

THE UNIVERSITY OF MICHIGAN
COLLEGE OF LITERATURE, SCIENCE, AND THE ARTS
Communication Sciences Program

Technical Report

TAPE MACHINE REALIZATIONS OF COMMUTATIVE-REGULAR EVENTS

Richard Laing

ORA Projects 03105 and 07363

under contract with:

U. S. ARMY RESEARCH OFFICE (DURHAM)
GRANT NO. DA-ARO(D)-31-124-G665
DURHAM, NORTH CAROLINA

and

DEPARTMENT OF THE NAVY
OFFICE OF NAVAL RESEARCH
CONTRACT NO. Nonr-1224(21)
WASHINGTON, D.C.

administered through:

OFFICE OF RESEARCH ADMINISTRATION ANN ARBOR

September 1965

ABSTRACT

In this paper various infinite tape machine realizations of classes of commutative-regular events are explored. In particular, a class of events which requires for machine realization an infinite-state machine with an infinite number of final states, and a class, somewhat more intractable, which requires an infinite state machine with an infinite number of final states, and which in addition is not strongly connected, are considered. Procedures for constructing deterministic machines with infinite counter tapes, which realize the events under examination are given.

Table of Contents

I.	Introduction	1
II.	Integer Lattice Representation of Commutative Events	5
III.	Extended Forms of CREA, CREB, CREC	12
IV.	Tape Machines	18
V.	TM for Events of the Form CREA	23
VI.	TM for Events of the Form CREB	26
VII.	TM for Events of the Form CREC	39

Tape Machine Realizations of Commutative-Regular Events

I. Introduction

In [LW] the automaton behaviors denoted by expressions of the form $C(RE)$ (where RE is any regular expression, and where C is an operator having the property that applied to any RE , for any word w in the set of words denoted by RE , C yields the words of all permutations of the letters of w) are investigated. A "quasi-normal form" for expressions denoting these commutative-regular events was obtained; all commutative-regular events can be denoted by expressions consisting of a finite union of constituents of the following kinds: $C(w_1)$, $C(w_2^*)$, $C(w_3w_4^*)$, $C(w_5^*w_6^*)$, $C(w_7w_8^*w_9^*)$. That is, the constituents are expressions denoting sets of the words of all permutations of the letters of words of sets given by: a single word (or by extension, a finite set of words), a word to which is applied the Kleene star operator (perhaps prefixed by a single word), or two concatenated words each starred (and perhaps prefixed by a single word). The important features of this quasi-normal form are that the maximum star-height of any constituent is 1 (the commutativity

leading to collapsing of the effect of any repeated star applications), and the maximum "star-length" (numbers of concatenated starred words in a constituent) is 2 (we are here speaking of words constructed on an alphabet of 2 letters; for words in an alphabet of 3 letters, star-length is 3, etc.; in this paper, unless otherwise stated, we confine ourselves to the case of words constructed on an alphabet of two letters.).

Being interested in machines which accept sets of words of the sort denoted by each of the constituents of the quasi-normal form we find that the required machines present a considerable range of complexity, namely:

0. Finite state machines accepting a finite set of words.
- 0'. Finite state machines accepting an infinite set of words.
1. Infinite state machines, accepting an infinite set of words, the machine having a finite number of final states
2. Infinite state machines accepting an infinite set of words and having an infinite number of final states, the machine also being strongly connected

3. Infinite state machines accepting an infinite set of words,
the machines having an infinite number of final states, and
not strongly connected.

These machines are described in [LW] and their construction is discussed in some detail (along with results on some of the algebraic and graph-theoretic properties of the transition systems of these machines and the relationship between commutative-regular events and the Presburger number-theoretic logical formalism from which several decidability results for commutative-regular events and associated machines follow).

The present paper is mainly concerned with the realization by tape machines (TM) consisting of a finite-state read-head and one or more infinite storage tapes, of the infinite-state commutative-regular machines (CM) mentioned in 1, 2 and 3 above. The associated forms of CRE (for "commutative-regular expression", or, also, when confusion is not likely to result, the "commutative regular event" denoted by the expressions) are, respectively:

- A. $C(w^*)$ (where w is any word in two letters, e.g., $C((ab)^*)$).

This form will be called a CREA.

B. $C(w_1^*w_2^*)$ (where w_1 is a word in one letter and w_2 is a word in two letters, e.g., $C((aa)^*(ab)^*)$). This form will be called a CREB.

C. $C(w_3^*w_4^*)$ (where both w_3, w_4 are words in two letters, and with the additional condition that the ratios of the kinds of letters in the two words are not the same -- otherwise it becomes a CREA, e.g., $C((aab)^*(ab)^*)$). This form will be called a CREC.

All of the sets of words under consideration are recursive and thus "recognizable" by some suitable Turing machine. The goal of this present research has been to obtain TM which appear to have rather "natural" structure and behavior and which operate in "real-time" in the sense that acceptance or not, of an input word is registered immediately upon completion of the basic machine operation which results from the reading of the final letter of the input word.

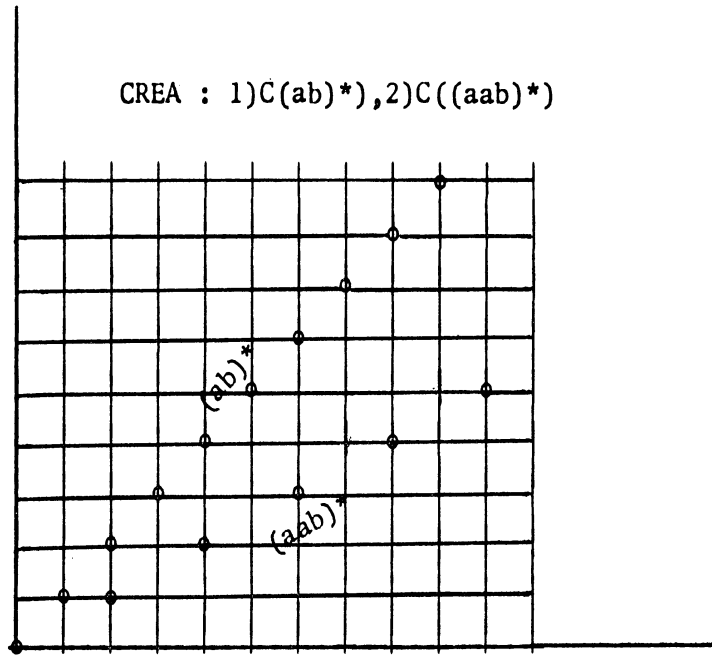
Much of the first four sections as well as parts of the later sections have been explicitly or implicitly presented in [LW].

II. Integer Lattice Representation of Commutative Events.

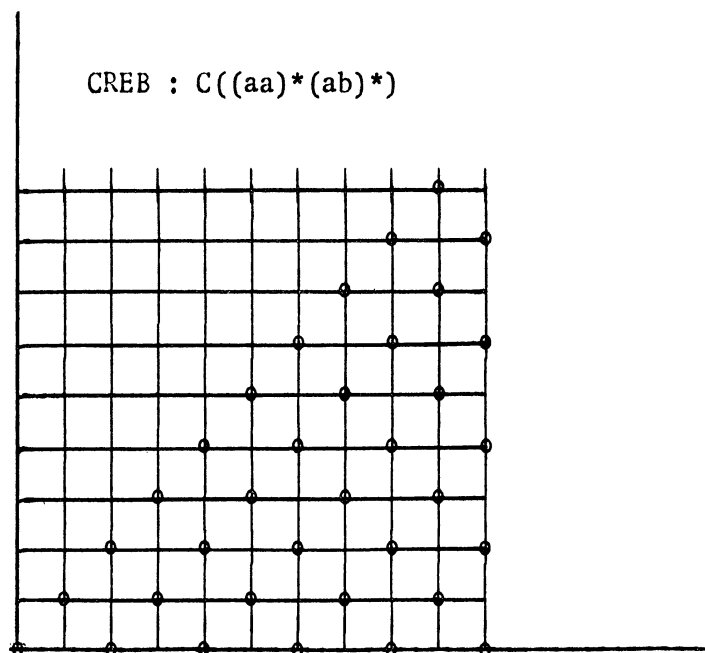
The discussion will be greatly facilitated by introducing a geometric representation of the commutative events under consideration.

Commutative events can be aptly represented as designated points in a first quadrant integer lattice of points L . Letting the x -axis be the a axis, and the y -axis the b axis, the designated point $2,2$ for example, denotes all words of length 4 which consist of all permutations of 2 a s and 2 b s, viz. the set of words {aabb, abab, bbaa, baba, abba, baab}. Thus any designated point x,y will denote the set of words or cardinality $N = \frac{(x + y)!}{x! \cdot y!}$ consisting of all permutations of words of length $x + y$, there being in each word x a s and y b s.

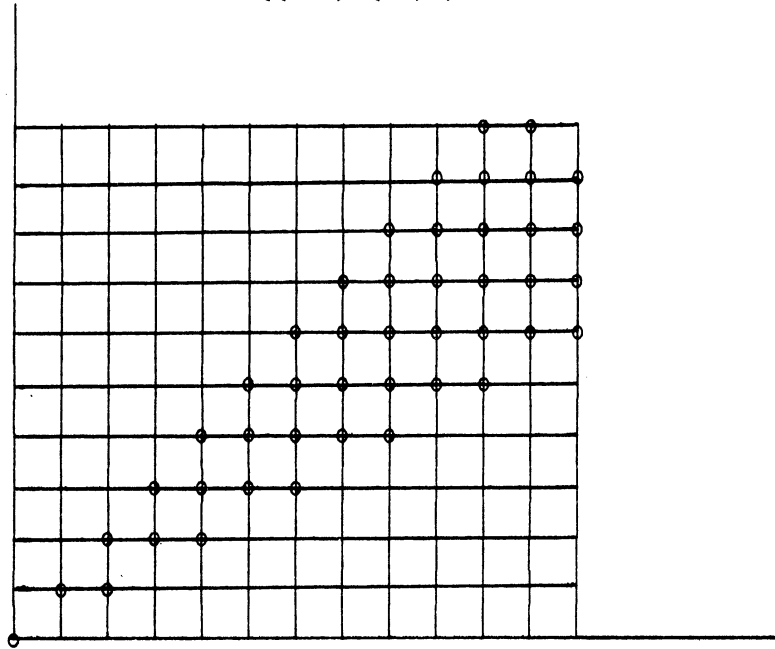
In such a geometric model of automaton events, the behavior of finite-state automata is denoted by designated finite sets of points in L or by orthogonal infinite periodic sets of points. The events which are the behavior of infinite-state automata are denoted by oblique infinite periodic sets of points. Examples of the events which are the particular subject of study in this paper are given below.



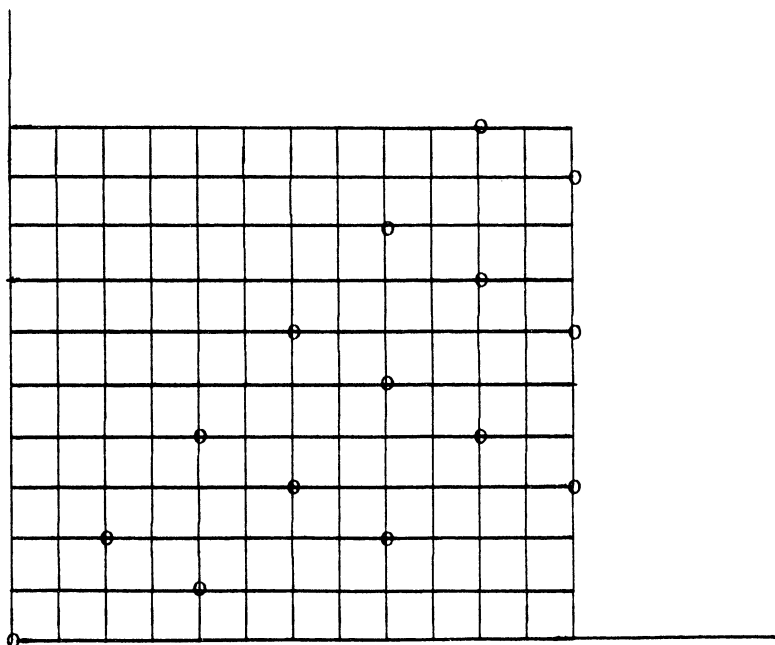
That is, $C((ab)^*)$ is denoted in L by designating every point on the main diagonal ($C((aabb)^*)$ would be denoted in L by designating alternate points on the main diagonal) and $C((aab)^*)$ is denoted in L by designating all points on a line from the origin having slope $1/2$.



CREC : C((aab)*(ab)*)



CREC : C((aaaab)*(aabb)*)



Note that this representation of CRE in L can also be viewed either as the behavior tree of the machine associated with the event or as the machine itself, the origin point in L being associated with the word of zero length in the one case, and with the initial state of the machine in the other. The designated points of L are, in the machine interpretation the final states of the machine; upon receipt of an a, the state transition is to a state (point in L) directly to the right, upon receipt of a b the state transition is to a state (point) directly above.

For events of the CREA and CREB type, uniform homomorphic mappings of points (alternatively, merging of states) can be performed which preserve the desired relationships between designated and non-designated points of L (final and non-final states of the associated CM (commutative-machines)).

Observation 1: Two states (points in L) s_1, s_2 can be merged if and only if there is no input word w whatsoever such that w applied beginning in s_1 leads to a final state (designated point) while w applied beginning in s_2 leads to a non-final state (non-designated point).

Observation 2: For any two states (points in L) s_1, s_2 which are not on a line parallel to the oblique process of final states (designated points) of a CREA, CREB or CREC a word w can always be found such that w applied to s_1 , yields a final state (designated point) while w applied to s_2 yields a non-final state (non-designated point).

Observation 3: A merging-mapping for CREA, CREB, CREC events in L (all of which have oblique processes in L) which preserves the desired properties can take place only on lines parallel to the oblique lines in L generated by the CRE.

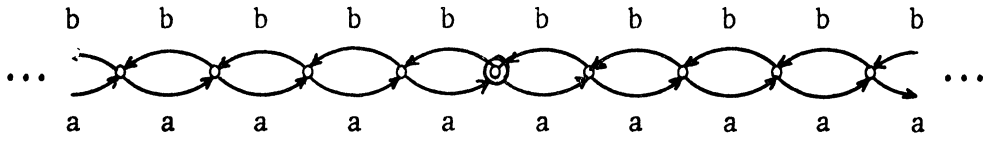
A CREC event has two non-parallel oblique processes, therefore:

Observation 4: For CREC, no states (points) in L can be uniformly merged (mapped) relative to the desired properties.

Thus the CM for $C((ab)^*(aab)^*)$ and $C((aaaab)^*(aabb)^*)$ are necessarily as given in the figures above.

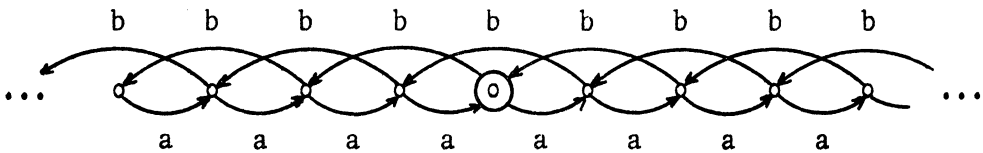
The results of merging-mapping that can be performed relative to CREA and CREB events are shown in the examples below.

CREA: $C((ab)^*)$

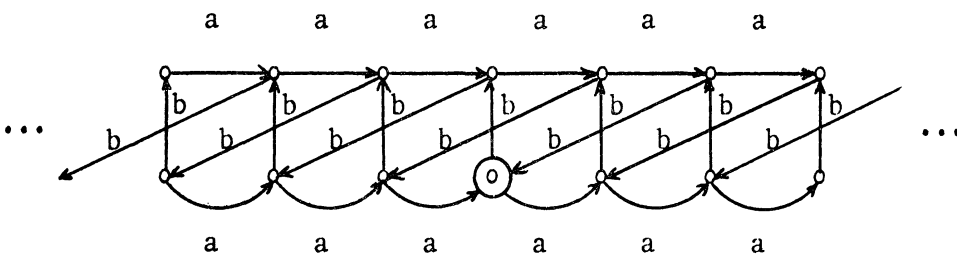


(\odot is both origin and designated point, alternatively initial state and final state.)

CREA: $C((aab)^*)$



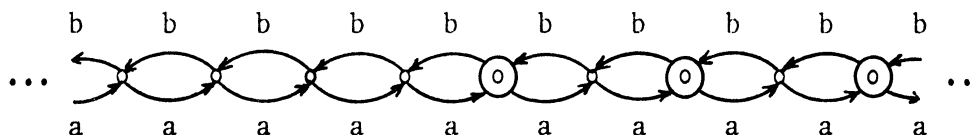
CREA: $C((aabb)^*)$



In the case of $C((ab)^*)$ which was denoted in L by designating all points on the main-diagonal, these main-diagonal points were all mapped-merged into the origin-initial state, and all points on lines parallel to the main-diagonal were mapped-merged into their respective

representatives on the axes (then the axes were "folded down flat").

CREB: $C((aa)^*(ab)^*)$



Above, the points on the main diagonal and on all lines parallel to the main diagonal have been mapped into their representatives at the origin and on the axes; alternate lines of points below the main diagonal were designated points of L (final states) and they have been mapped into alternating states to the right of the initial state (the leftmost of the final states is also the initial state).

III. Extended Forms of CREA, CREB, CREC

If the pattern of designated points in L determined by a CREB or CREC is extended over all of L , we will speak of the new event denoted as the "extended event" and employ the notation Ext CREB , Ext CREC to denote the event. For events denoted by expressions of the CREA form we define an analogous Ext CREA .

Definition: A CREA is of the form $C(a^k b^l)^*$ (where a^k means k as and b^l means l bs); then $\text{Ext } (C(a^k b^l)^*) = C((a^k)^*(b^l)^*)$

The Ext operation can be viewed as a "paving" or tessellation of the complete plane of L with "basic" blocks or tiles the characteristics of which (size, shape, orientation) are determined by the particular CREA, CREB, or CREC. A "basic tile" is defined to be any of the quadrilaterals which have designated points of L only at their vertices. (Note that in paving all of L , some tiles may overlap the axes, encroaching on other quadrants; we are interested only in the points -- tile vertices -- which fall in L proper.)

By translations of the tessellation of designated points of L

all points of L may be accounted for; the number of different placements of the fundamental tessellation necessary to encounter all points of L will be called the index of the determining CRE.

Observation 5: The index is the same as the area of the lattice polygon of the basic tile.

Observation 6: This index or area can be found by employing G. Pick's formula for the area of lattice polygons, viz: "The area of a simple lattice polygon is one-half the number of lattice points on its perimeter, plus the number of interior points, minus one."

Observation 7: The minimal state finite automaton which realizes Ext CRE has a number of states equal to the index.

Observation 8: The finite state automata realizing the events denoted by Ext CREA are orthogonal, and these orthogonal machines are minimal. For events denoted by Ext CREB and Ext CREC there are finite state orthogonal machines realizing the events denoted but these machines are not in general minimal.

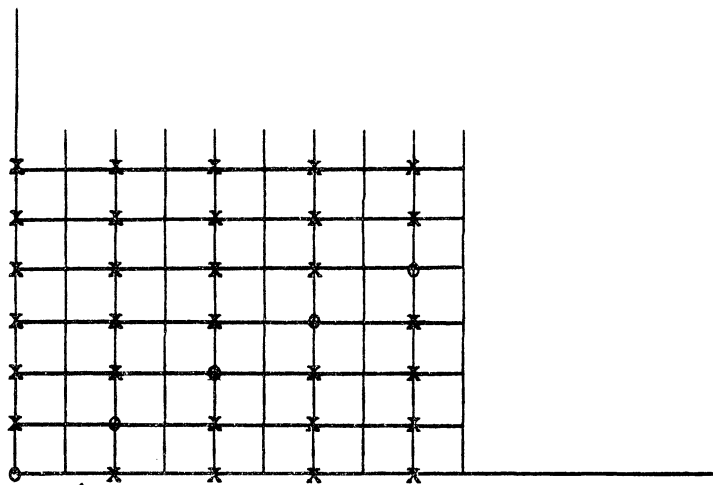
Observation 9: The set of words denoted by a CRE is included in Ext CRE;

the behavior of a machine realizing a CRE is included

in the behavior of a machine realizing Ext CRE.

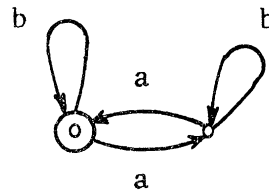
The following examples should make clear the above Observations.

CREA: $C((aab)^*)$

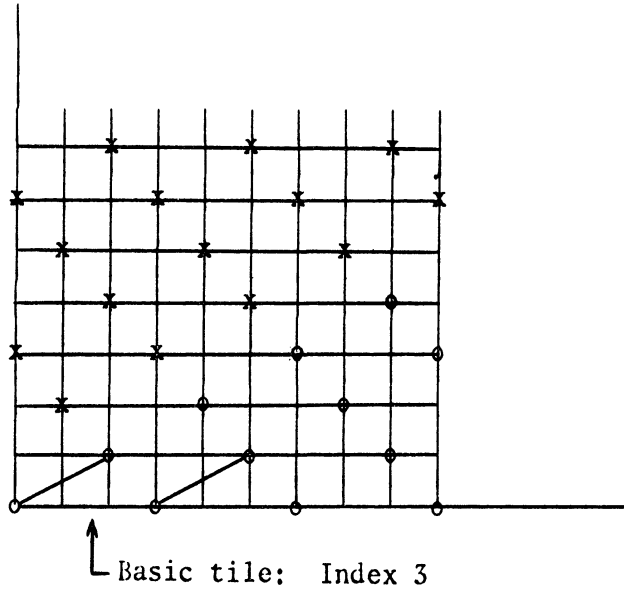


Basic tile: Index 2 for Ext $C((aab)^*) = C((aa)^*b^*)$

Ext. Machine:

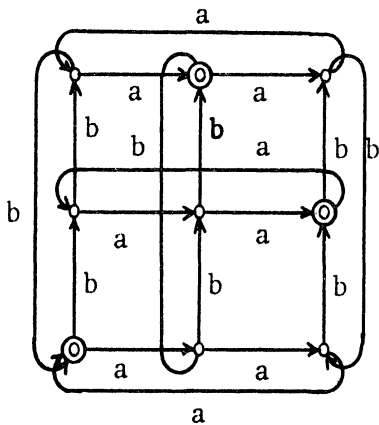


CREB: $C((aaa)^*(aab)^*)$

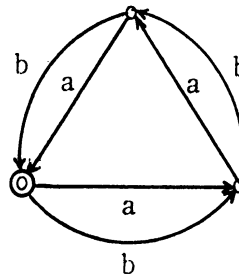


$$\text{Ext } C((aaa)^*(aab)^*) = C((\Lambda \cup aab \cup abb)((aaa)^*(bbb)^*) =$$

$$C((aaa)^*(bbb)^*) \cup C((aab)(aaa)^*(bbb)^*) \cup C((abb)(aaa)^*(bbb)^*)$$

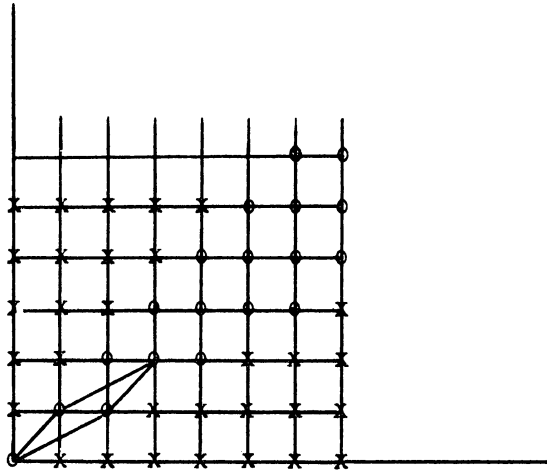


Orthogonal Ext Machine



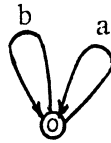
Minimal Ext Machine

CREC: $C((aab)^*(ab)^*)$



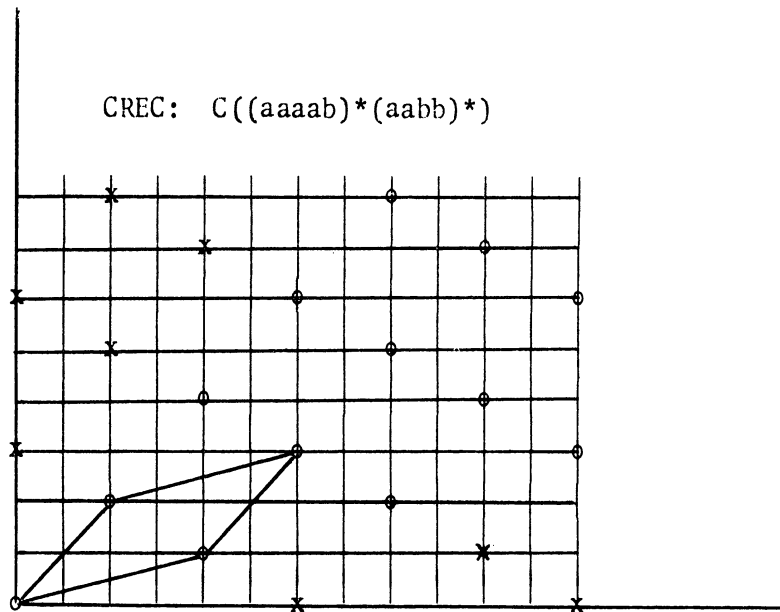
Basic tile: Index 1

$$\text{Ext } C((aab)^*(ab)^*) = C((\Lambda \cup aab \cup ab)((aaa)^*(bb)^*)) = C(a^*b^*)$$



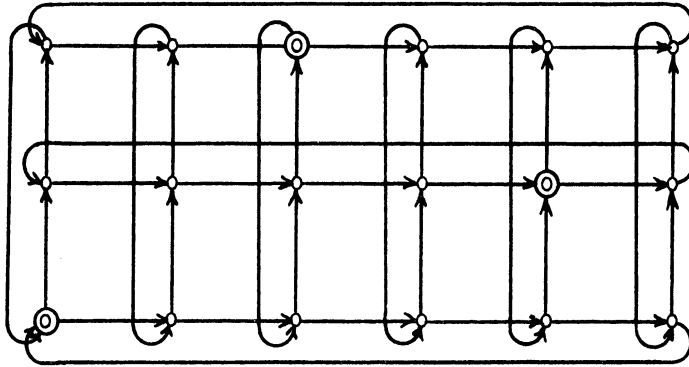
Ext Machine

CREC: $C((aaaab)^*(aabb)^*)$

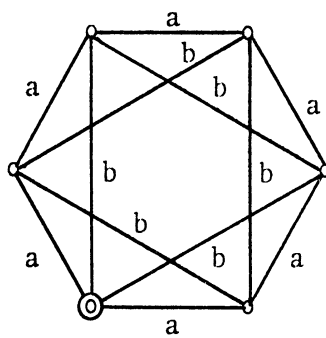


Basic tile: Index 6

$$\text{Ext } C((aaaab)^*(aabb)^*) = C((\Lambda \cup aabb \cup aaaab)(aaaaaa)^*(bbb)^*)$$



Orthogonal Ext Machine



Minimal Ext Machine

IV. Tape Machines

A tape machine TM consists of a read-head to which is attached one or more indefinitely extendible storage tapes. At the beginning of operation one storage tape may contain marked squares. The read-head of the TM receives input signals (which in the cases to be studied here may be viewed as appearing on a separate single input tape which at each moment of time exposes one new square). Upon receipt of an input symbol the TM responds by successively

- a) undergoing a change in internal state (a transition returning to the same state also being allowable)
- b) moving each of the storage tapes independently one square right, one square left, or not at all.
- c) possibly printing (or over-printing) a letter of the storage tape alphabet (which may include "blank," the "erase" letter) upon any of the storage tape squares now under scan

The sequence of actions -- undergoing read-head state transition,

moving, and then printing storage tapes -- will be called an "atomic act" of the TM.

The set of all n-tuples in which the first member is an internal state of the read-head and the remaining n-1-tuple is the contents of the storage tape squares under scan, will be called a configuration of the TM. A subset of the configurations of a TM will be called final configurations. When having read through an input tape the TM is placed in a final configuration the tape will be said to be accepted. The set of all tapes accepted by a given TM will be called the behavior of the TM, and the TM will be said to have realized the "event" denoted by the set.

Many kinds of TM can be distinguished; we focus on the following:

1. Those in which no atomic act is ever a function of the contents of any storage tape square under scan; these will be called TM without control. Conversely, those in which an atomic act may be a function of the contents of a storage-tape square under scan will be called TM with control.

2. If for all input sequences, the TM performs exactly one atomic act for each successive input letter scanned, then the TM will be called real-time.
3. If a TM can make atomic moves which include the printing of storage-tape squares, then it will be called a printing TM. (Conversely, if no atomic move of a TM can result in storage-tape printing, the TM will be called non-printing.)
4. If the atomic moves of the TM are a single valued function of the input tape symbol, internal state of the read head, and (possibly) storage tape square under scan, then the TM will be called deterministic (otherwise non-deterministic).

All the TM to be considered here operate in real-time; in addition, the storage-tape alphabet will consist of marks and blanks only.

The main TM results to be presented in the remaining section are summarized below:

- A. Events denoted by expressions of the form CREA are realizable by deterministic TM without control or printing, having a single storage tape with a single marked square.

B. Events denoted by expressions of the form CREB are realizable by:

1. Non-deterministic TM without control or printing and with a single storage tape having a single marked square.
2. Deterministic TM without control or printing but having a single storage tape whose squares have been "pre-marked" with an infinite periodic sequence.

(2'. Alternatively, pairs of machines, consisting of a deterministic TM without control having a single storage tape with, initially, a single marked square, and an input-free deterministic finite-state machine which operates by independently moving and printing on the squares of the storage tape of the TM an infinite periodic sequence.)

2''. A deterministic TM with control having a single storage tape with, initially at most one marked square, with printing (in effect combining the pairs of machines of 2' above into a single machine).

2'''. A deterministic TM with control, having 2 storage tapes, one an "ordinary" infinite storage tape with a single marked square and the second a finite "loop" tape with a single marked square.

C. Events denoted by expressions of the form CREC are realizable by:

1. Non-deterministic TM without control and with two storage tapes,

each of which tapes has a single marked square.

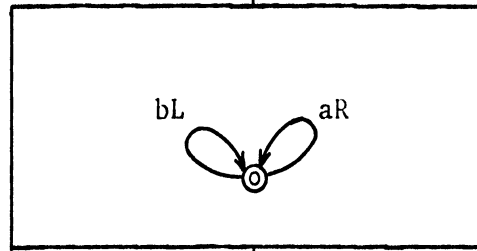
2. Deterministic TM with control and with two storage tapes, each of

which tapes has a single marked square.

The remaining sections of this paper are largely devoted to discussion of these TM and their construction.

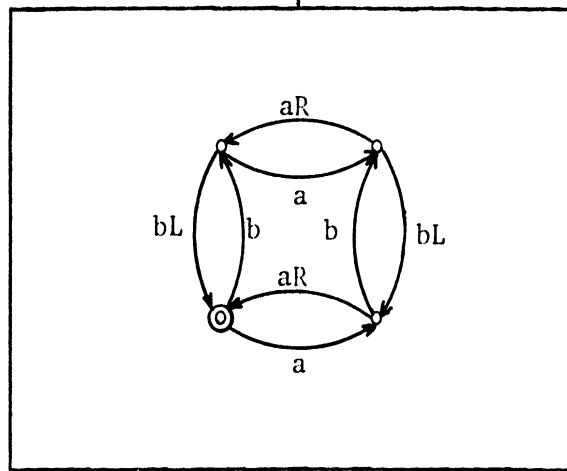
CREA: $C(ab)^*$

storage tape

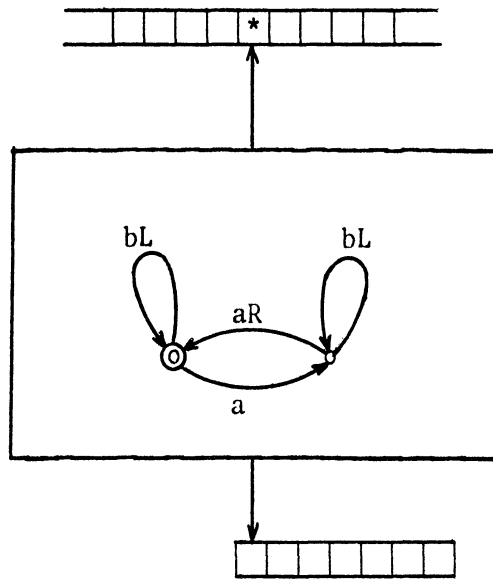


input tape

CREA: $C(aabb)^*$



CREA: $C(aab)^*$

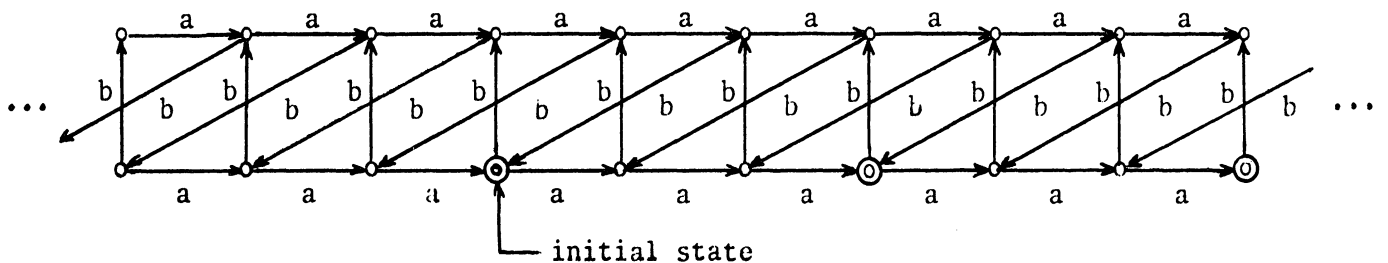


VI. TM for Events of the Form CREB

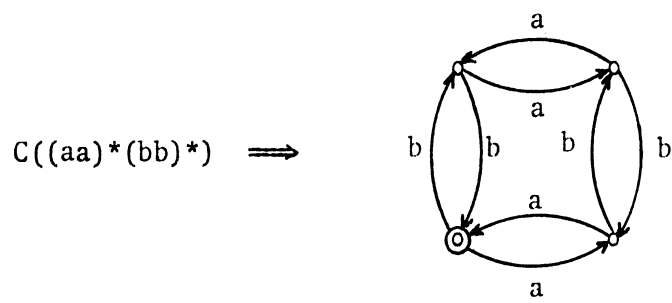
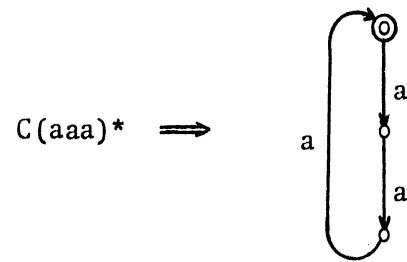
Events of the form CREB are generated by expressions composed of two parts; a "pure" word starred concatenated with a "mixed" word starred.

Non-deterministic TM realizing events of the CREB form can easily be obtained. The read-head structure consists of copies of the CREA read-head structure for the "mixed" word portion of the CREB expression; the number n of copies is equal to the number of letters in the "pure" word portion of the CREB expression.

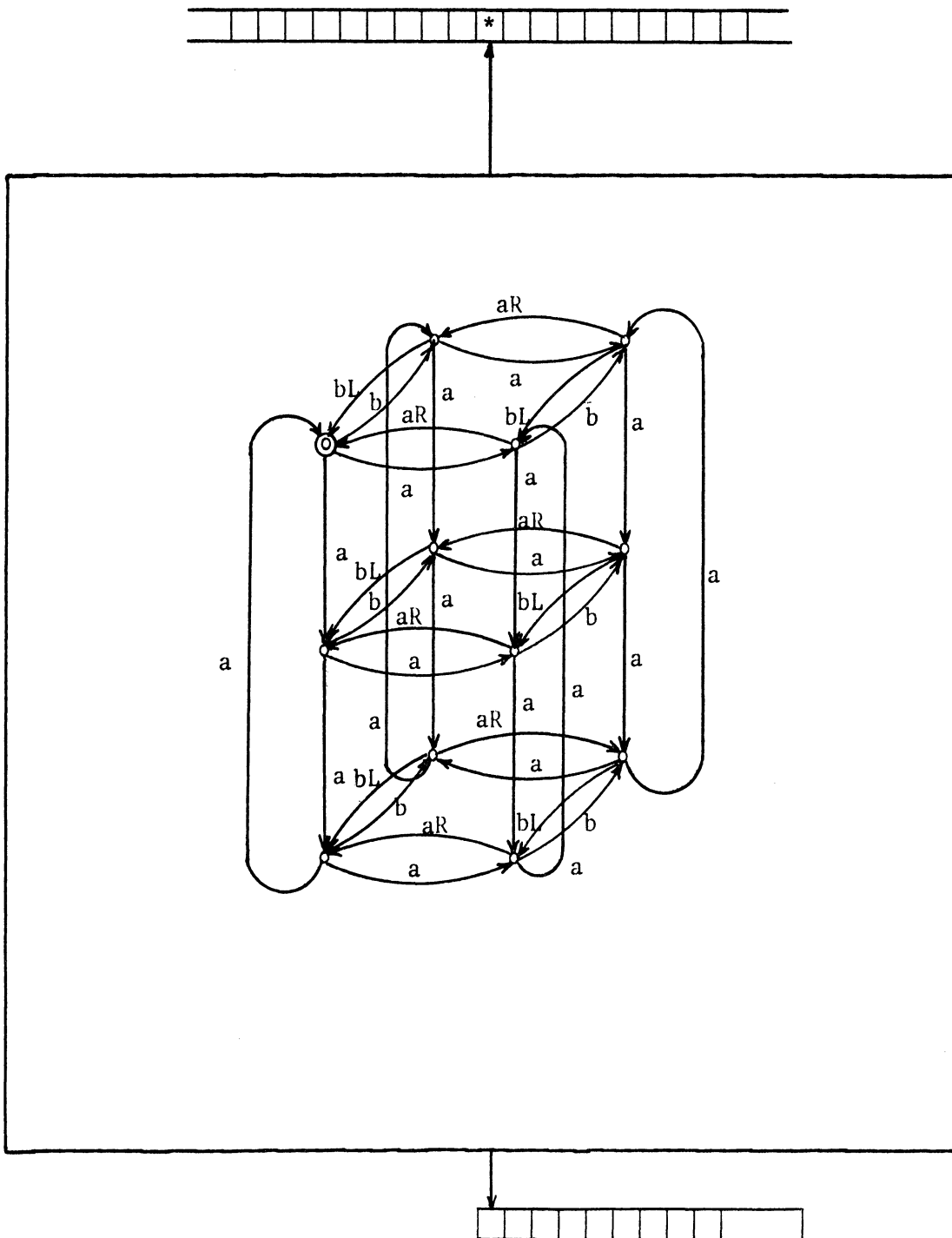
In the case $CREB = C((aaa)^*(aabb)^*)$ for example, three copies of the $(aabb)^*$ structure are required.



Infinite Commutative Machine for $C((aaa)^*(aabb)^*)$



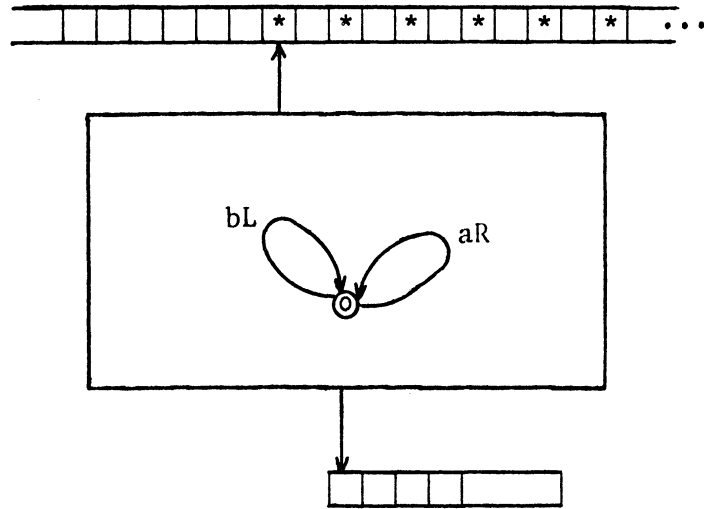
Non-deterministic TM for $C((aaa)^*(aabb)^*)$



At every state, there are two "out-transitions" labelled a, hence the non-deterministic nature of the TM. (The operation of the machine may be viewed as deterministic by assuming that some of the as of the input sequence are genuinely as but that some are a's of which the latter are to be treated as going to satisfy the conditions imposed by the "pure" word while the former are to be checked off as eventually going to make up elements derived from the mixed word. Since, however, this is precisely what we do not know and cannot assign ahead of time, the operation of the TM, as we have here constructed it, is non-deterministic.)

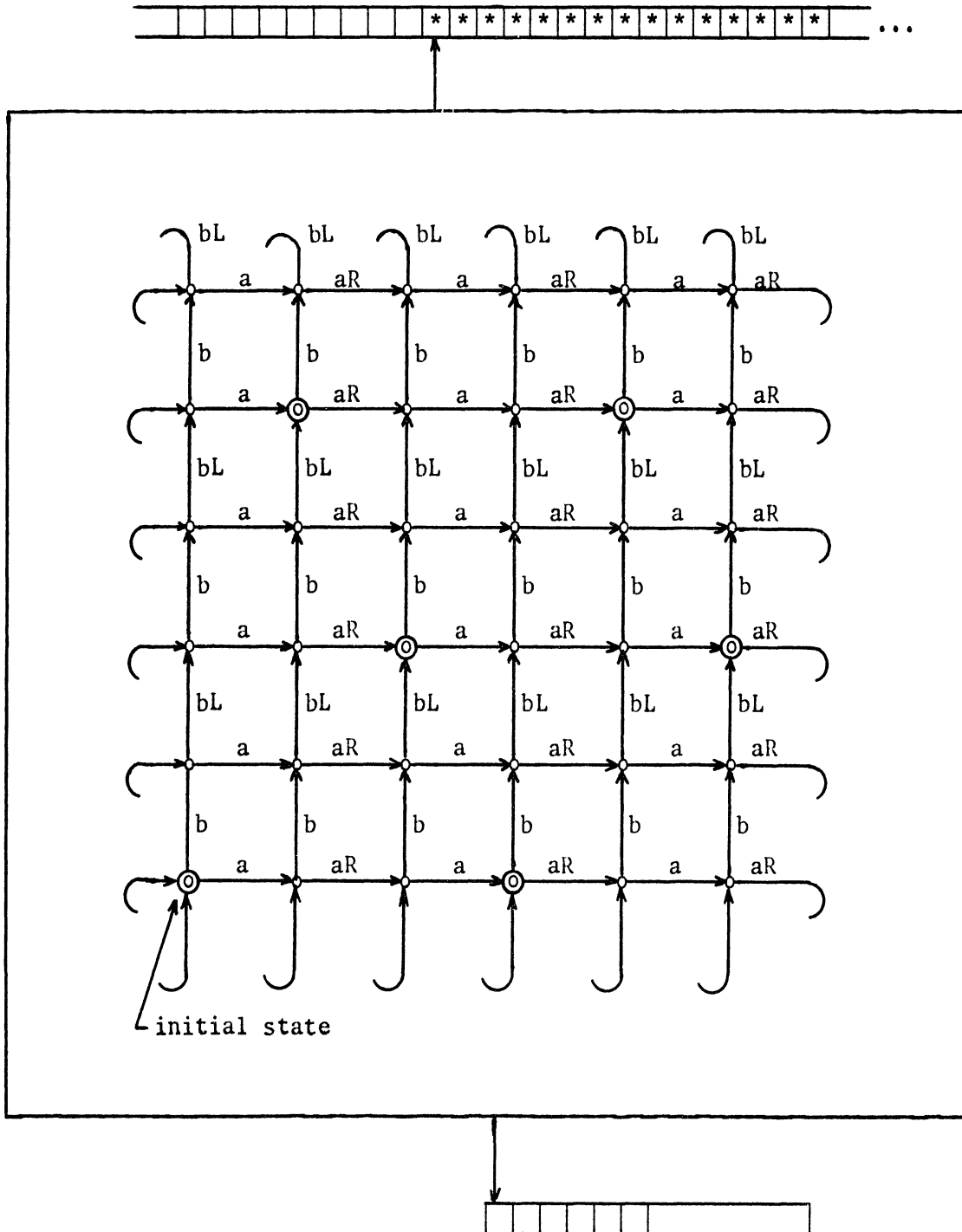
The non-determinism of the TM thus obtained will give it the property that $C((aaa)^*(aabb)^*)$ will be realized in the sense that the only sequences the TM will accept will be elements of the event but that the TM may frequently fail to accept genuine elements of the event.

How can this limitation be overcome? One solution is to "pre-print" upon the storage tape squares an infinite periodic pattern of marks. Thus the CREB $C((aa)^*(ab)^*)$ can be realized by:



Often in order to obtain the correct behavior some "change in the scale" of the read-head structure may have to be made to obtain a common periodicity between the numbers of letters in the pure word and the numbers of the same letter in the mixed word.

Pre-printed storage tape TM for CREB: $C((aaa)^*(aabb)^*)$



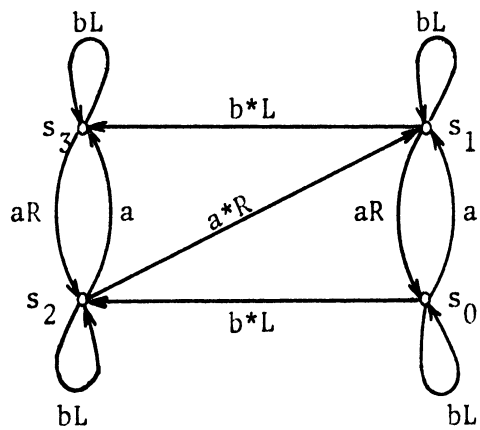
It is apparent then, that at the "cost" of infinite pre-printing, a 1-storage-tape deterministic real-time TM without control can be obtained for events of the CREB form. Note that the presence of the read-head in the left wing of the storage tape means that the event is located in L, the integer lattice model, above the oblique line (in the present example above the oblique line determined by (aabb)*) while presence of the read-head in the right wing of the storage tape means the event is located in L below the oblique line.

If we consider means of realizing CREB events in a fashion similar to the above but avoiding infinite pre-printing, one solution would be to have a pair of machines, one consisting of a TM identical to that for the "pre-print" case above, and the second an input-free finite automaton which is set to work independently to print the storage tape in the desired periodic fashion. Another alternative would be to incorporate the printing automaton in the TM read-head structure. In this latter case however, the TM would have to be a TM with control as well as with printing since the read-head would have to be able to distinguish and act upon the presence of (for example) the first marked square or the beginning

of the yet blank but to-be-marked portions of the storage tape. If however control is going to be required, then printing or infinite pre-printing ought to be eliminable, the correct count being kept in the read-head structure, this count being re-set whenever the read-head passes the sole pre-marked square.

The one storage tape, real-time deterministic, controlled realization of CREB events can be expressed as the intersection of the behavior of two machines. One machine would have as its read-head structure two copies of the structure required to realize the "oblique line" events which are a part of any CREB. This machine would accept all events which are, for example, on or above, or on or below the oblique line. The second machine would realize Ext CREB. The intersection of the two behaviors -- the events in Ext CREB which are on or above or on or below the line in L determined by the mixed word of the CREB -- is the desired CREB event.

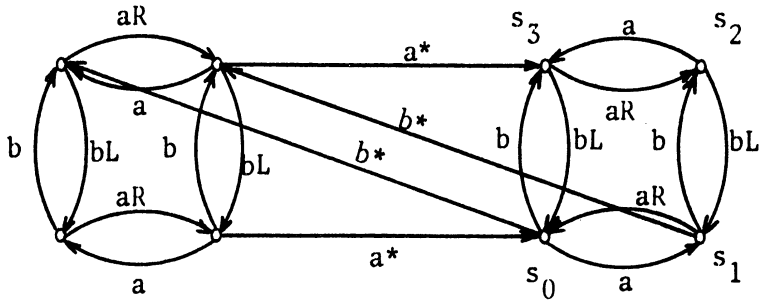
(In the following figures "a*" means "input symbol a while scanning * on storage tape" similarly for b*, mutatis mutandis.)



Initial configuration: s_0^*

Final configurations: $(s_0^*), (s_1^*), (s_2^*)$

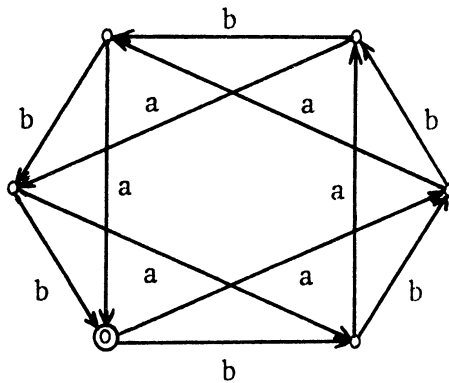
Read-head structure for TM realizing "on or below" $(aab)^*$ in L.



Initial configuration: (s_0^*)

Final configurations: $(s_0^*), (s_1^*), (s_2^*), (s_3^*), (s_4^*)$

Read-head for TM for $(aabb)^*$ or below

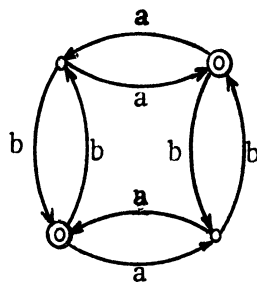
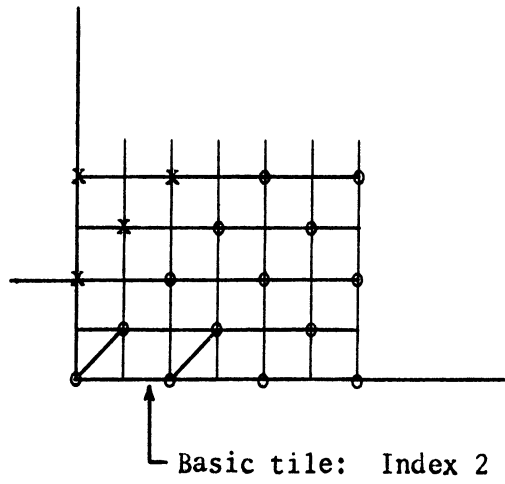


Finite-state automaton for $\text{Ext } C((aaa)^*(aabb)^*)$

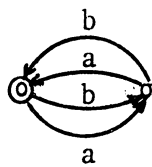
The intersection of the behaviors of the two above machines is the CREB event $C((aaa)^*(aabb)^*)$.

Since both of the intersect machines operate in the same manner and time-scale (they register acceptance or not in synchrony) the CREB events should be realizable not only by pairs of machines but there should also be a single TM having the desired properties: viz. a deterministic, one storage tape real-time TM with control. This TM will operate in the following fashion: its read-head structure will have two halves, each of which will operate identically in respect to the storage tape moves: transitions from one half to the other of the read-head structure will take place as a function of the storage tape squares under scan (thus the TM is "with control"); if the marked square is under scan a transition to the other half of the read-head structure will take place if the next input symbol takes the event to the "opposite" side of the oblique line in L determined by the mixed word of the CREB event. As the read-head passes through the sole marked square of the storage tape, the read-head will undergo a transition which will take it from one "half" of the read-head to the other. In addition at least one of the halves of the read-head structure has the additional property that it can count the letters of the "pure" word of the CREB.

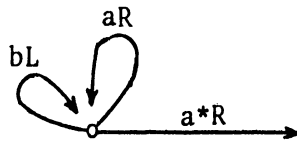
CREB: $C((aa)^*(ab)^*)$



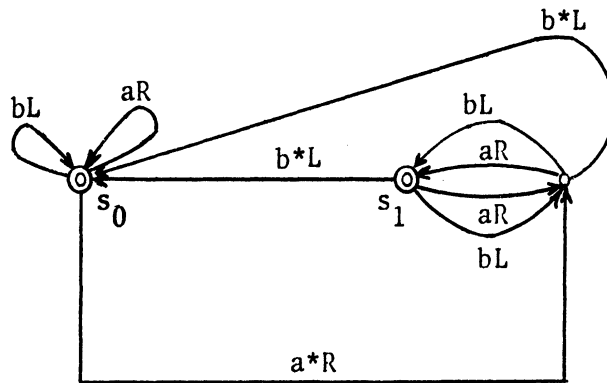
Orthogonal machine for Ext $C((aa)^*(ab)^*) = C((\Lambda \cup ab)(aa)^*(bb)^*)$



Minimal Ext machine



Read-head structure for registering on or below $(ab)^*$



Final configurations: s_0^*
 s_1^*
 s_1^-

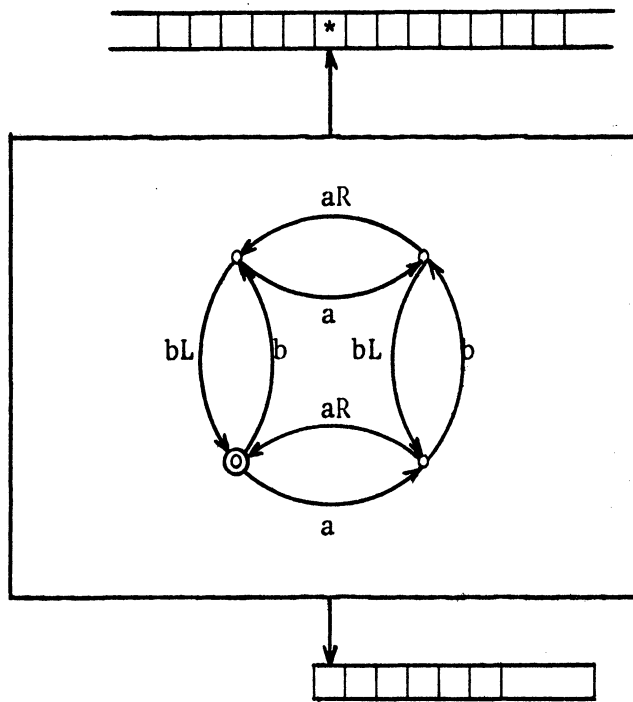
Read-head for TM realizing $C((aa)^*(ab)^*)$

VII. TM for Events of the Form CREC

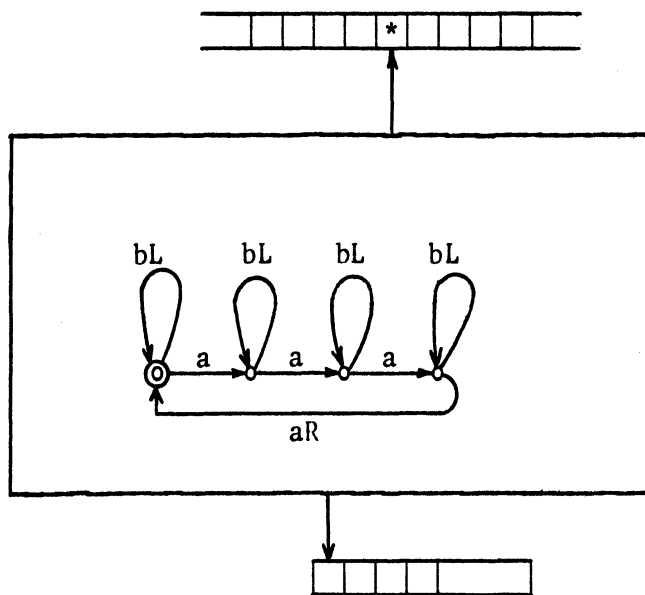
As with events denoted by expressions of the CREB form, finding a non-deterministic TM realizing the desired behavior is quite straight-forward. CREC events are realizable by a 2-storage tape non-deterministic TM in which for each state of the read-head there are two possible a-transitions and two possible b-transitions. (Again this can be seen readily by conceiving the input sequence to the TM as being composed not just of as and bs but of these along with a's and b's (where, a CREC being of the form $C(w_1 * w_2^*)$, the w_1 contains as and bs and the w_2 contains a's and b's). The general method of constructing a desired TM will be apparent from the example to follow.

CREC: $C((aabb)^*(aaaab)^*)$

From our discussion of expressions of the CREB form, the TM for the separate w_1 , w_2 parts of our example event are:



$$w_1 = C(aabb)^*$$



$$w_2 = C(aaaab)^*$$

These deterministic TM can be combined to form a non-deterministic 2-tape TM without printing or control having the desired properties.

The compound machine is obtained by first constructing the "product" read-head for the two original read-heads. In this "product" read-head:

- 1) The new states consist of pairs, $s_i r_j$ of the states of the original read-heads (where the s_i are states of the first original read-head, and the r_j are states of the second original read-head); thus, if there are a total of k s -states and l r -states, then there will be $k \times l$ states in the new compound read-head.
- 2) The new initial and final state is the pair, s_0, r_0 composed of the states which were both original and final states in the component read-heads.
- 3) For each state of the new read-head there will be two possible transitions under an a input and two possible transitions under a b (thus machine operation will be non-deterministic).
- 4) Transitions may be assigned by first taking all pairs for which the first member is s_0 , and filling in transitions by reference to the second member, r_j , of each pair and the transition-structure

of the original second read-head; similarly for all pairs in

which s_1 is first member, etc.

Complete the transitions by fixing in turn the second member of each pair, and following the transition-structure of the original first read-head.

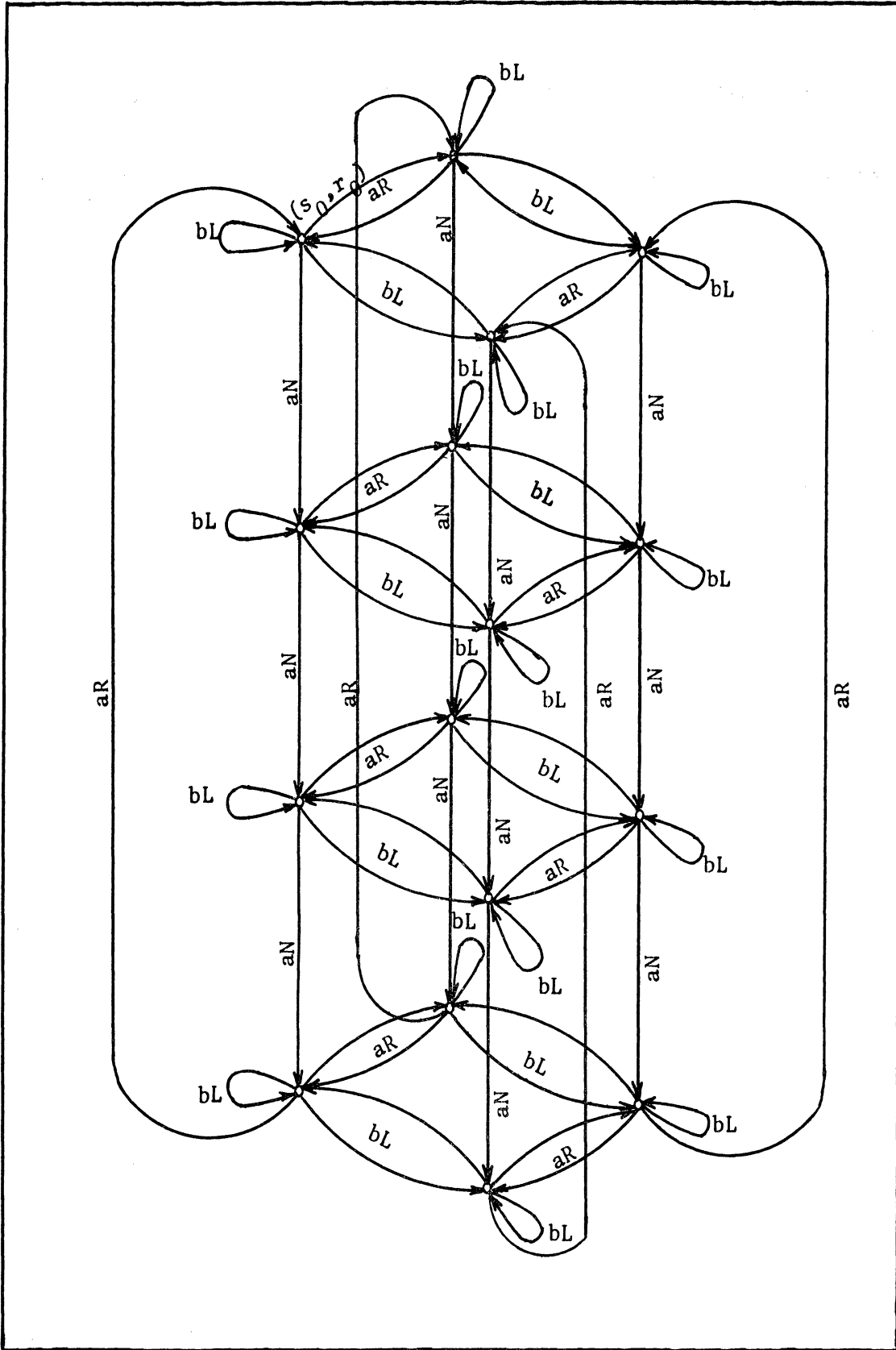
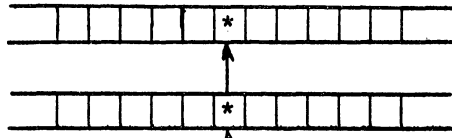
Append to the compound read-head so obtained an input-tape and two storage-tapes, T_1, T_2 . Whenever the new read-head is going through transitions assigned from the original first read-head, tape instructions are for T_1 ; Whenever the new read-head is going through transitions assigned from the original second read-head, tape instructions act on T_2 .

The single acceptance configuration is $((s_0, r_0)T_1(1), T_2(1))$.

The non-deterministic, two storage-tape machine for the example $C[(aabb)^*(aaaab)^*]$ is given below.

T₂

T₁



A deterministic two-storage tape TM without printing but with control realizing CREC events can also be obtained. From results already discussed we know that having any CREC: $C(w_1^*w_2^*)$ (where w_1^* is chosen to be the upper oblique line in L and w_2^* the lower oblique line in L) we can obtain the following three machines:

1. The finite state machine having behavior Ext CREC.
2. The 1-storage tape (deterministic, non-printing, with control) TM which will accept all sequences leading to points in L which lie on or below the line formed by the w_1^* of the CREC.
3. The 1-storage tape (deterministic, non-printing, with control) TM which will accept all sequences leading to points in L which lie on or above the line formed by the w_2^* of the CREC.

Observation 10: The intersection of the behaviors of these three machines is the desired CREC behavior.

Observation 11: The three machines can be constructed so as to operate in synchrony; a single combined machine is therefore possible.

Conceptually, the desired combined, single machine is simple enough.

A transition structure having Ext CREC behavior must be employed. In order

to be able to count correctly the number of as and bs which determine the tape moves the Ext CREC transition structure may have to be enlarged beyond the minimal state machine required for Ext CREC alone.

Observation 12: The orthogonal form of a finite automaton for Ext CREC expanded so that it possesses common a-cycle and b-cycle for the w_1 and w_2 of the CREC can always be found.

Let us call the resulting transition structure the "essential" structure of the read-head for the desired TM. The TM will have a finite-state read-head with three copies of this essential structure -- a Right, a Left, and a Center copy. Each of these will be a (possibly expanded orthogonal) Ext CREC machine.

If the read-head of the TM is in any of the final states of any of the three "essential" structures, then the input sequence is an element of Ext CREC. If the read-head is in the Left essential structure, then the input sequence, expressed in L, is at a point on or above the w_1^* line. If the machine is in the Right essential structure, then the input word expressed in L lies on or below w_2^* . If the read-head is in the Center essential structure, then the input word lies on or between the

oblique lines formed by w_1^* , w_2^* .

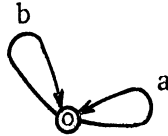
The operation of such a TM is as follows: the TM begins with its read-head in its initial state in the Center essential structure and scanning the sole marked squares on T_1, T_2 its 2 storage tapes. Upon receipt of input letters, transitions within the read-head structure are undergone. The storage tape moves for T_1, T_2 are determined separately; whenever a number of as has been received to equal the a-cycle of w_1 the w_1 's storage tape T_1 is moved one square right; similarly for b-cycles with left moves of T_1 ; w_2 's cycles are similarly recorded by moves of the T_2 storage tape. Whenever an input letter is about to take the event, expressed in L, above the w_1^* line, then the read-head transition calls for a "jump" to the corresponding state of the Left essential structure of the read-head; similarly to the Right essential structure for inputs leading to events in L below w_2^* .

There is one final state for the Right and one final state for the Left essential structure; the Center structure may have several final states. Whenever the read-head is in the final state of the Right or the Left structures and its associated storage tape is being scanned,

then the TM is in an acceptance configuration; whenever the read-head is in the Center structure and is in any of the Center's final states, then, regardless of storage tape squares under scan, the TM is in an acceptance configuration.

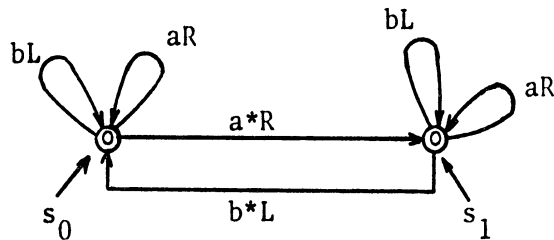
The complete TM for a simple CREC event is given below.

CREC: $C((aab)^*(ab)^*)$



Ext $C((aab)^*(ab)^*)$:

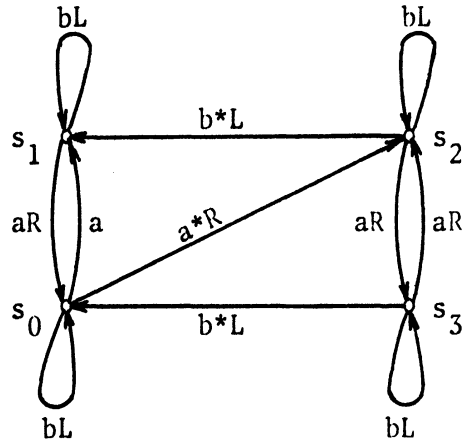
Read-head for TM realizing points on or below $(ab)^*$:



Initial configuration: (s_0^*)

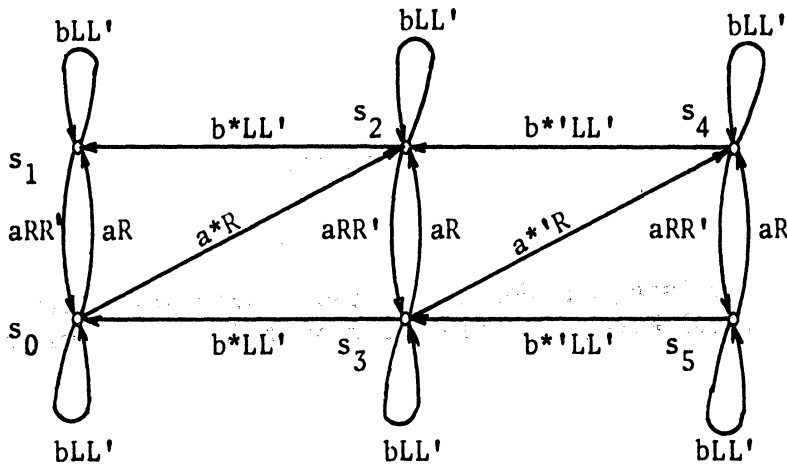
Final configurations: $(s_0^*), (s_1^{\bar{*}})$

Read-head for TM realizing points on or above $(aab)^*$:



Initial configuration: (s_0^*)

Final configurations: $(s_0^*), (s_1^*), (s_2^*), (s_3^*)$



Initial configuration: $(s_3^{**'})$

Final configurations: $(s_0^{**'})$, (s_1^{*-})
 $(s_2^{\bar{**}'})$, $(s_3^{\bar{**}'})$
 (s_4^{*-}) , $(s_5^{**'})$

Three "layer" read-head for TM realizing $C((aab)^*(ab)^*)$

(Explanation of notation: "b*'LL'" is to be read "upon receipt of a b while viewing the marked square of storage tape T' take this transition, moving one square to the Left on both storage tape T and T''"; " $s_2^{\bar{**}'})$ " is the notation for TM configurations in which the read-head is in state S_2 and either marked or unmarked squares are under scan on both T, T'.)

References

- [LW] Richard Laing and Jesse B. Wright, "Commutative Machines,"
Technical Note, The University of Michigan, December 1962.

UNIVERSITY OF MICHIGAN



3 9015 03466 4246