Int. J. General Systems
1978, Vol. 4, pp. 179–187

NETWORK SYNTHESIS AND VARELA'S CALCULUS

LOUIS H. KAUFFMAN

Department of Mathematics, University of Illinois at Chicago Circle, Chicago, Illinois, U.S.A.†

(*Received December 13, 1976; in final form May* 18, 1977)

Network models are given for self-referential expressions in the calculus of indications (of G. Spencer Brown). A precise model is presented for the behavior of such expressions in time. The extension of Brown's calculus by F. Varela is then shown to describe behavior invariant properties of these networks. Network design is discussed from this viewpoint.

INDEX TERMS Self-reference, networks, calculus of indications, calculus for self-reference.

"A man sets himself the task of portraying the world. Through the years he peoples a space with images of provinces, kingdoms, mountains, bays, ships, islands, fishes, rooms, instruments, stars, horses, and people. Shortly before his death he discovers that the patient labyrinth of lines traces the image of his face."

Borges.

1 INTRODUCTION

Starting with the notion of distinction, G. Spencer Brown in his book *Laws of Form*[1] develops a calculus of indications through which one may contemplate the genesis of form and the play of paradox. In perfect balance there is no distinction. That which acts and that which is acted upon are one. If this be paradox, then it had best be faced in all of its irreducibility. Brown teases apart this state of condensation until his language is sophisticated enough to mirror the antinomy within itself. In words it is the paradox of self-reference: I am that.

More concretely, Brown allows self-referential expressions in a Boolean algebra and suggests that rather than generating paradox they generate time! In other words, he suggests that such expressions make sense when interpreted as indicating processes occurring in time. This calls up an analogy with electrical circuits with feed-back. The circularities predispose behavior (memory, oscillation, . . .) and paradox never enters. The form of this circularity becomes a symbol not of contradiction, but of a self-sustaining whole with new properties to be observed and appreciated.

Francisco Varela in his paper *A calculus for self-reference*[3] develops an algebraic approach that avoids mention of time or sequence. In this paper we return to the temporal viewpoint. Self-referential expressions will be considered as networks. A network is an interconnected collection of elements (Brownian operators, cells, observers, atoms, . . .) each receiving information from the others. In a balanced state no information moves in the net: delicate poise. A slight disturbance creates conditions of local imbalance through the net. The net preserves itself by correcting these imbalances, but in the process may create further disturbance. We make no *a priori* conditions on how or in what order the balances are restored; a set of transition rules simply states that restoration occurs by some choice again and again. A net whose eventual behavior is independent of such choice is called determined.

Using this description of behavior for a net we are able to discuss the design of modulation as in Chapter 11 of *Laws of Form*[1]. We believe that our approach lends clarity to this aspect of Brown's work. Furthermore, Varela's calculus is intimately related to our approach. His calculus for self-reference provides an algebra for relating nets with the same behavior (Section 5). This relationship is remarkable, particularly since it gives tractable procedures for transforming networks locally, without changing their global properties.

While we hope that the ideas discussed here may be useful in many areas, the networks we use are very special. Composed of Brownian "marks", they are a mathematical abstraction of digital networks consisting entirely of "nor" gates. For digital networks this is not a fundamental restriction since other standard elements are logically equivalent to appropriate combinations of nor gates. Our meth-

†This paper was written in 1976/77 when the author was visiting the Department of Mathematics of the University of Michigan in Ann Arbor, Michigan. The research reported in this paper was partially supported through NSF Grant No. 76–07187.

ods are useful for the synthesis and simplification of hazard-free digital networks.

Nevertheless, I believe that these Brownian networks are potentially valuable in contexts wider than the analysis of digital circuitry. They provide a metaphor for the creation of numbers (counting, modulation, . . .) from processes of indication and self-indication. There is some value in letting Boolean algebra turn on itself.

The author wishes to thank David Solzman, Jerry Swatez, Paul Uscinski and all the members of the Chicago, Laws of Form, group for many conversations.

2 RECOLLECTION OF THE CALCULUS OF INDICATIONS

The calculus of indications[1] is based on one symbol, $\urcorner$. As a shorthand for $\square$, the mark makes a distinction in the plane on which it is scribed. It may also be seen as the name of the outside (unbounded) part of this division *or* as an instruction to change state, to cross from the state indicated within. Thus $\urcorner\;\urcorner = \urcorner$ expresses the redundancy of naming twice, while $\overline{\urcorner|} =$ (unnamed) says that to cross from the marked state is to enter the unmarked state. Hence $\urcorner$ has dual roles of operator and operand. Acting upon itself it generates an arithmetic of forms with initials $\urcorner\;\urcorner = \urcorner$ and $\overline{\urcorner|} =$

One might suspect that this confusion of roles would lead to contradiction. Not so! There is a unique reduction of any concatenated expression to either the marked or unmarked state. Even the simplest equations are sensible under role exchange. For example $\urcorner = \urcorner$ may be read: To cross from the unmarked state is to arrive at the marked state.

This primary arithmetic of forms is then seen to satisfy various general patterns ($\overline{\overline{a|}|} = a, \overline{a|}\,a = \urcorner, \ldots$) and an algebra (basically Boolean) is born. Its axioms are

$$J1.\quad \overline{\overline{a|}\;a|} =$$

$$J2.\quad \overline{\overline{ar|}\;\overline{br|}|} = \overline{\overline{a|}\;\overline{b|}|}\;r$$

(plus implicit commutativity and associativity). The algebra is complete: any arithmetical identity is a consequence of the axioms.

Just as arithmetic examining itself has borne algebra, the algebra may also look inward and find its own equations thus: $f = \overline{f|}$. This equation can be allowed no arithmetic solution without the collapse of indication to void. If we must persist, an extension is required. Brown posits an *imaginary* value for f. In Varela's notation[3] one writes $f = \boxdot$ such that $\overline{\boxdot|} = \boxdot$; the little hook indicates that f reenters its own space. Thus $\boxdot$ is a Boolean counterpart of $\sqrt{-1}$.

Note that from a formal viewpoint, if we set $\overline{\boxdot|} = \boxdot$, then we must sacrifice the rule $\overline{f|}\,f = \urcorner$ since

$$\overline{f|}\,f = \urcorner \Rightarrow \overline{\boxdot|}\,\boxdot = \urcorner \Rightarrow \boxdot\boxdot = \urcorner \Rightarrow \boxdot = \urcorner.$$

Strictly speaking, we would need to sacrifice either $ff = f$ or $\overline{f|}\,f = \urcorner$. A new algebraic investigation is suggested. This has been elegantly done by Francisco Varela.[3] We shall have more to say about his work later.

Yet $f = \overline{f|}$ describes itself and in so doing leads to a temporal interpretation. If marked, it flips to the unmarked state and vice versa, so on and forever. It is a prototype for condensation of active and passive modes. First it names its interior space by reentry; then it becomes an operator and cancels itself, but not quite. Ready to indicate, it jumps up from the void state only to fall, again and again.

For more complicated situations it is helpful to think of Brownian expressions as modifiers of signals or as functions of their variables. For this, a graphical notation due to Brown is convenient.

Let —|— stand for $\urcorner$ as an operator. The left side of the vertical line denotes the interior space of the mark; the right side stands for the exterior space. Lines (leads) going to the left side of a mark designate variables or expressions in the interior space. Leads emanating from the right side indicate spaces where the mark is placed. The examples in Figure 1 should suffice to make this clear.

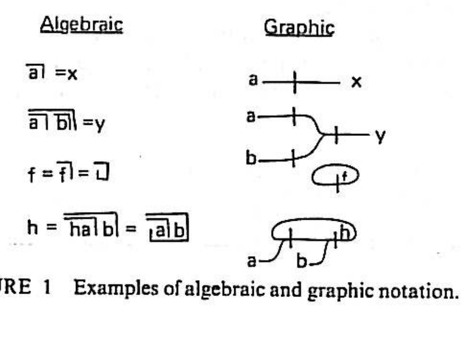
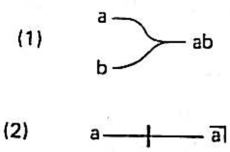
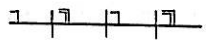


FIGURE 1 Examples of algebraic and graphic notation.

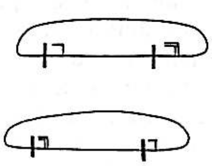
Computations may be performed graphically according to the following two rules:



Thus



corresponds to $\overline{\overline{a}|}| = \overline{a}|$ for $a = \urcorner$. During a computation accounts may not be settled across a mark, but in the end matters must balance according to (1) and (2). Here $f = \overline{f}|$ is never balanced but $g = \overline{\overline{g}|}|$ has two distinct balanced states:



Thus $g = \overline{\overline{g}|}|$ represents a simple memory function. If one considers $h = \overline{\overline{a}|\, \overline{b}|\, h}|$, it is natural to consider the behavior as a and b change. Thus if $a = \urcorner$ and $b = \overline{\urcorner}|$ then $h = \overline{\overline{h}\, \urcorner}| = \urcorner$. What happens if a changes to $\overline{\urcorner}|$? It is clear from the graph that changing a to the unmarked state leaves the expression balanced in state (ii). Hence it remembers $(a, b) = (\urcorner, \overline{\urcorner}|)$.

In the next sections these ideas about balance and transition will be made more precise.

3 BALANCE AND TIME

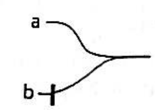
Let N be any network as described in the last section. Suppose that N has n marks; label the right hand sides of these marks by letters $x_1, x_2, \ldots x_n$. Algebraic variables in a corresponding expression correspond to letters $a_1, a_2, \ldots, a_r$ with leads connecting them to the left sides of certain marks.

We make the following assumptions about N:

i) Leads may come together only when leaving or entering a mark or a variable a_i. Thus the vertices of the network are the variables $a_1, \ldots, a_r$ and the marks $x_1, x_2, \ldots, x_n$.

ii) A lead is said to be an entry lead if it begins at the right side of a mark and terminates at the left side of another mark. Only such entries are allowed.

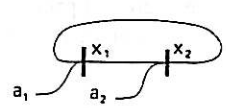
Note that these stipulations forbid certain common nets. For example



is not allowed since these two leads do not terminate at a mark. We make this restriction primarily for ease of description. In fact, we may regard all expressions on the page as standing under an unwritten cross (variously interpreted as the observer or reader). Extending the network in this way and allowing no re-entry from the unwritten cross back into the network on the page, then lets us use networks of the above type.

Let an equation of the form $x_i = \overline{y_1 y_2 \ldots y_k}|$ be associated to each mark in the net. The y_i's denote the exit points of the leads terminating at the mark labeled x_i.

For example, the network below has equations



$x_1 = \overline{a_1 x_2}|$ and $x_2 = \overline{a_2 x_1}|$

Specifying such a set of equations is equivalent to specifying the network.

Suppose that specific values for $x_1, \ldots, x_n$ and $a_1, \ldots, a_r$ have been chosen. We say that the net is *balanced at the ith mark* if these values satisfy the equation for that mark. The net is *balanced* if it is balanced at each mark.

We now seek behavior. Suppose the net is balanced and then some of the a_i are changed and held at their new values. If one imagines that each mark has a certain reaction time to imbalance and that information is transmitted between marks instantaneously, then the behavior may be computed, but it is dependent upon specific choices of reaction time. A discrete analog of this will now be given with the choices made as arbitrary as possible. Every time the net reacts we shall assume that all marks must react within a certain discrete time period. This time period (B below) is arbitrarily chosen. We shall be particularly interested in nets that have behavior independent of such choices.

The transition rules will be given as a program of steps to be carried out. This sequence may terminate in a balanced state, or it may cycle forever. By a *transition* we mean a pair $((a, x), a')$ where $a = (a_1, \ldots, a_r)$, $x = (x_1, \ldots, x_n)$ is a balanced state for the network N and a' is a new choice for some or all of the variables a_i.

Transition Rules: Let a transition $((a, x), a')$ for a network N be given. Suppose that in (a', x) the net is unbalanced at a subset of the marks corresponding to $S \subseteq \{1, 2, \ldots, n\}$.

i) Choose one positive integer B and label each mark with B (call this the *tag* of the mark).

ii) Choose $T \subseteq S$. All marks in S with tag$=0$ must be included in T.

iii) Change all x_i for $i \in T$. (That is, replace x_i by $\overline{x_i}|$).

iv) Set the tags of all marks in T back to B. Subtract 1 from the tags of all marks in $S - T$.

v) There will now be a new subset $S' \subseteq \{1, 2, \ldots, n\}$ at which the net is unbalanced. If S' is empty, *stop* and let x' be the vector of present mark values. If S' is not empty, replace S by S' and repeat steps (ii) to (v).

If a transition $((a, x), a')$ always leads to the same balanced state (a', x') independent of the choices in (i) and (ii) we say that the transition is *determined* and write $(a, x) \rightarrow (a', x')$. A network is said to be determined if every transition is determined.

In undetermined situations, the choices involved may or may not lead to a balanced state. For example, let $f = \overline{\overline{a}|\,a}|$ so that $x_2 = \overline{x_1 a}|$ and $x_1 = \overline{x_2 a}|$ are the network equations. Then $a = \urcorner \Rightarrow x_1 = x_2 = \overline{\urcorner}|$ so that both equations become unbalanced under $a: \urcorner \rightsquigarrow \overline{\urcorner}|$. So $S = \{1, 2\}$. If $T = \{1, 2\}$ then we reset $x_1 = x_2 = \urcorner$ and the net is still unbalanced. If $T = \{1\}$ then we set $x_1 = \urcorner$, $x_2 = \overline{\urcorner}|$ and this is balanced. If $T = \{2\}$ we set $x_1 = \overline{\urcorner}|$, $x_2 = \urcorner$ and again the net is balanced. Thus depending upon the choices, the net can go to either of its two stable states or it can oscillate. In practice, oscillation is very unlikely here since no two marks should be expected to have precisely the same reaction time.

In the light of this last remark, it is interesting to formulate a simpler set of transition rules as follows:

Simplified Transition Rules: Let the notation be the same as that given for the regular transition rules.

i) Choose $i \in S$.

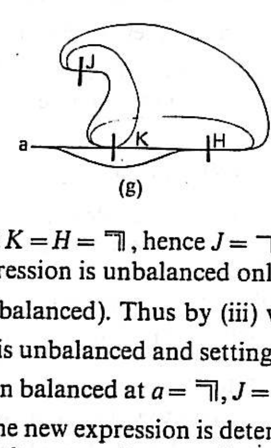
ii) Replace x_i by $\overline{x_i}|$.

iii) There will now be a new subset S' at which the net is unbalanced. If S' is empty, *stop* and let x' be the vector of present mark values. If S' is not empty, replace S by S' and repeat steps (i) and (ii).

Under the simplified rules only one mark resets at any given time. We mention these rules because they provide a very instructive solitaire game that can be played to illustrate network properties. The game is played by placing tokens on the marks of a circuit diagram to indicate marked and unmarked states. A transition can then be played through to its conclusion by simply changing tokens on the game board. Balanced and unbalanced states are apparent to visual inspection.

These definitions apply perfectly well to expressions without re-entry. Such expressions are determined (since the simplification of an arithmetic expression is unique (see [1] p. 14)).

Return to the example $H = \overline{\overline{a}|\,a}|$. The transition $a: \urcorner \rightsquigarrow \overline{\urcorner}|$ is ill-determined. Imagine an observer who prejudices the choice, with appropriate timing. Let the expression be expanded to *include the observer*, thus:



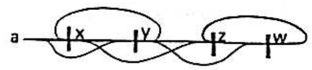
(g)

If $a = \urcorner$ then $K = H = \overline{\urcorner}|$, hence $J = \urcorner$. If $a: \urcorner \rightsquigarrow \overline{\urcorner}|$ then the expression is unbalanced only at H ($J = \urcorner$, $K = \overline{\urcorner}|$ is unbalanced). Thus by (iii) we set $H = \urcorner$. Then only J is unbalanced and setting $J = \overline{\urcorner}|$ leaves the expression balanced at $a = \overline{\urcorner}|$, $J = \overline{\urcorner}|$, $K = \overline{\urcorner}|$, $H = \urcorner$. Thus the new expression is determined.

One can often transform an indeterminate expression to a determined expression by adding new markers who observe the original expression and prejudice the choice in transition.

4 MODULATION

In this section we outline the design of a self-referential expression E that modulates an input. E will have one variable, a, and four balanced states, two for each value of a. As a changes, the expression E will cyclically pass through the four states in a determined manner. We begin with the expression below.



This expression has exactly four balanced states as given by the chart in Figure 2.

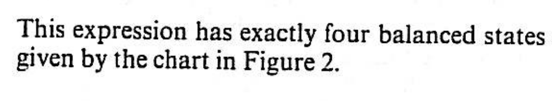
a	x	y	z	w	
⌉	⌉⌉	⌉⌉	⌉	⌉⌉	α
⌉	⌉⌉	⌉⌉	⌉⌉	⌉	β
⌉⌉	⌉	⌉⌉	⌉⌉	⌉	γ
⌉⌉	⌉	⌉	⌉	⌉⌉	δ

FIGURE 2 A list of balanced states.

Let these states be labeled α, β, γ and δ.

As it stands, the expression is not determined. States α and β may fall to either γ or δ when a changes from ⌉ to ⌉⌉. On the other hand, γ does become β when a: ⌉⌉ ⇝ ⌉. Thus we might try to obtain the transition sequence $\alpha \rightsquigarrow \gamma \rightsquigarrow \beta \rightsquigarrow \delta \rightsquigarrow \alpha$.

To accomplish $\alpha \rightsquigarrow \gamma$, expand the expression so that it sees $Z =$ ⌉ (identifying state α): This can be done by a re-entry from Z to the left side of Y.



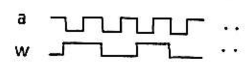
To accomplish $\beta \rightsquigarrow \delta$ we recognise β by $X = Y = Z =$ ⌉⌉, and for $\delta \rightsquigarrow \alpha$ we recognize δ by $X = Z = W =$ ⌉⌉. Thus, encoding these self-observations, we obtain

The mark J recognizes state β and prejudices the transition $\beta \rightsquigarrow \delta$. The mark K recognizes state δ and prejudices $\delta \rightsquigarrow \alpha$.

Hence if we tabulate W as a changes and begin in α, we have

a	⌉	⌉⌉	⌉	⌉⌉	⌉	⌉⌉	⌉	...
w	⌉⌉	⌉	⌉	⌉⌉	⌉⌉	⌉	⌉	...

or



This expression is one of Brown's modulators (see [1] p. 68); our description of its behavior coincides with his.

Note that one really should check this new expression against our description of transitions to see that it indeed behaves properly. This is easily done. (For example, use black and white markers placed on the circuit diagram and play through the transitions.)

It is striking to observe how closely the creation of this modulator parallels one's own experience. For example, try simultaneously beating the rhythms of a and W and note how the process is learned and stabilized by self-observations insuring the desired transitions.

5 NETWORK SIMPLIFICATION AND VARELA'S CALCULUS FOR SELF-REFERENCE

Francisco Varela[3] has constructed a calculus that incorporates the imaginary value □. He begins with arithmetical rules $\overline{□}$ = □, □ □ = □, and ⌉ □ = ⌉ (along with ⌉⌉ = , ⌉ ⌉ = ⌉). Then an algebra consonant with these rules is developed; its initials are

*A*1. $\overline{\overline{p}\,q}\,p = p$

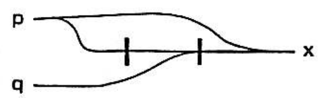
*A*2. $\overline{\overline{pr}\,\overline{qr}} = \overline{\overline{p}\,\overline{q}}\,r$

*A*3. [redacted] p □ = p $\overline{p\,□}$

Certain consequences in the calculus of indications are not allowed here. For example, $\overline{a}\,b$

$=\overline{ab\rceil}\, b$ is false in Varela's arithmetic since $\neg\,\square = \neg$ while $\overline{\square}\rceil\,\square = \square\,\square = \square$. He then assumes that all self-referential expressions satisfy these axioms and he derives certain simplification rules. The calculus for self-reference is complete with respect to the arithmetic including $\square$.

To see the relationship between Varela's calculus and our considerations note that the network for $X = \overline{\overline{p\rceil}\; q\rceil}\; p$ is given by



This may occur inside a larger network as indicated schematically in Figure 3.

*

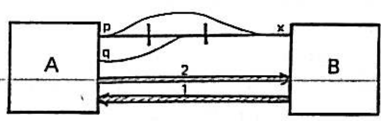


FIGURE 3 Network containing $\overline{\overline{p\rceil}\; q\rceil}\; p$.

This larger network (call it N) consists of two parts A and B with some marks in B (re)entering marks in A (via shaded arrow 1). There may also be other channels from A to B (shaded arrow 2). Suppose that we are analyzing the behavior of N. Then we can reason as follows: X is certainly opaque to changes in q. Hence if q changes during a transition process this change cannot be transmitted along the lines corresponding to $X = \overline{\overline{p\rceil}\; q\rceil}\; p$. Even if q never stops changing it is possible that other parts of the network will go into a useful stable balance. Thus this particular instance of q can be eliminated without changing the behavior of the network in parts A and B. The network N may be replaced by N' as shown in Figure 4.

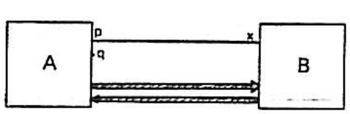


FIGURE 4 $\overline{\overline{p\rceil}\; q\rceil}\; p$ is replaced by p.

Similarly $\overline{\overline{pr\rceil}\;\overline{qr\rceil}\rceil} = \overline{\overline{p\rceil}\;\overline{q\rceil}\rceil}\; r$ allows replacements of type specified in Figure 5.

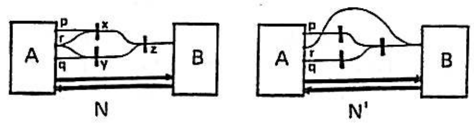
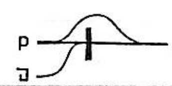


FIGURE 5 Replacement of $\overline{\overline{pr\rceil}\;\overline{qr\rceil}\rceil}$ by $\overline{\overline{p\rceil}\;\overline{q\rceil}\rceil}\; r$.

Here the reasoning is as follows: In the expression on the left a change in r may unbalance both x and y. However, resetting either x or y will have the same effect on z. Thus there is no essential difference between the transition behaviors of the left and right networks.

Case 3 ($A3$) says that if a given line is known to be always off balance (symbolized by $\square$) then



can be replaced by

Thus Varela's calculus gives a system of allowable transformations that can be performed on parts of a network without disturbing the behavior of the rest.

We sum this up in the following theorem.

THEOREM *Let N be a network containing a subnetwork (without re-entries) of the form $f = f(a_1, \ldots, a_r, x_1, \ldots, x_n)$ where f is an expression in primary algebra. If $f = f'$ is an equivalence in the calculus for self-reference, then f' may be substituted for f in the network N without changing the behavior of the rest of N.*

This theorem contains wonderful possibilities for simplifying networks. Its power rests on the completeness of Varela's calculus. One may always refer to the extended arithmetic to see whether a given substitution is valid. Thus the apparently very difficult problem of discovering valid network transformations is reduced to calculations in a 3-valued logic.

Invalid substitutions arise if one makes unrestricted use of Brown's algebra. Thus $g = \overline{\overline{g\rceil}\,a\rceil}\; a$ and

* This is an error. $\overline{\overline{a\rceil}\, b\rceil}\, a$ can transmit signals from b under delay conditions. One algebra that does this job is a transposition algebra with initials:
I) $aa = a$
II) $\overline{\overline{a\rceil}\rceil} = a$
III) $\overline{\overline{a\rceil}\,\overline{b\rceil}\rceil}\, c = \overline{\overline{ac\rceil}\;\overline{bc\rceil}\rceil}$

$g=\overline{\overline{g}\rceil\, a}\rceil$ have different behavior. Yet such substitutions can sometimes be made. If the network has the form shown in Figure 6 and the inputs to f are *opaque to transmission from B*, then we may substitute f' for f whenever $f=f'$ in Brown's algebra.

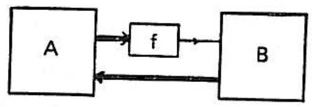


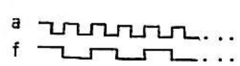
FIGURE 6 Brown's calculus applies under conditions of opacity.

Thus simplification involves recognition of possible substitutions, use of Varela's calculus and/or recognition of opacity and the use of Brown's calculus.

In the next section we shall apply these principles to the design and simplification of Brown's other modulator.

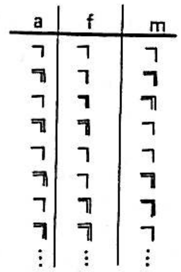
6 ANOTHER MODULATOR

In this section we show how to design a modulator with behavior given by the chart below:



This time the design idea is as follows. The value of f will be stored in a memory of type $f=\boxed{\overline{\alpha}\rceil\,\beta}$. Another memory m will, in conjunction with a, tell f when to change. Note that if $f=\rceil$ we change it by setting $\beta=\rceil$, and if $f=\overline{\rceil}\rceil$ we change it by setting $\alpha=\rceil$.

Let m contain the next value of f. Thus we have the chart given in Figure 7.



a	f	m
⌉	⌉	⌉
⌉⌉	⌉	⌉⌉
⌉	⌉⌉	⌉⌉
⌉⌉	⌉⌉	⌉
⌉	⌉	⌉
⌉⌉	⌉	⌉⌉
⌉	⌉⌉	⌉⌉
⌉⌉	⌉⌉	⌉
⋮	⋮	⋮

FIGURE 7 Modulaιor chart.

The value of f changes whenever a becomes marked. Since we wish f to change to m we must set $\alpha=\overline{\overline{a}\rceil\, m}\rceil$ and $\beta=\overline{\overline{a}\rceil\, m}\rceil$ (refer to Figure 7 to see this). Similarly, the equations for m are $m=\boxed{\overline{\alpha'}\rceil\,\beta'}$ where $\alpha'=\overline{af}\rceil$ and $\beta'=\overline{a\,\overline{f}\rceil}\rceil$.

If, at this point, we translate these equations into a network we obtain the following list of network equations:

$$
\begin{array}{lll}
f=\overline{f'\beta}\rceil & f'=\overline{f\alpha}\rceil & m=\overline{m'\beta'}\rceil \\
m'=\overline{m\alpha'}\rceil & \alpha=\overline{a'M}\rceil & \beta=\overline{a'm}\rceil \\
\alpha'=\overline{af}\rceil & \beta'=\overline{aF}\rceil & a'=\overline{a}\rceil \\
M=\overline{m}\rceil & F=\overline{f}\rceil. &
\end{array}
$$

Thus eleven marks seem to be required. However, simplifications can be made.

In the first place one can see by inspection that f and f' are never both marked and that $f'=\overline{f}\rceil$ (similarly for m and m'). To see this in the context of Varela's calculus, we have the following lemma.

MEMORY LEMMA *Let* $g=\overline{\overline{gX}\rceil\; Y}\rceil$ *where* $X=\overline{ah}\rceil$ *and* $Y=\overline{a\,\overline{h}\rceil}\rceil$. *Let* $g'=\overline{gX}\rceil$. *Then* $g'=\overline{g}\rceil$ *in the calculus for self-reference.*

Proof In what follows we shall use CSR as an abbreviation for the calculus for self-reference. Note that $a\,\overline{h}\rceil\;\overline{ah}\rceil=a\,\overline{h}\rceil$ is a CSR identity. Hence $\overline{\overline{Y}\rceil\;\overline{X}\rceil}\rceil=\overline{\overline{\overline{a\,\overline{h}\rceil}\rceil}\rceil\;\overline{\overline{ah}\rceil}\rceil}\rceil=\overline{\overline{a\,\overline{h}\rceil\;\overline{ah}\rceil}\rceil}\rceil=\overline{a\,\overline{h}\rceil}\rceil=Y$. Thus $\overline{\overline{Y}\rceil\;\overline{X}\rceil}\rceil=Y$. Therefore $gX=\overline{\overline{gX}\rceil\; Y}\rceil\, X=\overline{\overline{gX}\rceil\;\overline{\overline{Y}\rceil}\rceil}\rceil\, X=\overline{\overline{gXX}\rceil\;\overline{\overline{Y}\rceil\;\overline{X}\rceil}\rceil}\rceil$ (using $A2.$ for CSR). Thus we conclude that $gX=\overline{\overline{gX}\rceil\; Y}\rceil=g$. Hence $g'=\overline{gX}\rceil=\overline{g}\rceil$, proving the lemma.

Thus, by the memory lemma, we may eliminate the variables M and F, thereby reducing the list by two marks.

A further simplification results from the CSR identity $\overline{\overline{X}\rceil\; Y}\rceil\, Z=\overline{\overline{XZ}\rceil\; Y}\rceil\, Z$. For $f=\overline{\beta f'}\rceil=\overline{\overline{a'm}\rceil f'}\rceil=\overline{\overline{\overline{a}\rceil m}\rceil f'}\rceil=\overline{\overline{af'}\rceil m}\rceil f'}\rceil=\overline{\overline{\beta' m}\rceil f'}\rceil$. Similarly, $f'=\overline{\overline{\alpha' m'}\rceil f}\rceil$ (using $M=\overline{m}\rceil=m'$). Thus we may replace β by $B=\overline{\beta' m}\rceil$ and replace α by $A=\overline{\alpha' m'}\rceil$. This leads to a new list of network equations as follows:

$$f=\overline{f'B}| \quad f'=\overline{fA}| \quad m=\overline{m'\beta'}| \quad m'=\overline{m\alpha'}|$$
$$A=\overline{\alpha'm'}| \quad B=\overline{\beta'm}| \quad \alpha'=\overline{af}| \quad \beta'=\overline{af'}|.$$

These equations give the eight mark net depicted in Figure 8. It is Brown's other modulator (*Laws of Form*[1], p. 67).

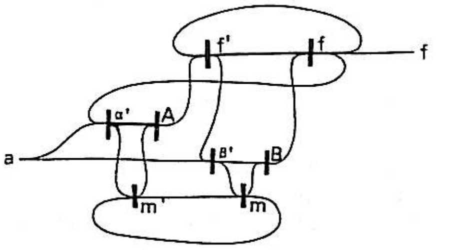
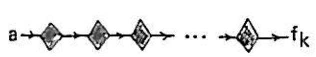


FIGURE 8 Modulator corresponding to Figure 7.

7 OTHER FREQUENCIES

We have exhibited two modulators that cut the incoming frequency in half. There is a standard method for using these to obtain frequency division by powers of 2 (for digital applications the structures we have called modulators are usually called frequency dividers). For example, let f denote the modulator of Section 6 (with input a). Let this circuit with its input a and output f be symbolized by a diamond as follows:

Then

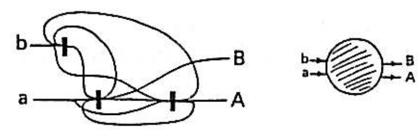


(k diamonds) divides an incoming frequency by 2^k. Its internal states can be used to count (in binary) from 0 to 2^k-1.

Another method of making modulators stems from the special memory network indicated in Figure 9. This network is a modification of the net discussed at the end of Section 3. The behavior of this net is also indicated in Figure 9. When either a or b is marked then the outputs A and B have strictly determined values. When a and b are both unmarked then the expression has two balanced states and $B=\overline{A}|$. Thus the interesting transitions occur when a and b become un-marked. The only undetermined transition is $(a, b)=$(⌉, ⌉)→(⌉⌉, ⌉⌉). This transition is automatically avoided in the class of determined nets that we are about to discuss.

We let the network of Figure 9 be indicated by a disk, and form a network G_n from n of these disks as shown in Figure 10. By using the transitions for the special memory network it is not hard to work out the behavior of G_n.

In doing this analysis, it is important to note the re-setting effect of the feed-back from the last disk to the first. The analysis (which we omit) shows that G_n is a determined modulator. In fact G_2 is identical (as a net it is topologically identical) to the modulator of Section 4. In general, G_n divides frequency by $([n/2]+1)$ for n even, and by $(([n-1/2])+1)$ for n odd. Thus G_6 counts to three, G_8 counts to four, and so on.



a	b	A	B
⌉	⌉	⌉⌉	⌉⌉
⌉	⌉⌉	⌉⌉	⌉⌉
⌉⌉	⌉	⌉⌉	⌉
⌉⌉	⌉⌉	α	$\overline{\alpha}$⌉

States

a	b	A	B
⌉	⌉⌉	⌉⌉	⌉⌉
↓		↓	
⌉⌉	⌉⌉	⌉	⌉⌉
⌉⌉	⌉	⌉⌉	⌉
↓		↓	
⌉⌉	⌉⌉	⌉⌉	⌉

Transitions

FIGURE 9 A special memory network.

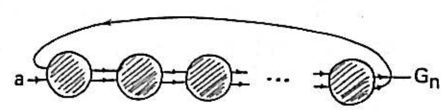


FIGURE 10 A frequency divider.

8 EXTENSIONS

There are a number of lines for further investigation suggested by our study of self-reference. We have, following Brown and Varela, regarded self-reference as a natural extension of Boolean algebra. Our model for behavior is an abstraction of the behavior of digital networks. This invites a more detailed comparison with the existent methods for analyzing such networks.

The major difference between our approach and the usual approaches for digital nets lies in the treatment of time delays. It is common to separate the time delays from the logic as a first approximation to the circuit design. Thus one uses a lumped time delay Δt and considers equations (in standard Boolean algebra) of the form $x(t+\Delta t)=f(a, x(t))$. This leads to an input–output matrix analysis and a corresponding circuit which may contain various hazards (race conditions and so on). The hazards are then found and eliminated on an *ad-hoc* basis.

Our approach avoids this separation of time and logic. Varela's calculus provides a method for global analysis of the network as it actually behaves. Nevertheless, this does not change circuit design from an art to a mechanical exercise. Going from desired behavior to a network realization remains in the domain of invention. Minimization problems are particularly subtle and should be attacked with whatever techniques are available.

There are many other networks (biological, physical, linguistic, geographical, mathematical) that are characterized by a self-referential interdependence of parts and the whole. The behavior is a matter of balance among the parts and the preservation of this balance under internal and external perturbation. But what distinguishes parts from the whole is a matter of choice on the part of an observer. Thus one wants a logic that can deal with the whole independent of any particular decomposition into hierarchies of sub-assemblies. Our admittedly abstract study of self-referential logic may at least suggest some approaches to more concrete problems.

There is one glaring difference that comes to light when one compares our nets to biological systems. Computer networks do not decay. They preserve certain internal states, but the organization is given; the active components of the network are not processed from and returned to the environment. Mathematical modelling of this aspect for Brownian networks might shed light on many real situations.

Another area of inquiry arises by analogy with catastrophe theory.[2] In catastrophe theory one considers generic cases of transitions (discontinuous change) and balance (stability) for systems that have continuous parameters. This has led to the discovery of basic cases (such as the cusp catastrophe) that occur in many situations. The stable states of a system are described by the extrema of a "potential function" $V(a, x)$ where a denotes a set of control parameters and x a set of internal variables. In our situation we consider a Boolean function $f(a, x)$ and the stable states are the fixed points x such that $x=f(a, x)$ for a given choice for a. What is a topology appropriate for studying the qualitative behavior of networks? What networks are "generic" in some fashion analogous to the stable unfoldings of catastrophe theory?

9 EPILOG

A self-referential expression is a description describing itself. That is its form. If we desire its behavior, this description unfolds the time-aspect of the original expression. We have chosen specific rules for the unfolding in time. They involve concepts of balance, and choice in the correction of imbalance. The moment choice enters we have expanded this closed form to include an observer to perform the choice. If, on the other hand, each point of imbalance leads to a predictable conclusion, then we observe a causal frame-work and so call the expression determined.

We define the line between the human observer and the network as automaton. Network synthesis is the process of deciding that boundary. The behavior is our behavior, the choice our own.

REFERENCES

1. G. Spencer Brown, *Laws of Form*. Bantam Books, 1972.
2. René Thom, *Structural Stability and Morphogenesis*. W. A. Benjamin Inc., 1975.
3. Francisco J. Varela G., "A calculus for self-reference". *Int. J. General Systems*, Vol. 2, 1975, pp. 5–24.

Louis Hirsch Kauffman was born in Potsdam, New York, on February 3, 1945. He received his B.A. from the Massachusetts Institute of Technology, and his Ph.D. from Princeton University in 1972.

He has taught at the University of Illinois at Chicago Circle since 1971 and was appointed Associate Professor of Mathematics in 1976. He is currently (1976–1977) on leave at the University of Michigan in Ann Arbor, Michigan.

His scientific activity is centered around problems in differential topology and the knot theory associated with singularities of mappings, and has contributed several articles to these problems.