

RBAdler

PIECEWISE-LINEAR NETWORK THEORY

THOMAS EDWIN STERN

TECHNICAL REPORT 315

JUNE 15, 1956

RESEARCH LABORATORY OF ELECTRONICS
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS

The Research Laboratory of Electronics is an interdepartmental laboratory of the Department of Electrical Engineering and the Department of Physics.

The research reported in this document was made possible in part by support extended the Massachusetts Institute of Technology, Research Laboratory of Electronics, jointly by the U. S. Army (Signal Corps), the U. S. Navy (Office of Naval Research), and the U. S. Air Force (Office of Scientific Research, Air Research and Development Command), under Signal Corps Contract DA36-039-sc-64637, Project 102B; Department of the Army Projects 3-99-10-022 and DA3-99-10-000.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
RESEARCH LABORATORY OF ELECTRONICS

Technical Report 315

June 15, 1956

PIECEWISE-LINEAR NETWORK THEORY

Thomas Edwin Stern

This report is based on a thesis submitted to the Department of Electrical Engineering, M.I.T., May 14, 1956, in partial fulfillment of the requirements for the degree of Doctor of Science.

Abstract

A systematic approach to the problems of analysis and synthesis of piecewise-linear systems that do not contain memory is presented. These systems provide a link between the general studies of nonlinear systems, exemplified by the work of Wiener, Zadeh, and others, and the needs of the practical circuit designer. In the area of analysis, straightforward procedures are developed for handling resistive piecewise-linear networks. The methods are based upon an algebra of inequalities. Examples of applications to analysis are given. In the area of synthesis, techniques are developed by using diode networks for the construction of general piecewise-linear driving-point functions, as well as generators of piecewise-linear voltage transfer functions of several variables. Some of the properties of nonlinear resistive networks, in general, and diode networks, in particular, are discussed. Applications of the inequality algebra to the synthesis problem are also considered. Two forms of the transfer-synthesis problem are treated: arbitrary function synthesis, and particular function synthesis. Examples of the practical application of the techniques that are discussed to the construction of generators of functions of one and two variables are given.

•
•
•

•
•
•

Table of Contents

I. Introduction	1
II. Symbolism: An Algebra of Inequalities	2
2.1 Motivation	2
2.2 Definitions and Theorems	4
2.3 Symbolic Representation of Piecewise-Linear Functions	11
III. Applications to Analysis	15
3.1 Symbolic Description of Network Elements	15
a. Elements of a diode network	15
b. The vacuum tube	16
3.2 Series-Parallel Networks	16
3.3 Non-Series Parallel Networks	21
a. Bridge diode network	21
b. Triode feedback amplifier	23
IV. General Properties of Piecewise-Linear Networks	25
4.1 The Resistive Diode Network as a Basis For Synthesis	25
4.2 Theorems Concerning The Behavior of Diode Networks	26
4.3 Duality in Nonlinear Resistive Networks	30
V. Applications To Synthesis	34
5.1 Introduction	34
5.2 Driving-Point Function Synthesis	35
a. Strictly convex or concave functions	35
b. Arbitrary functions	38
5.3 Transfer Function Synthesis	41
a. General purpose function generation	41
Tabulation, tessellation, and interpolation	41
Unit functions and function generators	48
b. Special purpose function generation	56
VI. Suggestions For Future Work	63
Appendix I. Proofs Of The Theorems Of Section II	63
Appendix II. Proofs Of The Theorems Of Section IV	67
Appendix III. Tessellation Theorem	69
Bibliography	75

•
•
•

•
•
•

I. INTRODUCTION

Within the past ten years, the field of nonlinear network theory has been attacked on a large scale for the first time. The contributions to the theory have been many and varied, indicating the intense interest that has developed since the end of the second World War. One of the principal reasons for this interest is that the linear-system theorists succeeded in setting upper bounds to their own capabilities. For example, if a filter is desired to separate a signal from its associated noise, for which statistical descriptions are given, Wiener and Lee have shown that a certain optimum linear filter can perform this task within a certain degree of perfection, and no other type of linear filter can come any closer to the desired performance. Naturally, as soon as an upper limit is recognized, the question is immediately asked, "How can this limit be exceeded?" The answer, of course, is to use a nonlinear system.

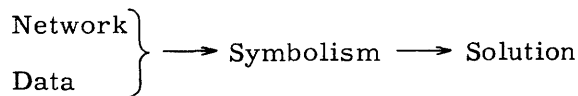
Many techniques have recently been developed for dealing with certain specific nonlinear problems. As a rule, they are interesting as far as their limited applications are concerned, but they cannot be generalized. The reason for this limitation is clear. Since linear systems constitute only a minute fraction of the complete class of physical systems, it is to be expected that the class of nonlinear systems will be of enormous size and complexity.

Wiener (28), Zadeh (29), and Singleton (22) made important contributions to the general theory, especially with regard to classifying nonlinear systems. Most of their efforts were concerned with analyzing, synthesizing, and classifying two terminal-pair "black boxes." Although these general contributions are of fundamental importance, they are often too unwieldy to be of much practical value.

The methods of analysis and synthesis given in this report are intended to bridge the gap between the specific and general studies of nonlinear systems. Since piecewise-linear systems can be used to approximate almost any type of nonlinearity, and still retain some of the simplicity of linear systems, a thorough investigation of their properties and capabilities appears to be very appropriate. The scope of this work includes the development of a general systematic approach to the problems of piecewise-linear network analysis and synthesis, as well as an approach to those problems that can be approximated.

It is readily apparent from past experience that the concise mathematical formulation of a problem is often the most important step in proceeding to its solution. The application of operational calculus to linear electrical networks, and more recently, of Boolean algebra to switching circuits, are two striking examples. So far, concise mathematical representation has been lacking in piecewise-linear networks. The first step in this investigation is, therefore, the representation of piecewise-linear problems by a concise, easily manipulated, algebraic symbolism. In Section II, an "algebra of inequalities" is presented. This symbolism establishes an efficient means of characterizing, analyzing, and synthesizing piecewise-linear networks and systems. Inequalities play a fundamental role in these problems.

Section III describes applications of the symbolism to problems of analysis. The "flow diagram" of the analysis problem is



In the case of networks with no energy storage elements (the only type considered here), the mechanization of the first arrow is quite simple. Mechanization of the second arrow is perfectly systematic and straightforward but requires more labor, as is to be expected.

Section IV deals with some of the general properties of diode networks. Since the diode network has been selected in this work as a basis for piecewise-linear synthesis, a study of these general properties gives a useful preamble for the development of synthesis procedures. In addition, some of the properties discussed, such as an extension of the duality principle to nonlinear resistive networks, are of interest in their own right.

The algebraic characterization of synthesis problems introduces a new philosophy of diode network synthesis. Section V deals with both driving-point and transfer synthesis. The basic emphasis, however, is placed upon synthesis of voltage transfer functions of several input variables: that is, the design of analog function generators. The synthesis procedures involve (a) expressing the function to be synthesized in terms of the inequality algebra, and (b) mechanizing the algebraic operations with simple diode networks. Numerous examples of the broad possibilities offered by this method in the field of general zero memory function generation are given in Section V.

II. SYMBOLISM: AN ALGEBRA OF INEQUALITIES

2.1 MOTIVATION

In developing an efficient mathematical method of analyzing a broad class of problems, the first question that arises is "What are the basic properties peculiar to this class of problems?" The fundamental properties of piecewise-linear systems are:

1. They are characterized by functional relationships composed of a finite number of linear regions adjoining one another.

2. The change-over from one linear region to the next is determined by the point at which some quantity becomes greater or less than some other quantity.

Although those systems appear to be closely related to linear systems, it is clear that the superposition principle is not valid in piecewise-linear systems. This fact alone increases enormously the difficulties of analysis and synthesis, and makes the development of an algebraic method of handling them, which differs from conventional techniques, worth while. It may be observed from property 2 that the words "greater" and "less," i. e., inequalities, play important roles in these systems. It was the recognition of this fact that led to the development of a symbolism that would enable

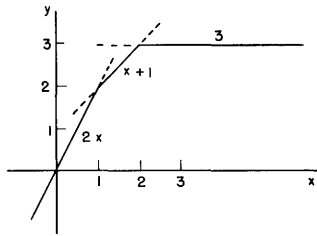


Fig. 1. Piecewise-linear function.

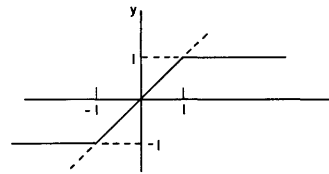


Fig. 2. Piecewise-linear function.

the handling of such concepts algebraically: in effect, an "algebra of inequalities."

The basic feature of the algebra is the symbolic representation of the words "greatest" and "least." After attempting various symbolic methods of describing piecewise-linear functions, it appeared that two very simple transformations were useful and efficient, both in indicating a systematic method of analysis, and in forming the basis of a productive synthesis technique. They are both many-to-one transformations, which operate on sets of numbers or functions. The first, represented by $(\) \phi^+$, selects the greatest of the set of elements appearing as its argument. Similarly, the second, represented by $(\) \phi^-$, selects the least of the set of elements appearing as its argument. Suitable combinations of these transformations enable the algebraic expression of the behavior of any piecewise-linear function without resorting to writing several equations with inequality relationships in order to indicate the region of validity of each equation. Two examples serve to illustrate the convenience of this symbolism in representing piecewise-linear functions analytically.

EXAMPLE 1. Consider the relationship of Fig. 1. In conventional notation it is described by

$$y = \begin{cases} 2x & x \leq 1 \\ x + 1 & 1 \leq x \leq 2 \\ 3 & 2 \leq x \end{cases}$$

If the lines are extended beyond the breakpoints, it is clear that the function is everywhere given by the particular line that is less than the others. Thus its algebraic representation is

$$y = (2x, x+1, 3) \phi^-$$

EXAMPLE 2. Consider the limiter curve of Fig. 2. A conventional description is

$$y = \begin{cases} -1 & x \leq -1 \\ x & -1 \leq x \leq 1 \\ 1 & 1 \leq x \end{cases}$$

The symbolic description is

$$y = [(1, x) \phi^-, -1] \phi^+$$

or equivalently,

$$y = [(x, -1) \phi^+, 1] \phi^-$$

It is clear from example 2 that the symbolic representation is not necessarily unique. It will become apparent later that the variety of possible, equivalent representations of a particular function allows for a considerable amount of flexibility in synthesis techniques.

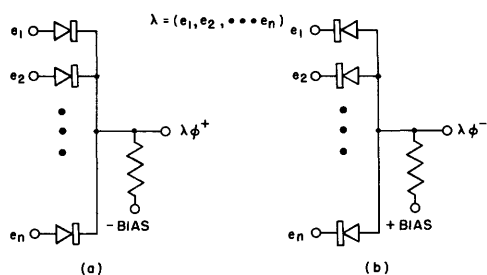


Fig. 3. ϕ^+ and ϕ^- circuits.

One important reason for choosing these particular transformations is the ease with which they can be mechanized as voltage transfer functions, when diode networks are used. The circuit of Fig. 3a performs a ϕ^+ transformation on its input voltages; that is,

$$e_o = \lambda \phi^+ = (e_1, e_2, \dots, e_n) \phi^+$$

Similarly, the circuit of Fig. 3b performs a ϕ^- transformation on its input voltages; that is,

$$e_o = \lambda \phi^- = (e_1, e_2, \dots, e_n) \phi^-$$

Note that the bias voltages in each circuit should be greater in magnitude than the most negative value of the input voltages in the first case, or the most positive value of the input voltages in the second case. These two circuits form the basis for piecewise-linear voltage transfer function synthesis. It should be clear from these illustrations that, once a network transfer characteristic is prescribed in terms of the inequality algebra, it is theoretically a simple matter to synthesize it. With the foregoing background and motivation, we are now ready to proceed with the formal structure of the algebra.

2.2 DEFINITIONS AND THEOREMS

The elements of the algebra are known as scalars and vectors. (The structure of the inequality algebra is similar to the algebra of vector spaces.) They are formed from the elements of an ordered field, $R^\#$; the real number system (for definitions of unfamiliar

terms see ref. 1).

DEFINITION 1. A scalar is any member of the field. (Scalars will be denoted by lower-case Roman letters or by numbers.)

DEFINITION 2. A vector is any proper subset of the field. A vector will be denoted by a single Greek letter, ξ , to indicate the whole set of elements, or by (a, b, \dots, n) to enumerate each element. The elements of a vector are scalars. Note that, unlike ordinary vectors, the order in which the elements of ξ appear is unimportant.

A scalar can be either a constant, (a) , a variable, (x) , or a function of one or more variables, $(a + bx)$. Likewise, a vector can contain members which are any of these three.

It should be observed that, according to the definitions, a single element standing alone may be either a vector or a scalar. In the development that follows, single elements will be treated as vectors or scalars interchangeably, but their status at any time will be clear from the context.

DEFINITION 3. Scalar multiplication. The product of a scalar, c , and a vector, $\lambda = (\ell_1, \ell_2, \dots, \ell_n)$, is denoted by $c\lambda$, where $c\lambda = (c\ell_1, c\ell_2, \dots, c\ell_n)$.

DEFINITION 4. Vector addition. The sum of two vectors, $\alpha = (a_1, a_2, \dots, a_n)$ and $\beta = (b_1, b_2, \dots, b_n)$, is denoted by $\alpha \oplus \beta$, where $\alpha \oplus \beta$ is the set of all scalars, $a_p + b_q$, a_p in α , and b_q in β .

EXAMPLE. Let $\alpha = (0, 3, 3 - 2x)$, $\beta = (0, -2x)$. Then $\alpha \oplus \beta = (0, 3, 3 - 2x, -2x, 3 - 4x)$.

DEFINITION 5. The union of two vectors, α and β , is denoted by (α, β) , where (α, β) is a set of scalars that is the union of the set of all scalars in α , and the set of all scalars in β .

EXAMPLE. For the α and β used above,

$$(\alpha, \beta) = (0, 3, 3 - 2x, -2x)$$

Clearly, definitions 3 and 4 reduce to the ordinary rules for adding and multiplying real numbers when the vectors involved contain only one element. This is essential, since a one-element vector can be assumed also to be a scalar, the rules of combination of scalars being the familiar rules for addition, subtraction, multiplication, and division.

DEFINITION 6. Let α be a vector of which p is the greatest element. Then the transformation, ϕ^+ , takes α into p . Or, symbolically, $\alpha \phi^+ = p$.

DEFINITION 7. Let α be a vector of which q is the least element. Then the transformation, ϕ^- , takes α into q . Or, symbolically, $\alpha \phi^- = q$. Note that a transformed vector becomes a scalar.

With the basic definitions set forth, we can now proceed to the various theorems that facilitate the application of the algebra to practical problems. Naturally, there are innumerable theorems which can be derived. The few that follow are the ones that have

most frequent application in the solutions of typical problems. Proofs are presented in Appendix I.

- THEOREM 1.** 1. Commutative law: $a \oplus \beta = \beta \oplus a$.
 2. Associative laws: $(a \oplus \beta) \oplus \gamma = a \oplus (\beta \oplus \gamma)$.
 $c(da) = (cd)a$.
 3. Distributive law: $c(a \oplus \beta) = ca \oplus c\beta$.

- THEOREM 2.** $(ca)\phi^\pm = c(a\phi^\pm)$ $c \geq 0$
 $(ca)\phi^\pm = c(a\phi^\pm)$ $c \leq 0$

or equivalently,

$$(ca)\phi^\pm = (0, c)\phi^+(a\phi^\pm) + (0, c)\phi^-(a\phi^\mp)$$

A special case of theorem 2 is

$$a\phi^\pm = - [(-a)\phi^\mp]$$

THEOREM 3. Let $a = (a)$. Then

$$a\phi^+ = a\phi^- = a$$

THEOREM 4. $(a \oplus \beta)\phi^\pm = a\phi^\pm + \beta\phi^\pm$

THEOREM 5. $(a\phi^\pm, \beta)\phi^\pm = (a, \beta)\phi^\pm$

THEOREM 6. Inversion theorem. Let

$$y = F(x) = [f_1(x), f_2(x), \dots, f_n(x)]\phi^\pm$$

If each $f_p(x)$ is a strictly monotonic, increasing (decreasing), and continuous function, then

$$x = F^{-1}(y) = [f_1^{-1}(y), f_2^{-1}(y), \dots, f_n^{-1}(y)]\phi^\mp(\pm)$$

in which $f_p^{-1}(y)$ is the inverse of $f_p(x)$, that is,

$$y = f_p[f_p^{-1}(y)] \text{ and } y = f_p^{-1}[f_p(y)]$$

Here and in the following discussion, the words and symbols in parentheses constitute alternative statements of the theorem. For example, in this case the ϕ^\mp transformation applies to increasing functions and the ϕ^\pm transformation to decreasing functions. It should be observed (and it will be pointed out in the illustrations that follow), that theorem 6 establishes sufficient conditions for inversion. This does not imply that a function that does not satisfy the above conditions cannot be inverted.

EXAMPLE 1. Given the network of Fig. 4, find its impedance and admittance functions. (In the discussion of piecewise-linear resistive networks, driving-point

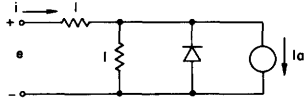


Fig. 4. Diode network.

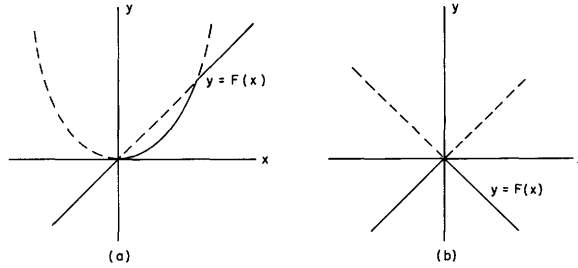


Fig. 5. Non-invertible functions.

voltage-current and current-voltage relationships will be known as impedances and admittances, following the terminology of linear network theory.)

First, by methods that will be described in Section III, the impedance function, $e = z(i)$, is easily found to be

$$e = (i, 2i - 1)\phi^+$$

To find the admittance, $i = y(e)$, theorem 6 can be applied to yield

$$i = \left[f_1^{-1}(e), f_2^{-1}(e) \right] \phi^- = \left(e, \frac{e+1}{2} \right) \phi^- = y(e)$$

EXAMPLE 2. Given the function $y = F(x) = \left[f_1(x), f_2(x) \right] \phi^-$

Find $x = F^{-1}(y)$ when

(a). $f_1(x) = x^2$ and $f_2(x) = x$ (See Fig. 5a.)

In this case, F^{-1} exists, since the over-all function is a one-to-one transformation; f_1^{-1} does not exist, since it is double-valued in x , and f_2^{-1} exists. Although the function does not satisfy the conditions of theorem 6, an expression for its inverse can be found in the form,

$$x = F^{-1}(y) = \left[g(y), y \right] \phi^+$$

where

$$g(y) = \begin{cases} -\infty & y \leq 0 \\ |\sqrt{y}| & y \geq 0 \end{cases}$$

Note that $g(y)$ must be defined in this somewhat artificial manner, since it must be less than y for all negative values of y .

Application of theorem 6 to this example would yield the meaningless result,

$$x = (\pm\sqrt{y}, y)\phi^\pm$$

Actually there would be no way of determining which sign should be assigned to the transformation, since the functions are both increasing and decreasing.

(b). $f_1(x) = x$, and $f_2(x) = -x$ (See Fig. 5b.)

In this case each f is monotonic and continuous, but one is increasing while the other is decreasing. Again, if theorem 6 were applied, the result would be ambiguous concerning the sign of the transformation. Obviously, this should be expected, since the over-all function, being double-valued in y , cannot be inverted.

These examples demonstrate that the conditions on theorem 6 provide a check on the invertability of a function. However, if it is found that a particular function does not satisfy the conditions, it is worth while to examine it more closely before deciding that it cannot be inverted. In all of the practical problems that follow, the functions will always be piecewise-linear so that the individual elements will be of the form $(a + bx)$. Therefore, strict monotonicity is assured if $b \neq 0$, and the function is always invertible if all the coefficients of x are nonzero and of the same sign.

THEOREM 7. Implicit Equation theorem. Let

$$F(x, y) = [f_1(x, y), f_2(x, y), \dots, f_n(x, y)] \phi^\pm = 0$$

If

1. Each f_p is continuous in x and y ;
 2. Each f_p is strictly monotonically increasing (decreasing) in y for any constant value of x ;
 3. For each x there is some value of y of such a kind that $f_p(x, y) = 0$, for any p ;
- then, the implicit equation can be solved explicitly for y in the form, $y = G(x) = [g_1(x), g_2(x), \dots, g_n(x)] \phi^{\mp(\pm)}$, where $y = g_p(x)$ is the explicit solution of the equation,

$$f_p(x, y) = 0$$

Again, an example will clarify the statement of the theorem.

EXAMPLE. Consider the equation,

$$F(x, y) = [(-x+y, -x+2y-2)\phi^+, x+y+1] \phi^- = 0$$

Note first that all of the coefficients of y are of the same sign, so that condition 2 of theorem 7 is satisfied if we attempt to solve for y . However, the coefficients of x are not of the same sign, so that difficulty should be anticipated in attempting to solve explicitly for x . The problem is clarified by reference to Fig. 6, which shows a portion of the surface, $z = F(x, y)$. The intersection of this surface with the x - y plane is the desired explicit solution. It can be seen from Fig. 6 that this intersection is single-valued for y as a function of x , but not for x as a function of y . Clearly, solution for x is impossible, which is the reason why theorem 7 does not apply in this case.

Now, to solve for y , the equation can first be written as

$$[f_1(x, y), f_2(x, y)]\phi^- = 0$$

where

$$f_1(x, y) = (-x+y, -x+2y-2)\phi^+$$

$$f_2(x, y) = x+y+1$$

To apply theorem 7, the equation, $f_1(x, y) = 0$, must first be solved explicitly for y . A preliminary application of theorem 7 performs this operation, giving

$$y = (x, \frac{x+2}{2})\phi^- = g_1(x)$$

For the equation, $f_2(x, y) = 0$,

$$y = -x - 1 = g_2(x)$$

Thus the explicit solution for y is

$$y = G(x) = [(x, \frac{x+2}{2})\phi^-, -x - 1]\phi^+$$

From the foregoing example, a corollary to theorem 7, which applies only to piecewise-linear functions, is readily deduced.

COROLLARY. The implicit equation,

$$F(x, y) = (a_1+b_1x+c_1y, a_2+b_2x+c_2y, \dots, a_n+b_nx+c_ny)\phi^\pm = 0$$

is solvable explicitly for y as a function of x , if and only if all of the c_p 's are nonzero and of the same sign.

Theorems 6 and 7 require strict monotonicity and continuity. However, in many analysis and synthesis problems we deal with monotonic functions that do not fulfill the conditions of being strictly monotonic and continuous. A simple example of this is the voltage-current characteristic of an ideal diode, which has one region of zero slope and another of infinite slope. It is useful to be able to deal analytically with such functions, and, to this end, the two following functions will be defined.

DEFINITION 8. The function, $y = 0(x)$, (read zero of x) is defined as

$$y = \lim_{n \rightarrow \infty} \left(\frac{x}{n} \right) = 0 \text{ (for all } x \text{)}$$

DEFINITION 9. The function, $y = \infty(x)$, (read infinity of x) is defined as

$$y = \lim_{n \rightarrow \infty} (nx) = \begin{cases} +\infty & x > 0 \\ 0 & x = 0 \\ -\infty & x < 0 \end{cases}$$

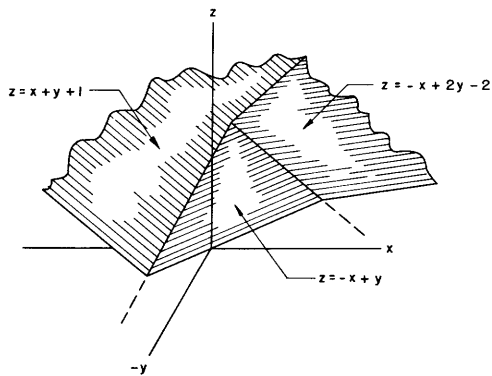


Fig. 6. Implicit function.

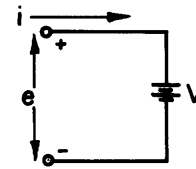


Fig. 7. Voltage source.

These two functions were defined by a limiting process rather than by writing the limiting values directly, because it is their behavior for very large but finite n which is of interest. A function that is constant over a region, or has infinite slope, is merely an idealization of a function derived from a physical problem, which is nearly constant or has a very large slope. For example, the characteristic of the ideal diode that has just been mentioned is actually an idealization of a physical diode which has a very high forward conductance and back resistance.

Thus, the two functions just defined can be used to represent idealized functions of zero or infinite slope, if we always keep in mind that they will be treated in the algebraic manipulations as if n were very large but finite. One consequence of this is that, for any finite n , they are inverses, although the two functions are not inverses in the limit. They will be treated as inverses in the discussion that follows, and functions containing them will be treated as if they were strictly monotonic and continuous. Some of their properties are:

1. $0(x) = \infty^{-1}(x)$
2. $\infty(x) = 0^{-1}(x)$
3. $0(x) + f(x) = f(x)$ (for any f)
4. $\infty(x) + f(x) = \begin{cases} \infty(x) & x \neq 0 \\ f(x) & x = 0 \end{cases}$

EXAMPLE. The impedance of the voltage source of Fig. 7 is

$$e = z(i) = V$$

Its admittance is

$$i = y(e) = z^{-1}(e)$$

But

$$z(i) = V = V + 0(i) = e$$

$$0(i) = e - V$$

$$i = \infty(e - V)$$

Note that the function, $0(i)$ was added to V rather than subtracted, because the ideal voltage source is actually an approximation of a source with a finite, positive resistance. As a result, the admittance function shows that, if e becomes slightly greater than V , a large positive current will flow, which is in keeping with the physics of the problem. On the other hand, if $0(i)$ had been subtracted (or, equivalently, added to the other side of the equation), the admittance function would be $i = \infty(V - e)$, indicating a large negative current when e is slightly greater than V , a characteristic of a source with a small negative resistance. Thus, when using these two functions it is wise to make sure that the chosen function corresponds to the actual physical situation.

2.3 SYMBOLIC REPRESENTATION OF PIECEWISE-LINEAR FUNCTIONS

In the previous section the monotonic nature of the functions under discussion played an important role. In the case of functions which are everywhere differentiable (piecewise-linear functions are not), this property is associated with the sign of the derivative. Another important property, which plays a vital part in synthesis procedures, is the convexity or concavity of a function. The concept of convex and concave functions (not to be confused with convex sets) was originally developed by Jensen (10) and his definitions will be used here. However, his convention regarding convexity and concavity will be reversed to correspond with our intuitive concepts of convex and concave shapes.

DEFINITION 10.

(a) For functions of a single variable, a function, $f(x)$, is convex (concave), if

$$f\left(\frac{x_1 + x_2}{2}\right) \geq (\leq) \frac{f(x_1) + f(x_2)}{2}$$

for all x_1 and x_2 .

(b) For functions of several variables, a function, $f(x_1, x_2, \dots, x_n)$, is convex (concave) if

$$f(p_3) \geq (\leq) \frac{f(p_1) + f(p_2)}{2}$$

where p_1 and p_2 are any two points in the independent-variable space, and p_3 is the midpoint of the chord joining them.

If a function of a single variable is everywhere twice differentiable and its second derivative is always non-negative (nonpositive) the function is concave (convex). Although this test cannot be applied to piecewise-linear functions, the convexity or concavity of a piecewise-linear function or local regions of the function can be

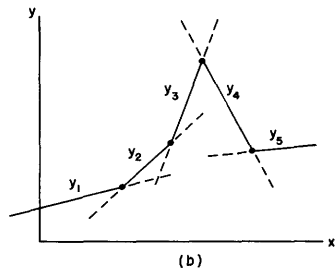
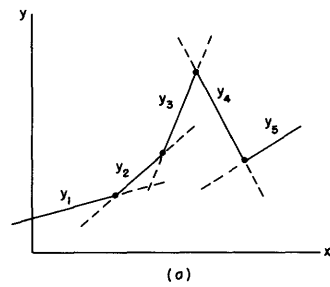


Fig. 8. Piecewise-linear function and modification.

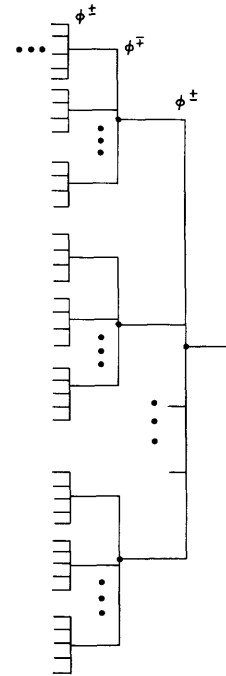


Fig. 9. Cascade-parallel transformations.

determined in an analogous manner by examination of its breakpoints. A breakpoint may be classed as convex (concave) if the slope of the function decreases (increases) in passing through the breakpoint from left to right. Breaklines on piecewise-linear surfaces can be classified in a similar manner: a ridge type of intersection like the peak of a sloping roof being convex, and a trough or valley type of intersection being concave. A piecewise-linear function, all of whose breakpoints or lines are convex (concave), will be called strictly convex (concave).

The examples of section 2.1 give an indication of the role that the classifications of the breakpoints play in determining the symbolic representation of the function. Each convex (concave) breakpoint must be associated with a ϕ^- (ϕ^+) transformation. Thus, the strictly convex function of Fig. 1 was represented by a single ϕ^- -transformed vector. The function of Fig. 2, possessing both types of breakpoints, required two cascaded transformations. It would be convenient if the classifications of the breakpoints were enough to prescribe the symbolic representation of the function. Unfortunately, this is usually not the case. The classifications of the breakpoints prescribe the kinds of transformations which are required but they do not indicate the order in which they must occur. Since it has already been pointed out that the symbolic representation is not unique, we should not be surprised that the order of the transformations is not specified. It will be shown in the following discussion that the relative magnitudes of the slopes and intercepts of the function are the factors that decide the order of the transformations.

As a point of departure for rendering arbitrary functions into symbolism, a list of several basic algebraic forms is useful.

1. The Simple Form. This is merely a single transformed vector, $a\phi^\pm$. This form is capable of representing any strictly convex or concave piecewise-linear function of any number of variables.

2. The Cascade Form. This is the simplest form which is capable of representing functions that have both types of breakpoints. Its general structure is

$$(a\phi^\pm, \beta)\phi^\mp, \gamma)\phi^\pm, \delta)\phi^\mp, \dots,)\phi^\pm$$

where each element of each vector is a linear function. This form is clearly nonreducible to any simpler form because of the alternation of the signs of the transformations.

EXAMPLE. The function of Fig. 8a is represented as

$$y = (y_1, y_2, y_3)\phi^+, y_4)\phi^-, y_5)\phi^+$$

where each y_n is of the form, $a_n + b_n x$. To illustrate the effects of the magnitudes of slopes and intercepts on this function, let us change the slope of the last segment, y_5 , as in Fig. 8b. In the original function, the extension of y_5 beyond its breakpoint lay below the rest of the function. Now, however, its extension intersects the rest of the function somewhere along y_2 . The above representation is, therefore, incorrect for the function of Fig. 8b, since it leads to a spurious intersection. In order to find an appropriate representation of the new function, we must go to a more general form.

3. The Cascade-Parallel Form. The structure of this form is best described by an iterative process. Starting from the outermost transformation and working inward, we see a single transformed vector, $a\phi^\pm$. The elements of a are also transformed vectors, $\beta_1\phi^\mp, \beta_2\phi^\mp, \dots, \beta_n\phi^\mp$; the elements of the β 's are in turn transformed vectors, and so forth. Again, the alternation of the signs of the transformations indicates that this form is nonreducible. Figure 9 is a graphical illustration of the general cascade-parallel form. Each vertical line in the diagram indicates a transformed vector. The type of transformation is indicated at the head of each column. The horizontal lines joined by each vertical line represent the elements of that particular vector.

Just as the cascade form includes the simple form as a special case, the cascade-parallel form includes all other forms, and thus it is the most general representation of a piecewise-linear function, subject to the qualification that a function which is the sum of several piecewise-linear functions is certainly not cascade-parallel. However, such a function can always be rearranged into a cascade-parallel form through the application of the theorems of section 2.2.

EXAMPLE. Consider the function, $y = (y_1, y_2)\phi^+ + (y_3, y_4)\phi^-$. The following procedure converts it to cascade-parallel form:

$$y = (y_1, y_2)\phi^+ + [(y_3, y_4)\phi^-] \phi^+ \quad (\text{Theorem 3})$$

$$= \left\{ (y_1, y_2) \oplus [(y_3, y_4)\phi^-] \right\} \phi^+ \quad (\text{Theorem 4})$$

$$= \left[y_1 + (y_3, y_4)\phi^-, y_2 + (y_3, y_4)\phi^- \right] \phi^+ \quad (\text{Definition 4})$$

$$= \left[(y_1)\phi^- + (y_3, y_4)\phi^-, (y_2)\phi^- + (y_3, y_4)\phi^- \right] \phi^+ \quad (\text{Theorem 3})$$

$$= \left\{ \left[(y_1) \oplus (y_3, y_4) \right] \phi^-, \left[(y_2) \oplus (y_3, y_4) \right] \phi^- \right\} \phi^+ \quad (\text{Theorem 4})$$

$$= \left[(y_1 + y_3, y_1 + y_4)\phi^-, (y_2 + y_3, y_2 + y_4)\phi^- \right] \phi^+ \quad (\text{Definition 4})$$

Although six steps were necessary to perform this conversion, such operations can be performed by inspection after some facility in handling the algebra is developed. This conversion operation occurs quite often in analysis, since an analysis problem often calls for addition of two or more piecewise-linear functions followed by some other operation such as inversion or implicit equation solution. The form of the various theorems makes them applicable to functions only in the cascade-parallel form. Therefore, consolidation to this form is often required before the analysis can proceed. In the applications to analysis in Section III, several of the intermediate steps in these operations will often be omitted.

For an additional example of an application of the cascade-parallel form, let us return to the function of Fig. 8b. A valid, symbolic representation can now be presented in the form,

$$y = \left[(y_1, y_2, y_3)\phi^+, (y_4, y_5)\phi^+ \right] \phi^-$$

No great difficulty should be experienced in finding a convenient cascade-parallel representation for any reasonable piecewise-linear function. In fact, it requires considerable ingenuity to construct a function for which it is difficult to find such a representation. In the unusual cases, it is always possible to utilize the methods that will be discussed in Section V, which yield representations as sums of simple piecewise-linear functions. By the methods of the first example of the cascade-parallel form, these sums can be converted to cascade-parallel form.

The definitions, theorems, and descriptions of the various forms of representation of piecewise-linear functions constitute the basis for the applications to analysis and synthesis set forth in the succeeding sections. Although these applications take many diverse forms, it should be kept in mind that they all stem either directly or indirectly

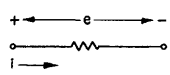
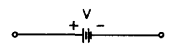
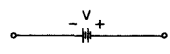
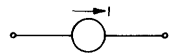
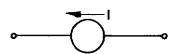
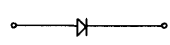
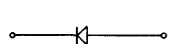
ELEMENT	IMPEDANCE	ADMITTANCE
	$e = Ri$	$i = \frac{1}{R}(e)$
	$e = V$	$i = \infty(e - V)$
	$e = V$	$i = \infty(e + V)$
	$e = \infty(i - I)$	$i = I$
	$e = \infty(i + I)$	$i = -I$
	$e = [\infty(i), 0] \phi^-$	$i = [\infty(e), 0] \phi^+$
	$e = [\infty(i), 0] \phi^+$	$i = [\infty(e), 0] \phi^-$

Fig. 10. Elements of a diode network.

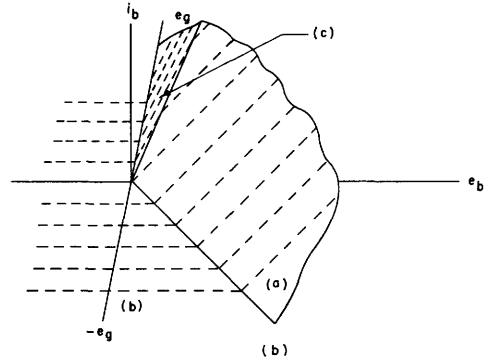
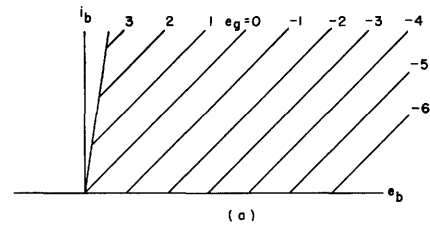


Fig. 11. Piecewise-linear triode characteristics.

from the algebra of inequalities, or more specifically, from the ϕ^+ and ϕ^- transformations.

III. APPLICATIONS TO ANALYSIS

3.1 SYMBOLIC DESCRIPTION OF NETWORK ELEMENTS

As a prerequisite to the application of the algebra of inequalities to network analysis, the network elements must be approximated piecewise-linearly and then represented algebraically. In this section some typical network elements will be considered. These examples are presented for two purposes: 1. many of the elements will be used in the applications to follow, and 2. the development of the algebraic expressions illustrates the general method of describing any network device.

a. Elements of a Diode Network

Of all the elements of a diode network, constant sources, resistances and diodes, only resistances have impedances that are odd functions of current, i.e., $z(i) = -z(-i)$. This fact makes it essential to establish a reference convention for defining their driving-point functions. This convention is indicated in connection with the first element in Fig. 10. Impedance and admittance functions are given for each element in both possible orientations to emphasize the nonsymmetric nature of the functions.

b. The Vacuum Tube

Undoubtedly, the most common nonlinear element appearing in electrical engineering problems is the vacuum tube. The crudest and most widely used approximation of the vacuum tube is the linear incremental model, derived from the second term of the Taylor series expansion of the tube characteristics about the quiescent operating point. Naturally, this model is valid for small-signal behavior only. A more refined approximation, which is usually acceptable for large signals, is the piecewise-linear representation of the tube characteristics. (In this case a more descriptive term would be "piecewise-planar," rather than piecewise-linear, since functions of two independent variables are being considered.) A procedure for handling vacuum tubes, or, more generally, multiterminal devices, is illustrated here with a triode.

Figure 11a is a plot of the plate characteristics of a triode that is approximated as piecewise-linear. Figure 11b shows these same characteristics in three dimensions. The surface describing the behavior of the tube consists of three intersecting planes:

$$\begin{aligned} \text{(a)} \quad i_b &= \frac{1}{r_p} (\mu e_g + e_b) && \text{(Normal operating region)} \\ \text{(b)} \quad i_b &= 0 && \text{(Cutoff)} \\ \text{(c)} \quad i_b &= \frac{1}{r_s} e_b && \text{(Saturation) } (r_s \ll r_p) \end{aligned}$$

It can be observed from Fig. 11b that planes (a) and (c) intersect in a convex breakline and that both these planes intersect the zero plane in concave breaklines. Thus, the plate current may be expressed in the cascade form as

$$i_b = \left[\left(\frac{\mu}{r_p} e_g + \frac{1}{r_p} e_b, \frac{1}{r_s} e_b \right) \phi^-, 0 \right] \phi^+$$

Viewing the triode from the grid, it is reasonable, for most purposes, to ignore the plate-to-grid transconductance, assuming the grid current to be independent of plate voltage, and to neglect grid current for negative grid-to-cathode voltages. The resultant expression for grid current is,

$$i_g = (0, \frac{1}{r_g} e_g) \phi^+$$

These two piecewise-linear functional relationships completely define the behavior of the tube as it affects its associated circuitry. In any problem involving a piecewise-linear triode, these equations can be combined with the equations that describe the external circuit and the combination can be solved simultaneously.

3.2 SERIES-PARALLEL NETWORKS

One important class of analysis problems is the evaluation of driving-point or

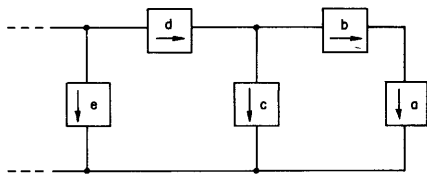


Fig. 12. General ladder network.

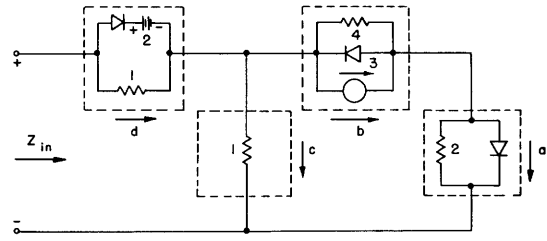


Fig. 13. Diode ladder network.

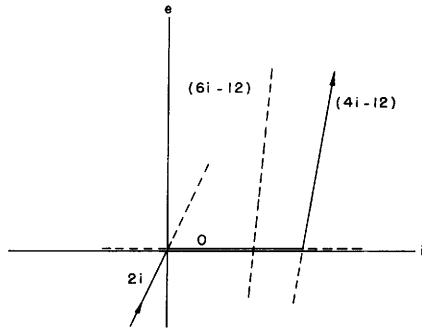


Fig. 14. Preliminary driving-point impedance of ladder.

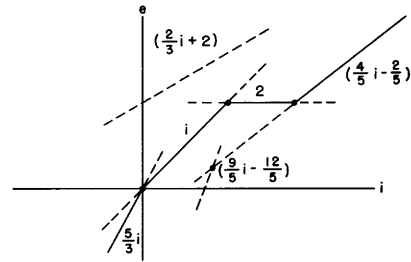


Fig. 15. Driving-point impedance of ladder.

transfer functions of networks containing two-terminal piecewise-linear elements. Such problems can be attacked in two different ways: 1. by combining the impedance and admittance functions of the individual elements; and 2. by writing loop or node equations for the network and solving them simultaneously. The first method is limited in application to series-parallel networks. To illustrate method 1, the driving-point impedance of a series-parallel diode network will be calculated in this section. Method 2 will be illustrated in subsequent sections.

As an example of a series-parallel network, consider the ladder network of Fig. 12. (For rigorous definitions of series-parallel graphs, see Appendix II.) For the moment, let us assume that the elements can have any type of nonlinear impedance functions so long as they are monotonically increasing (in order to ensure the existence of corresponding admittance functions, and the stability of the network). The driving-point impedance of this network can be found by utilizing techniques of impedance and admittance combination which are exactly analogous to those used for linear networks, that is, impedance functions of elements appearing in series are added, and admittance functions of elements appearing in parallel are added. The addition is accomplished through the application of the vector addition theorem and the procedure of section 2.3.

Impedances are converted to admittances and vice versa by using the inversion theorem. The impedance of a ladder network can be found by alternate additions and inversions, starting from the end opposite the driving point.

Thus, for the network of Fig. 12, z_a and z_b are combined to form

$$Z_1(i) = z_a(i) + z_b(i)$$

It should be observed that the impedances can be combined in this manner because the reference arrows associated with the two elements point in the same direction. If box (b) were inserted into the network with its connections reversed, then the expression would be

$$Z_1(i) = z_a(i) - z_b(-i)$$

This illustration again brings out the necessity of assigning reference directions when calculating impedances of nonlinear networks. If z_b were a linear passive element, it would not make any difference which expression was used, since

$$z_b(i) = -z_b(-i)$$

The next step is to combine $Z_1(i)$ with $z_c(i)$. We must, therefore, add the inverses of these two functions in order to obtain

$$Y_2(e) = Y_1(e) + y_c(e)$$

where Y_1 and y_c are the inverses of Z_1 and z_c . Then Y_2 is inverted and added to z_d , and the cycle is repeated on the next portion of the ladder.

To illustrate this method more explicitly, let us refer to Fig. 13, which is part of the ladder of Fig. 12, with diode networks inserted in the various boxes. With a little practice, the expression for the impedance or admittance of each box can be written by inspection. However, for the sake of clarity, almost every step in the derivation of the driving-point impedance will be written explicitly. Starting from the right,

$$i = y_a(e) = \frac{e}{2} + [\infty(e), 0] \phi^+ = \left(\frac{e}{2}\right)\phi^+ + [\infty(e), 0] \phi^+ \quad (\text{Theorem 3})$$

$$i = \left\{ \frac{e}{2} \oplus [\infty(e), 0] \right\} \phi^+ \quad (\text{Theorem 4})$$

$$i = \left[\infty(e), \frac{e}{2} \right] \phi^+ \quad (\text{Definition 4})$$

$$e = z_a(i) = (0, 2i)\phi^- \quad (\text{Theorem 6})$$

By using the same technique, or by inspection, we obtain

$$z_b(i) = (0, 4i - 12)\phi^+$$

$$z_c(i) = i$$

$$z_d(i) = (i, 2)\phi^-$$

Combining z_a and z_b yields

$$Z_1(i) = z_a(i) + z_b(i) = (0, 2i)\phi^- + (0, 4i-12)\phi^+ = \left[(0, 2i)\phi^- \right] \phi^+ + (0, 4i-12)\phi^+ \quad (\text{Theorem 3})$$

$$Z_1(i) = \left[(0, 2i)\phi^- \oplus (0, 4i-12) \right] \phi^+ \quad (\text{Theorem 4})$$

$$Z_1(i) = \left[(0, 2i)\phi^-, 4i-12+(0, 2i)\phi^- \right] \phi^+ = \left[(0, 2i)\phi^-, (4i-12, 6i-12)\phi^- \right] \phi^+ \quad (\text{Definition 4, Theorems 3, 4, Definition 4})$$

Part of this function is superfluous, as we can see by drawing a sketch of the expression (see Fig. 14). It can be simplified to

$$Z_1(i) = \left[(0, 2i)\phi^-, 4i-12 \right] \phi^+$$

Inverting Z_1 yields

$$Y_1(e) = \left\{ \left[\infty(e), \frac{e}{2} \right] \phi^+, \frac{e}{4} + 3 \right\} \phi^- = \alpha \phi^- \quad (\text{Theorem 6})$$

Combining Y_1 and y_c yields

$$Y_2(e) = e + \alpha \phi^- = (e \oplus \alpha) \phi^- \quad (\text{Theorems 3, 4})$$

$$Y_2(e) = \left\{ e + \left[\infty(e), \frac{e}{2} \right] \phi^+, \frac{5}{4}e + 3 \right\} \phi^- \quad (\text{Definition 4})$$

$$Y_2(e) = \left\{ \left[\infty(e), \frac{3}{2}e \right] \phi^+, \frac{5}{4}e + 3 \right\} \phi^- \quad (\text{Theorems 3, 4, Definition 4})$$

Inverting Y_2 yields

$$Z_2(i) = \left[(0, \frac{2}{3}i)\phi^-, \frac{4}{5}i - \frac{12}{5} \right] \phi^+ = \beta \phi^+ \quad (\text{Theorem 6})$$

Combining Z_2 and z_d yields

$$e_{in} = Z_{in}(i) = z_d(i) + Z_2(i) = (i, 2)\phi^- + \beta \phi^+ = \left[(i, 2)\phi^- \oplus \beta \right] \phi^+ \quad (\text{Theorems 3, 4})$$

$$e_{in} = \left[(i, 2)\phi^- + (0, \frac{2}{3}i)\phi^-, \frac{4}{5}i - \frac{12}{5} + (i, 2)\phi^- \right] \phi^+ \quad (\text{Definition 4})$$

$$e_{in} = \left[(i, 2, \frac{5}{3}i, \frac{2}{3}i + 2)\phi^-, (\frac{9}{5}i - \frac{12}{5}, \frac{4}{5}i - \frac{2}{5})\phi^- \right] \phi^+ \quad (\text{Theorems 3, 4, Definition 4})$$

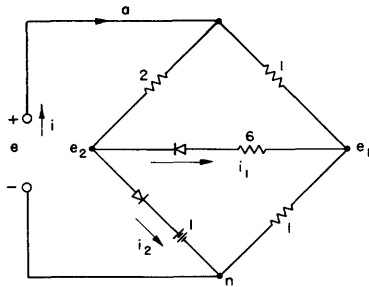


Fig. 16. Bridge diode network.

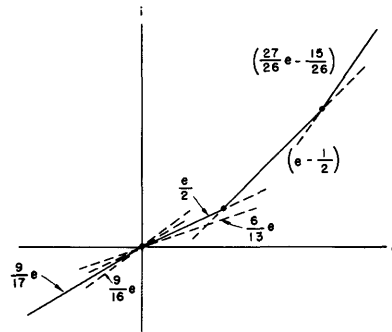


Fig. 17. Driving-point admittance of bridge.

This impedance function is plotted in Fig. 15. It can be seen from the figure that some of the terms in the above expression for $Z_{in}(i)$ are superfluous. An alternative form for expressing Z_{in} , without the superfluous terms, is

$$Z_{in}(i) = \left[i, \frac{5}{3} i, \left(2, \frac{4}{5} i - \frac{2}{5} \phi^+ \right) \phi^- \right]$$

It should be noted that familiarity with the algebra enables one to skip many of the steps listed in the above derivation, so that the technique is not as cumbersome in practice as it might at first appear from the illustrative example. The frequent use of vector addition in the derivation often introduced superfluous terms, since the vector sum always contains a number of terms equal to the product of the number of terms in each of the summands. These extra terms are not incorrect, but their presence needlessly complicates the algebra. Therefore, the superfluous terms were eliminated as quickly as they occurred by the artifice of sketching the function and then rewriting the functional relationship in a more efficient form. Generally, if superfluous terms are not removed, the method of impedance combination will result in an expression containing 2^n elements, where n is the number of diodes in the network. If the values of the network parameters are given only in literal form, we cannot tell from the expression which elements are redundant. For different combinations of parameter values, different elements become redundant. Thus, there is no redundancy in the original literal expression; the redundancies are the result of particular combinations of values of voltages, currents, and resistances.

A further note in reference to series-parallel networks is in order. The method just illustrated can be applied equally well to transfer ratios or impedances. For example, the transfer ratio for the network of Fig. 12 could be calculated by assuming the output voltage, e_o , across branch a , and then working back to the driving point, adding and inverting impedance and admittance functions, until the driving-point voltage is obtained as a function of the assumed e_o . The desired transfer ratio is then obtained by inverting this function. It happens that this inversion is always possible in a series-parallel

diode network that contains no control sources.

3.3 NON-SERIES PARALLEL NETWORKS

a. Bridge Diode Network

Consider the network of Fig. 16, a bridge containing two ideal diodes. The driving-point admittance looking into branch a is to be calculated. In this case, the method of impedance combination will not suffice, since the elements do not appear in series and parallel combinations. If this were a linear network, two alternative methods of solving the problem would be possible: reduce the network to a series-parallel form by a succession of Y- Δ transformations; or write loop or node equations for the network and solve them simultaneously. Lacking a convenient method of extending the Y- Δ transformation to nonlinear networks, we must use the second alternative. The general method is to write an independent set of equations that describe the system and solve these equations through direct substitution, utilizing the implicit equation theorem. Unfortunately, direct substitution appears to be the only method available for solving simultaneous piecewise-linear equations. Matrix methods and other linear techniques are not generally valid in this situation.

Referring to Fig. 16, we can write the three following node equations:

$$\frac{e_2 - e}{2} + i_1 + i_2 = 0 \quad (1)$$

$$i = \frac{e - e_2}{2} + e - e_1 \quad \text{or} \quad e_1 = \frac{3}{2}e - i - \frac{e_2}{2} \quad (2)$$

$$e - e_1 + (-e_1) + i_1 = 0 = e - 2e_1 + i_1 \quad (3)$$

The branch currents, i_1 and i_2 , can be expressed in terms of the admittance functions of their branches, as follows:

$$i_1 = y_1(e_1, e_2) = \left(\frac{e_2 - e_1}{6}, 0 \right) \phi^-$$

$$i_2 = y_2(e_2) = \left[\infty(e_2 - 1), 0 \right] \phi^+$$

Now, substituting the above expressions in Eq. 1, we obtain

$$\frac{e_2}{2} - \frac{e}{2} + \left(\frac{e_2 - e_1}{6}, 0 \right) \phi^- + \left[\infty(e_2 - 1), 0 \right] \phi^+ = 0$$

$$\left[\infty(e_2 - 1), \frac{e_2}{2} - \frac{e}{2} + \left(\frac{e_2 - e_1}{6}, 0 \right) \phi^- \right] \phi^+ = 0 \quad (\text{Theorems 3, 4, Definition 4})$$

$$\left[\infty(e_2 - 1), \left(\frac{2e_2}{3} - \frac{e_1}{6} - \frac{e}{2}, \frac{e_2}{2} - \frac{e}{2} \right) \phi^- \right] \phi^+ = 0 \quad (\text{Theorems 3, 4, Definition 4})$$

$$e_2 = \left[1, \left(\frac{e_1}{4} + \frac{3e}{4}, e \right) \phi^+ \right] \phi^- \quad (\text{Theorem 7}) \quad (4)$$

Substituting the expressions for the branch currents in Eq. 3, we obtain

$$2e_1 = e + i_1 = e + \left(\frac{e_2 - e_1}{6}, 0 \right) \phi^- \quad (5)$$

Substituting Eq. 2 in Eq. 4, we obtain

$$e_2 = \left[1, \left(\frac{9}{8} e - \frac{e_2}{8} - \frac{i}{4}, e \right) \phi^+ \right] \phi^-$$

$$\left[1 - e_2, \left(\frac{9}{8} e - \frac{9}{8} e_2 - \frac{i}{4}, e - e_2 \right) \phi^+ \right] \phi^- = 0 \quad (\text{Theorems 3, 4, Definition 4})$$

$$e_2 = \left[1, \left(e - \frac{2}{9} i, e \right) \phi^+ \right] \phi^- \equiv (1, \eta \phi^+) \phi^- \quad (\text{Theorem 7}) \quad (6)$$

Substituting Eq. 2 in Eq. 5,

$$3e - e_2 - 2i = e + \left[\frac{e_2}{6} - \frac{1}{6} \left(\frac{3e}{2} - \frac{e_2}{2} - i \right), 0 \right] \phi^-$$

$$\left(\frac{5}{4} e_2 - \frac{9}{4} e + \frac{13}{6} i, e_2 + 2i - 2e \right) \phi^- = 0 \quad (\text{Theorems 3, 4, Definition 4})$$

$$i = \left(\frac{27}{26} e - \frac{15}{26} e_2, e - \frac{1}{2} e_2 \right) \phi^+ \quad (\text{Theorem 7}) \quad (7)$$

Substituting Eq. 6 in Eq. 7, we obtain

$$i = \left\{ \frac{27}{26} e - \frac{15}{26} \left[1, \eta \phi^+ \right] \phi^-, e - \frac{1}{2} (1, \eta \phi^+) \phi^- \right\} \phi^+$$

$$i = \left\{ \frac{27}{26} e + \left[-\frac{15}{26}, \left(-\frac{15}{26} \eta \right) \phi^- \right] \phi^+, e + \left[-\frac{1}{2}, \left(-\frac{1}{2} \right) \phi^- \right] \phi^+ \right\} \phi^+ \quad (\text{Theorem 2})$$

$$i = \left\{ \left[\frac{27}{26} e - \frac{15}{26}, \left(\frac{27}{26} e - \frac{15}{26} e + \frac{5}{39} i, \frac{27}{26} e - \frac{15}{26} e \right) \phi^- \right] \phi^+, \right.$$

$$\left. \left[e - \frac{1}{2}, \left(e - \frac{1}{2} e + \frac{i}{9}, e - \frac{1}{2} e \right) \phi^- \right] \phi^+ \right\} \phi^+ \quad (\text{Theorems 3, 4, Definition 4}) \quad (8)$$

By theorem 5, the ϕ^+ 's appearing inside the braces can be omitted. Then, adding $-i$ to both sides yields

$$\left[\frac{27}{26} e - \frac{15}{26} - i, \left(\frac{6}{13} e - \frac{34}{39} i, \frac{6}{13} e - i \right) \phi^-, e - \frac{1}{2} - i, \left(\frac{1}{2} e - \frac{8}{9} i, \frac{1}{2} e - i \right) \phi^- \right] \phi^+ = 0$$

$$\quad (\text{Theorems 5, 3, 4, Definition 4})$$

$$i = \left[\frac{27}{26} e - \frac{15}{26}, \left(\frac{9}{17} e, \frac{6}{13} e \right) \phi^-, e - \frac{1}{2}, \left(\frac{9}{16} e, \frac{1}{2} e \right) \phi^- \right] \phi^+$$

$$\quad (\text{Theorem 7}) \quad (9)$$

This is an expression for the driving-point admittance of the bridge circuit. From its

sketch (Fig. 17), we can see that some terms are superfluous. A more concise equivalent expression is

$$i = \left[\frac{27}{26} e - \frac{15}{26}, e - \frac{1}{2}, \left(\frac{9e}{17}, \frac{e}{2} \right) \phi^- \right] \phi^+$$

b. Triode Feedback Amplifier

The previous section demonstrated that networks of two-terminal, piecewise-linear elements can be analyzed by solving sets of simultaneous equations, whether they are series-parallel or not. In this section it will be shown that these identical techniques are also applicable to networks that contain multiterminal elements, such as vacuum tubes, transistors, and so forth.

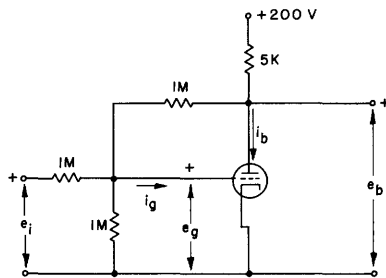


Fig. 18. Triode feedback amplifier.

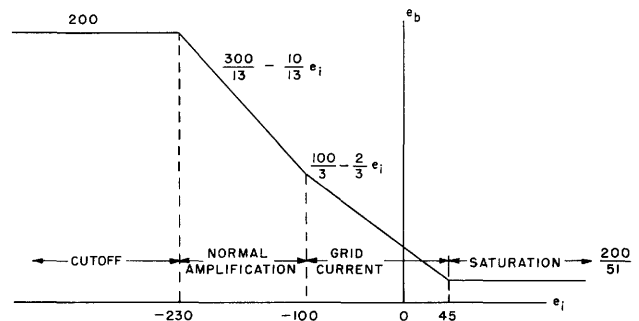


Fig. 19. Transfer function of amplifier.

The circuit of Fig. 18 serves as an illustrative example. In this simple, triode, negative-feedback amplifier, it will be assumed that the tube can be approximated piecewise-linearly by characteristics of the form of Fig. 11. The following numerical values will be assigned to the tube parameters:

$$\begin{aligned} r_p &= 5000 \text{ ohms} \\ \mu &= 20 \\ r_s &= 100 \text{ ohms} \\ r_g &= 500,000 \text{ ohms} \end{aligned}$$

(The last value is an unrealistic one; it was chosen to make the problem more interesting.) Substituting these values in the algebraic expressions for the triode characteristics given in section 3.1a, we obtain

$$i_b = \left[\left(\frac{20}{5000} e_g + \frac{e_b}{5000}, \frac{e_b}{100} \right) \phi^-, 0 \right] \phi^+ \quad (10)$$

$$i_g = \left(0, \frac{e_g}{500,000} \right) \phi^+ \quad (11)$$

The transfer function $e_b = f(e_i)$ will be determined as follows: First, assuming the grid circuit to be a negligible load on the plate circuit, we write node equations about node e_g and e_b :

$$\frac{200 - e_b}{5000} - i_b = 0 \quad (12)$$

$$\frac{e_i - e_g}{10^6} + \frac{e_b - e_g}{10^6} - i_g - \frac{e_g}{10^6} = 0 \quad (13)$$

Multiplying Eq. 12 by 5000 and substituting Eq. 10 in it, we obtain

$$200 - e_b - 5000 \left[\left(\frac{20}{5000} e_g + \frac{e_b}{5000}, \frac{e_b}{100} \right) \phi^-, 0 \right] \phi^+ = 0$$

Rearranging yields

$$200 - e_b + \left[(-20e_g - e_b, -50e_b) \phi^+, 0 \right] \phi^- = 0 \quad (\text{Theorem 2}) \quad (14)$$

$$\left[(200 - 20e_g - 2e_b, 200 - 51e_b) \phi^+, 200 - e_b \right] \phi^- = 0$$

(Theorems 3, 4, Definition 4)

Multiplying Eq. 13 by 10^6 and substituting Eq. 11 in it, we obtain

$$e_i + e_b - 3e_g - 10^6 \left(0, \frac{e_g}{500,000} \right) \phi^+ = 0$$

$$e_i + e_b - 3e_g + (0, -2e_g) \phi^- = 0 \quad (\text{Theorem 2})$$

$$(e_i + e_b - 3e_g, e_i + e_b - 5e_g) \phi^- = 0 \quad (\text{Theorems 3, 4, Definition 4})$$

Solving for e_g , we obtain

$$e_g = \left(\frac{e_i}{3} + \frac{e_b}{3}, \frac{e_i}{5} + \frac{e_b}{5} \right) \phi^- \quad (\text{Theorem 7}) \quad (15)$$

Substitution of Eq. 15 in Eq. 14 yields

$$\left\{ \left[200 - 20 \left(\frac{e_i}{3} + \frac{e_b}{3} \right), \frac{e_i}{5} + \frac{e_b}{5} \right] \phi^- - 2e_b, 200 - 51e_b \right\} \phi^+, 200 - e_b \Big\} \phi^- = 0$$

$$\left\{ \left[200 + \left(-\frac{20}{3} e_i - \frac{20}{3} e_b, -4e_i - 4e_b \right) \phi^+ - 2e_b, 200 - 51e_b \right] \phi^+, 200 - e_b \right\} \phi^- = 0$$

(Theorem 2)

$$\left\{ \left[\left(200 - \frac{20}{3} e_i - \frac{26}{3} e_b, 200 - 4e_i - 6e_b \right) \phi^+, 200 - 51e_b \right] \phi^+, 200 - e_b \right\} \phi^- = 0$$

(Theorems 3, 4, Definition 4)

$$\left[(200 - \frac{20}{3}e_i - \frac{26}{3}e_b, 200 - 4e_i - 6e_b, 200 - 51e_b)\phi^+, 200 - e_b \right] \phi^- = 0$$

(Theorem 5)

Solving for e_b , we obtain

$$e_b = \left[(\frac{300}{13} - \frac{10}{13}e_i, \frac{100}{3} - \frac{2}{3}e_i, \frac{200}{51})\phi^+, 200 \right] \phi^- \quad (\text{Theorem 7})$$

This is the desired expression for e_b in terms of e_i . It is plotted in Fig. 19.

The examples set forth in this section illustrate only a few of the representative problems in the analysis of piecewise-linear systems; they were chosen to illustrate some of the varied applications of the algebra. It can be applied equally well to many other types of problem, for example, to mechanical systems that contain stops and dead space, electrical systems that contain nonlinear elements, such as thyrite resistors which can be approximated as piecewise-linear, and so forth.

The examples were chosen to emphasize the systematic nature of the analysis procedure. They are not trivial examples; nor are they overly complicated. In many cases, a person familiar with piecewise-linear circuitry could arrive at the final answer by a shorter but less systematic route. The algebra was applied to several different types of problems with the intention of bringing out its universal applicability and flexibility. Its basic value lies in the fact that once one develops some confidence in, and facility with, the algebraic manipulations he can attack any piecewise-linear problem in a systematic rather than an intuitive manner.

IV. GENERAL PROPERTIES OF PIECEWISE-LINEAR NETWORKS

4.1 THE RESISTIVE DIODE NETWORK AS A BASIS FOR SYNTHESIS

In order to evolve a reasonable approach to nonlinear resistive network synthesis, attention must be restricted to certain more or less artificial "ideal" circuit elements. The choice of these elements is up to the circuit designer and is somewhat arbitrary. Factors influencing his choice are:

1. Availability of close approximations of the "ideal" characteristics.
2. Stability and reproducibility of these approximations.
3. Amenability of circuits containing the approximations to synthesis procedures.
4. Economic factors.
5. The size of the class of networks that can be synthesized by using these elements.

An appropriate candidate is the "ideal" diode. If its associated circuitry is properly designed, almost any inexpensive semiconductor diode will adequately reproduce the switching action required of an ideal diode. Here is a device that satisfies requirements 1 and 4 admirably. Similarly, factor 2 is satisfied, since stability and reproducibility of the characteristics are unimportant when the elements are used only as switches.

Diodes have already been used widely in digital computer logical networks, as well as in a variety of analog applications, not to mention miscellaneous uses, such as detectors, gating devices, rectifiers, and so forth — almost anywhere that some sort of nonlinearity is desired. However, no systematic synthesis procedures have been formulated for resistive diode networks. That diode networks are amenable to simple, efficient synthesis techniques, and therefore satisfy condition 3, will be shown in Section V.

As stated previously, the characteristics of a network containing ideal diodes and linear elements must be piecewise-linear. Thus, selection of the ideal diode as a building block immediately imposes a restriction to piecewise-linear synthesis rather than general nonlinear synthesis, just as restriction to lumped R's, L's, and C's confines us to rational function synthesis in the linear case. Clearly, this restriction is not particularly serious, since any reasonable function can be adequately approximated piecewise-linearly. Thus, condition 5 is satisfied.

In the following discussion of driving-point impedances, a network containing only positive resistors, constant current and voltage sources, and ideal diodes will be considered, and referred to as a diode network. It will be seen that many of the properties of such networks are similar to those of linear, lumped, passive networks and that many of the linear synthesis techniques can be carried over by analogy to the piecewise-linear case. This analogy should not be taken too seriously, however, since the fact that superposition has been discarded immediately eliminates the bulk of the linear techniques. The important analogies are to be found in the structure of the networks and their qualitative behavior.

In considering transfer function synthesis, the link to linear networks is more tenuous and will be virtually discarded. More flexible networks will be employed, which admit any linear resistive device, active or passive, but still restrict the nonlinear elements to ideal diodes.

4.2 THEOREMS CONCERNING THE BEHAVIOR OF DIODE NETWORKS

Before proceeding with synthesis techniques, it is well to consider some of the general properties of the diode network with a view toward utilizing these properties, or at least setting bounds upon the capabilities of the networks. The theorems that follow serve as a base from which to proceed. They are of importance in determining the structure of networks and they also point the way to new and unusual applications of the diode network. Proofs are presented in Appendix II.

THEOREM 1. A driving-point function containing $2^n - 1$ breakpoints requires at least n diodes for synthesis.

This establishes an extremely optimistic lower bound. This number is far from sufficient in the majority of cases, as will be shown in the next two theorems. Only in the cases wherein the successive types of breakpoints (convex or concave) follow special patterns, and the incremental resistance values fall within certain bounds, can this

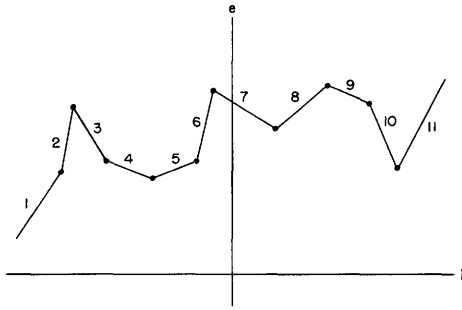


Fig. 20. Arbitrary driving-point impedance.

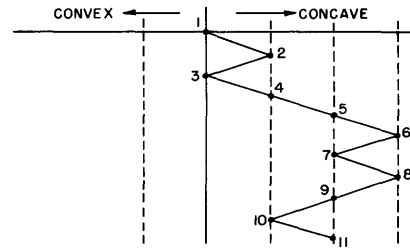


Fig. 21. Chart of breakpoints.

minimum be attained.

A more stringent lower bound can be determined by the following graphical procedure, which was suggested by Professor D. A. Huffman. Consider the impedance function of Fig. 20. Each concave breakpoint indicates that one diode in the network has switched from closed to open, and vice versa for the convex breakpoints. (It is assumed that only one diode switches at a time.) If the network contains n diodes, we can represent the "state" of the network, that is, the condition of each diode, by an n -digit binary number. Each digit is associated with a particular diode, being a zero when the diode is open and a one when it is closed. The order in which the concave and convex breakpoints of the impedance function of Fig. 20 occur will indicate something about the network that is needed to synthesize it. To keep track of these states it is convenient to make a chart, as in Fig. 21. The numbered points of Fig. 21 correspond to the numbered regions of Fig. 20. Starting from region 1, associated with the point at the origin of Fig. 21, we move one step to the right in passing through a concave breakpoint to the next region, and one step to the left if the breakpoint is convex. This procedure produces the chart shown as Fig. 21. Since each step to the right corresponds to the opening of a diode, and each step to the left, the closing of one, and no state can appear more than once, all of the points appearing in the same column of the chart correspond to different states with the same number of diodes open and closed, or binary numbers with the same number of ones and zeros. Also, since each column must have one more open diode than the one immediately to its left, a chart containing n columns must correspond to a network containing at least $n-1$ diodes. Thus, the impedance that is being discussed requires at least three diodes. However, from theorem 1, we also observe that it requires at least three. The chart also tells us that there must be enough diodes to provide the required number of states in each column. For example, if column 3 represents two diodes open, then there must be at least four different ways of having two diodes open in the network. This is clearly impossible in a network containing only three diodes. In general, the number of different n -digit binary numbers containing n zeros is the binomial coefficient, $\binom{n}{m}$. In this case $\binom{2}{3} = \binom{2}{3} = 3$. Now, we do not know how many diodes are necessary to synthesize the given function;

therefore n is unknown. However, a lower bound to n can be determined by picking a trial n and writing the binomial coefficients associated with it; then sliding this binomial distribution back and forth until it "fits" over the columns in the chart, that is, the sum of the states in each column is equal to or less than the binomial coefficient under that column.

This can be adequately demonstrated with the following example. The column sums are 2, 3, 4, 2. Now, $n = 3$ was previously shown to be too small, so we shall try $n = 4$. The binomial coefficients are 1, 4, 6, 4, 1. If we try fitting this distribution to the column sums, the best that can be done is

$$\begin{array}{cccc} \left. \begin{array}{c} 2 \\ 1 \end{array} \right\} & 3 & 4 & 2 \\ & 4 & 6 & 4 & 1 \end{array}$$

which does not fit over the first column. Going to $n = 5$, we obtain a successful fit.

$$\begin{array}{cccc} & 2 & 3 & 4 & 2 \\ 1 & 5 & 10 & 10 & 5 & 1 \end{array}$$

Thus, the lower bound has been raised from 3 to 5 diodes. A direct consequence of this procedure is

THEOREM 2. One diode per breakpoint is a necessary and sufficient number to synthesize any strictly concave or convex driving-point function.

THEOREM 3. One diode per breakpoint is a necessary and sufficient number to synthesize any driving-point function in a series-parallel development.

This theorem indicates that to approach the lower bounds described previously, bridge-type networks must be used. (Such networks have been developed in the form of cascaded lattices.) The proof of the necessary part of this theorem follows from the fact that in a series-parallel network (with no negative resistances or control sources), when the driving-point current or voltage is increased monotonically from $-\infty$ to $+\infty$, each diode can change state only once. The theorem is interesting because it illustrates the intimate connection between the topology of the network and its capabilities. Analogous connections also arise in the linear case. For example, it is impossible to produce transmission zeros in the right half-plane when a series-parallel network is used. The connection shows up again in switching circuits, in which a preliminary design of a combinatorial switching circuit is usually made as a series-parallel development. However, modifications are generally made to minimize relay contacts and these usually lead to non-series parallel networks. This relationship between the structural form of a network and its electrical behavior appears to be of fundamental importance.

The sufficiency of theorems 2 and 3 is proved in Section V, in which networks of this kind are constructed. All the synthesis procedures given there result in series-parallel developments that use one diode per breakpoint. This may, at first, seem rather extravagant, considering the lower bound mentioned in theorem 1. However, the

actual number of diodes required for a given synthesis problem depends so much on the relative magnitudes of the various incremental resistances, that it is impractical to determine sufficiency conditions for numbers of diodes less than the number of breakpoints of the function. Also, synthesis using non-series parallel networks usually requires solution of large numbers of simultaneous equations, making it somewhat cumbersome. Non-series parallel developments appear to be most useful in special cases, in which a large number of breakpoints are required, and a dramatic saving of diodes can be made (19).

THEOREM 4. Given a resistive diode network the behavior of which at some arbitrary terminal pair is described by $e = z(i)$ or equivalently, $i = y(e)$. If all voltage sources and all resistances are multiplied by the same positive constant, k , then the new impedance and admittance functions are,

$$e = k[z(i)] \quad \text{and} \quad i = y\left(\frac{e}{k}\right)$$

Theorem 5 is the dual of theorem 4, and the proof of both of these theorems follows directly from theorem 2 of Section II (see Appendix II). A useful corollary of these two theorems follows.

COROLLARY. If all voltage sources and current sources in a given diode network are multiplied by the same positive constant, k , the resultant impedance and admittance functions are

$$e = k \left[z \left(\frac{i}{k} \right) \right]$$

$$i = k \left[y \left(\frac{e}{k} \right) \right]$$

The corollary is just the result of applying theorems 4 and 5 successively. Since, in this process, the resistances are all multiplied by a constant, and then the conductances (their reciprocal) are again multiplied by the same constant, the result is a new network with the sources modified but the resistances unchanged. Since the control of sources is a common operation in linear networks, while control of resistances is more difficult, one might expect some applications of the corollary in terms of time-varying sources. An example follows.

EXAMPLE. Variable admittance function. Consider a resistive diode network whose admittance function is a piecewise-linear approximation of some analytic nonlinear function over a given range, for example, the function, $i = y(e) \approx f(e) = e^3$.

Assuming that all bias voltages are obtained from a common supply, let this supply voltage be proportional to another independently variable voltage, u . Then, from the corollary just stated,

$$i = ku \left[y \left(\frac{e}{ku} \right) \right] \approx ku \left(\frac{e}{ku} \right)^3 = \frac{e^3}{k^2 u^2} \quad u \geq 0$$

Figure 22 shows this function, the approximation being valid in the range $|e| \leq 50$,

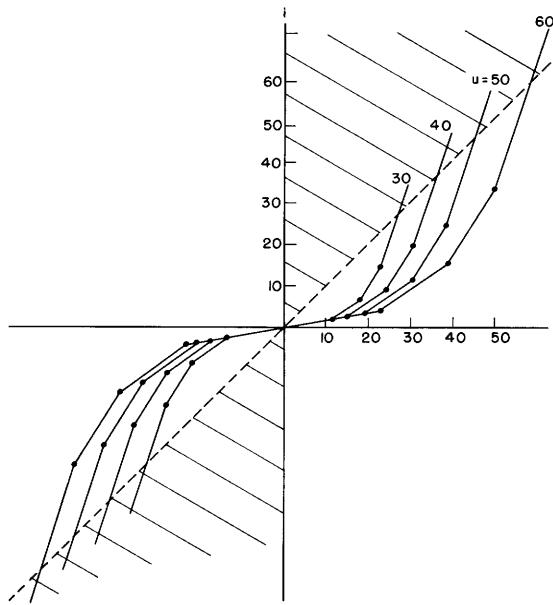


Fig. 22. The function, $i \approx \frac{e^3}{u}$.

when $u = 50$. The family of admittance characteristics demonstrates the effect of variation of u , but at the same time sharply points up the disadvantages of such a scheme. It will be observed that as $u \rightarrow 0$, the region of valid approximation also goes to zero because of the crowding of the breakpoints toward the origin. However, the accuracy of the approximation in this region is commensurately increased. The shaded area in the figure indicates the region over which the approximation is invalid. Despite this disadvantage, such an arrangement affords a simple and economical method of obtaining a class of admittance functions of two variables over a limited dynamic range. Insertion of such a device into a suitable high-gain feedback network will convert the admittance to a transfer function of two voltage variables.

4.3 DUALITY IN NONLINEAR RESISTIVE NETWORKS

The duality principle being considered here applies only to networks representable by directed line graphs, that is, interconnections of two-terminal elements. Note that this eliminates consideration of mutual inductance unless it is possible to represent the coupled coils by an equivalent Tee. Wherever the isolating properties of the mutual coupling are important, this is clearly impossible.

Ordinarily, the dual of a planar linear network can be obtained quite easily without carefully considering polarities and directions of elements. This is so because linear elements (other than sources) have voltage-current characteristics which are odd functions. In other words, their terminals need not be marked to distinguish one from the other, since their behavior is identical no matter which terminal is assigned the positive reference direction. Thus, a network consisting of two-terminal linear

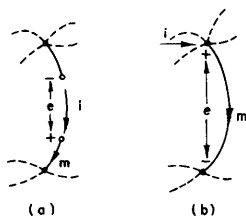


Fig. 23. Reference conventions.

elements other than sources can be represented by a line graph with nondirected line segments. Sources require arrows to indicate their direction because they have voltage-current characteristics which are not odd functions; not because they are active elements. An ideal negative resistance is an active element but it has no preferred reference direction. In calculating the duals of networks which contain sources, their directional nature is usually taken into account by assuming a simple reference convention and following it consistently throughout the calculations. The common conventions still lead to difficulties when nonlinear networks are considered. The addition to a network of nonlinear elements that must be represented by directed line segments necessitates a more careful consideration of polarity and direction. Thus, the first step in applying duality to nonlinear networks is to obtain a definition that is clear in this regard. A suitable definition (which does not conflict with the usual definitions for linear networks) is developed in the following discussion.

Consider a network, N , consisting of interconnected two-terminal resistive elements. (By a resistive element is meant one whose complete behavior can be described by a single-valued voltage-current relationship, independent of time.) Let us examine it by making "pliers" entries into each branch, and "soldering iron" entries across each branch. The voltage-current relationship looking into a pliers entry in branch m will be known as the short-circuit driving-point admittance for branch m , and will be denoted by $i = y_m(e)$. Similarly, the voltage-current relationship looking into a soldering iron entry across branch m will be known as the open-circuit driving-point impedance for branch m , and will be denoted by $e = z_m(i)$.

Figure 23a establishes reference polarities relative to the direction of branch m , for determining y_m , and Fig. 23b establishes the convention for z_m . It should be observed that this set of conventions was an arbitrary choice; other sets would have been perfectly acceptable. Now that the conventions have been established, duality can be defined.

DEFINITION. Given two networks, N and N' ; they are mutually dual if and only if:

1. To each branch, m , in N , there corresponds one and only one branch, m' , in N' ; and to each branch, n' , in N' , there corresponds one and only one branch, n , in N . In other words, the branches of the two networks can be put in one-to-one correspondence.

2. $y_m = z_{m'}$ for all m .

3. $z_m = y_{m'}$ for all m .

It is clear that this definition implies topological duality as a prerequisite for electrical duality, for example, all branches m, n, p, \dots that appear in series around a

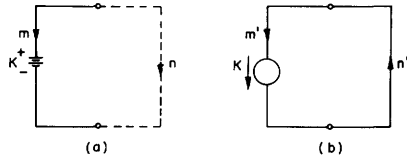


Fig. 24. Voltage source and its dual.

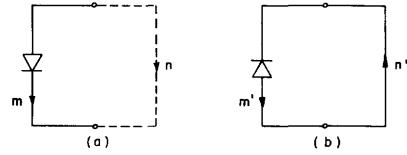


Fig. 25. Diode and its dual.

loop in N must correspond to branches m', n', p', \dots across a single node-pair in N' , since $y_m = y_n = y_p = \dots$; therefore, $z_{m'} = z_{n'} = z_{p'} = \dots$.

The dual of a planar nonlinear network can be constructed in the same manner as that of a linear network; replacing loops by nodes, and directed branches by their dual branches, always being careful to observe reference conventions.

As a first step in constructing duals of diode networks the dual of a voltage source, a current source, and an ideal diode must be determined.

a. Voltage and Current Source Duality

Consider the open-circuited voltage source (branch m) as being shunted by a branch, n , of infinite resistance (Fig. 24a). Then,

$$e = z_m(i) = K \qquad e = z_n(i) = K$$

$$i = y_m(e) = 0 \qquad i = y_n(e) = 0$$

A network of two branches, m' and n' , must be constructed in such a manner that

$$i = y_{m'}(e) = z_{m'}(e) = K \qquad i = y_{n'}(e) = z_{n'}(e) = K$$

$$e = z_{m'}(i) = y_{m'}(i) = 0 \qquad e = z_{n'}(i) = y_{n'}(i) = 0$$

The circuit of Fig. 24b fits this description and is, therefore, the dual of the circuit of Fig. 24a. Note that the current source generates a current which flows in the direction of the branch reference arrow (downward), while the voltage source generates a potential rise in a direction opposite to that of the reference arrow (upward). This is an important consequence of the chosen reference convention.

b. Dual of a Diode

Consider the diode of Fig. 25a, shunted by a branch of infinite resistance. Then,

$$e = z_m(i) = [0, \infty(i)]\phi^- \qquad e = z_n(i) = [0, \infty(i)]\phi^-$$

$$i = y_m(e) = 0 \qquad i = y_n(e) = 0$$

Therefore, we must construct a two-branch network in such a manner that

$$i = y_{m'}(e) = [0, \infty(e)] \phi^-$$

$$e = z_{m'}(i) = 0$$

$$i = y_{n'}(e) = [0, \infty(e)] \phi^-$$

$$e = z_{n'}(i) = 0$$

The circuit of Fig. 25b fits this description; hence it is the dual of that of Fig. 25a. Note that the diode in the dual circuit is pointing in a direction opposite to the branch reference arrow, while the diode in the original circuit is pointing in the same direction as its reference arrow. This is a second important consequence of the chosen convention.

Thus, to summarize the above examples, **THE DUAL OF A VOLTAGE SOURCE IS A CURRENT SOURCE GENERATING A CURRENT IN THE DIRECTION OF THE POTENTIAL DROP OF THE VOLTAGE SOURCE**, and **THE DUAL OF A DIODE IS ANOTHER DIODE POINTING IN THE OPPOSITE DIRECTION**. It should be noted that these rules are a consequence of the chosen convention, and would be different had a different convention been chosen. Using these rules, we may now proceed to the determination of the dual of a more general diode network.

c. Dual of a Diode Network

The dual of the bridge network of Fig. 26a will be determined. The procedure is:

1. Assign labels and reference directions to each branch, thus reducing it to a network of directed line segments.

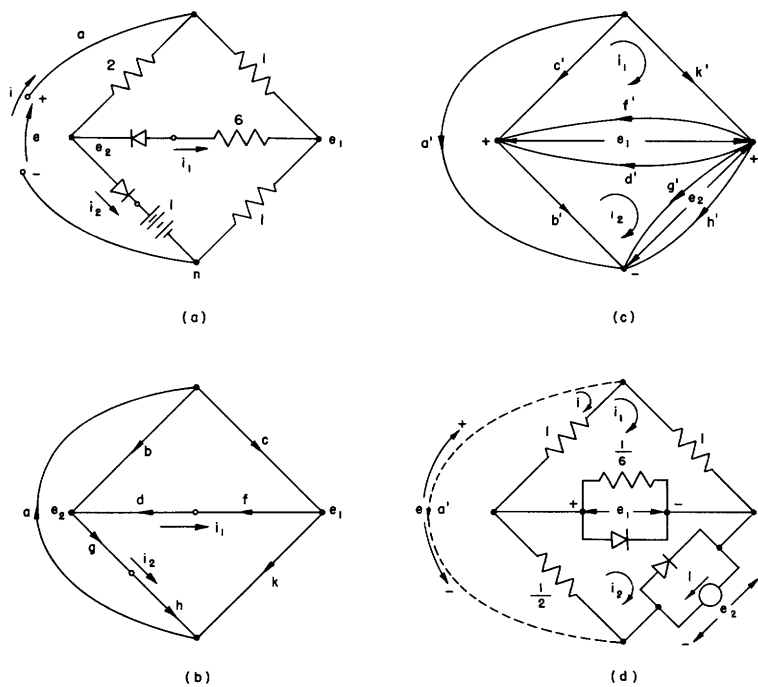


Fig. 26. Construction of the dual of a bridge diode network.

2. Draw the topological dual of this line graph, using any arbitrary reference convention, such as all branches converging on a node in the original graph will appear clockwise around a corresponding loop in the dual graph.

3. Insert, in each branch of the dual, a branch that is the dual of the original branch.

Figure 26 illustrates the procedure. Figure 26a is the original network; Fig. 26b is the network skeletonized to a line graph; Fig. 26c is the topological dual of that graph; and Fig. 26d is the dual network. As a check, the loop equations for the dual network are:

$$\frac{i_2 - i}{2} + e_1 + e_2 = 0$$

$$e = \frac{i - i_2}{2} + i - i_1$$

$$i - i_1 + (-i_1) + e_1 = 0$$

where

$$e_1 = z_1(i_1, i_2) = \left(\frac{i_2 - i_1}{6}, 0 \right) \phi^-$$

$$e_2 = z_2(i_2) = \left[\infty(i_2 - 1), 0 \right] \phi^+$$

If e and i are interchanged in these equations, we find that they are identical to the node equations for the network of Fig. 26a. (The original network is the same one used for illustration in section 3.3a, and the original node equations appear in that section.) Therefore, the open-circuit driving-point impedance of this network will be given analytically by the expression of section 3.3a for the driving-point admittance of the original network, and graphically by the plot of Fig. 19 with e and i interchanged in each case.

V. APPLICATIONS TO SYNTHESIS

5.1 INTRODUCTION

As it was stated in Section IV, the synthesis problem will be restricted to piecewise-linear resistive network synthesis. In the case of driving-point functions, only resistive diode networks in the strict sense will be considered, but for synthesizing transfer functions any zero-memory linear elements, such as active summing devices, will be admitted in conjunction with ideal diodes.

Section 5.2, covering driving-point synthesis, follows closely the lines of modern linear network synthesis, and is to a certain extent a classification of that which has gone before, since many of the network configurations mentioned have been applied in the past. These are included mainly for the sake of completeness.

Section 5.3 will cover transfer synthesis and the approximation problem. As we shall demonstrate, these two problems are intimately related. The general philosophy of transfer function synthesis expressed in this work is largely influenced by the art of analog computation. It was the investigation of a method of constructing an analog multiplier with diode networks that motivated this exploration of piecewise-linear network theory. The applications to analysis actually constituted a by-product of the synthesis techniques. Because of this influence, rather free use is made of active linear elements and little emphasis is placed upon the linear operations. Also, since generators of arbitrary voltage transfer functions of a single input variable have been in fairly common use, most of the emphasis here is placed upon generators of functions of more than one input variable. Although there is a great demand in the analog computing field, as well as in other fields, for such multivariable function generators, these components are very rare, and they have so far suffered from one or more of the following deficiencies:

1. Small bandwidth;
2. Costly and cumbersome equipment;
3. Setting up the function requires a large investment in time and money;
4. Equipment restricted to certain limited types of functions;
5. Large static or dynamic inaccuracies.

Transfer function synthesis can be resolved into two parts: (a) arbitrary function synthesis (general-purpose function generation); and (b) particular function synthesis (special-purpose function generation). While it is desirable that a general-purpose machine have none of the aforementioned deficiencies, a special-purpose machine might be perfectly satisfactory even though it possessed deficiencies 3 and 4. Thus, it is clear that the synthesis techniques should be directed specifically toward these two kinds of components.

5.2 DRIVING-POINT FUNCTION SYNTHESIS

a. Strictly Convex or Concave Functions

In Section IV it was indicated that one diode per breakpoint is a necessary and sufficient number of diodes to synthesize any strictly convex or concave driving-point function. The sufficiency of this number is shown here by the construction of several canonical, or minimum, forms for the realization of such functions. Only monotonically increasing functions will be considered. This does not restrict the generality of the synthesis procedures, however, since any function containing regions of negative resistance can be constructed through the series or parallel combination of a monotonically increasing impedance function with an ideal negative resistance, whose value is equal to or greater than the value of the greatest negative resistance segment that appears in the desired function.

A typical concave driving-point admittance function is shown in Fig. 27a. Figure 28a illustrates a canonical form for realizing any concave driving-point admittance. For

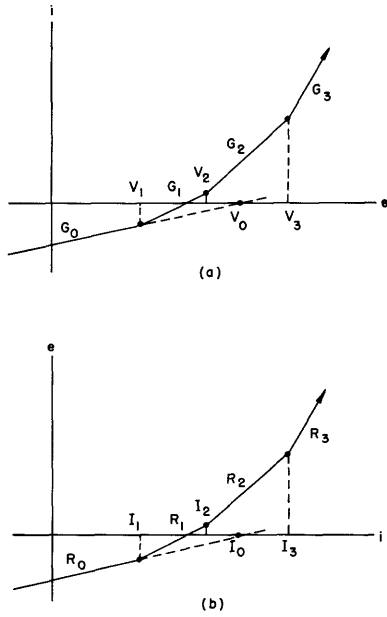


Fig. 27. Concave driving-point functions.

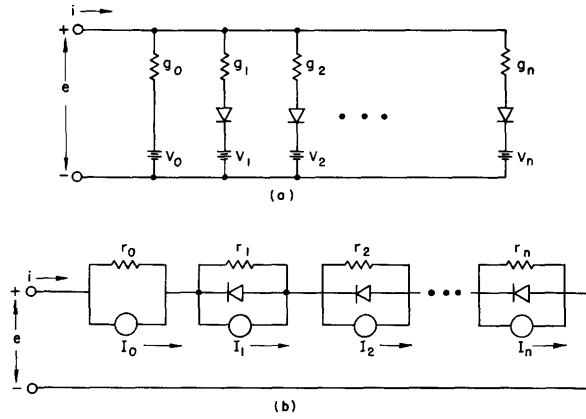


Fig. 28. Foster canonical forms.

monotonically increasing functions, a concave driving-point admittance is equivalent to a convex driving point impedance. (This statement is only true for increasing functions.) Inspection of the network indicates that for all applied voltages less than some sufficiently large negative value, all of the diodes will be open and the incremental conductance of the network will be just g_0 . If the batteries are arranged in order of increasing voltage (with the exception of V_0) as the applied voltage is increased beyond V_1 , the incremental conductance will increase by an amount, g_1 , producing a concave breakpoint in the driving-point admittance. Similarly, other concave breakpoints will occur as the applied voltage reaches each successively higher battery voltage. Clearly, there will be one breakpoint for each diode and the numbered breakpoint voltages of Fig. 27a will correspond to the numbered battery voltages in Fig. 28a. The battery voltage, V_0 , corresponds to the V_0 intercept shown in Fig. 27a. As indicated in the figure, the first segment may have to be extrapolated beyond the first breakpoint to determine its intercept, and therefore V_0 may be larger than some of the breakpoint voltages. The various slopes are given by,

$$\begin{aligned}
 g_0 &= G_0 \\
 g_1 &= G_1 - G_0 \\
 g_2 &= G_2 - G_1 \\
 &\vdots \\
 &\vdots \\
 &\vdots \\
 g_n &= G_n - G_{n-1}
 \end{aligned}$$

The g 's refer to the network elements of Fig. 28a. The G 's refer to the slopes of the function of Fig. 27a. Or generally, $g_p = \Delta G_p$, where ΔG_p is the first backward difference of the G 's.

A typical concave driving-point impedance is shown in Fig. 27b. It is merely the curve of Fig. 27a with its voltage and current axes interchanged, and therefore, according to the definitions of Section IV, it is the dual of Fig. 27a. Thus, to realize this function, we have only to construct the dual of the network of Fig. 28a. This is shown in Fig. 28b. The duality principle saves a great deal of effort here, since everything that was said about the network of Fig. 28a can be carried over to its dual, replacing currents by voltages, conductances by resistances, and so forth. Thus, the network of Fig. 28b is constructed so that its current sources, I_1, I_2, \dots, I_n , correspond to the breakpoints of the desired driving-point impedance, and its resistances, r_1, r_2, \dots, r_n , correspond to the first backward differences of the incremental resistances of the desired driving-point impedance.

The structural form of the networks of Fig. 28 bears a marked similarity to the so-called Foster canonical forms that are used in linear network synthesis for the construction of arbitrary RL, LC, or RC driving-point functions. They will be referred to here as Foster forms. These forms are sufficient to synthesize any strictly concave or convex driving-point function that uses a minimum number of diodes. Note also, that reversal of the reference directions at the terminals changes the driving-point function from concave to convex or vice versa. Thus, any one of these forms can be used for both types of function.

To complete the analogy to linear networks, consider the network of Fig. 29a, and its dual, shown in Fig. 29b. Assume that all voltage sources in Fig. 29a, except V_0 , are arranged in order of increasing voltage. Therefore, for a sufficiently negative value of the input voltage, all diodes are open, and the incremental conductance of the network is just g_0 . As the applied voltage is increased beyond V_1 , the incremental conductance will increase by an amount, g_1 , producing a concave breakpoint in the driving-point admittance, and so forth, for higher voltages. Hence, this network

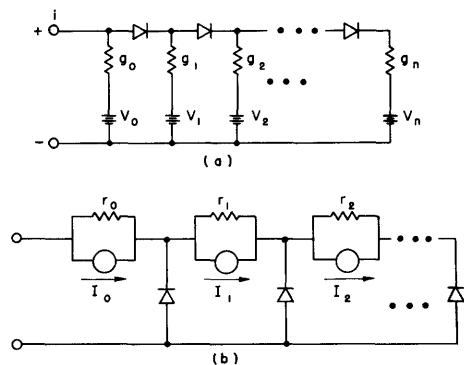


Fig. 29. Cauer canonical forms.

behaves exactly like the network of Fig. 28a, and the expressions which relate its element values to the slopes and breakpoints of the associated admittance function are identical to those given for the first Foster form. This ladder structure will be known as the first Cauer form, because of its similarity to the Cauer forms of linear network theory. The form shown in Fig. 29a is applicable to concave admittances, or with its terminals reversed, to concave impedances. Similarly, its dual, the second

Cauer form, is applicable to concave impedances, or with its terminals reversed, to concave admittances. The expressions for the network parameters of the dual are, of course, the duals of the expressions already given. Note that all of the circuits mentioned can be synthesized with the use of either voltage or current sources by making suitable source transformations on the given network forms.

The basic Foster form and its three variations, and the basic Cauer form with its three variations constitute a total of eight canonical forms available for synthesis of strictly convex or concave functions. The particular one to be chosen depends upon practical considerations: e.g., the circuit of Fig. 28a would be convenient if the available sources had a common ground, or if the diodes had a common cathode or plate.

b. Arbitrary Functions

With the aid of the previously derived canonical forms, a method will now be presented by which any nondecreasing piecewise-linear driving-point function can be synthesized with the use of one diode per breakpoint. The method utilizes a ladder development and produces a minimum form when only series-parallel configurations are considered.

Consider the general impedance function pictured in Fig. 30a. The synthesis procedure is initiated by partitioning the function into strictly convex or concave sections. The partitions are indicated in the figure by breaks in the function. The points of partition are located by proceeding along the function from left to right and observing the types of breakpoints encountered. The function is partitioned between each pair of breakpoints that differ in direction, i.e., between a convex breakpoint followed by a concave breakpoint, or vice versa. Some partitions may contain only one breakpoint, as in the fourth partition of Fig. 30a.

The next step in the procedure is to form the auxiliary impedance functions, $Z_1(i)$, $Z_2(i)$, ..., $Z_n(i)$, in which $Z_p(i)$ equals the original function, $z(i)$, over the p^{th} partition and is linear elsewhere, its linear portions being extrapolations of the first and last linear segments of the p^{th} partition. The first four Z_p 's are shown in Fig. 30b-e. Note that all the odd-numbered Z 's are strictly concave and the even ones strictly convex. (Of course, if the function had started with a convex partition, the odd-numbered Z 's would have been convex and the even ones concave.) The inverse of an impedance function, $Z_p(i)$, will be denoted by $Y_p(e)$.

Next, the Z -functions must be modified as follows:

$$z_1(i) = Z_1(i)$$

$$y_2(e) = Y_2(e) - (I_2 + G_2 e)$$

$$z_3(i) = Z_3(i) - (V_3 + R_3 i)$$

$$y_4(e) = Y_4(e) - (I_4 + G_4 e)$$

$$z_5(i) = Z_5(i) - (V_5 + R_5 i)$$

...

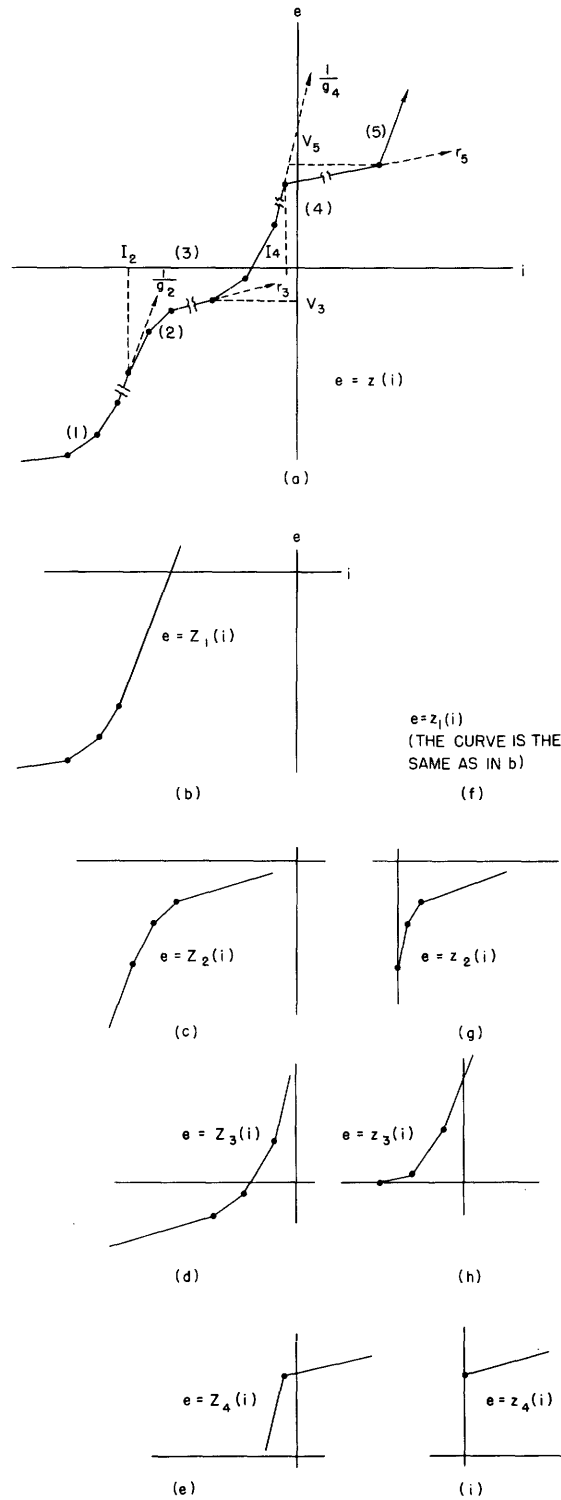


Fig. 30. An arbitrary function and its decomposition.

The I's, V's, G's, and R's are slopes and breakpoint coordinates, as indicated in Fig. 30a. Note that each concave $Z_p(i)$ (except the first) has a linear term subtracted from it, whose slope is equal to the slope of the first segment of Z_p . Thus, the resultant $z_p(i)$ is still nondecreasing and concave, and therefore realizable by the methods of section 5.2a. Each convex Z_p is first inverted to form a concave $Y_p(e)$ from which is subtracted a linear term. Again the slope of the linear term equals the slope of the first segment of Y_p , and therefore the resultant $y_p(e)$ is realizable by the methods of section 5.2a. The first four z_p 's for the function of Fig. 30a are plotted in Fig. 30f-i for comparison with the unmodified Z_p 's. It will be observed that the modification performs two functions; first, it reduces the slopes of each segment of the concave impedance or admittance functions until the first segment of each has zero slope; second, it displaces each function so that the impedance functions (except the first) each have a region of zero voltage and the admittance functions each have a region of zero current. These "zero" regions make possible the combination of the z_p 's to form the over-all function, $z(i)$.

The complete function is formed in the following manner. First, each of the z_p 's is synthesized with the use of any one of the Foster or Cauer forms. Then,

1. $y_1 + y_2 = y_{12}$ is formed (Parallel addition)
2. $z_3 + z_{12} = z_{123}$ is formed (Series addition)
3. $y_4 + y_{123} = y_{1234}$ is formed (Parallel addition)
4. . . .
- .
- .
- .
- n. $z_n + z_{12\dots n-1} = z_{12\dots n} = z(i)$ is formed

This process can be clarified by description and reference to the ladder development of Fig. 31. To build up the total function from the left, y_1 and y_2 are added, that is, the impedances, z_1 and z_2 , are added "current-wise," so that the networks realizing them appear in parallel in the over-all network. This is the beginning of the ladder network of Fig. 31. The first series branch is z_1 , and z_2 is the first shunt branch. Examination of the method by which they were constructed indicates that the expression, $y_1 + y_2 = y_{12}$, is identical to the desired function, $z(i)$, over the first two partitions

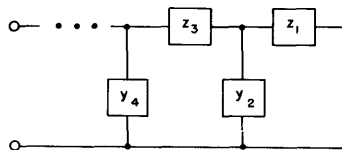


Fig. 31. Ladder development for an arbitrary driving-point function.

and is a linear extrapolation of the function beyond the first two partitions.

The next step involves adding the network realizing y_{12} to the network that realizes z_3 , "voltage-wise" (in series), forming the function, $z_3 + z_{12} = z_{123}$. Thus, z_3 appears as the next series branch in the ladder. Now, the resultant network, z_{123} , realizes the desired function over the first three partitions. Next, a parallel addition is performed and the process is continued until the complete function is realized. Note that the series branches of the ladder are each strictly concave impedances, while the shunt branches are strictly convex. If the given function had started with a convex partition, the ladder would have begun with a shunt rather than a series branch.

This completes the discussion of the driving-point function problem. Regarding minimization of diodes, at first, one might think that a non-series parallel development of the function of Fig. 30 would result in a saving of diodes. This is not the case, however. Since there is no quantitative information available about the actual values of the slopes and breakpoints of the function (other than the convexity or concavity of the breakpoints), little progress can be made toward reducing the number of diodes. Such a reduction always requires a certain amount of information regarding the values of these quantities.

5.3 TRANSFER FUNCTION SYNTHESIS

a. General Purpose Function Generation

A general purpose analog function generator can be defined roughly as a network (or system) with some adjustable parameters that can be controlled or "programmed" to produce one of an infinite variety of functions of its input voltage variables as an output voltage. Thus, it is a flexible piece of equipment which has an infinite repertoire of possible output functions that may be called upon by convenient adjustments. A probe riding on a three-dimensional cam is an example of such a device. In this case, the adjustable parameter is the shape of the cam, a new cam being inserted in the machine for each different function. A special-purpose device, on the other hand, is inflexible. It is designed for the purpose of producing only one type of functional output, and, in general, has no external adjustments for modifying this function. An example of such a machine is a sine potentiometer. Its output is fixed by the construction of the potentiometer card and winding.

Tabulation, Tessellation, and Interpolation

In considering general-purpose generation of functions of n variables, $y = f(x_1, x_2, \dots, x_n)$, some restrictions must be made immediately upon the class of functions to be produced, in order to bring the problem within the range of possible solution. We have already restricted the class to piecewise-linear (and for practical reasons, continuous) functions. However, this still leaves too much freedom. Any

arbitrary function will, in general, be presented in terms of a finite amount of tabulated data, and, in general, this will be a regular tabulation, i. e., the points of tabulation will be regularly spaced throughout the n-dimensional independent variable space. The most reasonable type of grid of tabulation would be the vertices of a set of n-dimensional hypercubes. (This is not the only possibility, however.) It is, therefore, reasonable to propose a machine that will produce a piecewise-linear and a continuous function, taking on a particular arbitrary value at each point of a hypercubical grid of tabulation. This machine would have a set of adjustment knobs for the operator, one per tabulated point, for the purpose of programming any particular function. Since nothing has been said, so far, about the behavior of the function at nontabulated points, it is necessary to define some sort of interpolation scheme to prescribe the function at these points. Consideration of this interpolation scheme is the crucial step in solving the synthesis problem. That "linear" interpolation cannot, in general, be used, will be made clear by observing the somewhat surprising fact that linear interpolation, when generalized to functions of several variables, is a nonlinear operation. Let us examine this operation for functions of 1, 2, and n variables.

A general linear interpolation formula can be defined as a function of n variables that is linear in each variable, and takes on an arbitrarily assigned value at each of the 2^n vertices of an n-cube in the independent variable space. (It should be noted that the space referred to here is the independent variable space. If the dependent variable were included, the space would be of dimension $n + 1$. For example, a function of two variables is often represented as a surface in three-space. However, its independent variable space is only two-dimensional.)

Applying this definition to a function of one variable, $y = f(x)$, tabulated at all integral multiples of an interval, Δ , we have

$$y(x) = y_n + \frac{(y_{n+1} - y_n)(x - n\Delta)}{\Delta} \quad n\Delta \leq x \leq (n+1)\Delta$$

where y_p is the tabulated value of $f(x)$ at $x = p\Delta$. This is the widely used linear interpolation formula and can be expressed more simply as

$$y = a_0 + a_1x$$

Now consider a function, $y = f(x_1, x_2)$, tabulated at all points, $(m\Delta, n\Delta)$, of a uniform square grid of side Δ . For interpolation throughout any one of these squares, the function can be put in the form,

$$y = a_0 + a_1x_1 + a_2x_2 + a_{12}x_1x_2$$

that is known as a "bilinear" or "double interpolation" formula. Figure 32 shows such a surface, in which the rulings indicate intersections of the interpolation surface with planes $x_1 = \text{constant}$ and $x_2 = \text{constant}$. This is one of a class of functions known as doubly ruled surfaces; it is generated by either of two possible straight lines. As in

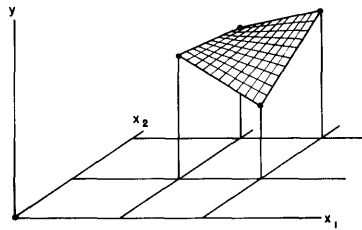


Fig. 32. Surface of bilinear interpolation.

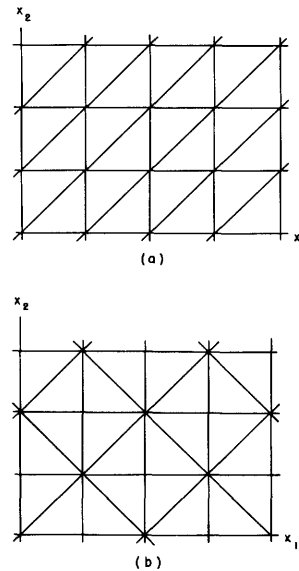


Fig. 33. Two simplicial subdivisions of the plane.

this case, these surfaces are generally nonplanar or "warped." Clearly, the presence of the product term in the bilinear formula makes it unacceptable in a system that is restricted to piecewise-linear functions. Generalization to n variables produces additional product terms.

Because of the limitation to piecewise-linear elements, none of the aforementioned interpolation formulas are usable except the formula for functions of a single variable. The fact that the linear interpolation leads to an acceptable piecewise-linear function in this special case is doubtless an important reason why much work has been done in constructing these functions with the use of diode networks but almost no generalizations to more variables have been attempted.

Let us then consider a usable type of interpolation function, $y = f(x_1, x_2, \dots, x_n)$. The restrictions that it must satisfy are:

1. f must be a piecewise-linear and continuous function, defined over the interior and boundaries of an n -cube;
 2. f must assume some arbitrary value at each of the 2^n vertices of this n -cube;
 3. f must not have any maxima or minima anywhere in the interior of the cube.
- (This restriction is meant to rule out any functions which obviously do not perform a good interpolation, although they still allow freedom of choice of the actual function.)

At this point, it is appropriate to give special attention to functions of a small number of variables, starting with two, and gradually generalize to the n -variable case. This is done mainly for the sake of clarity, for the important points are brought out best by geometrical visualization of particular examples. Visualization in more than three dimensions is close to impossible. Furthermore, the two-variable case is by far the most appealing in the practical sense; for this reason, greater emphasis will be

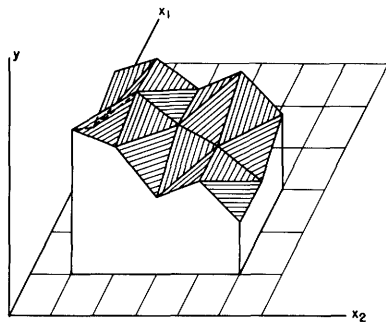


Fig. 34. Piecewise-linear surface.

placed on it in the following sections.

Consideration of the two-variable case provides a clue to the form that the interpolation function must take. The problem here is to construct a continuous surface that consists of segments of planes passing through each point of tabulation with the correct ordinate. The simplest interpolation function in this case would be a plane, and, at first glance, this might look like the logical extension of the straight-line interpolation for functions of a single vari-

able. However, the grid of tabulation consists of squares, and it is obviously impossible, in general, to pass a plane through four ordinates corresponding to the four vertices of a square. The most efficient way out of this dilemma is to divide each square of tabulation into two triangles by drawing one diagonal; then pass a triangular segment of a plane through the ordinates centered over the vertices of each triangle. This produces a piecewise-linear, continuous function which takes on the prescribed values at the tabulated points, and approximates "bilinear" interpolation. Figure 33 shows a portion of the independent-variable plane that is divided into triangles and uses two different systems: (a) with all diagonals drawn upward and to the right, using lines of the form, $x_1 - x_2 + k = 0$; (b) with directions of the diagonals alternated to form a crisscross pattern, using lines of the form, $x_1 \pm x_2 + k = 0$. Although these two systematic methods of subdivision are most convenient as a basis for synthesis, there is no reason why any other scheme could not have been used, such as drawing either diagonal of each square at random. Figure 34 shows a portion of a surface that performs piecewise-linear interpolation among several arbitrary points. The function is specified arbitrarily at the tabulated points and is linear (or planar) over each triangular section of the independent-variable plane; the type (a) subdivision has been used as a basis for this example. The jagged appearance of the function is due to the fact that widely different values of neighboring ordinates were used in the example to illustrate the piecewise-linear nature of the function. In practice, neighboring points would be more nearly equal in height.

Before passing from the two-variable case to higher-order functions, it is convenient to pause and review some of the definitions and terminology that are necessary in dealing with n-dimensional geometrical problems.

1. Linear independence. A set of points in n-space is linearly independent if no subset of $n + 1$ of these points lies in a hyperspace of dimension $n - 1$. For example, four points in three-dimensional space are linearly independent if they are not all coplanar.

2. Polytope. The generic term in the series: polygon, polyhedron, A polytope

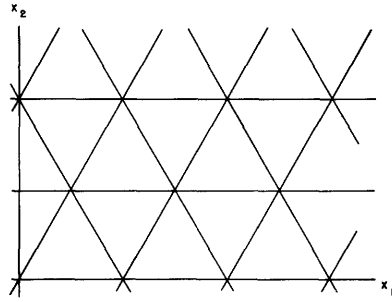


Fig. 35. Regular simplicial subdivision of the plane.

is an n -dimensional polygon; the set of points in n -space enclosed by a set of intersecting $n - 1$ dimensional hyperplanes.

3. Simplex. An n -simplex can be defined as the least convex set containing $n + 1$ linearly independent points; or, in other terms, a closed polytope bounded by $(n + 1)$ $n - 1$ dimensional hyperplanes. It is, in a sense, the simplest n -dimensional figure. For $n = 0$ it is a point; $n = 1$, a line segment; $n = 2$, a triangle; $n = 3$, a tetrahedron, and so on. Thus, it may be thought of as a generalized tetrahedron.

4. Tessellation. A tessellation of a space is a set of polytopes that "fill up" or cover the space, without overlapping. A regular tessellation is a tessellation that uses regular polytopes. An example of the latter is the construction of a floor covering that has regular, hexagonal tiles.

In postulating a hypercubical grid of tabulation, the independent-variable space is divided into a regular tessellation of hypercubes. It is interesting to note that for functions of two variables, three regular tessellations of the plane are possible: triangular, square, and hexagonal. However, it can be shown (2) that the hypercubical tessellation is the only possible regular one for n greater than 2. Figure 35 shows the equilateral, triangular tessellation of the plane. To utilize such a subdivision of the plane as a basis for a piecewise-linear function, the function must, of course, be tabulated at the triangle vertices. Although this form of tabulation is somewhat unusual, it has the advantage that no further subdivision into simplices is necessary; the function can be linear over each triangle. In the case of the square tabulations of Fig. 33, a further simplicial subdivision (partitioning of the squares into triangles) was necessary to provide for piecewise-linear interpolation.

The role of the n -simplex in this discussion becomes evident when one recalls that a linear function of n variables is uniquely specified by the values it assumes at $n + 1$ linearly independent points in the independent-variable space. Since the n -simplex contains just that many vertices, it is clear that it should form the basis of the construction of a piecewise-linear function of n variables. Thus (considering hypercubical grids), if each n -cube of tabulation can be divided into simplices, as was done in the two-variable case, in which all the simplex vertices correspond to n -cube vertices, a

function can be constructed that is linear over each simplex and assumes the prescribed values at the simplex vertices. This is an acceptable interpolation scheme, since it is piecewise-linear, continuous, takes on the correct values at the points of tabulation, and has no maxima or minima within each n-cube (because of the linearity of each segment, and because there are no extra vertices within the cube). The appearance of vertices at points other than at the tabulated ones is to be avoided. To illustrate its consequences, consider the function of two variables tabulated on a square grid. Suppose both diagonals were drawn for each square instead of just one. This would produce an extra vertex in the center of the square, and divide the square into four instead of two simplices. The piecewise-linear function cannot be defined uniquely over these four triangles because of the freedom left in specifying its value at the center vertex. Of course, this center point could be specified as, say, the average of the four surrounding points, but this would just be equivalent to forming a finer grid of tabulation: one whose interval is $1/\sqrt{2}$ times the original interval, and which is rotated 45° . This finer grid could have been established originally, so nothing has been gained by the addition of the extra vertex.

Now let us proceed to the simplicial subdivision of a three-dimensional, independent-variable space. Obviously, the method of Fig. 35 cannot be generalized to three variables, since three-space cannot be tessellated with regular tetrahedra, the logical generalizations of equilateral triangles. Thus, for higher dimensions, the hypercubical grid must be used as a basis for tabulation; each hypercube must, therefore, be simplicially subdivided.

It is possible to generalize either of the two types of subdivisions appearing in Fig. 33 to three-cubes. In order to generalize the method of subdivision pictured in Fig. 33a, the independent variable space should be subdivided in a manner such that projections of the space into the x_1 - x_2 plane, x_1 - x_3 plane, or x_2 - x_3 plane should all appear as shown in Fig. 33a. This can be arranged by passing planes of the form, $x_1 - x_2 + k = 0$, $x_1 - x_3 + k = 0$, and $x_2 - x_3 + k = 0$, through each cube. Figure 36 shows this type of subdivision of one cube. It is divided by three planes into six tetrahedra (three-simplices), which have been slightly separated for illustrated purposes. Similarly, generalization of the type of subdivision pictured in Fig. 33b can be constructed by passing planes of the form, $x_1 \pm x_2 + k = 0$, $x_1 \pm x_3 + k = 0$, and $x_2 \pm x_3 + k = 0$. The pattern formed by this type of subdivision is shown in Fig. 37. Several of the cubes of tabulation are shown; one is drawn in more detail to illustrate the intersection of three planes that divide it into six tetrahedra. Note that when the type (a) subdivision is used, each square or cube undergoes the same type of partitioning, but when the type (b) subdivision is used, two types of square partitioning and four types of cube partitioning occur. This nonuniformity in the internal structure of the cubes has an important bearing upon the methods that will be employed in constructing piecewise-linear functions over these spaces.

In generalizing these methods of simplicial subdivision to functions of n variables,

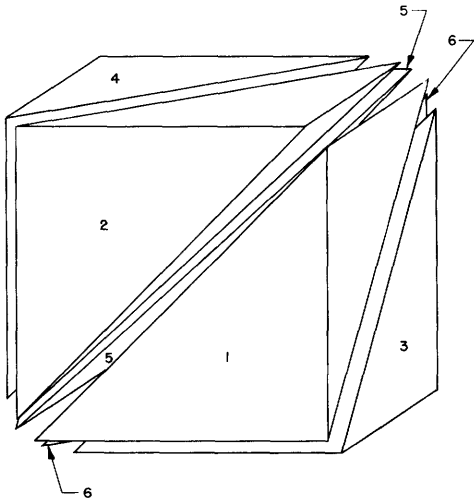


Fig. 36. Type (a) simplicial subdivision of the cube.

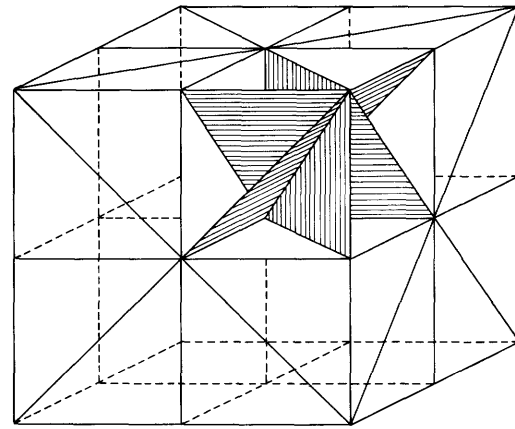


Fig. 37. Type (b) simplicial subdivision of three-space.

it is convenient to use the simplest and most systematic method. Therefore, the type (b) subdivision will be abandoned for generalization purposes, in favor of the type (a) method. The progressions, one simplex for $n = 1$, two simplices for $n = 2$, six simplices for $n = 3, \dots$, suggests that this method of subdivision, when it is generalized, would lead to $n!$ simplices for each n -cube. This, indeed, is the case. It might be further conjectured that $n!$ is the minimum number of simplices that will tessellate an n -cube. Unfortunately, subdivision of a three-cube into five tetrahedra provides a counter example for this.

By using the type (a) method of subdivision, each n -cube in the n -dimensional, independent-variable space is divided into n -simplices by passing all possible $n - 1$ dimensional hyperplanes of the form

$$\begin{aligned}
 x_i - x_j + m\Delta = 0 & & i \neq j \\
 & & i = 1, 2, \dots, n \\
 & & j = 1, 2, \dots, n \\
 & & m = 0, 1, 2, \dots
 \end{aligned}$$

with Δ as the interval of tabulation.

It is proved in Appendix III, that this method of subdivision always results in $n!$ simplices, which are non-intersecting except for their bounding surfaces, whose vertices correspond to the points of tabulation. The proof in Appendix III is of interest because it indicates how each n -cube is actually divided. It happens that this method of partitioning separates the set of points of the n -cube into subsets that are based upon the ordering of their coordinates: e.g., all points whose coordinates are ordered in the form, $x_1 \geq x_2 \geq \dots \geq x_n$, fall within one simplex. There are exactly $n!$ such orderings, corresponding to the $n!$ simplices.

It has now been shown that there is a systematic procedure available for subdividing an independent variable space of any dimension into simplices whose vertices correspond with the vertices of the hypercubes of tabulation. Thus, it is now possible to define a piecewise-linear function that satisfies the conditions stated for a generator of arbitrary functions of several variables. The next section indicates alternative methods of synthesizing such functions.

Unit Functions and Function Generators

Figure 34 reveals that an arbitrary piecewise-linear function of two variables can become rather complicated; indeed, if it were possible to draw such a function of three variables, its intricacy would be quite staggering. It is, therefore, desirable to resolve the function into a summation of simpler functions in order to make it amenable to synthesis. These simpler functions, which will be called "unit functions", will be discussed first in relation to general functions of a single variable, and then will be generalized to functions of several variables, as was done in the previous section.

Consider the piecewise-linear function of a single variable that is shown in Fig. 38. It is tabulated on a regular grid with linear interpolation used between the points.

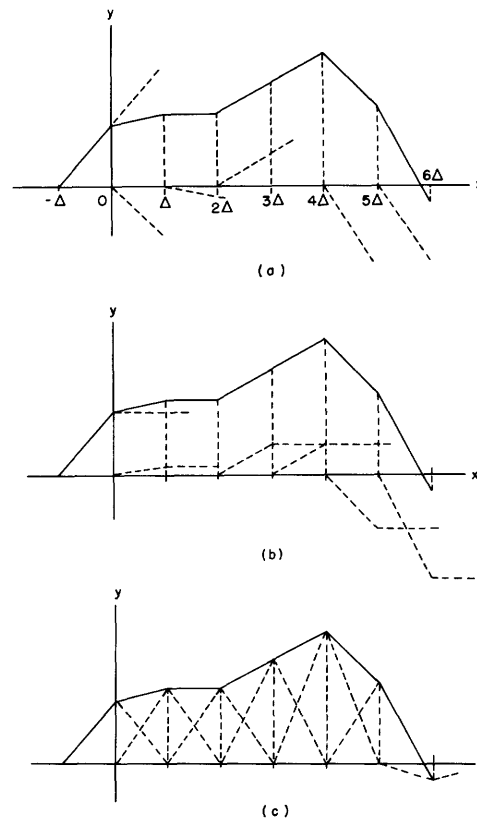


Fig. 38. A function of a single variable decomposed into unit functions.

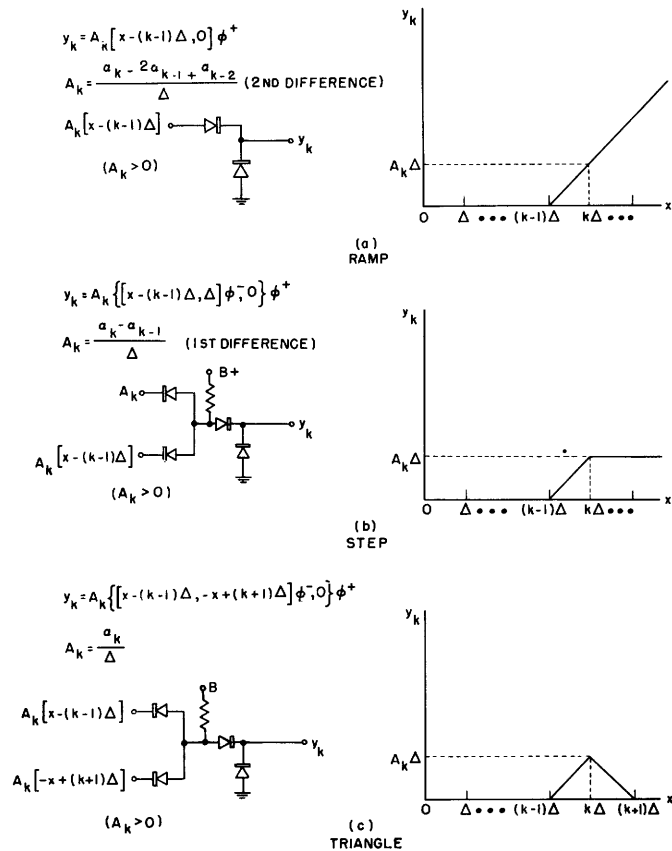


Fig. 39. General unit functions of one variable.

Figure 38a illustrates one method of decomposing this function into (or conversely, constructing the function from) a summation of simpler piecewise-linear functions. Starting from the left, we construct a function which is zero for $x \leq -\Delta$, and is linear for $x \geq -\Delta$, with a slope equal to the slope of the complete function in the interval $-\Delta \leq x \leq 0$. Part of the function is shown in the figure by the dashed line that extends above the original function. Similarly, a second function is constructed which is zero for $x \leq 0$, and linear thereafter, with a slope equal to the difference in the slopes of the first and second segments of the original function (indicated by the first dashed line of negative slope in Fig. 38a). This procedure is continued with the construction of one of these simpler functions for each breakpoint of the original one (indicated by the various dashed lines in the figure). These simpler functions will be known as "ramp functions". When they are constructed as indicated, their sum is the original function; each ramp accounts for the change in slope through the point at which its breakpoint occurs.

Figure 39a presents a clearer interpretation of the ramp. A general ramp function is shown, centered over the k^{th} point of tabulation (it is convenient to define the center of the ramp as the first tabulated point after its breakpoint) with a slope, A_k . Its

algebraic expression, y_k , is given, as is a diode network that realizes the ramp as a voltage transfer function. An expression for each A_k in terms of the ordinates, a_k , of the arbitrary function which constitutes their sum is also given. Note that the network realization is merely a ϕ^+ network (the bias is unnecessary in this case and has, therefore, been omitted). The "unit ramp function" will be defined as a ramp function of unity positive slope. The general ramp is merely a unit ramp multiplied by an appropriate positive or negative scale factor.

Figure 38b and c illustrates two other types of unit function that can be used in constructing functions of a single variable. In Fig. 38b, a set of "step" functions is used, in which each step accounts for the change in ordinate of the original function from one point to the next. Similarly, in Fig. 38c, a set of "triangle" functions is used. In this case, only the particular triangle function that is centered over a tabulated point contributes to the total function at that point. All the other triangles are zero at this point. The over-all function can, therefore, be constructed very simply with the use of triangles, merely by adjusting the height of each triangle to the desired height of the function at that point. The sides of adjacent triangles add to perform linear interpolation between points. Figure 39b and c provides the pertinent information regarding realization of the general step and triangle functions. Just as in the case of the ramp, the "unit" step and triangle functions are defined as the functions that have unity slope over their second linear interval.

These unit functions constitute three possible "building blocks" for piecewise-linear functions of a single variable. Although they were postulated over a regular grid, they can, with slight modifications, be applied equally well to functions tabulated over irregular grids. While we are still considering functions of a single variable, it is instructive to note some of the ramifications of this method for the construction of a function. First, it should be noted that decomposition of the desired function into a set of unit functions is a straightforward procedure, given by the expressions for the A_k 's. Second, synthesis of any unit function as a voltage transfer function is also quite simple, involving, first, the symbolic representation of the function and, then, mechanization of the symbolism by using diode networks. Note that, in the figures, the inputs to the diode networks are given as linear combinations of the input voltage variables and a constant. In practice, these linear combinations can be formed in several ways, with passive summing networks, active components that use high-gain amplifiers, batteries, and so forth. In the remainder of this section it will be assumed that devices for performing this linear operation are available, but they will be omitted from the discussion and the diagrams.

The structure of a general-purpose function generator that is based on unit functions is shown in the form of a block diagram in Fig. 45. The figure is meant to portray a generator of functions of n variables, but it can be specialized to a single variable by considering only one input, x_1 , and a bias, and by replacing the word "pyramid" by "ramp", "step", or "triangle". It will be shown in the following discussion that the pyramid

function is the n-dimensional generalization of the triangle. Each of the unit functions in the diagram is centered over a different point of tabulation, and the potentiometers and switches provide a means of controlling the scale factor of each function. Thus, any function can be "programmed" into the machine by adjusting the appropriate potentiometers and switches.

It is interesting to note how the various unit functions compare as a basis for

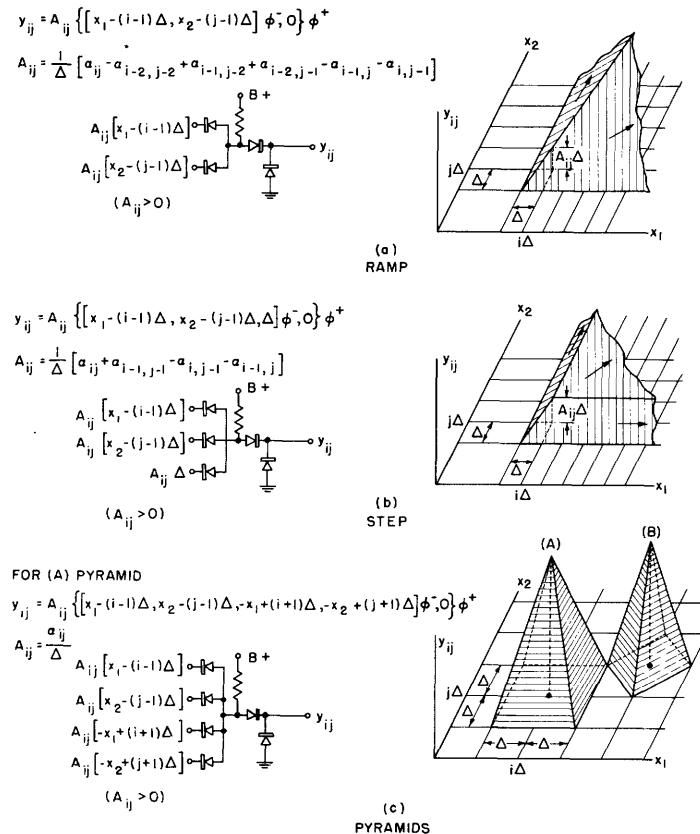


Fig. 40. General unit functions of two variables.

transfer function synthesis. Reading from top to bottom in Fig. 39, it can be seen that the functions become more complex and require more diodes for synthesis. However, it can also be observed that the process of setting up the over-all function becomes simpler toward the bottom of the list. If we use ramps, adjustment of one ordinate disturbs all points to the right of that ordinate; if we use triangle functions, each ordinate of a function can be adjusted independently of the others. This independence of adjustment considerably simplifies the problem of setting up a function and reduces the propagation of errors. These advantages of the triangle function become more pronounced in generalizing to functions of several variables.

As many alternative methods of producing functions of a single variable (many of

them based upon the ramp function) are in existence at present, the subject will not be pursued further. The main value of the unit functions of a single variable is that they provide a basis for generalization to functions of several variables. Figure 40 shows three unit functions of two variables, which are generalizations of the functions just described. The projections of the two variable unit functions, the ramp, step, and pyramid, into either the $y-x_1$ - or $y-x_2$ -plane, reduce to single-variable, ramp, step or triangle functions, respectively. Thus, these are the generalizations of the latter to functions of two variables. This generalization can be given further justification by referring to the tessellations of the independent-variable plane, as shown in Fig. 33. Clearly, any function that is a summation of ramp functions of two variables can only have breaklines parallel to either axis or along lines of unity slope in the $x_1 - x_2$ plane. Thus, a function formed by the summation of one ramp that is centered over each tabulated point in a square grid would, in general, appear as shown in Fig. 33a when it is projected onto the $x_1 - x_2$ plane. Viewed in three dimensions, it would have the appearance of Fig. 34. A summation of step functions must also take this form. Conversely, an arbitrary function that is tabulated on a square grid can be resolved into a summation of either ramps or steps that will produce the piecewise-linear interpolation defined by the tessellation of Fig. 33a. It can be shown that the expressions in Fig. 40a and b give the required height, A_{ij} , of the ij^{th} unit function in terms of the prescribed ordinates, a_{ij} , of the desired function.

Now suppose the crisscross type of tessellation, shown in Fig. 33b, is to be used as a basis for piecewise-linear interpolation. Obviously more than one type of unit function is necessary, since the pattern of breaklines differs from point to point. A summation of the pyramidal functions of Fig. 40c is appropriate in this case. To produce the desired pattern, the type (A) pyramids must be centered over all points that are intercepted by diagonals, while the type (B) pyramids are centered over the rest of the points and are alternated with each other. Just as the triangular function was zero at every tabulated point except at its center, and was nonzero only over the two line segments adjoining its center, the pyramidal functions are zero at every tabulated point except at their centers, and are nonzero only over the four triangles adjoining their center. The resultant interpolation function in any particular triangular region of the independent-variable plane is composed of the superposition of the three pyramids which are nonzero in that region; namely, the ones that are centered at the three vertices of that particular triangle. In the case of the pyramids we again have the opportunity of adjusting ordinates independently of each other. For example, a machine can be constructed which has, say, a 10×10 array of knobs that control the scale factors on a 10×10 array of unit pyramidal functions, summed to form an arbitrary function of two variables. The tabulated function is fed into the machine by adjusting each knob to the correct tabulated value. The function can then be displayed on an oscilloscope and a visual check can be made while various tabulated points are being changed. This can be done with the assurance that no tabulated points other than those that are being adjusted

will change. Figure 40 indicates that the ramp and step functions are both nonzero over a complete quadrant of the independent-variable plane. Hence, a change in the amplitude of the ij^{th} ramp or step effects every tabulated ordinate, a_{pq} , $p \geq i$, $q \geq j$. This indicates that any errors in the construction or scale factoring of a ramp or step function are propagated throughout a considerable portion of the complete function. Such a phenomenon is undesirable but tolerable in the case of functions of one variable; for functions of two variables, it presents extreme difficulties in programming the function; for more than two variables, the effect would undoubtedly be intolerable. It seems, therefore, that the unit pyramidal functions, and their generalizations to functions of n variables are the most desirable building blocks for piecewise-linear function generators, both from the point of view of ease in programming and of reduction of static errors. The advantages gained appear to be well worth the increase in circuit complexity.

All of the unit functions of two variables that have been discussed thus far have certain common geometrical characteristics: they are all formed by a convex surface, which intersects the zero plane in concave intersections. Thus, they can all be synthesized by networks of the same form: a ϕ^- network, followed by a ϕ^+ network which compares the output of the ϕ^- network with zero. The only difference between the various circuits is in the number and form of their linear inputs. It can be shown that this general form is retained when these functions are generalized to n variables. Before making this generalization, however, it is useful to consider two additional varieties of unit pyramidal functions of two variables. Since the method of tessellation determines the admissible forms of unit pyramidal functions, a different type of pyramid must be associated with the equilateral tessellation of Fig. 35. First, it can be seen from the regularity of the pattern that, in this case, only one type of pyramid is necessary. Also, from the fact that the pyramid function must be nonzero at one tabulated point, zero at all the others, and linear over each triangle, it can be seen that the appropriate function is a regular hexagonal pyramid that covers the six triangular regions adjoining its center (Fig. 41). The equilateral tessellation is of interest mainly because its result is these regular pyramidal functions. It cannot be generalized, however.

Another form of pyramidal function, which can be generalized, is shown in Fig. 42 together with a network that realizes it. In this case, six linear inputs, y_1, y_2, \dots, y_6 , representing the six faces of the pyramid, are necessary. Observe that this pyramid is defined on the type (a) tessellation of Fig. 33a. Since it can be used over every point in the plane, no alternation of pyramids as in the case of the type (b) tessellation, is necessary. Although this function requires two more diodes than the pyramids of Fig. 40c, it has a decided advantage because of the simplicity of its generalization. Figure 43 shows the pyramid based upon the three-variable case of the type (a) tessellation. Further generalizations will be confined to this type of tessellation.

Generalization of the type (a) tessellation, as described above, results in $n!$

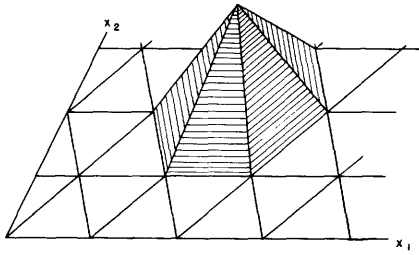


Fig. 41. Hexagonal pyramid on a regular triangular grid.

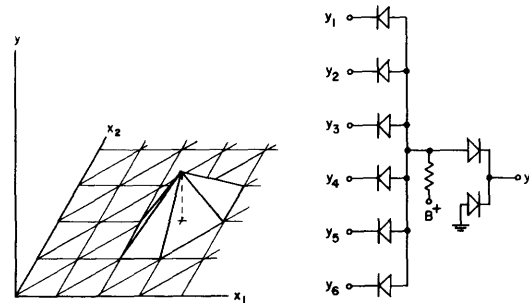


Fig. 42. Unit pyramid on type (a) tessellation.

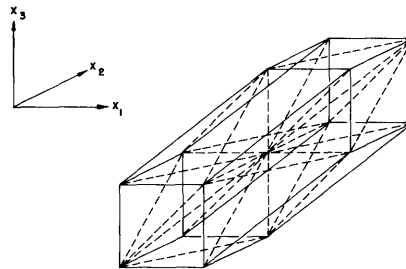


Fig. 43. Pyramidal functions of three variables on type (b) tessellation.

simplices for each n -cube. By referring to the proof of the tessellation theorem in Appendix III, it can be shown that each generalized n -pyramid function is nonzero over $(n + 1)!$ simplices that constitute portions of the 2^n n -cubes adjoining the tabulated point over which the pyramid is centered. As mentioned previously, but not generalized, the definition of a pyramidal function, f_p , of n variables is:

(a) f_p is nonzero at one point in the grid of tabulation and zero at every other tabulated point;

(b) f_p is linear over each simplex in the variable space, when the simplices have been defined by one of the aforementioned methods of tessellation.

This definition uniquely specifies f_p once the system of tessellation has been established. The form of the network that realizes this function is illustrated in Fig. 44. To construct a generator of arbitrary functions of n variables, a set of unit pyramidal functions is constructed; one is centered over each tabulated point, with their outputs fed through attenuating potentiometers and reversing switches to two summing and inverting amplifiers. The block diagram is shown in Fig. 45.

Before concluding this discussion of the theoretical aspects of arbitrary function generation, the n -variable generalizations of the unit ramp and step functions should be mentioned. They have not been emphasized for n greater than 2 for two reasons: 1. difficulties in programming the function and controlling errors, as mentioned previously, and 2. cumbersome relations between the coefficients of the unit functions and

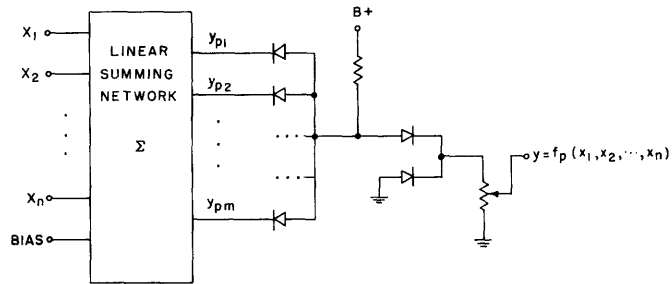


Fig. 44. Circuit for the general pyramid.

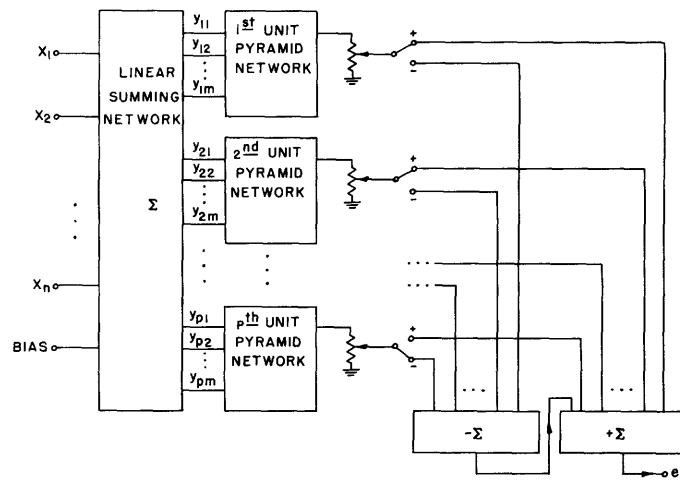


Fig. 45. General-purpose function generator.

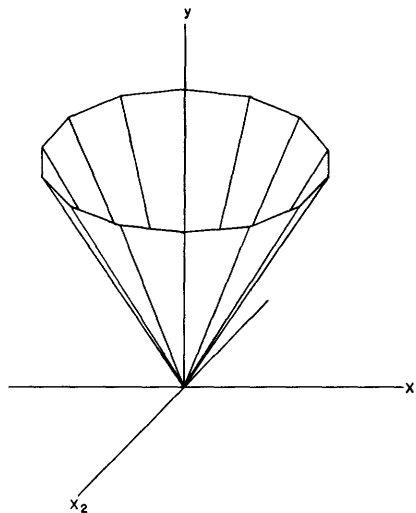


Fig. 46. Approximation of the cone.

the specified ordinates of the tabulated function. (In the two-variable case it was seen that the coefficient of any step function was a function of the ordinates of four tabulated points; the ramp was a function of six points. These numbers appear to increase rapidly as n increases, and their derivation becomes a fairly difficult matter.) The expression for the unit ramp for n variables is

$$f_r = \left[(x_1 - k_1\Delta, x_2 - k_2\Delta, \dots, x_n - k_n\Delta)\phi^-, 0 \right]\phi^+$$

For the unit step,

$$f_s = \left[(x_1 - k_1\Delta, x_2 - k_2\Delta, \dots, x_n - k_n\Delta, \Delta)\phi^-, 0 \right]\phi^+$$

For large n , these functions may have certain advantages over the pyramids because they require, at most, $n + 1$ linear inputs, while the pyramid requires $(n + 1)!$ inputs. (However, many of these are identical; e.g., in the case $n = 3$, of the 24 possible inputs, only 12 are distinct.)

Many of the methods described in this section have been applied successfully to practical function generators. (See references 12, 23, 15.)

b. Special Purpose Function Generation

In designing a generator of a particular function or class of functions of several variables, an approach that differs somewhat from the above procedures is desirable. Clearly, a simpler and more efficient machine can be designed if it is to be used only for a single function rather than to be adaptable to all types of functions. Usually, any particular function will have certain geometrical properties that will suggest special types of piecewise-linear approximations. This is especially true of functions that are given in analytic form. Discussion of these properties will be confined to functions of two variables.

The surface, $y = f(x_1, x_2)$, can be approximated and realized easily if it is: (a) completely convex or concave, and/or (b) a ruled surface. Examples of functions of type (a) are:

$$y = \left| (x_1^2 + x_2^2)^{1/2} \right| \quad (\text{half cone})$$

$$y = x_1^2 + x_2^2 \quad (\text{paraboloid of revolution})$$

$$y = \left| (x_1^2 + x_2^2 + a^2)^{1/2} \right| \quad (\text{half hyperboloid of revolution})$$

Examples of type (b) functions are:

$$y = \left| (x_1^2 + x_2^2)^{1/2} \right|$$

$$y = x_1 x_2 \quad (\text{multiplier} - \text{doubly ruled})$$

$$y = \frac{x_1 x_2}{a^2 + x_2^2}$$

$$y = \tan^{-1} \frac{x_2}{x_1}$$

Type (a) functions are easily realizable because they can be approximated piecewise-linearly to any degree of precision by an expression of the form

$$y = (a_1 + b_1 x_1 + c_1 x_2, a_2 + b_2 x_1 + c_2 x_2, \dots) \phi^\pm$$

This expression can be realized electrically with a single ϕ^\pm network. Figure 46 shows the cone approximated in this manner by using 16 planar segments. In the approximation, intersections with planes parallel to the $x_1 - x_2$ plane are regular polygons instead of circles. As the number of segments used in the approximation is increased, the number of sides of the polygons are increased and they tend toward true circles. (The accuracy of the approximation that is shown is better than 1 per cent.) Figure 47 shows a circuit realizing the approximation (13). Since the function is even in both variables, the network was simplified by using two full-wave rectifiers. Note that each linear function is formed by a passive summing network, and no bias is required because each plane passes through the origin. An interesting result of this type of approximation is that the accuracy is maintained as a constant percentage of the output voltage, even in the vicinity of the origin; this is due to the fact that the breaklines all converge toward the origin. The extreme simplicity of this function, and its wide application, make it a powerful example of the advantages to be gained by attacking synthesis problems from a more general point of view. A more conventional method of forming this function would involve two square-law devices, an adder, and a square-root device; clearly, an inferior system.

Now let us consider the realization of type (b) functions, ruled surfaces. Such functions can be approximated to any degree of accuracy by segments of planes that intersect along lines that coincide with the generatrix of the surface. The approximation is exact along these intersections. In general, additional diagonal intersections must be added so that the surface is divided into triangular segments of planes. Figure 48 illustrates such an approximation. Once the approximation has been made, the resultant intersections can be classed as either convex or concave, and the surface can be resolved into a linear combination of two functions, one containing all the convex intersections, and the other containing all the concave intersections. These functions, being of type (a), are readily realized.

Realization of a more general type of function, such as might be given graphically by a family of curves with one independent variable as a parameter, is not so well-defined

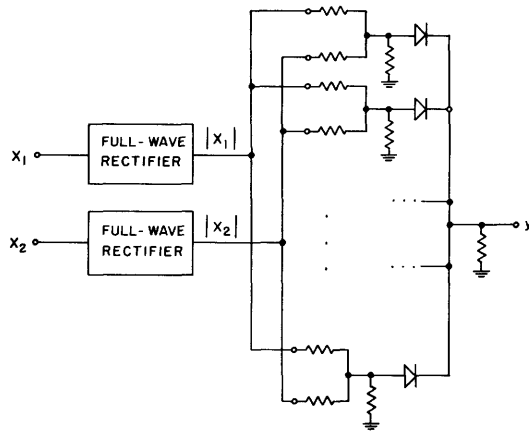


Fig. 47. Network realizing the cone approximation.

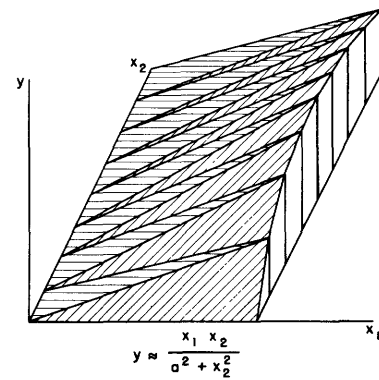


Fig. 48. Approximation of a ruled surface.

a problem. In most cases the contours of the function will lend themselves to certain convenient types of piecewise-linear approximations; however, no general methods exist for determining an "optimum" surface of approximation. Roughly, the steps in the approximation and realization of a function given in this graphical form are:

1. approximation of each curve in the family piecewise-linearly, attempting to make similar types of approximation for neighboring curves,
2. connection of the break-points of each piecewise-linear approximating function with those of its neighboring one by straight lines, until a piecewise-linear surface consisting (in general) of triangular segments of planes has been formed,
3. the obtaining of an algebraic expression for the surface of approximation by examining the breaklines for convexity or concavity, and
4. the mechanizing of this expression by using diode networks. This procedure is more or less of an art, and the ability to obtain the best approximation with the fewest segments of planes is largely a matter of experience.

As an illustrative example, consider the thermodynamic surface, $P = f(V, T)$, for a real gas (water). The family of curves in the P - V plane, with temperature as a parameter, is shown in Fig. 49a. Figure 49b shows a piecewise-linear approximation. The isotherms appear as dashed lines; the solid lines joining the breakpoints indicate the plane intersections. Note that the planar segments near the center of the P - V plane are mostly triangular, a consequence of the irregularity of the function in that region. Toward the lower and higher temperatures, however, the isotherms become more nearly parallel. It is, therefore, possible to find sets of four adjacent coplanar breakpoints that determine planar segments in the form of quadrilaterals or, in many cases, parallelograms. Such situations are desirable, since they reduce the required number of planar segments. Often a small perturbation of a breakpoint to a new location can create such a situation, and thereby simplify the approximation.

The surface of approximation is shown in three dimensions in Fig. 49c. Figure 49d shows a model of the original surface for the purpose of comparison. To get a true

picture of the nature of the approximation, it should be observed that the surface of Fig. 49d extends over a larger area of the V-T plane than the piecewise-linear surface of Fig. 49c. Most of the breaklines can be classified by inspection of Fig. 49c. For example, the intersection between planes 2 and 12 is obviously concave, and the composite surface formed by planes 1, 4, and 5 is also concave. Calculation of the convexity or concavity of the doubtful intersections can be accomplished in a straightforward manner by returning to Fig. 49b. Since the temperature coordinate of each breakpoint is indicated, the equation for each plane can be derived by solving a set of three linear equations. Now, suppose the classification of the breakline between planes 8 and 9 is to be determined. This can be done by substituting the V and T coordinates of some point in plane 8 (not on the breakline) in the expression for plane 9, this expression being put in the form,

$$P_9 = a_9 + b_9V + c_9T$$

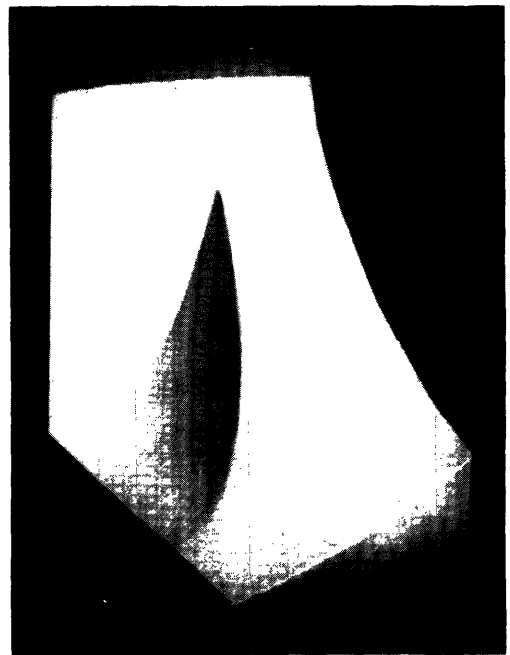
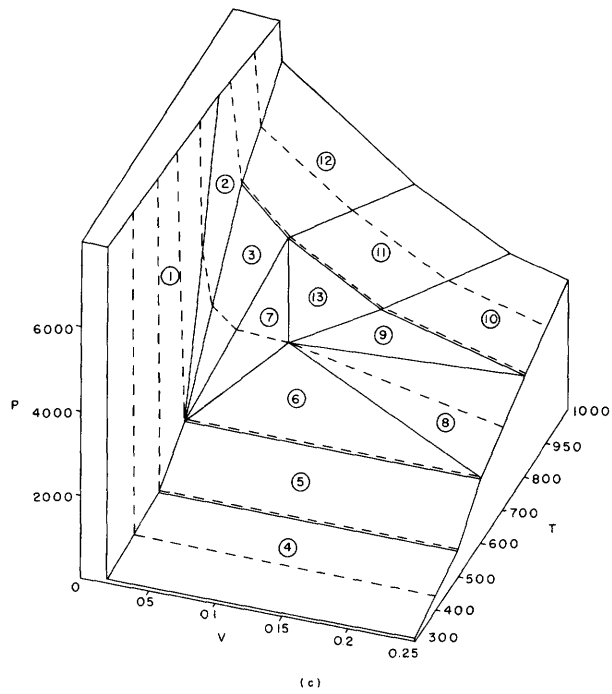
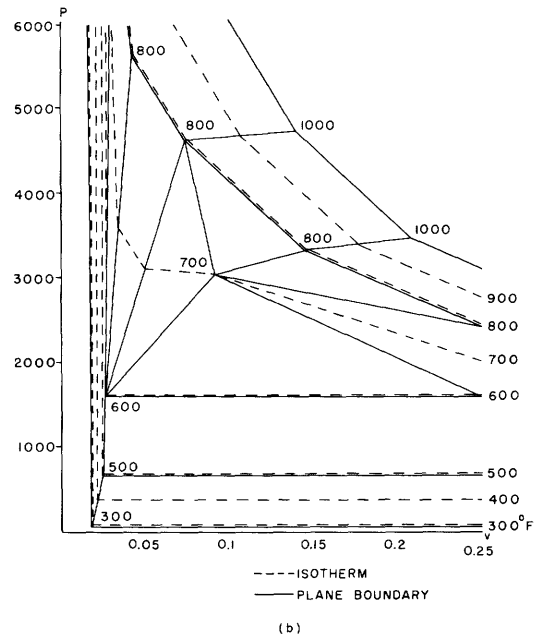
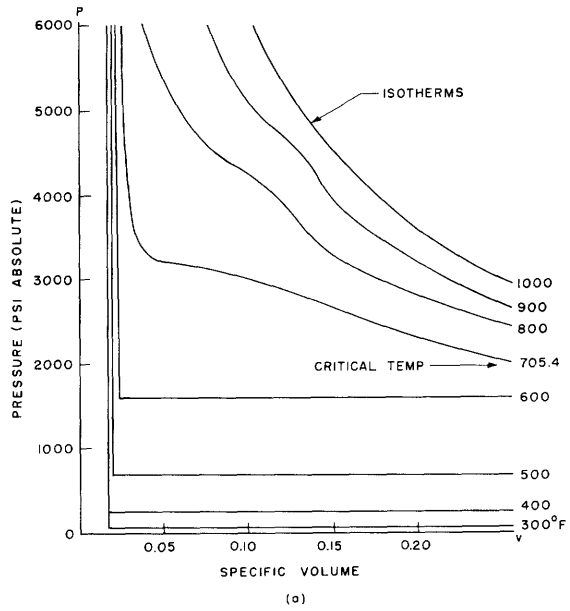
If the value of P given by this substitution is greater than the P coordinate of the point in plane 8, it means that plane 9 lies above plane 8 when extended into region 8. Therefore the intersection is convex. In this case, it happens that plane 9 lies below plane 8 when it is extended into that region, so that the intersection is concave. Note that this procedure is applicable to functions of any number of variables. It is usually convenient to use the breakpoints as test points for this procedure, since their coordinates can be read directly from the figure. In the case of this function, all of the breaklines could be classified without any laborious calculations by observing the locations of the various breakpoints.

Once the intersections have been classified, the function must be put in symbolic form. Usually several alternative forms are possible and can be derived without too much difficulty. Figure 49e gives an algebraic representation of the function under discussion, where each P_i is a linear function of the form,

$$P_i = a_i + b_iV + c_iT$$

The same figure also shows the diode network that realizes the function. Note that eighteen diodes were used here, a quantity comparable with the number of diodes used in constructing a reasonably accurate square-law device. If this function were to be programmed into a general-purpose function generator in a 10×10 tabulation, approximately 600 diodes would be required (using pyramid functions) or a minimum of 400 (using ramp functions).

If methods similar to the unit function superposition procedures previously described are used, we can decompose any irregular piecewise-linear function of several variables into simpler functions. For example, considering an irregular function of a single variable, we can make a decomposition into irregularly spaced ramps, steps, or triangles; in this case the triangles will no longer be isosceles. For functions of two variables, a generalization of the pyramidal function is a useful building block. This



(Fig. 49 is continued on the facing page.)

$$P = \left\{ \left[(P_8, P_9, P_{13}) \phi^+, (P_1, P_2, P_4, P_5, P_6, P_7) \phi^+, (P_{10}, P_{11}) \phi^+, (P_3, P_{12}) \phi^- \right] \phi^+ \right\}$$

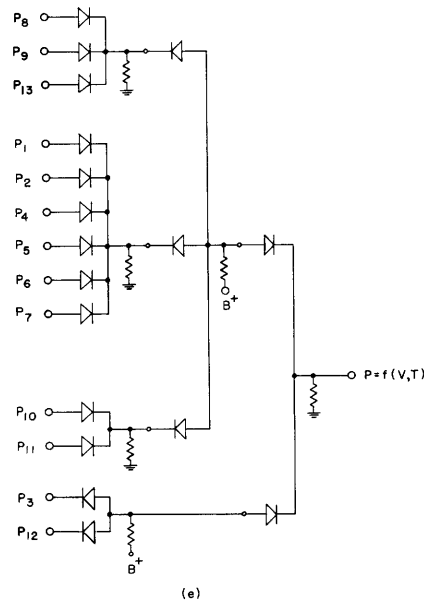


Fig. 49. A thermodynamic surface and steps in its approximation.

generalization can be demonstrated by reference to Fig. 50a. The figure shows the projection of an arbitrary piecewise-linear function of two variables onto the independent-variable plane. The linear regions appear as irregular polygons bounded by projections of the breaklines. Since the projection eliminates any information about the relative heights of the various breakpoints, it cannot be used to determine the classification of the breaklines. However, the complete surface can be realized by a superposition process without determining these classifications. To accomplish this, the independent-variable plane must first be simplicially subdivided. Note that the surface of Fig. 50a contains four-sided and five-sided polygons as well as triangles. The vertices of such polygons must all be coplanar. Simplicial subdivision of any polygon can always be accomplished by drawing a sufficient number of chords. This is demonstrated in Fig. 50b. The pentagon, 17856, is so divided by the dashed lines, 67 and 68; similarly, the quadrilateral, 7238, is divided into triangles by line 73.

The complete surface can be constructed by superposing a set of irregular pyramids, one being centered over each breakpoint. For example, a function can be defined that is equal to the prescribed height of the original function at point 7, is zero at every other breakpoint, and is linear over each simplex (triangle). (It is assumed that the ordinate of each breakpoint is positive.) This definition produces an irregular pentagonal pyramid whose sides are planes P_a , P_b , P_c , P_d , and P_e , corresponding to the lettered regions in Fig. 50b. Each plane is represented by a function of the form,

$$P_n = k_{n0} + k_{n1}x_1 + k_{n2}x_2$$

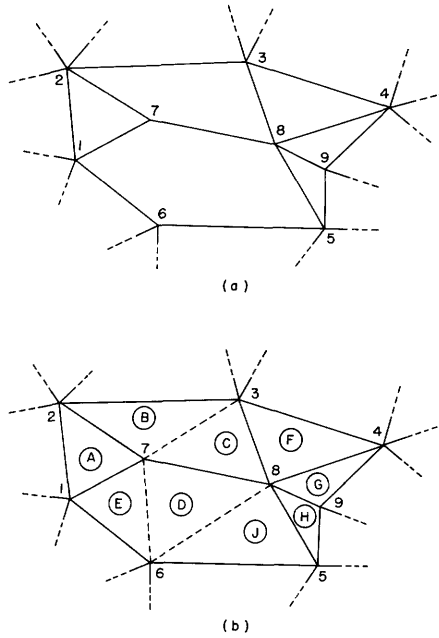


Fig. 50. Arbitrary piecewise-linear surface.

The algebraic representation of the pyramid, y_7 , centered over point 7, is clearly,

$$y_7 = [(P_a, P_b, P_c, P_d, P_e)\phi^-, 0]\phi^+$$

The circuit that realizes y_7 is, therefore, of the same form as the general unit-pyramid circuit.

Now let us proceed, in the same manner, to construct a pyramid that is centered over point 8. This time, we have an irregular hexagonal pyramid, but unlike the previous case, its base is not a convex polygon. (The exterior angle at vertex 9 is acute instead of obtuse.) That all the aforementioned pyramidal functions of several variables were representable algebraically by a ϕ^- , followed by a ϕ^+ transformation, which cut off the function at zero, was due solely to the fact that their bases were all convex polytopes. The indentation of vertex 9 causes breakline 89 to be concave. The rest of the breaklines that converge to point 8, are still convex. Then, an algebraic representation for the pyramid, y_8 , is

$$y_8 = \left\{ [(P_h, P_g)\phi^+, P_c, P_d, P_f, P_j]\phi^-, 0 \right\} \phi^+$$

Its realization is straightforward. The algebraic expression for any such irregular pyramid can be written by inspection, and, therefore, any irregular function can be synthesized systematically by this procedure. It should be observed that the dashed lines are canceled out when overlapping pyramids are superimposed; thus, only the

original breaklines appear in the resultant surface.

VI. SUGGESTIONS FOR FUTURE WORK

There is still much work to be done in exploring the general properties of nonlinear resistive networks: further study of the relation between network topology and possible and impossible performance; effects of oddness or evenness, convexity or concavity, and so forth, of the element characteristics on the terminal behavior of a network; and extension of more linear techniques to nonlinear networks.

Concerning synthesis, much more experimental work must be done in order to determine practical designs of the function generators described in Section V. Also, further analytical work could be done to determine techniques for making piecewise-linear approximations to nonlinear functions within a given percentage of error and utilizing a minimum number of linear segments. Geometrical properties, other than those discussed here, might be utilized to advantage in constructing special-purpose function generators. Also, the methods that have been described for functions of two variables could be extended to more than two variables.

One possibility worth examining is the analysis of systems that contain delay lines. If the lines are properly terminated, such a network has the advantage of possessing a "finite memory," i. e., any transient exists for only a finite time. Transients in lumped-parameter networks usually last for an infinite time. A further advantage of the piecewise-linear, delay-line network is that such a network will necessarily have a piecewise-linear response when excited with a piecewise-linear waveform. This is not the case in a lumped-parameter network. It may also be possible to make analyses of lumped-parameter networks by approximating them with delay lines.

There is a wide range of fairly unrelated areas to which applications of the inequality algebra might be interesting and useful. Problems involving solutions of simultaneous linear inequalities, i. e., linear programming problems, offer one possibility. Mechanical systems that change from one linear region to another, when particles come in contact with each other or separate, offer another possibility.

Although many of these suggestions may turn out to be "blind alleys," the possibilities for exploration are so numerous and varied, and the field so devoid of well-established techniques, that future applications of algebraic techniques to piecewise-linear problems appear to hold considerable promise.

APPENDIX I

PROOFS OF THE THEOREMS OF SECTION II

When a theorem can be stated in several alternative cases, proof will be given for only one case. Proofs for the other cases are of the same form. Only the particular statement of the theorem that pertains to the case that is being proved is given.

THEOREM 1. (a) $\alpha \oplus \beta = \beta \oplus \alpha$

PROOF. Let $\alpha = (a_1, a_2, \dots, a_m)$
 $\beta = (b_1, b_2, \dots, b_n)$

Then $\alpha \oplus \beta$ is the set of all scalars, $a_p + b_q$, a_p in α and b_p in β ; and $\beta \oplus \alpha$ is the set of all scalars, $b_p + a_p$, b_q in β and a_p in α .

Since $a_p + b_q = b_q + a_p$ for all p and q , $\alpha \oplus \beta = \beta \oplus \alpha$.

(b.1) $\alpha \oplus (\beta \oplus \gamma) = (\alpha \oplus \beta) \oplus \gamma$

PROOF. Let $\alpha = (a_1, a_2, \dots, a_m)$
 $\beta = (b_1, b_2, \dots, b_n)$
 $\gamma = (c_1, c_2, \dots, c_s)$

Then $\beta \oplus \gamma$ is the set of all scalars, $b_p + c_q$, b_p in β and c_q in γ ; $\alpha \oplus (\beta \oplus \gamma)$ is the set of all scalars, $a_r + (b_p + c_q)$, a_r in α and $(b_p + c_q)$ in $(\beta \oplus \gamma)$; $\alpha \oplus \beta$ is the set of all scalars, $a_r + b_p$, a_r in α and b_p in β ; and $(\alpha \oplus \beta) \oplus \gamma$ is the set of all scalars, $(a_r + b_p) + c_q$, $(a_r + b_p)$ in $(\alpha \oplus \beta)$ and c_q in γ .

But $(a_r + b_p) + c_q = a_r + (b_p + c_q)$ for all r , p , and q . Hence, $(\alpha \oplus \beta) \oplus \gamma = \alpha \oplus (\beta \oplus \gamma)$

(b.2) $c(da) = (cd)a$

PROOF. Let $\alpha = (a_1, a_2, \dots, a_n)$

Then, $da = (da_1, da_2, \dots, da_n)$; and $c(da) = (cda_1, cda_2, \dots, cda_n) = (cd)a$.

(c) $c(\alpha \oplus \beta) = c\alpha \oplus c\beta$

PROOF. Let $\alpha = (a_1, a_2, \dots, a_m)$
 $\beta = (b_1, b_2, \dots, b_n)$

Then $(\alpha \oplus \beta)$ is the set of all scalars, $a_p + b_q$, a_p in α , b_q in β ; $c(\alpha \oplus \beta)$ is the set of all scalars, $c(a_p + b_q) = (ca_p + cb_q)$, $(a_p + b_q)$ in $\alpha \oplus \beta$; and $c\alpha \oplus c\beta$ is the set of all scalars, $ca_p + cb_q$, a_p in α and b_q in β .

But these two sets are equal; therefore, $c(\alpha \oplus \beta) = c\alpha \oplus c\beta$.

THEOREM 2. $c(\alpha \phi^+) = (c\alpha)\phi^-$, when $c \leq 0$.

The elements of α can be treated as constants without any lack of generality, since the relation must hold at each point in the independent-variable space.

PROOF. Let $\alpha = (a_1, a_2, \dots, a_n)$; $c \leq 0$; and $\alpha \phi^+ = a_p$.

Then $a_p \geq a_1, a_2, \dots, a_n$ and $ca_p = c(\alpha \phi^+) \leq ca_1, ca_2, \dots, ca_n$.

But $(ca_1, ca_2, \dots, ca_n) = c\alpha$; therefore $ca_p = (c\alpha)\phi^- = c(\alpha \phi^+)$.

THEOREM 3. If $\alpha = (a)$, then $\alpha \phi^+ = \alpha \phi^- = a$.

The proof is trivial, since the greatest or least of a set of one element must be the element itself.

THEOREM 4. $(\alpha \oplus \beta)\phi^+ = \alpha \phi^+ + \beta \phi^+$

PROOF. Let $a = (a_1, a_2, \dots, a_m)$ $a_p = a\phi^+$

$\beta = (b_1, b_2, \dots, b_n)$ $b_q = \beta\phi^+$

Then $a_p \geq a_1, a_2, \dots, a_m$ (set of m inequalities); and $b_q \geq b_1, b_2, \dots, b_n$ (set of n inequalities). Select any one of the first set of inequalities and add it to any one of the second set to form

$$a_p + b_q \geq a_i + b_j$$

This can be done in mn possible ways to form a set of mn inequalities with $a_p + b_q$ on the left side of each. The terms on the right side are just the elements of $(a \oplus \beta)$.

(See definition 3.) Therefore, $a_p + b_q = a\phi^+ + \beta\phi^+ = (a \oplus \beta)\phi^+$.

THEOREM 5. $[(a_1, a_2, \dots, a_n)\phi^+, b]\phi^+ = (a_1, a_2, \dots, a_n, b)\phi^+$

PROOF. Let $a_p = (a_1, a_2, \dots, a_n)\phi^+$

Then $a_p \geq a_1, a_2, \dots, a_n$

First, consider the case $a_p \geq b$. Then $[(a_1, a_2, \dots, a_n)\phi^+, b]\phi^+ = a_p = (a_1, a_2, \dots, a_n, b)\phi^+$

Second, consider the case $a_p < b$. Then $[(a_1, a_2, \dots, a_n)\phi^+, b]\phi^+ = b = (a_1, a_2, \dots, a_n, b)\phi^+$.

THEOREM 6. Let $y = F(x) = [f_1(x), f_2(x), \dots, f_n(x)]\phi^+$

If each $f_p(x)$ is monotonically increasing and continuous, then

$$x = F^{-1}(y) = [f_1^{-1}(y), f_2^{-1}(y), \dots, f_n^{-1}(y)]\phi^-$$

PROOF. Clearly, if f_1, f_2, \dots, f_n are monotonically increasing and continuous, then $F(x)$ also has these properties and therefore has inverse $F^{-1}(y)$, as do each of the f_m .

Consider an interval, $a \leq x \leq b$, in which $F(x) = f_p(x)$. (There must be at least one f for which such an interval exists.)

On $[a, b]$,

$$f_p(x) \geq f_1(x), f_2(x), \dots, f_n(x) \tag{1}$$

Assume that $c \leq y \leq d$ on this interval.

Select a_{y_0} in the interval $[c, d]$ and let $x_0 = F^{-1}(y_0) = f_p^{-1}(y_0)$, where $a \leq x_0 \leq b$.

We then have $y_0 = f_p(x_0)$, $y_1 = f_1(x_0)$, $y_2 = f_2(x_0)$, \dots , $y_n = f_n(x_0)$, where

$$y_0 \geq y_1, y_2, \dots, y_n \tag{2}$$

Inverting the above, we have

$$x_0 = f_p^{-1}(y_0) = f_1^{-1}(y_1) = f_2^{-1}(y_2) = \dots = f_n^{-1}(y_n) \tag{3}$$

Now each f_q^{-1} is monotonically increasing. (If a function is monotonically increasing and continuous, its inverse also has this property.)

Therefore, from Eqs. 2 and 3, we have

$$x_0 = f_p^{-1}(y_0) \leq f_1^{-1}(y_0), f_2^{-1}(y_0), \dots, f_n^{-1}(y_0)$$

or

$$x_0 = \left[f_1^{-1}(y_0), f_2^{-1}(y_0), \dots, f_n^{-1}(y_0) \right] \phi^-$$

But this must be true for any x and y on any interval. Hence the theorem is proved.

THEOREM 7. Let $F(x, y) = [f_1(x, y), f_2(x, y), \dots, f_n(x, y)] \phi^+ = 0$.

If, for each $f_p(x, y)$:

1. $f_p(x, y)$ is monotonically increasing and continuous in y for any constant x ;
2. $f_p(x, y)$ is continuous in x for any y ;
3. for each x_0 there is some y_0 for which $f_p(x_0, y_0) = 0$; then $F(x, y) = 0$ can be solved explicitly for y in the form,

$$y = \left[g_1(x), g_2(x), \dots, g_n(x) \right] \phi^-$$

where $y = g_p(x)$ is the explicit solution of the equation, $f_p(x, y) = 0$.

PROOF. First, from the above postulates, it follows that $F(x, y)$ also possesses these properties and that each equation, $F(x, y) = 0, f_1(x, y) = 0, \dots, f_n(x, y) = 0$, has a unique explicit solution for y (4).

Let $y = G(x)$ represent the explicit solution of $F(x, y) = 0$. That is, $F[x, G(x)] \equiv 0$. Then $y = G(x)$, a single-valued function of x , defines some curve in the x - y plane.

Consider a portion, S , of this curve (where $a \leq x \leq b$) on which $f_p(x, y) = F(x, y) = 0$. (There must be at least one f_p for which such a region can be found.) On S ,

$$f_p(x, y) \geq f_1(x, y), f_2(x, y), \dots, f_n(x, y)$$

Select an x_0 in the interval $[a, b]$. Then on S , $y = g_p(x_0)$.

Consider any other function evaluated at the same point, $y = g_q(x_0)$. Then, by definition,

$$f_p \left[x_0, g_p(x_0) \right] = 0 = f_q \left[x_0, g_q(x_0) \right]$$

But

$$f_p \left[x_0, g_p(x_0) \right] \geq f_q \left[x_0, g_p(x_0) \right] \quad a \leq x_0 \leq b$$

because of the way in which region S was chosen. Therefore,

$$f_q \left[x_0, g_q(x_0) \right] = f_p \left[x_0, g_p(x_0) \right] \geq f_q \left[x_0, g_p(x_0) \right]$$

But f_q is monotonically increasing in its second variable. Hence, $g_p(x_0) \leq g_q(x_0)$ for any q , $a \leq x_0 \leq b$

or

$$y = G(x) = g_p(x) = \left[g_1(x), g_2(x), \dots, g_n(x) \right] \phi^{-1} \quad a \leq x \leq b$$

This argument can be repeated for each interval of x over which $F(x, y) = f_j(x, y) = 0$, until the complete x -axis is covered. This proves the theorem.

APPENDIX II

PROOFS OF THE THEOREMS OF SECTION IV

THEOREM 1. A driving-point function that contains $2^n - 1$ breakpoints requires at least n diodes for synthesis.

PROOF. $2^n - 1$ implies 2^n linear regions (or states of the network) which must occur as the driving current or voltage is varied from $-\infty$ to $+\infty$. Since 2^n is exactly the number of different states of n diodes, the only thing left to show is that no state can appear more than once. This has been shown elsewhere (19) and will not be repeated here.

THEOREM 2. One diode per breakpoint is a necessary and sufficient number to synthesize any strictly concave or convex driving-point function.

PROOF. The sufficiency part of the theorem is proved by a construction procedure given in Section V. Necessity is proved here.

Consider a strictly concave driving-point impedance that contains n breakpoints, which is associated with a network containing N diodes. (The proof is of the same form for convex functions.) The impedance function is to be examined by increasing the driving-point current monotonically from $-\infty$ to $+\infty$. Assume that at some highly negative current the network is in its first linear state, in which M of the total of N diodes in the network are closed. ($M \leq N$). Since it has been assumed throughout Section IV that only one diode switches at a time, each time we pass through a breakpoint in increasing the driving current, one of the M diodes must switch from closed to open. (An increase in incremental resistance implies an opening of a diode.) Therefore $M \geq n$ and hence $N \geq n$.

THEOREM 3. One diode per breakpoint is necessary and sufficient to synthesize any driving-point function in a series-parallel development.

To proceed with the proof of this theorem it is necessary first to define a series-parallel network or graph. Two equivalent definitions will be given (21):

- DEFINITION A.**
1. A single branch is a series-parallel graph.
 2. A parallel combination of two series-parallel networks is series-parallel.
 3. A series combination of two series-parallel networks is series-parallel.

This inductive definition is convenient for certain purposes and will be used in the proof to follow, but it has the disadvantage of failing to indicate a method of determining

whether or not a given graph is series-parallel. For this purpose a second definition is more appropriate.

DEFINITION B. Select any two nodes, \underline{a} and \underline{b} , in a given graph as driving-point nodes. Trace all possible paths through the graph from node \underline{a} to node \underline{b} , so that in any one path no node is traversed more than once. Now if all paths that pass through a given branch of the graph traverse it in the same direction, the graph is series-parallel.

As an illustration, definition B can be applied to a simple bridge network. Passing from the top node to the bottom node of the bridge, the detector arm can be traversed in either direction; hence the bridge is non-series parallel.

PROOF. The sufficiency part of theorem 3 is proved by a construction procedure that is given in Section V. In order to prove the necessary part of the theorem, it is sufficient to show that in a series-parallel network each diode can change state only once as the driving-point current or voltage is increased monotonically from $-\infty$ to $+\infty$. Then the total number of breakpoints of the function must equal the number of diodes that have changed state, which is equal to or less than the total number of diodes in the network.

To prove that any diode can change state only once in a series-parallel network, it is sufficient to show that the current through it or the voltage across it must be a monotonic function of the driving-point current or voltage. This is shown as follows.

The driving-point impedance or admittance of a diode network is a monotonically increasing function. Assume that the driving-point current (and voltage) is increased monotonically from $-\infty$ to $+\infty$. Now, using definition A, we can divide the network into either: 1. the series combination of two series-parallel networks, or 2. the parallel combination of two series-parallel networks. In case 1 we observe that the current through each of the new networks is again monotonically increasing, and in case 2 the voltage across each new network is monotonically increasing. We now have two new (and smaller) series-parallel networks, each of which is excited with a monotonically increasing driving-point current (and voltage). Now the subdivision procedure can be repeated on each of the new networks, and continued until the original network is resolved into a number of indivisible series-parallel networks (single branches) each of which has a monotonically increasing current (and voltage) associated with it. Thus any diode in the network, being driven by a monotonically increasing voltage or current, can switch only once.

It should be noted here that the definition of monotonicity used here is one that includes functions which may be constant over some range of the independent variable; that is, $f(x)$ is monotonically increasing if $f(x_2) \geq f(x_1)$ when $x_2 \geq x_1$. The inclusion of the equality admits constants to the class of monotonic functions.

THEOREM 4. Given a resistive diode network the behavior of which at some arbitrary terminal pair is described by $e = z(i)$ or equivalently, $i = y(e)$. If all voltage sources and all resistances are multiplied by the same positive constant, k , then the new impedance and admittance functions are,

$$e = k [z(i)] \text{ and } i = y\left(\frac{e}{k}\right)$$

PROOF. From the analysis procedures given in Section III, it is clear that any driving-point impedance of a diode network can be written as a set of linear elements

$$v_1 + r_1 i, v_2 + r_2 i, \dots, v_n + r_n i$$

operated on by a succession of ϕ^+ and ϕ^- transformations. Each linear element is of the form,

$$V_1 \frac{P_1(R)}{Q_1(R)} + V_2 \frac{P_2(R)}{Q_2(R)} + \dots + V_r \frac{P_r(R)}{Q_r(R)} + I_1 \frac{M_1(R)}{N_1(R)} + I_2 \frac{M_2(R)}{N_2(R)} + \dots + I_s \frac{M_s(R)}{N_s(R)} + i \frac{M_{s+1}(R)}{N_{s+1}(R)}$$

where the V's represent the voltage sources in the network and the I's the current sources. The P's, Q's, M's, and N's, are polynomials in the R's (the resistances) and each term of $P_x(R)$ must be of the same degree as each term of $Q_x(R)$. However, all the terms of $M_y(R)$ must be of the same degree, and one degree higher than those of $N_y(R)$. This follows from dimensional considerations. Therefore, if each V and R is multiplied by k, each of the above terms will be multiplied by k. But in Section II it was shown that if each element in a cascaded set of transformed vectors is multiplied by a positive constant, this constant can be removed and placed outside the transformations. Thus, the new impedance function is $e = k [z(i)]$. Inverting, we obtain

$$z(i) = \frac{e}{k}$$

$$i = y\left(\frac{e}{k}\right)$$

THEOREM 5. Since the proof of theorem 5 is of the same form as that for theorem 4, it will be omitted.

APPENDIX III

TESSELLATION THEOREM

In section 5.3 a method of simplicial subdivision of an n-cube is given which results in $n!$ simplices. Proof of this result and some of its ramifications are given here for a unit n-cube with one vertex at the origin; all coordinates of each point non-negative. No lack of generality results from this choice.

THEOREM. Given the n-cube whose vertices are: $(0, 0, \dots, 0)$, $(0, 0, \dots, 0, 1)$, \dots , $(1, 1, \dots, 1)$. (All coordinates either 0 or 1.) The hyperplanes, $x_1 - x_2 = 0$, $x_1 - x_3 = 0$, $x_2 - x_3 = 0$, \dots , $x_{n-1} - x_n = 0$ divide the n-cube into $n!$ n-simplices that are non-intersecting, except for their bounding surfaces.

PROOF. First, the n-dimensional space corresponding to the variables, x_1, x_2, \dots, x_n ,

will be partitioned into $2^{\lfloor n(n-1)/2 \rfloor}$ disjoint subspaces in the following manner.

Let h_{ij} represent one of the $n(n-1)$ half spaces defined by

$$x_i - x_j > 0 \quad (i \neq j)$$

Now consider the two sequences,

$$0 \rightarrow h_{12}, h_{13}, h_{14}, h_{23}, h_{24}, \dots, h_{n-1, n}$$

$$1 \rightarrow h_{21}, h_{31}, h_{41}, h_{32}, h_{42}, \dots, h_{n, n-1}$$

(The $\underline{0}$ sequence contains all h 's for which $i < j$. The $\underline{1}$ sequence is just the $\underline{0}$ sequence with its subscripts permuted.) A subspace, S_m , is defined as follows: let m be written as an $\lfloor n(n-1)/2 \rfloor$ digit binary number. Define S_m as the mutual intersection of a sequence of $\lfloor n(n-1)/2 \rfloor$ h 's chosen by replacing the p^{th} digit in the binary expression for m by the p^{th} \underline{h} in the first sequence if the digit is zero, and by the p^{th} \underline{h} in the second sequence if that digit is a one. (Each \underline{h} sequence is just $\lfloor n(n-1)/2 \rfloor$ elements long.) There will be $2^{\lfloor n(n-1)/2 \rfloor}$ of these subspaces, corresponding to the $2^{\lfloor n(n-1)/2 \rfloor}$ binary digits. Now it will be shown that all but $n!$ of these S_m 's are null, and they are all disjoint.

First, to show that they are all disjoint, note that $h_{ij} \cap h_{ji} = 0$ (since all points in h_{ij} satisfy $x_i - x_j > 0$; all points in h_{ji} satisfy $x_j - x_i > 0$, and they cannot be satisfied simultaneously). Now consider any two sets, S_p and S_q , $p \neq q$. Since p must differ from q in at least one binary digit, one \underline{h} , say h_{ij} , in the S_p sequence will differ from the corresponding \underline{h} , h_{ji} , in the S_q sequence. But $h_{ij} \cap h_{ji} = 0$, and therefore, $S_p \cap S_q = 0$.

Next, to investigate the nullity of the S 's, the following lemma will be proved.

LEMMA. The necessary and sufficient condition that S_m be null is that there must exist some cyclic subsequence of the h 's that define S_m , of the following form (the order, of course, is immaterial):

$$h_{ia} \cap h_{ab} \cap h_{bc} \cap \dots \cap h_{pq} \cap h_{qi}$$

PROOF. 1. Sufficiency. Adding all the inequalities,

$$x_i - x_a > 0$$

$$x_a - x_b > 0$$

$$\vdots$$

$$x_q - x_i > 0$$

$$x_i - x_i > 0$$

we obtain a contradiction. Thus no points satisfy the conditions and the set is null.

2. Necessity. Suppose S_m corresponds to a sequence of h 's which does not contain any closed cycle as above. The inequalities may be represented graphically as a directed line graph with the following properties:

- (a) The graph contains n nodes corresponding to the n variables.
- (b) Each half space, h_{ij} , is represented by a line segment directed from node i to node j .
- (c) The graph contains $[n(n-1)]/2$ directed paths.
- (d) If the arrow directions are ignored, the graphs corresponding to all the S 's are identical.
- (e) Closed loops on graphs correspond to closed cycles of h 's. (Therefore, the graph of the S_m that is under consideration has no closed loops.)
- (f) The graph has one and only one source. (Since this is a cascade graph, it must have at least one source. But $(n-1)$ lines must diverge from this source, one ending on every other node. Thus, none of the other nodes can be sources.) It has one and only one sink by the same argument.

A unique ordering of the variables can now be established from the graph. The source variable will be the greatest and the sink variable, the least. To find the next greatest and next least, we remove the source and sink and their associated branches. (This amounts to eliminating from consideration all the h 's that contain the source or sink variable.) We now have another graph whose properties are the same as those for the original graph, but it now contains two less nodes. We can again find one source and one sink that correspond to the next greatest and next least variable, respectively. This procedure must be repeated until the complete ordering is established. Any point whose coordinates satisfy the ordering will lie in the set S_m , since it will satisfy the condition set by each h . Hence S_m is non-null, and the lemma is now proved.

To illustrate the graphical procedure just outlined, consider the set (in which $n = 5$):

$$S_m = h_{21} \cap h_{31} \cap h_{41} \cap h_{51} \cap h_{23} \cap h_{24} \cap h_{25} \cap h_{43} \cap h_{35} \cap h_{45}$$

It will be investigated by referring to the flow graph of Fig. 51a. The graph has been

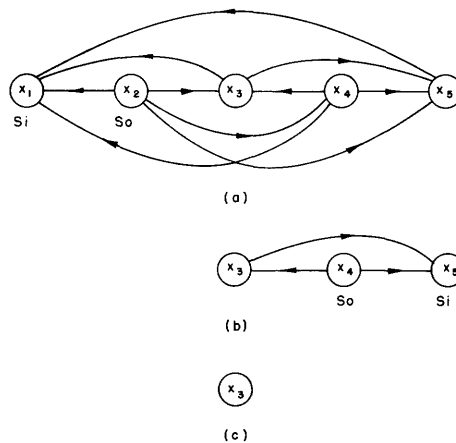


Fig. 51. Flow-graph analog of a set of inequalities.

constructed by the aforementioned procedure. It will be observed that node x_2 is a source (So), and node x_1 is a sink (Si). The process of decomposition of the graph will indicate whether or not it contains any closed loops. Clearly, if any such loops exist, they will still be present after the source and sink and their associated branches are removed, since a closed path cannot pass through a source or sink. Having determined that x_2 is to be ordered first, and x_1 last, our reduced flow graph appears as in Fig. 51b; x_4 is the new source and x_5 the new sink. Removal of these nodes yields only x_3 (Fig. 51c) as the intermediate variable. Thus the complete ordering is

$$x_2 > x_4 > x_3 > x_5 > x_1$$

Obviously, no closed loops exist, and any point whose coordinates satisfy the above ordering lies in S_m ; e.g. $(-3, 7, 0, 1, -1)$. Either by renumbering the nodes or by reversing the directions of some of the branches, new non-null sets corresponding to different orderings of the coordinates can be constructed. In fact, exactly $n!$ non-null sets can be constructed, corresponding to the $n!$ possible permutations of n things taken n at a time.

In order to prove the theorem it remains only to prove that 1. every point in the unit n -cube lies in one of the S 's, and 2. the intersection of each S with the unit n -cube is an n -simplex. Condition 1 is only true if the closure of each S (that is, S plus its limit points) is considered. This merely implies inclusion of the bounding surfaces in the set S , which can be done by redefining the h 's with \geq signs rather than with the strict inequality signs. Therefore the ordering of the coordinates for each S is now defined with the \geq sign rather than the $>$ sign. Now the S 's are mutually disjoint except for their bounding surfaces, and every point in the n -cube must fall within one of them. Note that, once the flow graph of a non-null set has been analyzed, all but a particular set of $n-1$ of the branches can be removed from the graph without changing anything. This is because these $n-1$ branches, which connect each variable to its immediate neighbors in the ordering relation, correspond to the inequalities that include all of the other inequalities.

To prove condition 2 above, let us represent the unit n -cube as follows.

Let h_p^0 be the half space defined by $x_p \geq 0$. Similarly, let h_p^1 be the half space defined by $x_p \leq 1$. Then the n -cube, C , takes the form,

$$C = h_1^0 \cap h_2^0 \cap \dots \cap h_n^0 \cap h_1^1 \cap h_2^1 \cap \dots \cap h_n^1$$

Now to investigate the intersection of the n -cube with a non-null set, say, $C \cap S_m$, we can again represent the bounding inequalities by branches in a flow graph. Figure 52 shows the graph of Fig. 51a with all but the essential $n-1$ branches removed. Two nodes that represent zero and one have been added, together with the $2n = 10$ branches that represent the n -cube boundaries. Again many of the added branches are redundant; these are shown by dashed lines in the figure. The only essential ones are those

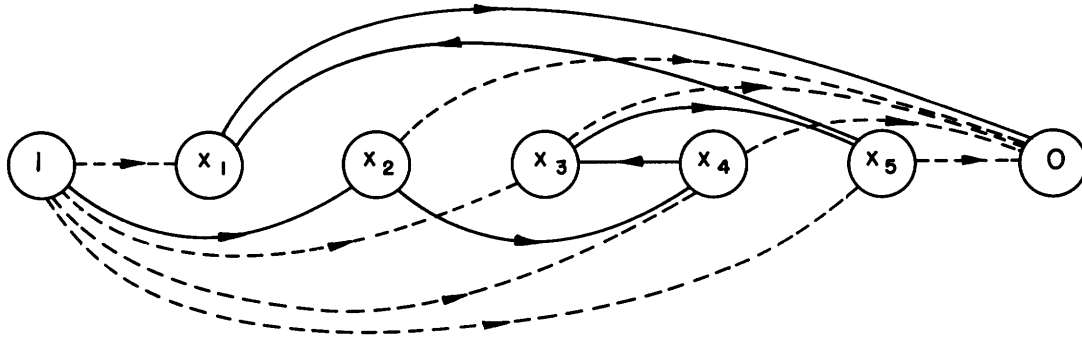


Fig. 52. Flow-graph analog of a set of inequalities.

corresponding to h_2^1 and h_1^0 . The redundant branches correspond to hyperplanes that are not needed to define the set and, therefore, are not bounding surfaces of the set. Thus we see that each non-null set, $C \cap S_m$, is bounded by $n-1$ of the subdivision hyperplanes and by two n -cube faces, a total of $n+1$ bounding surfaces. But according to the definition, $C \cap S_m$ must then be an n -simplex. The theorem is now proved.

The ordering relations derived in the proof are of value in determining the structure and orientation of each n -simplex formed by the subdivision procedure. For illustration, consider an n -space whose coordinates are x_1, x_2, \dots, x_n . Suppose we wish to determine the vertices of a particular n -simplex in the unit n -cube, which is defined by the ordering, $x_1 \geq x_2 \geq \dots \geq x_n$. Since the coordinates of each vertex are all either 0 or 1, the only vertices whose coordinates satisfy the ordering are:

x_1	x_2	x_3	x_4	\dots	x_n
0	0	0	0	\dots	0
1	0	0	0	\dots	0
1	1	0	0	\dots	0
1	1	1	0	\dots	0
		\vdots			
1	1	1	1	\dots	1

As we expected, there are just $n+1$ of these vertices. To find the vertices of any other simplex, the variables at the head of the columns are merely rearranged. It can be observed that the vertex containing all zeros is common to every simplex; each vertex containing one 1 is common to $(n-1)!$ simplices; each vertex containing two 1's is common to $2(n-2)!$ simplices; and so on. In general, each vertex containing m 1's ($m \leq n$) is common to $[m! (n-m)!]$ simplices.

It was stated in Section V that each tabulated vertex in a regular n -dimensional tabulation is common to $(n+1)!$ simplices. (When the simplicial subdivision is as defined

above.) This fact can be deduced from the classification of vertices just discussed. First, we observe that each tabulated point is the common vertex of 2^n n -cubes. It is oriented differently with respect to each n -cube; thus its coordinates, with respect to each adjoining n -cube referred to the origin, run through the complete list of binary numbers; a different number for each n -cube. Now, it was shown that a vertex represented by a binary number containing m 1's is common to $m! (n-m)!$ simplices. But in the list of binary numbers n digits long, $\frac{n!}{m! (n-m)!}$ of them contain m 1's. So the total number of simplices adjoining any tabulated point must be

$$N = \sum_{m=0}^n \frac{n!}{m! (n-m)!} [m! (n-m)!] = \sum_{m=0}^n n! = (n+1)n! = (n+1)!$$

Acknowledgment

The author wishes to express deep gratitude for the invaluable contribution to this work of Professor Ronald E. Scott, formerly of the Massachusetts Institute of Technology; presently Professor of Electrical Engineering at Northeastern University. Besides suggesting the thesis topic, and serving as thesis supervisor during his stay at M.I.T., Professor Scott provided assistance and encouragement in all phases of the author's graduate work.

The author is also indebted to Professors H. J. Zimmermann, T. F. Jones, and S. J. Mason who had the unenviable task of assuming the responsibility of thesis supervision in Professor Scott's absence.

In addition, thanks are due Mr. Samuel Giser for his cooperation in making available the facilities of the Applied Mathematics Group of the Instrumentation Laboratory, M.I.T.

Bibliography

1. G. Birkhoff and S. MacLane, A Survey of Modern Algebra (Macmillan Company, New York, Rev. ed., 1953).
2. H. S. M. Coxeter, Regular Polytopes (Methuen and Company, Ltd., London, 1948).
3. M. Fréchet and K. Fan, Introduction à la Topologie Combinatoire (Viubert, Paris, 1946).
4. P. Franklin, A Treatise on Advanced Calculus (John Wiley and Sons, Inc., New York, 1940).
5. S. M. Ganguli, Introduction to the Geometry of the Fourfold (Calcutta University Press, Calcutta, 1934).
6. P. R. Halmos, Finite Dimensional Vector Spaces (Princeton University Press, Princeton, 1942).
7. G. H. Hardy, J. E. Littlewood, and G. Pólya, Inequalities (Cambridge University Press, London, 1934).
8. D. Hilbert and S. Cohn-Vossen, Geometry and the Imagination (Chelsea Publishing Company, New York, 1952). (Translation of Anschauliche Geometrie.)
9. W. Hurewicz and H. Wallman, Dimension Theory (Princeton University Press, Princeton, Rev. ed., 1948).
10. J. L. W. V. Jensen, Sur les fonctions convexes et les inégalités entre les valeurs moyennes, Acta Math. 30, 175-193 (1906).
11. E. Jouffret, Geometrie à Quatre Dimensions (Gauthier-Villars, Paris, 1903).
12. J. Kehr, S.M. Thesis, Department of Electrical Engineering, M.I.T., 1956.
13. L. W. Massey, An electronic vector magnitude circuit, Quarterly Progress Report, Research Laboratory of Electronics, M.I.T., April 15, 1955, pp. 95-96.
14. H. Meissinger, An electronic circuit for the generation of functions of several variables, Paper presented at the National IRE Convention, New York, 1955.
15. R. I. Morgenstern, Piecewise-planar multiplier, Quarterly Progress Report, Research Laboratory of Electronics, M.I.T., April 15, 1955, pp. 93-95.
16. K. Reidemeister, Topologie der Polyeder und kombinatorische Topologie der Komplexe (Akad. Verlagsgesell., Leipzig, 1938).
17. W. Rudin, Principles of Mathematical Analysis (McGraw-Hill Book Company, Inc., New York, 1953).
18. R. E. Scott, S. Fine, and A. MacMullen, Nonlinear filtering and waveshape multiplexing, Electronics 25, 146 (Dec. 1952).
19. G. S. Sebestyen, Piecewise-linear resistive network synthesis, S.M. Thesis, Department of Electrical Engineering, M.I.T., 1955.
20. D. H. Schaefer, A rectifier algebra, AIEE Technical Paper 54-516, Oct. 1954.
21. C. E. Shannon, An algebra for theoretical genetics, Ph.D. Thesis, Department of Mathematics, M.I.T., 1940.
22. C. E. Shannon, J. Math. Phys. 21, 83-93 (1942).
23. H. E. Singleton, Theory of nonlinear transducers, Sc.D. Thesis, Department of Electrical Engineering, M.I.T., 1950.
24. F. Spada, S.B. Thesis (General Science), M.I.T., 1955.
25. T. E. Stern, Diode network synthesis, Quarterly Progress Report, Research Laboratory of Electronics, M.I.T., July 15, 1954, pp. 85-88.

26. T. E. Stern, Piecewise-linear network theory, Quarterly Progress Report, Research Laboratory of Electronics, M.I.T., April 15, 1955, pp. 89-93.
27. J. J. Stoker, Nonlinear Vibrations in Mechanical and Electrical Systems (Interscience Publishers Inc., New York, 1950).
28. G. R. Welts, Generator for functions of two variables (unpublished).
29. N. Wiener, Mathematical Problems of Communication Theory, Department of Mathematics, M.I.T. (unpublished notes).
30. L. A. Zadeh, A contribution to the theory of nonlinear systems, J. Franklin Inst. 255, 387-408 (May 1953).