + \ldots a_0 =$$

$$a_n \left\{x^n - f_1 x^{n-1} + f_2 x^{n-2} - \ldots (-1)^n f_n\right\}$$

Thus

$$f_1 = -\frac{a_{n-1}}{a_n} \qquad \text{A.2.1}$$

$$f_2 = \frac{a_{n-1}}{a_n}$$

$$\vdots$$

$$f_n = \frac{a_0}{a_n}$$

A.3 Fundamental Theorem of Symmetric Polynomials.

Theorem A.3.1: A symmetric polynomial can be expressed as a polynomial in the elementary symmetric functions.

Proof: It will be shown that the sums of powers function S_k can be written as a polynomial in the elementary symmetric functions. The theorem, then, follows directly from Theorem A.2.2.

A polynomial

$$G = (x - x_1)(x - x_2) \ldots (x - x_n)$$

of $n + 1$ variables is defferentiated with respect to x and re-arranged in the form

$$\frac{\partial G}{\partial x} = \frac{G}{(x - x_1)} + \frac{G}{(x - x_2)} + \ldots \frac{G}{(x - x_n)} \qquad \text{A.3.1}$$

G is now written in elementary function form

$$G = x^n - f_1 x^{n-1} + f_2 x^{n-2} - \ldots f_n$$

and divided by $(x - x_i)$ to obtain

$$\frac{G}{(x - x_i)} = x^{n-1} + (x_i - f_1)x^{n-2} + (x_i^2 - f_1 x_i + f_2)x^{n-3} + \ldots \qquad \text{A.3.2}$$

Differentiation of the elementary function form yields

$$\frac{\partial G}{\partial x} = nx^{n-1} - (n - 1)f_1 \, x^{n-2} + \ldots f_n \qquad \text{A.3.3}$$

Now replacing each term in A.3.1 by an appropriate form of A.3.2

$$\frac{\partial G}{\partial x} = nx^{n-1} + (S_1 - nf_1)x^{n-2} + (S_2 - f_1 \, S_2 + nf_2)x^{n-3} + \ldots \qquad \text{A.3.4}$$

Equating like terms in A.3.3 and A.3.4 yields

$$S_1 - nf_1 = -(n - 1)f_1$$

$$S_2 - f_1 \, S_1 + nf_2 = (n - 2)f_2$$

$$\vdots$$

$$S_{n-1} - f_1S_{n-2} + f_2S_{n-3} - \ldots (-1)^{n-1} \, nf_{n-1} = (-1)^{n-1}f_{n-1}$$

or upon rearranging

$$S_1 - f_1 = 0$$

$$S_2 - f_1S_1 + 2f_2 = 0$$

$$\vdots$$

$$S_{n-1} - f_1S_{n-2} + f_2S_{n-3} - \ldots (-1)^{n-1}(n - 1)f_{n-1} = 0$$

An additional general equation involving sums of powers greater than order $n - 1$ is obtained by multiplying each identity

$$x_i^{\,n} - f_1x_i^{\,n-1} + f_2x_i^{\,n-2} - \ldots (-1)^n f_n = 0 \qquad (i = 1, 2, \ldots n)$$

by $x_i^{\,k-n}$ and adding. The result, upon substituting S, is

$$S_k - f_1 S_{k-1} + f_2 S_{k-2} - \cdots + (-1)^n S_{k-n} = 0 \qquad (k > n)$$

The equations derived above are known as Newton's formulas. When they are solved, S is written as a polynomial in the elementary symmetric functions.

$$S_1 = f_1$$

$$S_2 = f_1^2 - 2f_2$$

$$\vdots$$

Thus the theorem is proved.

The coefficients of the polynomial of elementary symmetric functions in the theorem above are rational integer functions of the coefficients of the original symmetric polynomial. This results from the fact that the multipliers of the sigma functions (Theorem A.2.1) are coefficients of the symmetric polynomial.

APPENDIX B

EQUATIONS FOR EXAMPLES

B.1 EQUATIONS FOR EXAMPLE 1.

TABLE B.1.1

EXAMPLE 1
EQUATION SET 1

```
BEGIN                                                                  01.01
X0*1
(-A1)
X0*0
(+X4 X6 +X5 X6 )
END

BEGIN                                                                  01.02
X0*1
(-A0)
X0*0
(+X2 X4 +X2 X5 +X2 X6 +X3 X4 +X3 X5 +X3 X6 )
END

BEGIN                                                                  01.03
X0*1
(-B2)
X0*0
(+X4 X5 X6 )
END

BEGIN                                                                  01.04
X0*1
(-B1)
X0*0
(+X1 X4 X6 +X1 X5 X6 +X2 X4 X5 +X2 X5 X6 +X3 X4 X5 +X3 X4 X6 )
END

BEGIN                                                                  01.05
X0*1
(-B0)
X0*0
(+X1 X2 X4 +X1 X2 X5 +X1 X2 X6 +X1 X3 X4 +X1 X3 X5 +X1 X3 X6 +X2 X3 X4
 +X2 X3 X5 +X2 X3 X6 )
END
```

TABLE B.1.2

EXAMPLE 1
EQUATION SET 2

```
BEGIN                                                                      02.01
X1*0
(+B2 X4 +B2 X5 -A1 X4 X5 )
END

BEGIN                                                                      02.02
X1*0
(+B2 X2 X4 +B2 X2 X5 +B2 X2 X6 +B2 X3 X4 +B2 X3 X5 +B2 X3 X6 -A0 X4 X5 X6)
END

BEGIN                                                                      02.03
X1*1
(-B2 X4 X6 -B2 X5 X6 )
X1*0
(+B1 X4 X5 X6 -B2 X2 X4 X5 -B2 X2 X5 X6 -B2 X3 X4 X5 -B2 X3 X4 X6 )
END

BEGIN                                                                      02.04
X1*1
(-B2 X2 X4 -B2 X2 X5 -B2 X2 X6 -B2 X3 X4 -B2 X3 X5 -B2 X3 X6 )
X1*0
(+B0 X4 X5 X6 -B2 X2 X3 X4 -B2 X2 X3 X5 -B2 X2 X3 X6 )
END
```

TABLE B.1.3

EXAMPLE 1
EQUATION SET 3

```
BEGIN                                                                      03.01
X2*0
(+B2 X4 +B2 X5 -A1 X4 X5 )
END

BEGIN                                                                      03.02
X2*1
(+B2 X4 +B2 X5 +B2 X6 )
X2*0
(+B2 X3 X4 +B2 X3 X5 +B2 X3 X6 -A0 X4 X5 X6 )
END

BEGIN                                                                      03.03
X2*2
(-B2 X4*2 X5 -B2 X4 X5*2 -2 B2 X4 X5 X6 -B2 X5*2 X6 -B2 X5 X6*2 )
X2*1
(+B1 X4*2 X5 X6 +B1 X4 X5*2 X6 +B1 X4 X5 X6*2 -2 B2 X3 X4*2 X5
 -2 B2 X3 X4 X5*2 -2 B2 X3 X4 X5 X6 )
X2*0
(+B1 X3 X4*2 X5 X6 +B1 X3 X4 X5*2 X6 +B1 X3 X4 X5 X6*2 -B2 X3*2 X4*2 X5
 -B2 X3*2 X4 X5*2 -2 B2 X3*2 X4 X5 X6 -B2 X3*2 X4*2 X6 -B2 X3*2 X4 X6*2
 -B0 X4*2 X5 X6*2 -B0 X4 X5*2 X6*2 )
END
```

TABLE B.1.4

EXAMPLE 1
EQUATION SET 4

```
BEGIN                                                                    04.01
X4*1
(+B2 -A1 X5 )
X4*0
(+B2 X5 )
END

BEGIN                                                                    04.02
X4*4
(-A0*2 X5*3 X6 +A0 B1 X5*2 X6 -B2*2 X3*2 -B0 B2 X5 X6 )
X4*3
(-4 B2*2 X3*2 X5 +2 A0 B2 X3 X5*2 X6 -A0*2 X5*4 X6 -2 A0*2 X5*3 X6*2
 +2 A0 B1 X5*3 X6 +2 A0 B1 X5*2 X6*2 -3 B2*2 X3*2 X6 -3 B0 B2 X5*2 X6
 -2 B0 B2 X5 X6*2 )
X4*2
(-6 B2*2 X3*2 X5*2 +4 A0 B2 X3 X5*3 X6 +4 A0 B2 X3 X5*2 X6*2 -A0*2 X5*4 X6*2
 -A0*2 X5*3 X6*3 +A0 B1 X5*4 X6 +2 A0 B1 X5*3 X6*2 +A0 B1 X5*2 X6*3
 -9 B2*2 X3*2 X5 X6 -3 B0 B2 X5*3 X6 -3 B2*2 X3*2 X6*2 -4 B0 B2 X5*2 X6*2
 -B0 B2 X5 X6*3 )
X4*1
(-9 B2*2 X3*2 X5*2 X6 -6 B2*2 X3*2 X5 X6*2 +2 A0 B2 X3 X5*4 X6
 +4 A0 B2 X3 X5*3 X6*2 +2 A0 B2 X3 X5*2 X6*3 -4 B2*2 X3*2 X5*3 -B0 B2 X5*4 X6
 -2 B0 B2 X5*3 X6*2 -B2*2 X3*2 X6*3 -B0 B2 X5*2 X6*3 )
X4*0
(-B2*2 X3*2 X5*4 -3 B2*2 X3*2 X5*3 X6 -3 B2*2 X3*2 X5*2 X6*2
 -B2*2 X3*2 X5 X6*3 )
END
```

TABLE B.1.5

EXAMPLE 1
EQUATION SET 5

```
BEGIN                                                                    05.01
X3*2
(-A1*4 X5*6 +3 A1*3 B2 X5*4 X6 -3 A1*4 X5*5 X6 +6 A1*3 B2 X5*3 X6*2
 -3 A1*4 X5*4 X6*2 +A1 B2*3 X6*3 -3 A1*2 B2*2 X5 X6*3 +3 A1*3 B2 X5*2 X6*3
 -A1*4 X5*3 X6*3 -3 A1*2 B2*2 X5*2 X6*2 )
X3*1
(-2 A0 A1*2 B2 X5*5 X6 +2 A0 A1*3 X5*6 X6 +4 A0 A1 B2*2 X5*3 X6*2
 -8 A0 A1*2 B2 X5*4 X6*2 +4 A0 A1*3 X5*5 X6*2 -2 A0 B2*3 X5 X6*3
 +6 A0 A1 B2*2 X5*2 X6*3 -6 A0 A1*2 B2 X5*3 X6*3 +2 A0 A1*3 X5*4 X6*3 )
X3*0
(-A1*3 B0 X5*6 X6 +2 A1*2 B0 B2 X5*4 X6*2 -2 A1*3 B0 X5*5 X6*2
 -A1 B0 B2*2 X5*2 X6*3 +2 A1*2 B0 B2 X5*3 X6*3 -A1*3 B0 X5*4 X6*3
 +A0*2 B2*2 X5*4 X6*2 -A0*2 A1*2 X5*6 X6*2 -A0*2 B2*2 X5*3 X6*3
 +2 A0*2 A1 B2 X5*4 X6*3 -A0*2 A1*2 X5*5 X6*3 +A0 A1*2 B1 X5*6 X6
 -2 A0 A1 B1 B2 X5*4 X6*2 +2 A0 A1*2 B1 X5*5 X6*2 +A0 B1 B2*2 X5*2 X6*3
 -2 A0 A1 B1 B2 X5*3 X6*3 +A0 A1*2 B1 X5*4 X6*3 -A0*2 A1 B2 X5*6 X6 )
END
```

B.2 EQUATIONS FOR EXAMPLE 2.

TABLE B.2.1

EXAMPLE 2
EQUATION SET 1

```
BEGIN                                              01.01
X0'1
(-A0)
X0'0
(+X1 X2 +X1 X3)
END

BEGIN                                              01.02
X0'1
(-A1)
X0'0
(+X1 X5 +X4 X2 +X4 X3)
END

BEGIN                                              01.03
X0'1
(-A2)
X0'0
(+X4 X5)
END

BEGIN                                              01.04
X0'1
(-B0)
X0'0
(+X1 X2 X3)
END

BEGIN                                              01.05
X0'1
(-B1)
X0'1
(+X1 X2 X4 +X1 X3 X4 +X1 X3 X5 +X2 X3 X4)
END

BEGIN                                              01.06
X0'1
(-B2)
X0'0
(+X1 X4 X5 +X3 X4 X5)
END
```

TABLE B.2.2

EXAMPLE 2
EQUATION SET 2

```
BEGIN                                                   02.01
X1'1
(-A2 X2 -A2 X3 )
X1'0
(+A0 X4 X5 )
END

BEGIN                                                   02.02
X1'1
(-A2 X3 )
X1'0
(+A1 X4 X5 -A2 X2 X4 -A2 X3 X4 )
END

BEGIN                                                   02.03
X1'1
(-A2 X2 X3 )
X1'0
(+B0 X4 X5 )
END

BEGIN                                                   02.04
X1'1
(-A2 X2 X4 -A2 X3 X4 -A2 X3 X5 )
X1'0
(+B1 X4 X5 -A2 X2 X3 X4 )
END

BEGIN                                                   02.05
X1'1
(-A2 )
X1'0
(+B2 -A2 X3 )
END
```

TABLE B.2.3

EXAMPLE 2
EQUATION SET 5

```
BEGIN                                                          03.01
X4'0
(+B0 X2 +B0 X3 -A0 X2 X3 )
END

BEGIN                                                          03.02
X4'0
(+B0 X5'2 -A1 X2 X3 X5 +A2 X2'2 X3 +A2 X2 X3'2 )
END

BEGIN                                                          03.03
X4'1
(+B0 X2 X5 +B0 X3 X5 )
X4'0
(+B0 X3 X5'2 -B1 X2 X3 X5 +A2 X2'2 X3'2 )
END

BEGIN                                                          03.04
X4'1
(+B0 X5 )
X4'0
(-B2 X2 X3 +A2 X2 X3'2 )
END
```

TABLE B.2.4

EXAMPLE 2
EQUATION SET 4

```
BEGIN                                                          04.01
X2'1
(+B0 -A0 X3 )
X2'0
(+B0 X3 )
END

BEGIN                                                          04.02
X2'2
(+A2 X3 )
X2'1
(-A1 X3 X5 +A2 X3'2 )
X2'0
(+B0 X5'2 )
END

BEGIN                                                          04.03
X2'2
(-B2 )
X2'1
(+B1 X5 -B2 X3 +A2 X3'2 )
X2'0
(-B0 X5'2 )
END
```

TABLE B.2.5

EXAMPLE 2
EQUATION SET 5

```
BEGIN                                                                   05.01
X5'2
(+B0'2 -2 A0 B0 X3 +A0'2 X3'2 )
X5'1
(+A1 B0 X3'2 -A0 A1 X3'3 )
X5'0
(+A0 A2 X3'4 )
END

BEGIN                                                                   05.02
X5'2
(-B0'2 +2 A0 B0 X3 -A0'2 X3'2 )
X5'1
(-B0 B1 X3 +A0 B1 X3'2 )
X5'0
(-A2 B0 X3'3 -A0 B2 X3'3 +A0 A2 X3'4 )
END
```

TABLE B.2.6

EXAMPLE 2
EQUATION SET 6

```
BEGIN                                                                   06.01
X3'6
(+4 A0'6 A2'2 -A0'5 A1'2 A2 )
X3'5
(-20 A0'5 A2'2 B0 -4 A0'6 A2 B2 +5 A0'4 A1'2 A2 B0 +A0'5 A1'2 B2 )
X3'4
(+41 A0'4 A2'2 B0'2 +18 A0'5 A2 B0 B2 -A0'4 A1 A2 B0 B1 -10 A0'3 A1'2 A2 B0'2
 -4 A0'4 A1'2 B0 B2 +A0'5 A2 B1'2 -A0'5 A1 B1 B2 +A0'6 B2'2 )
X3'3
(-44 A0'3 A2'2 B0'3 -32 A0'4 A2 B0'2 B2 +4 A0'3 A1 A2 B0'2 B1
 +10 A0'2 A1'2 A2 B0'3 +6 A0'3 A1'2 B0'2 B2 -4 A0'4 A2 B0 B1'2
 +4 A0'4 A1 B0 B1 B2 -4 A0'5 B0 B2'2 )
X3'2
(+26 A0'2 A2'2 B0'4 +28 A0'3 A2 B0'3 B2 -6 A0'2 A1 A2 B0'3 B1
 -5 A0 A1'2 A2 B0'4 -4 A0'2 A1'2 B0'3 B2 +6 A0'3 A2 B0'2 B1'2
 -6 A0'3 A1 B0'2 B1 B2 +6 A0'4 B0'2 B2'2 )
X3'1
(-8 A0 A2'2 B0'5 -12 A0'2 A2 B0'4 B2 +4 A0 A1 A2 B0'4 B1 +A1'2 A2 B0'5
 +A0 A1'2 B0'4 B2 -4 A0'2 A2 B0'3 B1'2 +4 A0'2 A1 B0'3 B1 B2
 -4 A0'3 B0'3 B2'2 )
X3'0
(+A0 A2 B0'4 B1'2 -A1 A2 B0'5 B1 -A0 A1 B0'4 B1 B2 +A2'2 B0'6
 +2 A0 A2 B0'5 B2 +A0'2 B0'4 B2'2 )
END
```

B.3 EQUATIONS FOR EXAMPLE 3.

TABLE B.3.1

EXAMPLE 3
EQUATION SET 1

```
BEGIN                                                   01.01
X1*1
(+X2 X3 )
X1*0
(-B0 )
END

BEGIN                                                   01.02
X1*1
(+X2 X5 )
X1*0
(-B1 +X2 X3 X4 )
END

BEGIN                                                   01.03
X1*1
(+X3 X4 +X3 X5 )
X1*0
(-B2 +X2 X4 X5 )
END

BEGIN                                                   01.04
X1*1
(+X4 X5 )
X1*0
(-B3 +X3 X4 X5 )
END

BEGIN                                                   01.05
( -A3 +X4 X5 )
END
```

TABLE B.3.2

EXAMPLE 3
EQUATION SET 2

```
BEGIN                                                    02.01
X2*1
(+X3*2 X4 )
X2*0
(+B0 X5 -B1 X3 )
END

BEGIN                                                    02.02
X2*2
(+X4 X5 )
X2*1
(-B2 )
X2*0
(+B0 X4 +B0 X5 )
END

BEGIN                                                    02.03
X2*1
(-B3 X3 +X3*2 X4 X5 )
X2*0
(+B0 X4 X5 )
END

BEGIN                                                    02.04
( -A3 +X4 X5 )
END
```

TABLE B.3.3

EXAMPLE 3
EQUATION SET 3

```
BEGIN                                                    03.01
X3*4
(+B0 X4*2 +B0 X4 X5 )
X3*3
(-B1 B2 )
X3*2
(+B1*2 X5 +B0 B2 X5 )
X3*1
(-2 B0 B1 X5*2 )
X3*0
(+B0*2 X5*3 )
END

BEGIN                                                    03.02
X3*2
(+B1 X4 X5 )
X3*1
(-B0 X4 X5*2 -B1 B3 +B0 X4*2 X5 )
X3*0
(+B0 B3 X5 )
END

BEGIN                                                    03.03
( -A3 +X4 X5 )
END
```

TABLE B.3.4

EXAMPLE 3
EQUATION SET 4

```
BEGIN                                                                   04.01
X4*8
(+B0*3 X5*3 )
X4*7
(-3 B0*3 X5*4 )
X4*6
(+2 B0*3 X5*5 -2 B0*2 B1 B3 X5*2 )
X4*5
(+B0 B1*2 B2 X5*3 +2 B0*3 X5*6 )
X4*4
(+4 B0*2 B1 B3 X5*4 +B1*4 X5*4 -2 B0 B1*2 B2 X5*4 -3 B0*3 X5*7
 +B0 B1*2 B3*2 X5 +B0*2 B2 B3*2 X5 )
X4*3
(+B0 B1*2 B2 X5*5 -B1*3 B2 B3 X5*2 +B0*3 X5*8 +B0 B1*2 B3*2 X5*2
 -B0*2 B2 B3*2 X5*2 )
X4*2
(-B1*3 B2 B3 X5*3 +B0 B1*2 B3*2 X5*3 -2 B0*2 B1 B3 X5*6 -B0*2 B2 B3*2 X5*3
 -B0 B1 B2 B3*3 +B0*2 B3*4 )
X4*1
(-2 B0 B1 B2 B3*3 X5 +B1*2 B2*2 B3*2 X5 +B0 B1*2 B3*2 X5*4 +B0*2 B2 B3*2 X5*4
 +2 B0*2 B3*4 X5 )
X4*0
(-B0 B1 B2 B3*3 X5*2 +B0*2 B3*4 X5*2 )
END

BEGIN                                                                   04.02
X4*1
(+X5 )
X4*0
(-A3 )
END
```

TABLE B.3.5

EXAMPLE 3
EQUATION SET 5

```
BEGIN                                                                    05.01
X5*10
(+B0*3 A3*3 )
X5*9
(-2 B0*2 B1 B3 A3*2 )
X5*8
(-3 B0*3 A3*4 +B0 B1*2 B3*2 A3 +B0*2 B2 B3*2 A3 )
X5*7
(+B0 B1*2 B2 A3*3 -B0 B1 B2 B3*3 +B0*2 B3*4 )
X5*6
(+2 B0*3 A3*5 -B1*3 B2 B3 A3*2 +B0 B1*2 B3*2 A3*2 -B0*2 B2 B3*2 A3*2 )
X5*5
(+4 B0*2 B1 B3 A3*4 +B1*4 A3*4 -2 B0 B1*2 B2 A3*4 -2 B0 B1 B2 B3*3 A3
 +B1*2 B2*2 B3*2 A3 +2 B0*2 B3*4 A3 )
X5*4
(+2 B0*3 A3*6 -B1*3 B2 B3 A3*3 +B0 B1*2 B3*2 A3*3 -B0*2 B2 B3*2 A3*3 )
X5*3
(+B0 B1*2 B2 A3*5 -B0 B1 B2 B3*3 A3*2 +B0*2 B3*4 A3*2 )
X5*2
(-3 B0*3 A3*7 +B0 B1*2 B3*2 A3*4 +B0*2 B2 B3*2 A3*4 )
X5*1
(-2 B0*2 B1 B3 A3*6 )
X5*0
(+B0*3 A3*8 )
END
```

VITA

Louis Charles Thomason

Candidate for the Degree of

Doctor of Philosophy

Thesis: DRIVING-POINT IMPEDANCE SYNTHESIS WITH TOPOLOGICAL SPECIFICATIONS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born on November 26, 1938, in Oklahoma City, Oklahoma, the son of Louis C. and Ada B. Thomason.

Education: Attended primary and secondary schools in Enid, Oklahoma, and graduated from Enid High School in May, 1956; received the Bachelor of Science degree from the Oklahoma State University, with a major in Electrical Engineering, in January, 1961; received the Master of Science degree from the Oklahoma State University, with a major in Electrical Engineering, in August, 1961; completed requirements for the Doctor of Philosophy degree in May, 1966.

Professional Experience: Employed by General Electric Company, Louisville, Kentucky, as a Summer Engineer, May to September, 1960, specific assignment included specification of components for the power supply of a microwave oven. Employed as an Instructor in the School of Electrical Engineering, Oklahoma State University, September, 1961 to May, 1965.

Professional Organizations: Member of the Association for Computing Machinery, Member of the Institute of Electrical and Electronics Engineers, Associate Member of Sigma Xi.