MAC-M-424

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Project MAC

Computation Structures Group Memo 49

March 27, 1970

Information Flow in State Machines

Anatol Holt and Fred Commoner
Applied Data Research

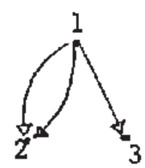
INFORMATION FLOW IN STATE MACHINES

Scope

This paper introduces a concept of information applicable to the analysis of the input/output behavior of state machines - a finite number of transitions and at least one transition path from any state to any state. By interpretation we shall assume that all transitions out of a given state are equi-probable. However, by regarding the transition from one state to another as represented several times we can effectively weight the probabilities of state transitions. For example:

$p_{1\,2} = 1/2$

$p_{1\,3} = 1/2$



$p_{1\,2} = 2/3$

$p_{1\,3} = 1/3$

In this way our information analyses become applicable to ergodic stochastic processes with a finite number of states, assuming that the transition probabilities are expressible as rational fractions.

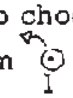
A. Semantic Preliminaries

We shall think of a state machine as a box which holds information. The states will be interpreted as states of box content; the transition as changes in box content. We will introduce an exact calculus which makes it possible to identify the information content corresponding to each state as well as to measure its quantity in a manner consistent with existing measures of information. Every transition can then be described as an input of information, identified both as to kind and quantity and an output of information, identified in these same ways. For particular transitions the input and/or output may be null.

It is natural to say that a state machine about to transit out of a state with several exits must input "information" from an outside source - information which identifies the next transition by which the present state is to be left. Thus the occasions of information input are associated with branch points in the transition diagram. It is equally natural to say that a state machine about to arrive at some state to which there are several entrances must output information to some outside sink - information which identifies the last transition by which the new state was reached. Thus the occasions for information output are associated with merge points in the transition diagram. That information is output at a merge point relates to this: that at a merge point (and only at a merge point) information must be input in order to back up along the path by which the machine stepped forward. That is the information which was taken out of the box as a result of merging.

Simple Examples

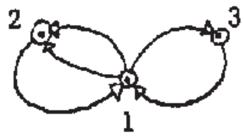
A1.

Exit From	Entrance To	Information Change
1		Input to choose between  and 
	2 or 3	No output since nothing needs to be input to back up
2 or 3		No input
	1	Output to identify the entrance transition

Comment: In this case it is easy to make a semantic connection between the information which is input and the information which is output: upon arrival at state 1 the machine "forgets" which way it branched at the last departure from 1. <u>And we are assuming</u>

that what it forgets is what it outputs. Thus, in our example the machine at state 1 accepts an input, stores the information at states 2 or 3 and puts it out upon return to state 1.

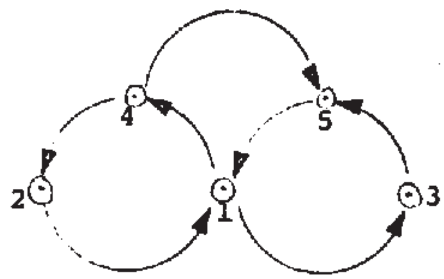
A2.



Exit From	Entrance To	Information Change
1		Input, to choose between 1, 1, or 1
	2	Output, to identify by which transition 2 or 2, state 2 was entere
3	3	no input, no output
	1	output to identify 1 or 1

In this case, part of the information which is input at 1 is output at 2 and another part is output upon return to state 1.

A3.



Comment: In this graph there are two vertices at which there is input - 1 and 4 - and two vertices at which there is output - 5 and 1 . Here it is more difficult than in examples (1) and (2) to see how input information relates to output information. Using the methods of analysis described below the answer to this kind of question becomes unambiguously clear.

B. Formal Preliminaries

B1. We shall analyze the operation of a state machine in regard to the flow of <u>information</u>. We shall restrict our attention to state machines which can be represented by finite, <u>strongly connected</u> graphs where the vertices represent the states and the arcs represent the transitions.

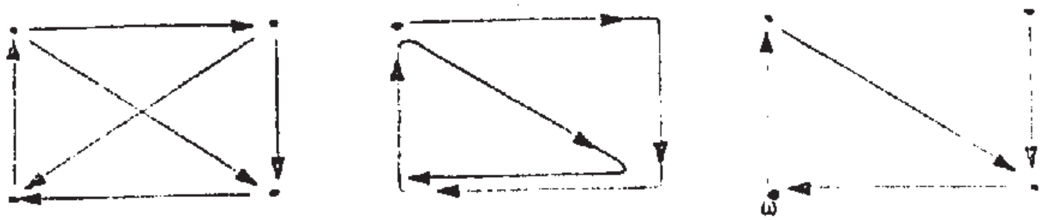
B2. We can think of <u>the history</u> of a state machine S as a path σ in the corresponding graph. We may either wish to draw attention to the state sequence (i.e., vertex sequence) which the path defines, or to the firing sequence (i.e., arc sequence). In either case, we will refer to the path σ and let context make clear which sequence is of interest.

B3. Let P be a finite directed path in a graph S . We can associate with every vertex x on the path P (except, perhaps, the terminal vertex) an arc, called <u>the last exit from x</u> , namely the latest arc on P incident out of x . Then we can call an arc of P a last exit if it is the last exit of some vertex. The last exits of P represent the latest decisions which were made at the vertices of P . A <u>Decision-graph</u> D or simply <u>D-graph</u>

is the set of last exits (E) of P , plus the vertices of these arcs with the terminal vertex of P (ω) distinguished. ω is called the root of the D-graph. We write $D = \langle E,\omega \rangle$.

Examples

(1)

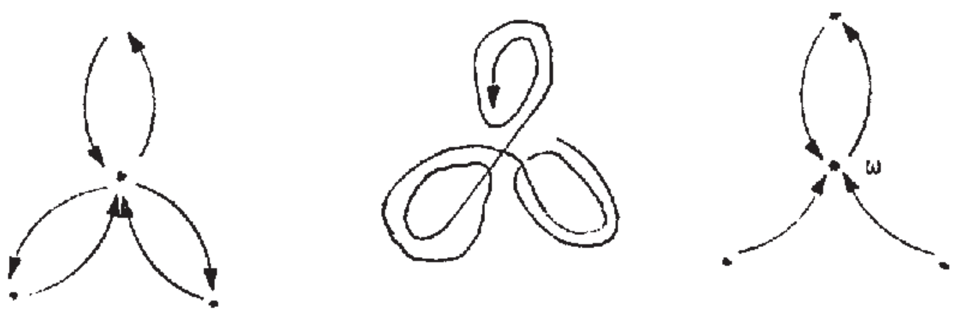


S P D

(2)

S P D

(3)



S P D

We will call a path in S <u>long</u> if it comes to and exits from every vertex of S .

<u>The D-graphs of S</u> are the D-graphs of all long paths of S .

B4. Let P' be a terminal segment of the path P . Then the D-graph of P' must be a subgraph of the D-graph of P .

<u>Proof</u>: This follows because, if a vertex lies on P' as well as P then its last exit in P' must be the same as its last exit in P .

B5. Let $\overrightarrow{P\|\ldots x}$ be a path and D its D-graph. Then, in D , there is a path from every vertex/on P to the vertex x , the root / of

<u>Proof</u>: This is easy to see with an induction on the number of arcs in P . This statement is obvious if P is a single arc.

Now consider a path, Q , with $n + 1$ arcs $(n \geq 1)$. Suppose that the k^{th} arc of Q is arc a from vertex y to vertex z , and that a is the first last exit of Q . If Q' is the terminal segment of Q which begins just after the k^{th} arc of Q , then its D-graph , D' is a subgraph of D , differing from D only in that it lacks vertex y and its exit arc. By inductive hypothesis our assertion is true for D' , and therefore obviously for D .

B6. If the path $\overrightarrow{P|\ldots x}$ never exists from x (i.e, is not of the form $\overrightarrow{|\ldots x \ldots x}$) then the D-graph D of P must be a tree rooted at x .

Proof: By B5, there is a path from every vertex in D to x . Since each vertex has at most one output arc in D , these paths are unique. Thus any circuit in D would have to pass through x -- but x , by hypothesis has no exit arc in D .

Conversely, if P is of the form $\overrightarrow{|\ldots x \ldots x}$ then D must contain a simple circuit, and in fact exactly one -- namely the simple circuit consisting of the last exit arc from x to some vertex y and the unique simple path in D from y to x .

We can now describe the D-graphs of a finite directed graph S . Since the generating paths are "long" (i.e., come into and out of every vertex of S) the resulting D-graph can be visualized as a maximal tree (directed toward the root), rooted at the terminal vertex of the path, together with one arc leaving the root and thus closing a circuit. For the rest of this discussion, D-graph means D-graph of a long path.

B7. Given a finite strongly directed graph S , we wish to show that any maximal directed tree together with one arc out of the root x is a D-graph of S with root x . In other words, given the maximal tree with root x and the additional arc one must construct a path terminating at x with that tree and that arc as its D-graph.

<u>Proof</u>: Let T be a maximal tree in S , x_o its root and α any arc leaving x_o . Now choose any path P_o which begins at x_o , exits x_o for the first time by the arc α , and contains every leaf node of T . This path surely exists since S is strongly connected.

$$\text{Now let } P_o = \overrightarrow{P_o | x_o, x_1, x_2 \ldots x_j}$$

Now define for $0 \leq i \leq j-1$

$P_o = P_o$

$P_i = \overrightarrow{P_i | x_o, x_1, \ldots x_{j-i}}$

(P_i is simply P with the last i vertices removed.)

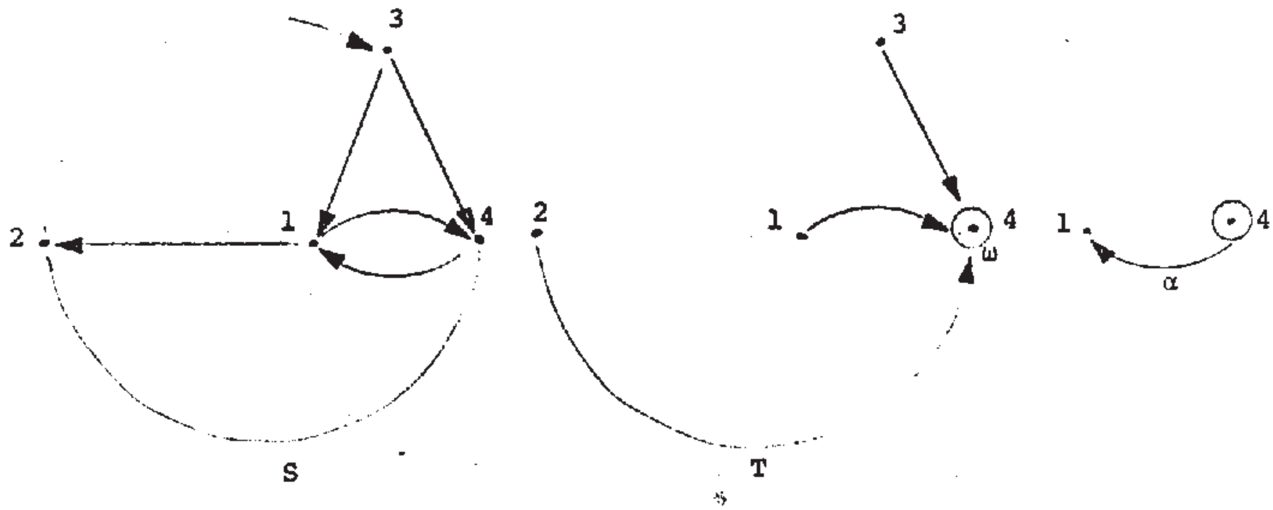
Now since T is maximal there exists a unique path Q in T from every vertex to x_o .

For $\overrightarrow{P_i | x_o, x_1, \ldots x_{j-i}}$ define Q_i to be the unique path in T from x_{j-i} to x_o .

Now we can define the path P which yields a D-graph consisting of T and α rooted at x_o .

$P = P_o Q_o P_1 Q_1 \ldots P_{j-1} Q_{j-1}$

The following is an example of this construction:



$P_o \| 4,1,2,3$

$P_1 \| 4,1,2$

$P_2 \| 4,1$

$P \| 4,1,2,3,4,1,2,4,1,4$

Now we will show that in fact the D-graph D of P is $\langle T + \alpha, x_o \rangle$. P terminates at x_o , thus D is rooted at x_o . The last exit arc in P from x_o is the path P_{j-1} , which is simply the first arc α of P_o , thus $\alpha \in D$.

Because P covers every path from a leaf of T to the root of T, it covers every arc of T. Since T contains every vertex, and since P begins with an exit from the root of the tree, P exits at least once from every vertex of S. Therefore P is long.

If a vertex does not lie on P then every exit from that vertex in P (particularly the last one) lies on a Q path and hence on the tree. If a vertex $y \neq x_o$ does lie on some path P_i then let P_k be the last such path in the order in which the paths are enumerated. Since P_k is the last path containing y, y must be the terminal vertex of P_k. Thus the last exit from y is on Q_k and hence on the tree.

B6 and B7 allow us to state the following theorems:

B8. $D = \langle A, \omega \rangle$ is a D-graph of S if and only if $A = T + \alpha$ where T is any maximal directed tree rooted at ω and α is any exit of ω.

B9. The number of different D-graphs rooted at a vertex x is equal to the product of the number of maximal trees with root x and the number of output arcs of x.

B 10. Let $\overrightarrow{P|\ldots\ldots y,x}$ be a long path in S, and D its D-graph. The arc entering x which lies on the unique circuit of D is the arc in P from y to x.

<u>Proof</u>: Consider the D-graph, D', of the path $\overrightarrow{P'\|\ldots\ldots y}$ with P' identical to P less its terminal arc. D' differs from D in at most one respect: it may contain a different exit arc from y or posibly no exit arc from y. Since P is long, P' exits from x and hence D' contains a single path from x to y which exists unaltered in D. In D, the arc a is a path from y to x and therefore the arc must lie on the unique circuit of D.

B 11. If S is interpreted as a state transition diagram and P as a state sequence for S then one can interpret the D-graph of P as <u>the record of all last decisions</u> made in generating P . The interpretation of B10 is then the following:

The records of all last decisions are sufficient to determine the last step of the state sequence (i.e., last arc of the path) and therefore sufficient to determine the next-to-last state. Put another way: the <u>information</u> for how to back up one step is contained in the record of last decisions.

We now wish to examine the connection between D-graphs and conventional probability measures in state machines. When we say <u>the present D-graph of S is D</u> we assume that the present state of S is the terminal vertex of a long path P and the D-graph of P is D. The <u>steady state probability</u> of a D-graph $D = \langle A, \omega \rangle$ in S is the probability that the present state of S is ω and A is the set of last exits from the vertices of S.

B 12. In the steady state of S, D-graphs are equiprobable:

PROOF:

From S we will construct a new state machine S' whose states are the D-graphs of S. Then we will show that an equiprobable distribution satisfies the steady-state equations of S'. Since these equations have a unique solution,[1] the theorem is proved.

Let S be a state machine at time t with state ω and present D-graph D_1. Clearly ω is the root of D_1. We write $P(D_1 \to D_2) = p$ if p is the probability that D_2 will be the D-graph of S at time $t + 1$.

---

[1]Feller, William. <u>An Introduction to Probability Theory and its Applications</u>, V.I, 2nd. ed. New York: Wiley, 1957, p. 408.

Let the states of S' be the D-graphs of S. The transition probability in S' from state D_i to D_j is defined as $P(D_i \to D_j)$.

Now for any $D = \langle E,x \rangle$ the number of different D-graphs D' for which $P(D \to D') \neq 0$ is simply the number A_x of output arcs of x. Furthermore since we are assuming equiprobable exits in S, these A_x output arcs are equiprobable, hence if $P(D \to D') \neq 0$, $P(D \to D') = \frac{1}{A_x}$.

By B10, the immediately preceding state of S is uniquely determined by the present D-graph of S. Thus, for any D, the D-graphs D_j for which $P(D_j \to D) \neq 0$ are all rooted at a unique vertex ω, which is simply the immediately preceding state determined by D.

Now let Δ be the set of all D_j such that $P(D_j \to D) \neq 0$.

If $D_j \epsilon \Delta$, D_j differs from D only in respect to the last exit from ω. Thus $D \cap D_j$ always contains a tree T rooted at ω. Furthermore any D-graph rooted at ω which contains T is in Δ. Hence Δ is exactly the set of D-graphs rooted at ω which contain T. The number of such D-graphs is the number A_ω of output arcs of ω. That is, $|\Delta| = A_\omega$.

Thus the steady state equations are satisfied if the D-graphs of S are equiprobable. Since their solution is unique, they are satisfied only if the D-graphs of S are equiprobable.

The preceding theorem states that D-graphs are equiprobable in steady state. Since S always has exactly one D-graph, $P(D) = \frac{1}{\Sigma}$ where Σ is the total number of D-graphs of S.

Results of theorem B12.

B 13. Let D_x be the number of D-graphs rooted at x. The steady state probability of x is $\frac{D_x}{\Sigma}$. This

Now define $P(D_j)$ as the steady state probability of D_j in S'. Then the steady state equations for S' are:

For all D

$$P(D) = \sum_j P(D_j)\, P(D_j \rightarrow D)$$

If $P(D_j \rightarrow D) \neq 0$ $\quad P(D_j \rightarrow D) = \frac{1}{A_\omega}$ where A_ω is the number of output arcs of the root of D_j. Furthermore every $D_j \in \Delta$ has the same root ω. Thus we may write:

$$P(D) = \sum_{D_j \in \Delta} P(D_j) \frac{1}{A_\omega}$$

Now assume all D-graph probabilities are equal and thus equal to $P(D)$. Then we have:

$$P(D) = \sum_{D_j \in \Delta} P(D) \frac{1}{A_\omega}$$

However, $|\Delta| = A_\omega$ thus

$$P(D) = \sum_{i=1}^{A_\omega} P(D) \cdot \frac{1}{A_\omega} = A_\omega \cdot P(D) \cdot \frac{1}{A_\omega} = P(D)$$

and we have $P(D) = P(D)$ which is always true.

follows immediately from B12 since S is in state x if and only if its present D-graph is rooted at x.

B 14. The probability of a D-graph D given that ω is the present state is 0 if ω is not the root of D and otherwise $\frac{1}{D_\omega}$ where D_ω is the number of D-graphs rooted at ω.

If ω is the root of D, $P(D) = P(\omega)\, P(D|\omega)$. However $P(D) = \frac{1}{\Sigma}$; $P(\omega) = \frac{D_\omega}{\Sigma}$ thus $\frac{1}{\Sigma} = \frac{D_\omega}{\Sigma} P(D|\omega)$ and $P(D|\omega) = \frac{1}{D_\omega}$.

B 15. Let D_{Ax} be the number of D-graphs rooted at x which contain the set of arcs A. Given the present state x, the probability that the present D-graph includes some set of arcs A is $\frac{D_{Ax}}{D_x}$. $P(A|x) = \sum_D P(D|x)\, P(A|D)$. By B14 $P(D|x) = \frac{1}{D_x}$; $P(A|D)$ is 1 if $A \subset D$ and 0 if $A \not\subset D$.

B 16. Let D_α be the number of D-graphs rooted at the vertex $\dot{\uparrow}(\alpha)$ and containing α on a circuit. (Note that D_α is also the number of D-graphs rooted at $\underset{\cdot}{\uparrow}(\alpha)$ and containing α on a circuit.) The steady state probability that an entrance to x is α where α is some arc leading into x is $\frac{D_\alpha}{D_x}$.

Let $P(\alpha x)$ be the steady state probability that x is entered through the arc α . By B10, if S is at state x , x was entered through α if and only if the present D-graph (rooted at x) is a member of a set containing α on a circuit. By B14, the probability of this set is $\frac{D_\alpha}{D_x}$.

B 17. We can now establish the relationship between state transitions in a state machine and changes in the amount of information contained in the state machine.

Assume S is at state x and transits to state y via arc α . We will associate the exit from x with input and the arrival at y with output.

The amount of information <u>input</u> required to leave x by α is defined as $-\log_2 p(x\alpha)$ where $p(x\alpha)$ is the probability of leaving x by α .

The amount of information which is <u>output</u> upon arrival at y is $-\log p(\alpha y)$ where $p(\alpha y)$ is the probability of entering y through α . This is simply the amount of information which would be required to <u>back up</u> from y along α .

Thus if the initial information content at x is

$-\log I$, the information content $-\log(I')$ after the transition from x to y is

$$-\log I - \log p(x\alpha) + \log p(\alpha y) =$$
$$-\log I\frac{p(\alpha y)}{p(x\alpha)} .$$

For a longer sequence of states and transitions $x_o\alpha_o x_1\alpha_1 \dots x_n$ we have

$$-\log(I') = -\log\left(I\frac{p(\alpha_o x_1)}{p(x_o\alpha_o)} \cdot \frac{p(\alpha_1 x_2)}{p(x_1\alpha_1)} \cdots \frac{p(\alpha_{n-1}x_n)}{p(x_{n-1}\alpha_{n-1})}\right)$$

B 18. Hereafter by <u>information content</u> we will mean the argument of the $-\log$ function rather than the value of that function.

If $x\alpha y$ is a state transition, $p(x\alpha) = \frac{1}{N}$ where N is the number of output arcs of x . By B8 and B9, $D_x = N \cdot T$ where T is the number of trees rooted at x . Furthermore, D_α , the number of D-graphs rooted at x and containing α on a circuit, is equal to T , thus $p(x\alpha) = \frac{1}{N} = \frac{D_\alpha}{D_x}$. Similarly (by B16) $p(\alpha y) = \frac{D_\alpha}{D_y}$. Thus

$$I' = I\,\frac{p(\alpha y)}{p(\alpha x)} = I\,\frac{D_\alpha}{D_y} \cdot \frac{D_x}{D_\alpha} = I\,\frac{D_x}{D_y} .$$

For a longer sequence $x_o, x_1, x_2, \dots x_n$

$$I' = I\,\frac{D_{x_o}}{D_{x_1}} \cdot \frac{D_{x_1}}{D_{x_2}} \cdot \frac{D_{x_2}}{D_{x_3}} \cdots \frac{D_{x_{n-1}}}{D_{x_n}} = I\,\frac{D_{x_o}}{D_{x_n}}$$

In particular for any sequence which is a circuit,

$D_{x_o} = D_{x_n}$ and $I' = I$. This measure allows us to speak of the information difference from state x to state y as $\frac{D_x}{D_y}$, which is independent of the path used in moving from x to y. Note that (by B13) $\frac{D_x}{D_y} = \frac{D_{x/\Sigma}}{D_{y/\Sigma}} = \frac{p(x)}{p(y)}$.

Thus it is consistent to assume that at the state x, $I = \frac{1}{p(x)}$ where $p(x)$ is the steady state probability of x. For if we move from x to y,

$$I' = I \frac{D_x}{D_y} = I \frac{p(x)}{p(y)} \quad \text{and} \quad \frac{I'}{I} = \frac{1/p(y)}{1/p(x)} .$$

B 19. In B10 it was shown that the present D-graph of a state machine uniquely determines the arc by which the present state was entered. For a specific state however, less information than the entire D-graph may be sufficient to uniquely determine this arc (for example a state with only one input arc). In this section we wish to determine exactly what information about last decisions is required to determine the entrance to a given state.

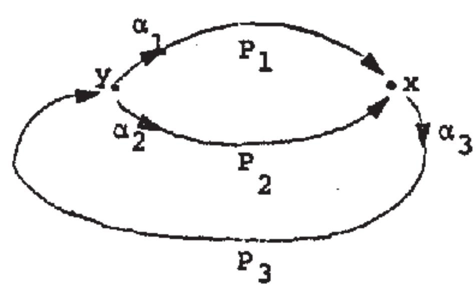
<u>Definition:</u> A <u>partial D-graph</u> of a set of vertices V is a D-graph less the last exits of vertices not belonging to the V set. We write $D(V) = \langle E,\omega \rangle$ for the partial D-graph of the set V rooted at ω where E is the set of last exits from V. (V may be null.)

<u>Definition</u>: Given a state x, $\overset{\nabla}{x}$ is the set of states such that:

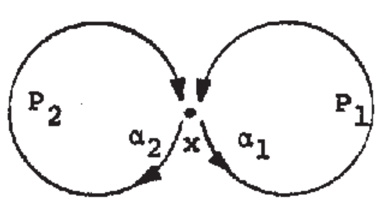
1. Any partial D-graph $D(\overset{\nabla}{x}) = \langle E,x \rangle$ uniquely determines the entry arc of x. More formally, for any partial D-graph $D(\overset{\nabla}{x}) = \langle E,x \rangle$, every D-graph $D = \langle A,x \rangle$, where $E \subseteq A$, has the same arc α entering x on its unique circuit.
2. $\overset{\nabla}{x}$ is minimal with respect to property 1.

B 20. <u>Theorem</u>: For a given state x, $\overset{\nabla}{x}$ consists of all vertices y with the following property:

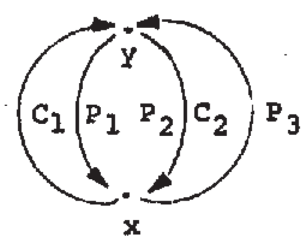
3. There exist two non-empty paths P_1 and P_2 from y to x which intersect only at x and y, and a path P_3 from x to y which intersects P_1 and P_2 only at x and y.



Note: P_3 may be null. This is the case if x itself satisfies property 3:



Proof: First we will show that any partial D-graph of the set of vertices V described in 3 uniquely determines the entry arc of x. To do this we will show that if $\langle A,x\rangle$ is a partial D-graph of V, there cannot exist D-graphs containing A whose circuits enter x by different arcs. Let C_1 and C_2 be two simple circuits which enter x through different arcs. Now traverse C_1 backwards from x. Let y be the first vertex encountered which is also on C_2



Since y is the first such vertex, path $P_1 \subseteq C_1$ from y to x intersects the path $P_2 \subseteq C_2$ from y to x only at the vertices y and x. Furthermore, the path $P_3 = C_2 - P_2$ intersects P_1 and P_2 only at x and y. Thus y satisfies property 3 and hence $y \in V$. Therefore any partial D-graph $D(V) = \langle A,x\rangle$ contains a specific exit arc of y. Since C_1 and C_2 exit y by different arcs, C_1 and C_2 cannot both be contained in D-graphs which contain A.

Now we will show that every vertex y which satisfies property 3 must be in $\overset{\nabla}{x}$. To do this, we will show that there exists a partial D-graph of the set $S - \{y\}$, (where

S is the set of vertices of the graph) , $D(S-\{y\}) = \langle A,x\rangle$ which does not uniquely determine the entry arc of x . This would imply that no set of vertices not containing y could satisfy property 1 of $\overset{\nabla}{x}$.

Assume y satisfies property 3. Let α_1 , α_2 and α_3 be the first arcs, if any, of P_1 , P_2 , and P_3 . Consider the set of arcs $\{P_1 \cup P_2 \cup P_3\}$. Delete from this set all arcs leaving x and the first arc α_1 of P_1 . (See arcs in the above diagrams.)

The remaining set of arcs is a tree rooted at x . Add arcs to this set until it becomes a maximal tree T .

Now define

$$E \overset{\Delta}{=} T - \{\alpha_2\} + \{\alpha_3\} .$$

Then $\langle E,x\rangle$ is a partial D-graph of the set $S - \{y\}$. However, $\langle E+\alpha_1 , x\rangle$ is a D-graph whose unique circuit contains the last arc of P_1 ; $\langle E+\alpha_2 , x\rangle$ is a D-graph whose unique circuit contains the last arc of P_2 . Since these arcs both enter x and cannot be the same, $\langle E,x\rangle$ does not uniquely determine the entry arc of x . Thus $y \varepsilon \overset{\nabla}{x}$.

Q.E.D.

B 21. By definition $\overset{\nabla}{x}$ is a minimal set of vertices whose last exits are always sufficient to determine the entry arc of x . The specification of this arc represents the information which is <u>output</u> when x is entered by the arc. The specification of this arc as a function of the past decisions or inputs at the vertices in $\overset{\nabla}{x}$ suggests that the information output at x is a function of the information input at the states in $\overset{\nabla}{x}$.

In this section we will further explore this connection.

Definition: If x is an arc or vertex, Cx is the set of simple circuits which contain x.

Note that Cx also specifies a class of D-graphs, namely all those D-graphs which contain circuits in Cx.

For each vertex x the Information set Ix is defined as the complement of Cx. Thus a large information set means a large set of excluded circuits and hence a small set of included D-graphs; and therefore a low steady state probability.

For an arc α from x to y,

$$I\alpha \triangleq Cx - C\alpha$$

$$O\alpha \triangleq Cy - C\alpha$$

Thus in the state transition $x \xrightarrow{\alpha} y$

$$Iy = Ix + I\alpha - O\alpha$$

If a circuit c is an element of $I\alpha$, we say c is input at x in the state transition $x \xrightarrow{\alpha} y$. If c is an element of $O\alpha$, we say c is output at y in the state transition $x \xrightarrow{\alpha} y$.

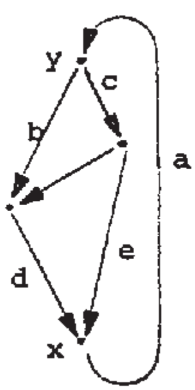
If c is output at y and if c was last input at a vertex z , we say $c(z)$ is output at y .

B 22. Theorem: $y \in \overset{\nabla}{x}$ if and only if there exists a simple circuit c such that c may be input at y and $c(y)$ may be output at x .

$\rightarrow$ If $y \in \overset{\nabla}{x}$ then by B 20 the paths P_1, P_2, and P_3 exist as defined in 3. Thus beginning at y we can move to x via P_1 and <u>input</u> the circuit $c = P_2 \cup P_3$ at y . Since $P_1 \cap c = \{x,y\}$, $c(y)$ is output at x .

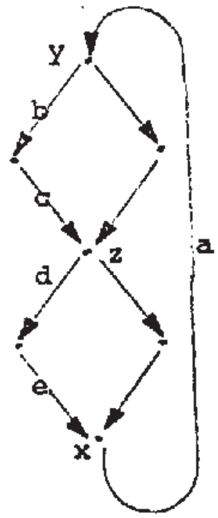
$\leftarrow$ Conversely if c may be input at y and $c(y)$ may be output at x , there exists a path P_1 from y to x such that $P_1 \cap c = \{x,y\}$. Now define P_2 and P_3 to be the paths in c from y to x and x to y respectively. The paths P_1, P_2 and P_3 satisfy requirement 3 for y . Thus by B 20, $y \in \overset{\nabla}{x}$.

<u>Examples:</u>



$y \in \overset{\nabla}{x}$

Both $c_1 = \{a,c,e\}$ and $c_2 = \{a,b,d\}$ may be input at y ; $c_1(y)$ and $c_2(y)$ may be output at x .



$y \not\in \overset{\nabla}{x}$

$C = \{a,b,c,d,e\}$ may be input at y and output at x, however $C(y)$ is output at z and $C(z)$ is output at x.

This means that the decision made at y is output at z and thus has no influence on the entry arc (and hence the output circuit set) at x.

B 23. With theorem B 22 in mind we define $\underset{\nabla}{x}$ to be the set of all states y such that there exists a curcuit c which may be input at x, such that $c(x)$ may be output at y.

Thus, while $\overset{\nabla}{x}$ is the set of states whose last inputs may influence the next output at x, $\underset{\nabla}{x}$ is the set of states whose next outputs may be influenced by the last input at x.

Hence $\underset{\nabla}{x} = \{y \mid x \in \overset{\nabla}{y}\}$.