the same number of states. $\mathfrak{M}_n$ is an example of a maximum memory machine.

Corollary: A maximum memory machine with memory μ exists if, and only if $(1+\sqrt{1+8\mu})/2$ is a positive integer.

Proof: Suppose $(1+\sqrt{1+8\mu})/2$ is a positive integer. Then machine $\mathfrak{M}_{(1+\sqrt{1+8\mu})/2}$ exists and it is a maximum memory machine of memory

$$\frac{1}{2}\left(\frac{1+\sqrt{1+8\mu}}{2}\right)\left(\frac{1+\sqrt{1+8\mu}}{2}-1\right)=\mu.$$

Suppose $\mathfrak{M}$ is any n-state maximum memory machine with memory μ. Then $\mu=n(n-1)/2$ or, solving for n, $n=(1+\sqrt{1+8\mu})/2$, which must be a positive integer.

It is an open question whether or not the existence of a maximum memory machine for any given n is dependent on the order of the input alphabet. In particular, it is not known whether a maximum memory machine with a binary input alphabet exists for every n.

An Improvement on a Theorem of E. F. Moore

GREGORY J. CHAITIN

Introduction

Following Moore [1], consider a sequential machine (discrete, synchronius automata) having n states or less, m inputs, and p-possible outputs. The state that the machine is in at a given time depends only on the state at the previous time and the previous input. The output at a given time depends only on the current state. In addition the machine is strongly connected. That is to say, it is possible to take it from any state to any other state by means of some sequence of inputs. Finally, the machine is in reduced form, which may be taken to mean that it is not possible to design a sequential machine having fewer states whose behavior, in so far as its inputs and outputs are concerned, is identical with the behavior of the original machine. As a gedanken experiment, one may attempt to determine the table of state transitions and outputs of this machine (to within a renaming of its states) by applying inputs and observing the corresponding outputs. It is assumed that one does not just open up the machine and trace its circuits. What will be the length of this gedanken experiment? That is, how many inputs are needed?

In [1] it is conjectured that the best upper bound on the length of this gedanken experiment on sequential machines is independent of the parameter p. Theorem 9 [1] yields an upper bound of

$$\left\{\frac{n(n-1)}{2}+2n-1\right\}\frac{n^{nm}p^n}{n!}\cdot$$

A simple algorithm is given here for effecting the gedanken experiment by means of which an upper bound of

$$\left\{2n-1+\frac{n^2-1}{n-2}\right\}(n-1)!(n-1)^{(m-1)n+2}+(n+1)\sum_{i=1}^{n-4}i!$$

is established. The latter bound is independent of p, and the ratio of the latter to the former approaches zero as m increases with n held fixed. The cardinality of the set of possible outputs is immaterial to the effectiveness of the procedure here given; it can be infinite. The record is given of a gedanken experiment for which $n=6$ and $m=3$ that was performed according to the algorithm given here.

Manuscript received February 7, 1964; revised February 8, 1965. The research reported here was awarded an A.A.A.S. Grant by the American Institute of the City of New York.

The author is located at 819 Madison Avenue, New York, N. Y. He was formerly with the School of Engineering and Applied Science, Columbia University, New York.

Preliminary Results

The following is an immediate corollary of Theorem 9 [1].

Corollary: If $R_{n,m}$ is the class of all strongly connected machines in reduced form having no more than n states and the inputs $a_1, a_2, \cdots, a_m$, then there exists a simple experiment of length at most

$$\left\{\frac{n(n-1)}{2}+2n-1\right\}\frac{n^{n(m+1)}}{n!}$$

which, when performed on a copy of any member S of $R_{n,m}$ will suffice to distinguish S from all other members of $R_{n,m}$.

Proof: By Theorem 9 [1], if $R_{n,m,n}$ is the class of all strongly connected machines in reduced form having no more than n states, the inputs $a_1, a_2, \cdots, a_m$, and the possible outputs $b_1, b_2, \cdots, b_n$, then there exists a simple experiment of length at most

$$\left\{\frac{n(n-1)}{2}+2n-1\right\}\frac{n^{nm}n^n}{n!}$$

which, when performed on a copy of any member S of $R_{n,m,n}$ will suffice to distinguish S from all other members of $R_{n,m,n}$. We can use the same algorithm given in the proof of Theorem 9 [1] for performing this experiment in performing the experiment previously mentioned; it is only necessary to follow these rules.

1) As each output is emitted by the machine being experimented upon, if it was (or is) the ith different output to be emitted in the course of the experiment, consider it to be instead b_i.
2) When the algorithm given in the proof of Theorem 9 [1] for distinguishing a member of $R_{n,m,n}$ from all other members of $R_{n,m,n}$ produces the conclusion that the machine experimented upon has the output b_i when in state j, conclude instead that the machine experimented upon has, when in state j, the ith different output that was actually observed to be emitted by the machine being experimented upon in the course of the experiment.

The same kind of reasoning may be applied in order to improve on Theorem 5.10 [2].

The Algorithm

The basic procedure for conducting the gedanken experiment is as follows: a sequence of inputs is generated which determines that only certain machines with specific initial states that are members of $R_{n,m}$ can be the machine being experimented upon. Each of these possible machines is taken through each of its (at most) nm-state transitions by this first sequence of inputs. The second part of the procedure consists, since the final states are determined, of experiments of (at most, by Theorem 7 [1]) length $2n-1$ distinguishing pairs of these states until only one is left.

A more detailed description of the first part of the procedure follows. Suppose the sequence of inputs generated by this part of the procedure is given. The steps by which it was generated will be apparent from a description of a method of analysis of the sequence and the corresponding sequence of outputs in order to determine which of the machines in $R_{n,m}$ is the machine being experimented upon. The state any such machine is in at time 1 may be arbitrarily numbered 1. There are two possibilities for the state at time 2; either the previous state 1, or any other state, which may be arbitrarily numbered 2, and so forth and so on, as the analysis proceeds states are assigned for each time interval. To get the state for any next time interval, it is either necessary to assign arbitrarily some state that has emitted the same output symbol, a new state (subject to the restriction that there cannot be more than n states), or it may be that this state transition has already occurred, and thus the next state is determined. When the next state is determined by the fact that the transition has occurred before, it is possible that the output that was previously observed to be emitted by the machine when in this state is not the one now observed. Thus, the state assignment turns out not to be a possible one any longer. In this case, go back to the last time interval for which a state was arbitrarily assigned, and assign another one, if there are any more left to be so assigned. If there are not, go back to the time interval before that for which a state was arbitrarily assigned, etc. This procedure of systematic analysis will eventually generate all the possible state assignments for the given sequences of inputs and outputs. From such, tables of state transitions and outputs are derived.

The process by which the first half of the experiment is performed is like this; but, at any time the procedure supplies inputs that take the hypothetical machine of the moment (the machine partially defined by the state assignments for the previous time intervals) through new state transitions.

Bounds on Lengths of Experiments

The number of inputs necessary to take—or try to, if it turns out while attempting to do so that the machine cannot be the one being experimented upon—any hypothetical machine through all its (at most) nm-state transitions is certainly not greater than $n+1$ multiplied by nm, for from any state, it is always possible to go to any other in n-time intervals, and another time interval must be spent in observing the output from the state the machine goes into as a result of this new state transition.

In addition, the number of machines for which this process must be gone through is certainly not greater than the number of possible arbitrary state assignments, which is

$$2\cdot 3\cdot 4 \cdots (n-1)\cdot n\cdot n\cdot n\cdot n \cdots n\cdot n = n!n^{(m-1)n+1}.$$

It is possible to decrease this by noting that unless the machine being experimented upon always gives the same output,[1] at any time when a state must be assigned, knowledge of the output from the machine being experimented upon must eliminate at least one state as a possibility. Thus we get

$$2\cdot 3\cdot 4 \cdots (n-2)\cdot(n-1)\cdot(n-1)\cdots(n-1) = (n-1)!(n-1)^{(m-1)n+2}.$$

Combining all the preceding results, we get an upper bound of

$$\{(n+1)mn + 2n - 1\}(n-1)!(n-1)^{(m-1)n+2}$$

for the length of the experiment.

This estimate will now be substantially improved. Although

$$(2n-1)(n-1)!(n-1)^{(m-1)n+2}$$

inputs may be necessary for the second part of the experiment, in the first part, there will be a great deal of overlapping in work due to the fact that the hypothetical machines "branch out" from each other, and such machines are often first dealt with after a great many of their state transitions have been gone through. The result of this consideration is to yield an upper bound of

$$(2n-1)(n-1)!(n-1)^{(m-1)n+2} + (n+1)\left\{\sum_{i=1}^{n-1} i! + (n-1)! \sum_{i=1}^{(m-1)n+2} (n-1)^i\right\}$$

which is equal to the bound mentioned in the Introduction of this paper. Q.E.D.[2]

It is easily seen that this bound has the properties ascribed to it in the Introduction. A numerical comparison may be of interest; for $m=2$, $n=4$ (and $p=2$), the upper bound given by Theorem 9 [1] is about 6×10^5 inputs, while this bound is about 6×10^4. An empirical investigation of this case suggests that such experiments probably never require more than 10^3 inputs.

[1] If this is the case, it will become apparent after $(n-1)!(n-1)^{(m-2)n+2}$ hypothetical machines fail to get to a state having a different output, but this just means that the experiment must be over so much the sooner.

[2] It is possible to do more with the idea of this proof than was done here.

An Example

An IBM 7094 Data Processing System was used to carry out the first part of the algorithm on the following machine which was initially in state 5, given only the information that the machine being experimented upon was in $R_{6,3}$. The following sequence of 69 inputs was successively generated:

Present State	Next State If Present Input Is 1	2	3	Present Output
1	3	3	4	1
2	5	1	1	1
3	2	2	6	1
4	1	5	3	2
5	6	5	4	2
6	6	2	4	2

112113121221231213133132123212133213333323333232211
332213331132213231

and the sequence of outputs was

222122211111121111221222111111222221221112121111 2
221111221122222222.

To complete the gedanken experiment, all but one of not more than 50 known machines with specified initial states would have to be eliminated in part two (this would require no more than 550 additional inputs). But for $n=6$ and $m=3$, the upper bound derived here is about 10^{13}. Thus it seems likely that it can be considerably improved upon.

References

[1] Moore, E. F., Gedanken-experiments on sequential machines, in *Automata Studies*, Princeton, N. J.: Princeton Univ. Press, 1956.
[2] Gill, A., *Introduction to the Theory of Finite-State Machines*, New York, McGraw-Hill, 1962.

Autonomous Clocks in Sequential Machines

S. S. YAU, member, ieee

Abstract

A necessary and sufficient condition for the existence of an autonomous clock in a sequential machine M is found to be the existence of a nontrivial input-independent partition on the set of internal states of M, no matter whether M is completely specified or incompletely specified. Two different techniques are given for generating the smallest input-independent partition π^I, from which all other input-independent partitions can be obtained. One is suitable for a sequential machine whose state behavior is specified in the form of a flow table, while the other is convenient for a sequential machine whose state behavior is specified in the form of a connection matrix. Both techniques are efficient, and give all possible assignments to the redundant conditions of an incompletely specified sequential machine to reach the same nontrival input-independent partition, and hence the same autonomous clock.

I. Introduction

In the design of a deterministic synchronous sequential machine, by using proper state assignment, there may exist a component machine whose state behavior is independent of the external inputs and

Manuscript received September 3, 1964; revised February 1, 1965. The work reported here was supported by the U. S. Air Force Office of Scientific Research.
The author is with the Department of Electrical Engineering, Northwestern University, Evanston, Ill.