NBS TECHNICAL NOTE

296

A Grammar for Component Combination In Chinese Characters

B. KIRK RANKIN III, STEPHANIE SIEGEL,
ANN McCLELLAND, AND JAMES L. TAN

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards

THE NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] provides measurement and technical information services essential to the efficiency and effectiveness of the work of the Nation's scientists and engineers. The Bureau serves also as a focal point in the Federal Government for assuring maximum application of the physical and engineering sciences to the advancement of technology in industry and commerce. To accomplish this mission, the Bureau is organized into three institutes covering broad program areas of research and services:

THE INSTITUTE FOR BASIC STANDARDS . . . provides the central basis within the United States for a complete and consistent system of physical measurements, coordinates that system with the measurement systems of other nations, and furnishes essential services leading to accurate and uniform physical measurements throughout the Nation's scientific community, industry, and commerce. This Institute comprises a series of divisions, each serving a classical subject matter area:

—Applied Mathematics—Electricity—Metrology—Mechanics—Heat—Atomic Physics—Physical Chemistry—Radiation Physics—Laboratory Astrophysics[2]—Radio Standards Laboratory,[2] which includes Radio Standards Physics and Radio Standards Engineering—Office of Standard Reference Data.

THE INSTITUTE FOR MATERIALS RESEARCH . . . conducts materials research and provides associated materials services including mainly reference materials and data on the properties of materials. Beyond its direct interest to the Nation's scientists and engineers, this Institute yields services which are essential to the advancement of technology in industry and commerce. This Institute is organized primarily by technical fields:

—Analytical Chemistry—Metallurgy—Reactor Radiations—Polymers—Inorganic Materials—Cryogenics[2]—Materials Evaluation Laboratory—Office of Standard Reference Materials.

THE INSTITUTE FOR APPLIED TECHNOLOGY . . . provides technical services to promote the use of available technology and to facilitate technological innovation in industry and government. The principal elements of this Institute are:

—Building Research—Electronic Instrumentation—Textile and Apparel Technology Center—Technical Analysis—Center for Computer Sciences and Technology—Office of Weights and Measures—Office of Engineering Standards Services—Office of Invention and Innovation—Clearinghouse for Federal Scientific and Technical Information.[3]

---

[1] Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted; mailing address Washington, D. C., 20234.

[2] Located at Boulder, Colorado, 80302.

[3] Located at 5285 Port Royal Road, Springfield, Virginia, 22151.

UNITED STATES DEPARTMENT OF COMMERCE • John T. Connor, Secretary
NATIONAL BUREAU OF STANDARDS • A. V. Astin, Director

TECHNICAL NOTE 296

A Grammar for Component Combination In Chinese Characters

B. Kirk Rankin III, Stephanie Siegel,
and Ann McClelland

Center for Computer Sciences and Technology
Institute for Applied Technology
National Bureau of Standards

and James L. Tan
George Washington University

NBS Technical Notes are designed to supplement the Bureau's regular publications program. They provide a means for making available scientific data that are of transient or limited interest. Technical Notes may be listed or referred to in the open literature.

For sale by the Superintendent of Documents, U.S. Government Printing Office
Washington, D.C., 20402 - Price 60 cents

Preface

This note describes and evaluates an attempt to analyze the pictorial structure of Chinese characters. The approach underlying this analysis is borrowed from the field of linguistics. The collection of objects under study is the set of all well-formed Chinese characters; that is, the set of all character-like structures which an informant accepts as actually occurring or possibly occurring. The set of these well-formed Chinese characters is considered to be a language. The corpus on which the grammar is based is Mathews' (3) (see Bibliography, page 118) minus a handful of entries which are deliberately excluded.1/

The task of analyzing this language is taken to be that of constructing a grammar to account for the pictorial structure of characters by generating objects which correspond to characters. These

---

1/ More precisely, our corpus consists of hand-written copies of the characters in Mathews'. See Appendix A for a listing and classification of the excluded entries.

objects are well-formed at a high (syntactic) level of structure only, and are properly viewed as requiring further processing by an output transducer which is yet to be constructed. This transducer will contain rules for size and shape variation in character sub-parts and rules for precise spatial placement of character sub-parts.

The grammar 1/ displayed in this note is a mixture of various formalisms which in turn are inspired by formalisms used in the fields of linguistics and computation. Although it is a mixture, the central property of the grammar is its capability of imposing a kind of phrase structure on its output characters.2/

1/ The grammar, GCC-3 (Grammar for Component Combination, number three), is a refinement and extension of the concepts developed in two previously constructed grammars: GCC-1, in Rankin, Sillars, and Hsu (5) and GCC-2, in Rankin (4).

2/ See Chomsky (2), Chapter 4 for a discussion of phrase structure grammars for natural languages.

Phrase structure grammars are well-suited for describing natural languages or sub-languages whose sentences show an internal structure in terms of hierarchical combinations of smaller units (e.g, words or morphemes) into larger units (e.g., phrases and ultimately sentences). Phrase structure grammars are also well-suited for describing such artificial languages as some of the logical calculi and some of the programming languages.

It is now suggested that modified phrase structure grammars can well describe languages other than the one-dimensional languages just mentioned. In particular, modified phrase structure grammars are well-suited for describing the two-dimensional language of Chinese characters.

The following discussion will be in three parts. First, there is a section dealing with the description of the various phenomena and processes in the language of Chinese characters. Second, there is a section in which the grammar constructed to account for this language is characterized and discussed. Finally, there is a section dealing with the evaluation of this grammar in terms of criteria originally proposed for evaluating grammars for natural languages.

Contents

A GRAMMAR FOR COMPONENT COMBINATION IN CHINESE CHARACTERS

by

B. Kirk Rankin III
Stephanie Siegel
Ann McClelland
Center for Computer Sciences and Technology
National Bureau of Standards

and

James L. Tan
George Washington University

ABSTRACT

A linguistic analysis of one aspect of the structure of Chinese characters is presented. The analysis is an extension into two dimensions of a general approach to one-dimensional language study. Results of the analysis are in the form of a three-level generative grammar. The first level formalizes restrictions governing the general complexity of well-formed Chinese characters; the second level formalizes co-occurrence constraints among character components and the particular spatial arrangement of these components in classes of characters; the third level constitutes a procedure for selecting actual components from a lexicon. Finally an evaluation of the grammar is presented, in terms of criteria used in evaluating natural language grammars.

Key-Words

Chinese characters
grammar
generative grammar
component combination
phrase structure grammar
linguistics
frame embedding
two-dimensional languages

I. THE LANGUAGE

A. INTRODUCTORY

This discussion of the language is based on two fundamental notions: component and frame. Chinese characters may be viewed as occupying a hypothetical square;1/ they may also be viewed as being composed of recurring sub-parts - here called components. The segmentation of characters into components segments the square in a variety of ways. These segmentations of the square are here called frames. Frames can thus be viewed as abstract representations of classes of characters.

With the notion of component and frame now introduced, the notion of positioning of components with respect to frames can be discussed. The following position classes are necessary in a description of Chinese character component combination: NORTH, SOUTH, WEST, EAST, BORDER, INTERIOR, and FREE. Thus, the character 相 is segmented as follows:  , and is represented by the frame  ;

1/ This is, in fact, a part of the traditional native Chinese view of Chinese characters.

the component 木 is a WEST and the component 目 is an EAST. The character 家 is segmented as follows:  , and is represented by the frame  ; the component 宀 is a NORTH and the component 豕 is a SOUTH. The character 困 is segmented as follows:  , and is represented by the frame  ; the component  is a BORDER and the component 木 is an INTERIOR. (BORDERS may occupy two, three, or four sides of the square. Other examples are 辶 , 匚 , and 勹 .) Thus, a frame for any two-component character consists of two sub-frames of equal area. The only such frames are  ,  , and  . The single-component character 也 is represented by the frame  , and the component 也 is a FREE.

Based on the discussion so far, in particular the notions of frame and position class, we can represent the above four characters as:  for 相 ,  for 家 ,  for 困 , and  for 也 , where the symbols in the frame are obvious abbreviations for the position class names. In fact, the grammar

provides further information for W, E, N, and S in the form of subscripts on these symbols and these will be discussed just below.

There are, to be sure, characters of greater complexity than those mentioned above. For example, there is the character 後, whose frame can be viewed as being derived from simple frames by means of a process of frame-embedding. 後 is represented by the frame  , which is derived by means of the embedding of  in the EAST part of  . Note that here embedding results in two sub-frames of equal area, the sum of their areas being equal to the area of the sub-frame not embedded in. Frames for other complex characters are derived by means of similar embeddings. 1/

A final word on positioning. Most components occur in many but not all positions. Only a few occur in only one position. As was seen above, the component 木 is a WEST and an INTERIOR; it is

---

1/ Actually, any frame except  is derived via frame-embedding. See page 11, ff.

also an EAST, a NORTH, a SOUTH, and a FREE. The component 冂 is a WEST and an EAST, and nothing else. The component 辶 is a BORDER only.

The broad outline of the phrase structure of Chinese characters arises out of this classification of components into position classes. However, if a grammar were constructed to represent this and only this classification, it would be inadequate because it would generate a great deal of unacceptable output. Further inspection of the co-occurrence properties of components has prompted us to sub-classify each position class according to the dimensions of strength and adjunctiveness.

Before discussing these two co-occurrence properties, we will have to introduce the concept of "in construction with". Two components are in construction with each other if they occur in sub-frames of equal area. For example, in 相 , 木 and 目 are in construction with each other. The frame for 相 , which is  , contains two sub-frames of equal area, one in which 木 occurs and

one in which 日 occurs. A component is in construction with a complex character sub-part if it occupies a sub-frame whose area is equal to the sum of the areas of the sub-frames occupied by the components in that complex sub-part. For example, 後 has the frame  . 彳 is in construction with the complex sub-part 夂 . 彳 occupies the shaded sub-frame:  , whose area is equal to the sum of the areas of the sub-frames occupied by 幺 and 夂 , namely, the shaded sub-frames:  .

We are now ready to introduce the concept of strength which was mentioned just above. A component is strong in a particular position if it can be in construction with many single components and with many complex character sub-parts while in that particular position. It is perhaps the case that many of the strong components in our lexicon are so limited in terms of the number of components with which they can occur that they do not fit this definition, under any reasonable interpretation of the term "many" (even though the notion "can be in construction with" is not intended to be corpus restricted). There

seems, in fact, to be a scale of strength, only the outlines of which are understood currently.1/ At the present time we therefore consider a component to be either strong or not strong with respect to a particular position. For example, 木 is strong in WEST because it can be in construction with 目 (in 相), 乚 (in 札) and with many other single components; and with 吉 (in 桔), 同 (in 桐), and with many other complex character sub-parts. Similarly, 灬 is a strong SOUTH, 刂 is a strong EAST, and 艹 is a strong NORTH. All BORDERS are strong. INTERIORS (since no INTERIOR can occur with many BORDERS) and FREES (since strength is a property of components in construction) are not strong. It is typical that a component which may occur in several positions is strong in one or more positions and not strong in others. A good example is 㔾 . It is strong in SOUTH and not strong in EAST, WEST, and NORTH.

1/ See page 55 for a discussion of this problem. Much more informant work is needed before this limitation can be corrected.

A property of certain strong components is adjunctiveness.1/

A strong component is either an adjunct or it is not regardless of position. That is, an adjunct is an adjunct wherever it occurs. An adjunct is a strong component which cannot occur in FREE. Examples of adjuncts are  and  in WEST, and  in EAST,  and  in SOUTH and  and in NORTH. Adjuncts have the further property that they may never be in construction with adjuncts. Thus the structure  is not an acceptable character since both and are adjuncts. In fact, it was in order to avoid generating such unacceptable structures that the dimension of adjunctiveness was established.

1/ Rankin (4) recognized both strong and non-strong adjuncts (p.32) Further informant work has prompted us to reassign non-strong adjuncts either to the class of strong adjuncts or to the class of FREE components. This solution, incidentally, seems to be not entirely satisfactory.

Since they must be strong components, adjuncts occur only in WEST, EAST, NORTH, SOUTH, and BORDER. Among the strong components in each of these position classes, some are adjuncts and some are non-adjuncts.

Based on the discussion so far, we can give the grammar's full representation of the four sample characters given on pages 2 and 3. For 相, there is 

; for 家, there is 

; for 困, there is 

, and for 也, there is 

.

W_{sf} determines a WEST component which is strong (by subscript s) and non-adjunctive (by subscript f indicating that the component can occur in FREE.) E determines any EAST component by lack of subscript, for any EAST can be in construction with a strong non-adjunctive WEST. $N_{s\bar{f}}$ determines a strong (by s) adjunctive (by $\bar{f}$) NORTH. S_f determines a non-adjunctive (by f) strong or non-strong (by lack of s) SOUTH. (Note how the constraints on the occurrence of strong components and adjuncts are governed by these subscripts.) B, I, and F are subscriptless in the grammar, B always being strong, I and F being insensitive to the strength dimension, but both I and F being non-adjunctive.

One of the processes involved in the formation of Chinese characters is that of component repetition. We here recognize four 1/ types of repetition; horizontal continuous, as in  ; vertical, as in  ; triangular, as in  ; and horizontal discontinuous, as in  . With the exception of the horizontal discontinuous cases, all 2/ instances of repetition function in the grammar as non-strong, non-adjuncts in WEST, EAST, NORTH, SOUTH, INTERIOR, and FREE. The horizontal discontinuous cases function as strong adjuncts

---

1/ For a treatment of repetition which recognizes the existence of more than four types and which is non-generative in nature see Rankin (4). In a generative treatment, such as we are presenting here, we feel that only these four should be recognized.

2/ There are a few exceptions in this observation. Some structures, like and  , could be generated by repetition rules, but are not, because they are strong components and it is desirable that strong components be listed in the lexicon as single components. This inclusion of complex structures in the lexicon is one of the limitations in the current treatment.

in BORDER and do not occur in any other class.

These instances of repetition are represented in the grammar as follows:  as  (horizontal continuous),  as  (vertical),  as  (triangular), and  as  (horizontal discontinuous).

B. DETAILS

What follows now is a rather more detailed account of some of the phenomena and processes that have just been introduced.

The process of frame-embedding has been mentioned briefly. In this detailed treatment, we will use the notions of host-frames, occupant-frames, and product-frames.

Further, we will highlight the process of frame-embedding by ignoring all marks which in the grammar may occur in frames or sub-frames - except for the mark H (for "host"), which is the recursive element in the grammar that activates the process of frame-embedding. We will refer to the sub-frame containing H as the "H sub-frame".

We will use the formula:

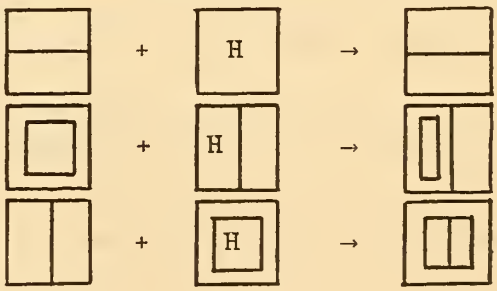
Occupant-frame + Host-frame → Product-frame

which is read "Embed the occupant-frame in the host-frame to produce the product-frame". The host-frame always has H in one of its sub-frames (or in its only sub-frame in 

). This one sub-frame may be thought of as the hypothetical square or size and shape transformations of it. The occupant frame (always of the form: 
, 
, 
, 
) may or may not have H in one of its sub-frames. The product-frame of any embedding will have H in one of its sub-frames if the occupant-frame that produced it has an H sub-frame. The product-frame of the first embedding is always identical to the occupant-frame of the first embedding.

Sample Single Embeddings:

Occupant-frame + Host-frame → Product-frame



In general, H may occur in any sub-frame except the border sub-frame, since border is not a possible transformation of the hypothetical square. (For purposes of this discussion, "border" sub-frames can be filled by components from both the BORDER position class and the horizontal discontinuous repetition class.) Thus there are no complex borders.1/

If a product-frame contains an H sub-frame, that product-frame is then redefined as a host-frame. This redefinition is shown in the following example.

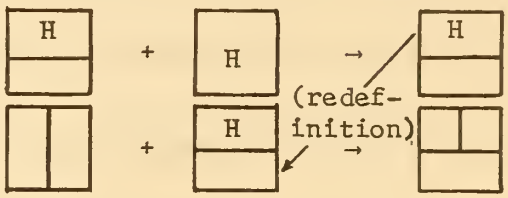
1/ Actually there are some BORDERS which appear to be complex. Examples are  and  . Complexity in BORDER is extremely rare and we circumvent the problem by listing such instances as the two above as single components in the lexicon. Further, all instances of horizontal discontinuous repetition appear to subdivide the border sub-frame. However, they do conform to our general definition of BORDER in that they occupy two sides of the hypothetical square (see page 3).

Sample first embedding followed by second embedding:

Occupant-frame + Host-frame → Product-frame

1st embedding:

2nd embedding:



Note that in the first embedding one of the sub-frames of the occupant-frame contains H. It is this occurrence of H that both allows the second embedding to take place and also marks the place where the second is to take place. If there were no H sub-frame in the occupant-frame, no second embedding could take place.

Further, no occupant-frame in the grammar may contain more than one H sub-frame. This restriction imposes a "blocking" of embedding.

Once an occupant-frame has been embedded in a host-frame in any embedding, the sub-frame which is equal in area to the H sub-frame is permanently blocked from embedding. Another way of saying this is that the grammar does not allow any host-frame to contain more than one H. In the following sample products the shaded sub-frames are blocked from embedding:

There are further constraints on the process of frame-embedding. First, there is an upper limit of four embeddings in the generation of any character--giving rise to a maximum of five sub-frames per frame. Second, there may be only two BORDERS in the generation of a frame for any character. (With respect to this constraint, cases of horizontal discontinuous repetition are treated as BORDERS). Finally, there is a limit of four sub-frames per frame if any sub-frame is occupied by a case of horizontal continuous, vertical, or triangular repetition. This is so because these cases of repetition are in a sense multi-componential. In contrast, cases of horizontal discontinuous repetition function as a single-component BORDERS.

Blocking of embedding is closely related to the functioning of strong components in the following sense. A strong component must occur in every blocked sub-frame of a frame. Further, a terminal frame contains within it two sub-frames of equal area (shaded in the following examples:  ,  , and ). A strong component must occur in at least one of these two sub-frames of equal area. It follows that an output character contains at most one non-strong component.

With respect to adjuncts, they may occur wherever any strong component may occur with one exception. There are no cases of adjuncts being in construction with adjuncts. So, not both of the sub-frames of equal area may be occupied by adjuncts. That is, for example, not both shaded sub-frames in any one of the following frames may be occupied by adjuncts:  ,  ,  . It follows that in any output character there must be at least one non-adjunct.

Finally, not both sub-frames of equal area in any frame may be occupied by the same component. If it were otherwise, such output characters as  would be ambiguous (or derivable in more than one way). One derivation of  would be an EAST plus a WEST with  filling both positions; the other derivation would be the correct one: that  is a case of horizontal continuous repetition.

II. THE GRAMMAR

A. INTRODUCTORY

The grammar is set up to generate output characters in three stages. The grammar also imposes a hierarchical classification on the

set of all output characters, and this can best be seen by first discussing the three stages of the generative process.

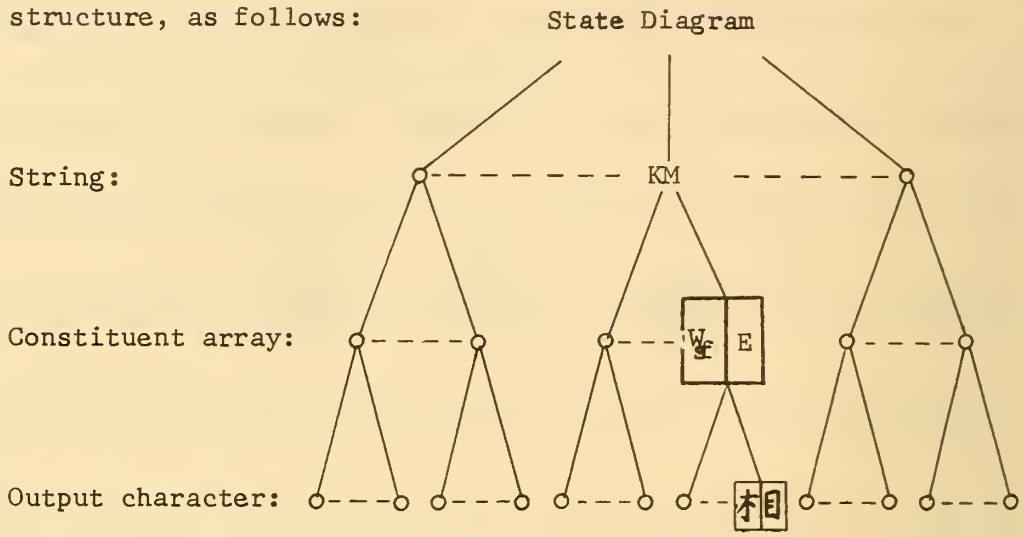
Stage one is a string of symbols generated by a finite state diagram. This string controls the number and type of embeddings in the output character. 1/ Stage two is a two-dimensional constituent array (generated via the process of frame embedding) which determines the two-dimensional phrase structure of the output character and governs the positioning, strength, and adjunctiveness constraints on the components to be selected. Stage 3 is the output character itself: a frame filled with components selected (via a lexical look-up procedure) from the lexicon.

The hierarchical classification is as follows. Any stage 1 string determines a number of constituent arrays and any stage 2 frame determines a number of output characters. Thus, the set of all strings

---

1/ The various constraints and phenomena discussed here in terms of the grammar have already, of course, been discussed in Section I, to which the reader is now referred.

permitted by the finite state diagram ultimately determines the set of all output characters. The hierarchy referred to can be depicted as a tree structure, as follows:



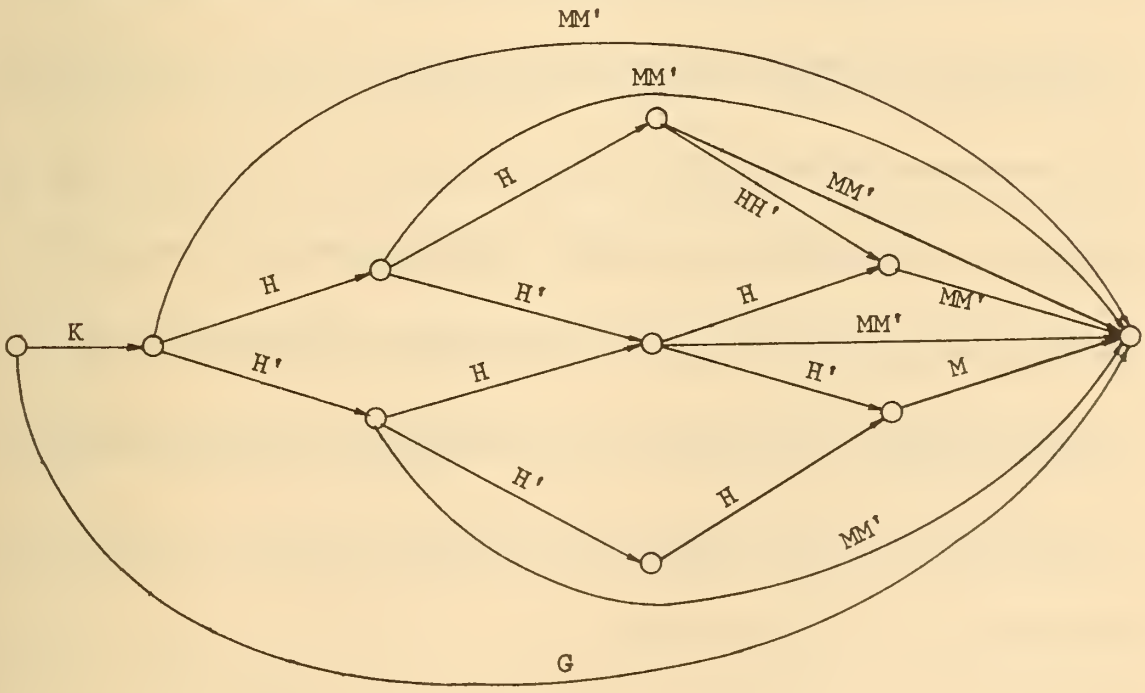
The path from KM to  to the output character  gives the derivation of this output character in terms of how the grammar generates it. The dashed line-----between nodes in the tree indicates that in general there are many (though finitely many) other nodes at the same stage dominated by the same node of the previous stage. 1/

1/ Or in the case of Stage 1, other nodes dominated or generated by the state diagram.

B. DETAILS

To produce Stage 1:

Stage 1 is an output string from a finite state diagram, which consists of a number of states with input and output transitions. One state is distinguished by having only output transitions, and this is the initial state. One state is distinguished by having only input transitions, and this is the final state. Along the transition arrows there are either one or two output symbols (K, G, H, H', M, M'). 1/



1/ These symbols are explained on pages 21 and 22.

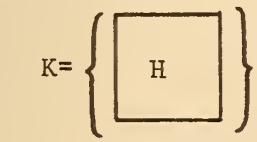
The initial state is entered and a process of random transition from the initial state to the final state (perhaps through intermediate states) is begun. At each transition from state to state one output symbol is produced. If there is only one symbol along the transition arrow, that symbol is necessarily chosen. If there are two symbols along the transition arrow, one is chosen at random. The next transition causes a symbol to be produced to the right of the last-produced symbol, and so on. As the final state is entered, the last (rightmost) symbol is produced, and the process terminates, producing the string which is Stage 1: $x_1 \; x_2 \ldots x_{n-1} \; x_n$. 1/

1/ Since n cannot exceed 5, it might seem inappropriate to set up the general case for this and all the other stages and processes in the grammar. However, by so doing, we have achieved ease of manipulation and generality for such future applications as the study of frame-embedding as an isolated process.

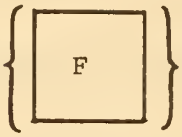
Sample output strings of the form $x_1 x_2 \ldots x_{n-1} x_n$, generated by the state diagram, are: KM' (by taking the topmost path through the diagram), G (by taking the bottom-most path through the diagram), KHM and KHHM (by taking two of the middle paths through the diagram).

The nature of any string generated by the state diagram determines the size and complexity of the final output character. For example no string may exceed length five, and this restriction guarantees that no output character shall have more than five components.

Each symbol in the string represents a set of frames (see the correspondence table on pages 23 and 24). The frames of each set have certain properties in common, related to the embedding process, and to the border-non-border distinction. Border here includes not only the position class BORDER but also the discontinuous repetition cases, which are represented by the same frame configuration as the BORDER position class and which function in the grammar subject to the same restrictions.

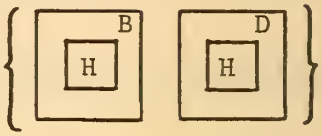
K=  is the one frame which is a host only. That is, it

is the initial step with respect to frame embedding.

G= 

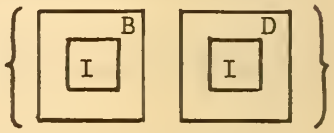
is the one case of non-embedding. It is neither a host nor an occupant.

H= 

, and H'= 

are the sets of frames which function as both hosts and occupants. They are the intermediate steps with respect to frame embedding. They are distinct sets in that H is non-border and H' is border.

M= 

, and M'= 

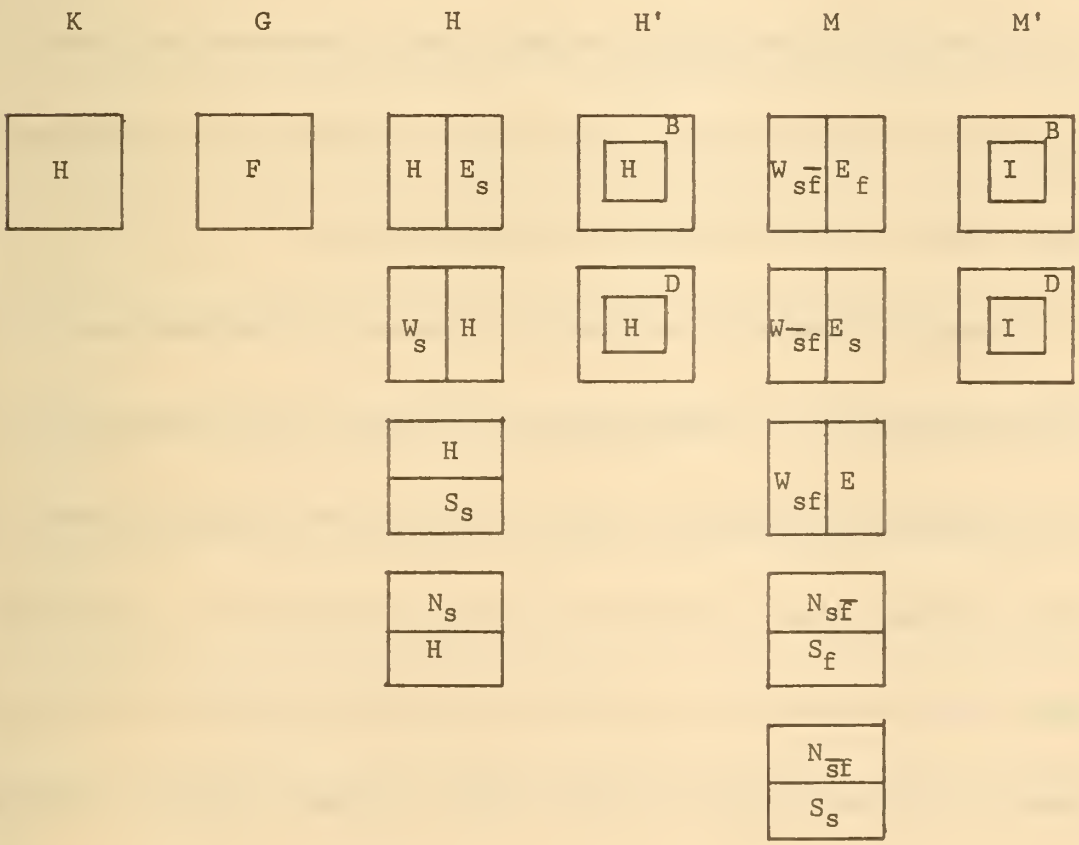
are the sets of frames which function as occupants only. They are the terminal steps with respect to frame embedding. They are distinct in that M is non-border and M' is border.

Finally, the state diagram (by distinguishing H from H' and M from M' and by prohibiting certain transitions) guarantees that no output string may contain more than two instances of "symbol-prime". Thus it guarantees that no output character may contain more than two borders.

To produce Stage 2:

And now, before discussing the process that generates Stage 2, we will display the output symbol-to-frame correspondence table referred to above. Each output symbol of the state diagram corresponds to a set of frames. These output symbols appear as column headings. The filled frames are listed underneath.

Table of Correspondences Between Output Symbols from Stage 1 and Frames:



K G H H' M M'

There are two steps in the process that generates a Stage 2 constituent array from Stage 1 string. First, a sequence of frames is created. Second, if there is more than one frame in the sequence, these frames are compiled into a constituent array.

First, each symbol in any string of Stage 1 corresponds to a column in the table of correspondences. To obtain the sequence of frames, select at random one frame from each column indicated by the Stage 1 string. Thus a sequence of frames is created: $E_1 \ldots E_n$. The frames of the sequence $E_1 \ldots E_n$ consist of sub-frames which contain either or both of two types of marks: (1) the mark H (for Host) which sets up the process of frame embedding and (2) the marks called con-

stituents which will later be used in the process of component selection. At this stage in the process, since we are concerned with frame-embedding, only the frame configurations and the mark H will be relevant to our discussion.

Second, when there is more than one frame in the sequence, the process of frame-embedding is initiated so as to compile the sequence of frames into a constituent array.

To produce the constituent array, apply the frame-embedding formula:

Occupant-frame + Host-frame → Product-frame

to the sequence of frames:

$E_1\ E_2\ \ldots\ E_n \qquad 1 \leq n \leq 5$

The frame-embedding process may be viewed as a recursive process initiated by E_1 which is the hypothetical square, or host-frame of the first embedding. Hence we will set $E_1 = H_1$; H_1 for the first host-frame. The subsequence $E_2 \ldots\ldots E_n$ is the set of occupant-frames which are called upon during the embedding process, one occupant-frame necessary for each embedding. Note that where there is only one frame

in the sequence, it is not possible to define either host or occupant frames.

Embed occupant-frame E_2 in host-frame H_1 to obtain product-frame H_2. Redefine H_2 as the host-frame of the second embedding. Embed E_3 in H_2 to obtain product-frame H_3. Continuing in this way, embed occupant-frame E_i in host-frame H_{i-1} to obtain frame H_i, etc., until E_n is embedded in H_{n-1} to obtain the final product-frame H_n, which is defined to be the constituent array.

A distinction should be made between H and H_i, i=1, 2, ..., n. H_i is a well-formed frame with an H in one sub-frame. H is a mark which indicates the precise location of embedding. $H_1 = E_1$ is the hypothetical square with an H inside, 

. Size and shape transformations of this hypothetical square are created automatically as we go through the recursive process. In other words, the sub-frame containing an H, of a product-frame H_i, is considered to be the transformed hypothetical square.

H, which is present in one sub-frame of each E_i except E_n and each

H_i except H_n (constituent array), acts as the indicator of the sub-frame in which the next embedding is to occur. In this way H is the activator of the frame-embedding process. When an H fails to appear in a product-frame after an embedding has taken place, that product-frame cannot be redefined as a host-frame, and the process stops. That final product-frame where H fails to appear is, of course, H_n, the constituent array.

This discussion of the recursive process of embedding occupant-frame is host-frame to produce product-frame and redefining product-frame of the previous embedding as host-frame and embedding again, until the constituent array (H_n) is obtained may be expressed in compact form as the following sequence of formulas where + is the embedding operation:

$$E_2 + H_1 \rightarrow H_2$$

$$E_3 + H_2 \rightarrow H_3$$

$$\vdots \qquad \vdots \qquad \vdots \qquad 1 \leq n \leq 5$$

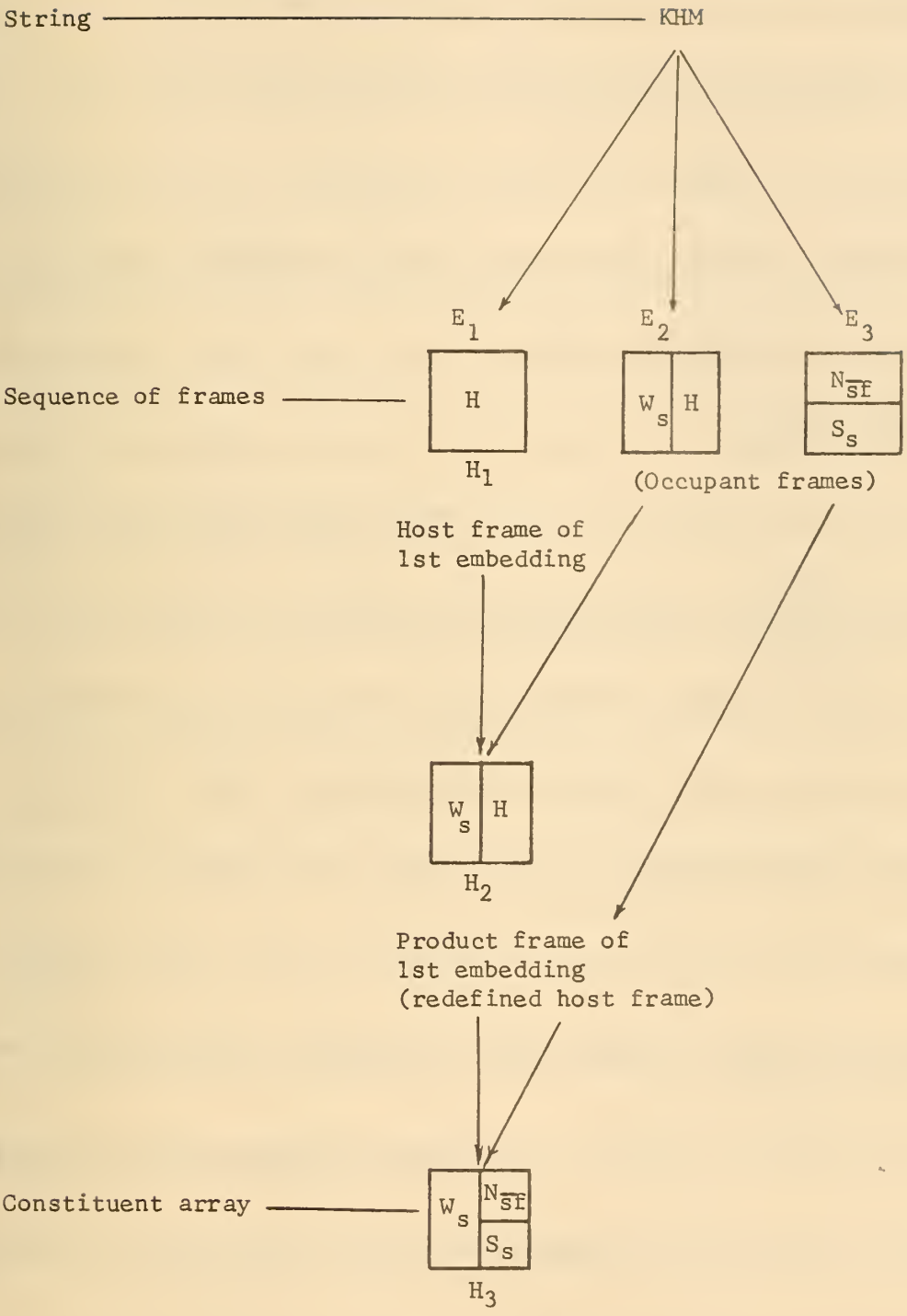
$$E_i + H_{i-1} \rightarrow H_i$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$E_n + H_{n-1} \rightarrow H_n$$

For each occupant-frame + host-frame embedding a product-frame is produced. This product-frame is then the host-frame of the next formula in the sequence. The recursive nature of the embedding process is evidenced by the generation of a sequence of formulas in which there is a successive production of product-frames and redefinition as host-frames. The sequence ends with the production of the final product-frame H_n, the constituent array.

An example of the generation of a constituent array from an output string is now given:


String
KHM
E1
E2
E3
Sequence of frames
H
Ws
H
N sf
Ss
H1
(Occupant frames)
Host frame of
1st embedding
Ws
H
H2
Product frame of
1st embedding
(redefined host frame)
Constituent array
Ws
N sf
Ss
H3
Product frame of
2nd embedding

The string KHM gives the sequence of frames E_1, E_2, E_3, and these frames then enter into the embedding process:

Occupant-frame + Host-frame → Product-frame

$$E_i + H_{i-1} \rightarrow H_i$$

The process is repeated in this case until E_3 is embedded in H_2 to produce H_3, the constituent array.

To produce Stage 3:

We will now describe the process which generates an output character from a Stage 2 constituent array. The process selects a component in the lexicon for each constituent which occurs in the constituent array of Stage 2. The constituents are handled in order according to the size of the sub-frames that they occupy - from largest to smallest. For constituent arrays having two sub-frames of equal size a diagonal order, from top-left to bottom right is followed.1/ Specifically, the

---

1/ Actually, this corresponds to the traditional stroke order. A person trained in traditional calligraphy writes most Chinese characters starting at top-left of the hypothetical square, and finishing at bottom-right.

"north" sub-frame precedes the "south" sub-frame; "west" precedes "east"; and "border" precedes "interior."

These constituents are of the form Σ_{α}, where Σ is a symbol corresponding to one of the position classes, or to one of the repetition classes, and α is a subscript which gives information on strength and adjunctiveness. The combination of Σ_{α}'s allowed guarantees that each output character has at least one non-adjunct, and at most one non-strong component (see pages 15 and 16).

For instance, for the Stage 2 constituent array 

, the constituents are (in order) W_s, N_{sf}, and S. In the case of W_s, we have Σ as W and α is the single subscript s. For N_{sf} we have Σ as N and α is the double subscript sf. Finally, S is an example of Σ with α as the null subscript.

Next it will be necessary to describe the lexicon, an essential tool used in the creation of stage 3.

The lexicon is a table with components 1/ down the side for each row, symbols across the top for column headings, and tallies at certain row-column intersection points signifying that the components may occur in the positions signified by column headings W, E, N, S, B, I, or F, 2/ or may undergo the repetition processes signified by V, C, T or D. 3/

1/ Each of the components has a unique number associated with it. These numbers are not a part of the formal apparatus of the grammar. We are including them for the purpose of making lexical look-up easier for those readers familiar with Chinese characters. The component numbers are of the following form: S.T.N, where S is the number of strokes in the component (S=1-19), T=1-8 characterizes the types of the last stroke (last in the sense of traditional stroke order) of the component: 1 for horizontal, 2 for vertical, 3 for dot, 4 for \ -like, 5 for / -like, 6 for / -like, 7 for hooked, and 8 for multi-directional. N is the number of the component in the sub-list.

2/ These column headings are abbreviations of the position classes discussed in Section I. W=WEST, E=EAST, N=NORTH, S=SOUTH, B=BORDER, I=INTERIOR, and F=FREE.

3/ These column headings are abbreviations of the repetition processes discussed in Section I. V=Vertical repetition, C=horizontal Continuous repetition, T=Triangular repetition, and D=horizontal Discontinuous repetition.

Sample Lexicon:

(The superscripts on some of the tallies are to be ignored. Their significance will be explained on page 74.)

		W	E	N	S	B	I	F	V	C	T	D
3.2.11	彳	s										
3.3.3	幺	$\bar{s}$	$\bar{s}$	$\bar{s}$	$\bar{s}$			x		x		x
4.4.4	攵		s	s^5	s^5	x^5		x				
6.3.13	糸	s^5			s		x	x				x
6.8.3	先	$\bar{s}$	$\bar{s}$				x	x		x		
7.1.5	言	s	$\bar{s}$		s		x	x		x		
7.3.3	貝	s	$\bar{s}$		s		x	x		x	x	

In general, the lexicon will be used in the following way. For any constituent Σ_α, we examine the column specified by Σ, where Σ is any one of the eleven column headings, to determine the set of rows which have tallies corresponding to the subscript α.

If α is f or includes either f or $\bar{f}$ (such as in S_f or $N_{s\bar{f}}$), we must further determine the subset of these rows which also have tallies in column F. Note that the column F has two functions. First, it is one of the columns denoted by Σ. Second (as in this part of the

discussion), it is a column to be searched for adjunctiveness information in conjunction with the search in column Σ.

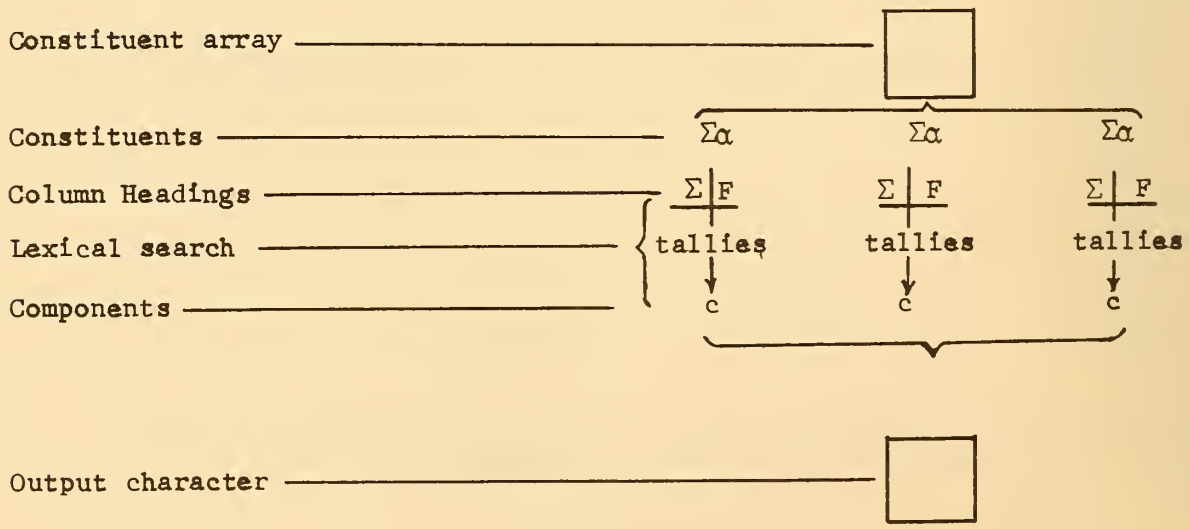
The following chart summarizes the types of subscripts which may occur in each constituent mark, and the types of tallies which match the subscripts.

Subscripts*	Tallies activated by α in: Column Σ	Column F	Explanation
$\emptyset$	s, $\overline{s}$, x		The null subscript specifies a row which has any tally in the Σ column.
s	s		The s subscript specifies a row which has an s tally in Σ.
f	s, $\overline{s}$	x	The f subscript specifies a row which has both an x tally in column F, and either an s tally or an $\overline{s}$ tally in column Σ.
$\overline{s}$f	$\overline{s}$	x	The $\overline{s}$f subscript specifies a row which has both an $\overline{s}$ tally in column Σ, and an x tally in column F.
sf	s	x	The sf subscript specifies a row which has both an s tally in column Σ, and an x tally in column F.
s$\overline{f}$	s	$\overline{x}$	The s$\overline{f}$ subscript specifies a row which has both an s tally in column Σ, and no x tally in column F.

* Note that $\overline{s}\overline{f}$ does not occur. See page 8 footnote.

For each of the first two cases (Ø and s) a set of rows is determined by searching column Σ. For each of the last four cases (f, $\overline{s}$f, sf, and s$\overline{f}$) a set of rows is determined by searching column Σ, and further, a subset of this set is determined by searching column F. At this point, a single row is randomly selected from the set or subset thus determined, and the component (if the symbol of the constituent is W, E, N, S, B, I, F) or repetition of the component (if the symbol of the constituent is V, C, T, or D) on that row replaces the constituent being processed. This restriction prevents the class of ambiguous derivations mentioned on page 16.

The lexical search procedure can be summarized in the following way:



An example of the generation of an output character from a constituent array is now given.

The stage 2 constituent array, 

, is processed to generate an output character in the following fashion.

First, the constituents are ordered according to the decreasing size of the sub-frames that they fill. Since there are two terminal sub-frames of equal size, the order of these last two is determined by the diagonal line procedure.

Then, the sets or subsets of rows which have tallies corresponding to each constituent are determined. In other words, column W will be searched for the set of rows which have tallies corresponding to subscript s. Column N will be searched for the set of rows which have tallies corresponding to subscript $\bar{s}$, and further, we determine the subset of this set of rows having x tallies in column F. Finally column S will be searched for the set of rows which have tallies corresponding to subscript s. To illustrate what is taking place at this point, the tallies corresponding to W_s will be circled once in

the sample lexicon, the tallies corresponding to N_{-sf} will be circled twice, and the tallies corresponding to S_s will be circled three times.

		W	E	N	S	B	I	F	V	C	T	D
3.2.11	彳	(s)										
3.3.3	幺	$\bar{s}$	$\bar{s}$	($\bar{s}$)	$\bar{s}$			x		x		x
4.4.4	攵		s	s^5	(s^5)	x^5		x				
6.3.13	糸	(s^5)			(s)		x	x				x
6.8.3	先	$\bar{s}$	$\bar{s}$				x	x		x		
7.1.5	言	(s)	$\bar{s}$		(s)		x	x		x		
7.3.3	貝	(s)	$\bar{s}$		(s)		x	x		x	x	

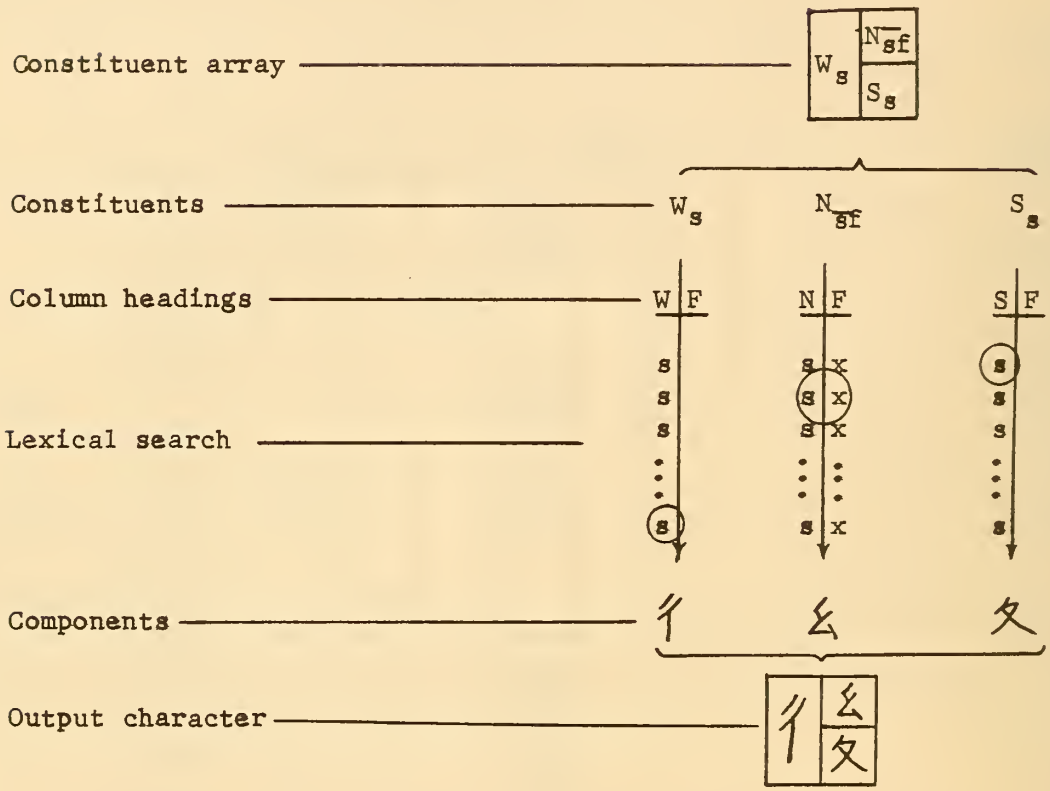
Then one of the rows in each set or subset is picked at random, with the constraint that the row selected for N_{-sf} may not be the same as the row selected for S_s, since they are in construction with each other. Again, one, two and three circles will be used to illustrate

the tallies corresponding to the sets or subsets of tallies.

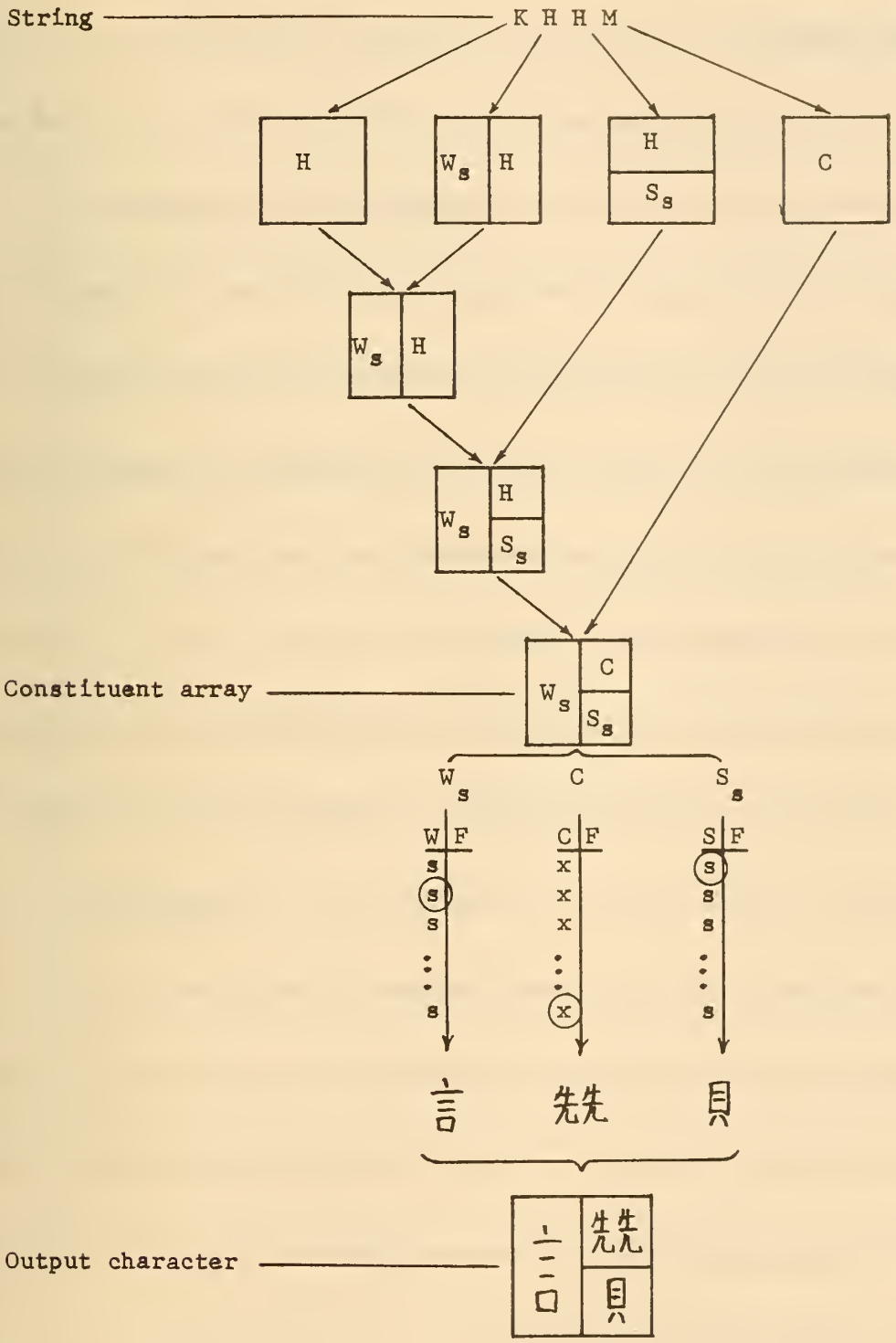
		W	E	N	S	B	I	F	V	C	T	D
3.2.11	彳	(s)										
3.3.3	幺	$\bar{s}$	$\bar{s}$	($\bar{s}$)	$\bar{s}$			x		x		x
4.4.4	攵		s	s^5	(s^5)	x^5		x				
6.3.13	糸	s^5			s		x	x				x
6.8.3	先	$\bar{s}$	$\bar{s}$				x	x		x		
7.1.5	言	s	$\bar{s}$		s		x	x		x		
7.3.3	貝	s	$\bar{s}$		s		x	x		x	x	

Finally, the component which occurs on the row of the tally just selected replaces Σ_α.

This example can also be represented in the following way:



Sample Derivation from Total Grammar:



III. EVALUATION

A. GENERAL REMARKS

How are grammars for component combination to be evaluated? There are two criteria which are used here: completeness and tightness. These criteria are well-known in the literature of natural language study. Tightness and completeness taken together in fact constitute the transformationists' "minimal requirements" on natural language grammars - that they should generate all and only the grammatical sentences of the languages under study. 1/

There are, of course, other conditions that might be put on a successful grammar of component combination - conditions having to do with simplicity, with the choice of a grammar model, with relative correctness of structural descriptions assigned by the grammar to output objects, with integratability of the grammar with other grammatical processes necessary in the total description of Chinese

---

1/ See, for example, Bach (1), page 5.

characters, with possible applicability of the grammar or grammar model to other two-dimensional languages, and so on. In this note, however, we will be concerned only with completeness and tightness.

GCC-3 constitutes an attempt to construct a simple grammar which is balanced between completeness and tightness. It seems to be nearly as simple as possible in terms of the size of the lexicon of components and the number and complexity of grammar rules. It is neither 100 per cent tight nor 100 per cent complete, but it would seem that any adjustment to make it tighter would result in less simplicity and/or completeness, and that any adjustment to make it more complete would also result in less simplicity and/or tightness.

1. Completeness

A complete grammar for the language of Chinese characters is one which generates all of the well-formed characters in the language. This is an extremely difficult condition to satisfy, if only because it is so difficult to determine in many cases exactly what is and what

is not in the language. 1/ The problem here is probably no worse than the problem in natural language study of deciding for border-line strings whether or not they are grammatical sentences of the language under study.

A less demanding condition is that a successful grammar generate all the well-formed characters in some target corpus.

Such a grammar will be called "corpus-complete", and a grammar that is complete with respect to the language will be called "language-complete." Only corpus-completeness will be used in evaluating GCC-3. The grammar might then be tested against other corpora for corpus-completeness. In this way, the language-complete grammar would be approximated by the grammar which is corpus-complete for several corpora, but it would never be known whether the grammar actually attains language-completeness.

1/ Decisions on what is and what is not in the language are currently based on judgments given by the informant.

2. Tightness

A tight grammar for the language of Chinese characters is one which generates only well-formed Chinese characters. Tight grammars are easy to come by. A trivial example would be a "grammar" which lists some small number of actually-occurring characters.

In a way analogous to the completeness situation, we may speak of corpus-tightness and language-tightness. If a GCC is language-tight, then it generates only those characters in the target corpus. It turns out that any serious GCC, i.e., one which attempts to reveal the internal structure of the characters (and does not simply list characters), is almost bound not to be corpus-tight. This is an empirical observation, and an analogous observation probably holds for any serious attempt at grammar construction for natural language corpora. Only language-tightness is used in this evaluation.

Lack of completeness implies insufficient output, or under-generation. Lack of tightness implies too much output, or over-generation. We have discussed over-generation of output characters so far, but

there is another type of over-generation, and that is the over-generation of derivations per character.

An output object from any grammar is said to be ambiguous if the grammar can generate it in more than one way, i.e., provides more than one derivation of it. The problem of whether or not there are ambiguous characters, or what would be meant by "ambiguous character" have by no means been solved. However, it seems that every character in the language is uniquely segmentable into components by the informant, and therefore our current position is that there are no ambiguous characters in the language of Chinese characters. Consequently, a grammar that generates any character is more than one way "over-generates".

B. EVALUATION OF GCC-3

1. Completeness

GCC-3 appears capable of accounting for 94% of the acceptable

characters in Mathews'. 1/ The characters that GCC-3 cannot account for fall into seven known small classes. 2/ The grammar could be adjusted so as to account for these classes, but only with a great loss in tightness and/or simplicity.

First, there are those characters which contain more than five components. Since the state diagram in the grammars sets a maximum of five components per character, such characters cannot be accounted for. Examples are: 攝 (扌 , ++ , 曰 , 勹 , 人 , and 乚) and 爇 (++ , 土 , 八 , 土 , 丸 , and 川).

1/ To arrive at an indication of how complete GCC-3 is, we conducted the following test. We selected (via a random number table) 200 characters from Mathews'. We then attempted to generate each character from GCC-3. We found that GCC-3 could generate 188 of the 200--some in more than one way. (See the discussion of ambiguity on pages 56-62).

2/ With the exclusion of possible clerical errors.

Second, there are those characters which are (or which contain sub-parts which are) complex in both NORTH and SOUTH (represented by the frame 

, or in both EAST and WEST (represented by the frame 

). Examples of 

are: 響 (鄉 in construction with 音) and 馨 (殼 in construction with 香). Examples of 

are 虢 (孚 in construction with 虎) and 鬻 (高 in construction with 粥).

Third, there are those characters which contain weak components in strong positions: either a weak component in construction with a weak component or a weak component in construction with a complex character sub-part. Examples of characters having a weak component in construction with a weak component are: 舅 , where 臼 and 男 are in construction but both weak, and 甥 , where 生 and 男 are in construction but are both weak. Examples of characters in which there is a weak component in construction with a complex character sub-part are: 疏 , where 正 is weak, but in construction with 流 , and 虧 , where 亏 is weak but in construction

with 虐 .

Fourth, there are those characters which contain more than two BORDERS. Since the state diagram prohibits any output character from having more than two BORDERS, such characters cannot be accounted for by the grammar. Examples are 遍 , which contain the three BORDERS 辶 , 户 , and 冂 , and 遏 , in which there are the three BORDERS 辶 , 勹 , and 匕 .

Fifth, there are those characters in which there is a BORDER in construction with a non-FREE component. Since in the grammar, the set of INTERIORS (those components which are in construction with BORDERS) form a proper subset of the set of FREES, BORDER plus non-FREE structures cannot be accounted for by the grammar. The only known examples are 衍 , in which the INTERIOR 氵 (in construction with BORDER 彳亍) is an adjunct, and 均 , in which the INTERIOR 冫 (in construction with BORDER 勹) is an adjunct.

Sixth, there are those characters which contain the form 一

(This form is not to be confused with —— , a component in the lexicon, and for structural reasons cannot be considered simply a shortened variant of —— .) There are doubts as to whether this form should be considered a component. If it were, it would seriously affect the tightness of the grammar. Therefore, all that can be said is that there are characters which contain it and cannot be accounted for by the grammar because the grammar does not list it as a component. Examples are 滅 , where — is between 戊 and 火 , and 睿 , where — is between 𠂤 and 八 .

Seventh, there are a few characters, which if segmented into components, would yield extremely irregular frames. 1/ Examples are: 𠓛 , whose frame would be something like  , and

---

1/ These constitute a type of component superimposition. Component superimposition is discussed in Rankin (4). However, the type of superimposition discussed there does not include these characters.

 , whose frame would be something like  . Since the only frames in the frammar are  ,  ,  , and  or results of combining these, characters segmentable into frames like those above are not accountable for by the grammar.

There are two final observations. First, some characters which cannot be accounted for by the grammar fall into more than one of the classes discussed above. Second, some characters which appear to fall into one of the above classes are actually accounted for by the grammar. This is so, because there is always the option of listing as a single component those complex sub-parts which give trouble. For example, the complex sub-part  is listed as a component, even though it might fall into class seven above, and the complex sub-part  is listed as a component even though it might be segmented into plus  , which segmentation would cause characters containing  to fall into class three above. Generally, decisions on segmentation are motivated by three considerations: (1) a structure is segmented into two components if the two components either must not

touch or need not touch; (2) a structure is segmented into two components if the segmentation generally corresponds to one of the three recognized frame types:  ,  , and  ; (3) a structure is segmented into two components if each sub-structure can occur in different environments. 1/

2. Tightness

A. Over-generation of output characters

Rather than sampling random output with a view to assigning a tightness percentage, we feel it is more to the point ot sub-classify the unacceptable output on linguistic grounds. The reason is that potentially there exist infinitely many grammars equivalent to GCC-3 in the sense that they all generate the same language, all having different tightness percentages associated with them. For example, we could construct a GCC-4 from GCC-3 by substituting the following set of trans-

---

1/ For a previous and more detailed discussion of segmentation, see Rankin (4), chapter 4.

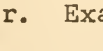
itions for the bottom-most transition which produces the single symbol G.

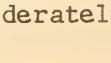
Then there would be added to GCC-3 rules by which G1 would ultimately determine the subclass of the current FREE class which contain, e.g., fewer than five strokes: G2, G3, and G4 would respectively determine subclasses containing 5-to-10 strokes, 11-to-15 strokes, and more than 15 strokes. GCC-4 would generate precisely the same language as that generated by GCC-3, but random output from GCC-4 would contain a higher percentage of acceptable characters than random output from GCC-3. This is so because GCC-4 has a higher probability of generating single-component characters, and all of these are by definition acceptable.

There are three classes of unacceptable output. The first class of unacceptable output contains characters of excessive size. Here,

there are three sub-classes. First there is a sub-class which contains characters of excessive overall size. Examples are (1) 1/ and (2). The problem with these output characters is that the frame for each has the maximum number of sub-frames, and each sub-frame is occupied by a component which is itself very large. Second, there is a sub-class containing characters which are excessively horizontal. Examples are (3) and (4). The problem here is that the frames for these characters are of the maximum horizontality allowed by the grammar, and the sub-frames are filled with "very horizontal" components. Then, there is a corresponding excessively vertical sub-class. Examples are (5) and (6). The frames for these characters are of the maximum verticality allowed by the grammar, and the sub-frames are filled with "very vertical" components.

1/ This output character and the following five (where there are numbers in parentheses) are found in Appendix D.

The second class of unacceptable output contains characters in which there are co-occurrence problems. That is, it is not the case that, say, every WEST can occur with every EAST (even given the strength and adjunctiveness constraints), and so on for other pairs of classes. The one pair of classes which gives rise to the greatest number of co-occurrence problems is the BORDER-INTERIOR pair, but there are co-occurrence problems for every pair. Examples are  ,  , and  , in which  and  , and  , and  and  , respectively simply should <u>not</u> co-occur.

Finally, there is a class of unacceptable output characters which contains certain components in strong positions, these components being only moderately strong. There seems, in fact, to be a scale of strength, Among WESTS, for example, the scale ranges from, say  (very strong) through  (moderately strong) to  (weak). Currently in the grammar, the class of strong components includes very strong and moderately strong components. This blurring of the scale of strength gives rise to such unacceptable output characters as 

and 傘 , where 斗 and 幸 are only moderately strong (我 and 几 being weak).

B. Over-generation of Derivations

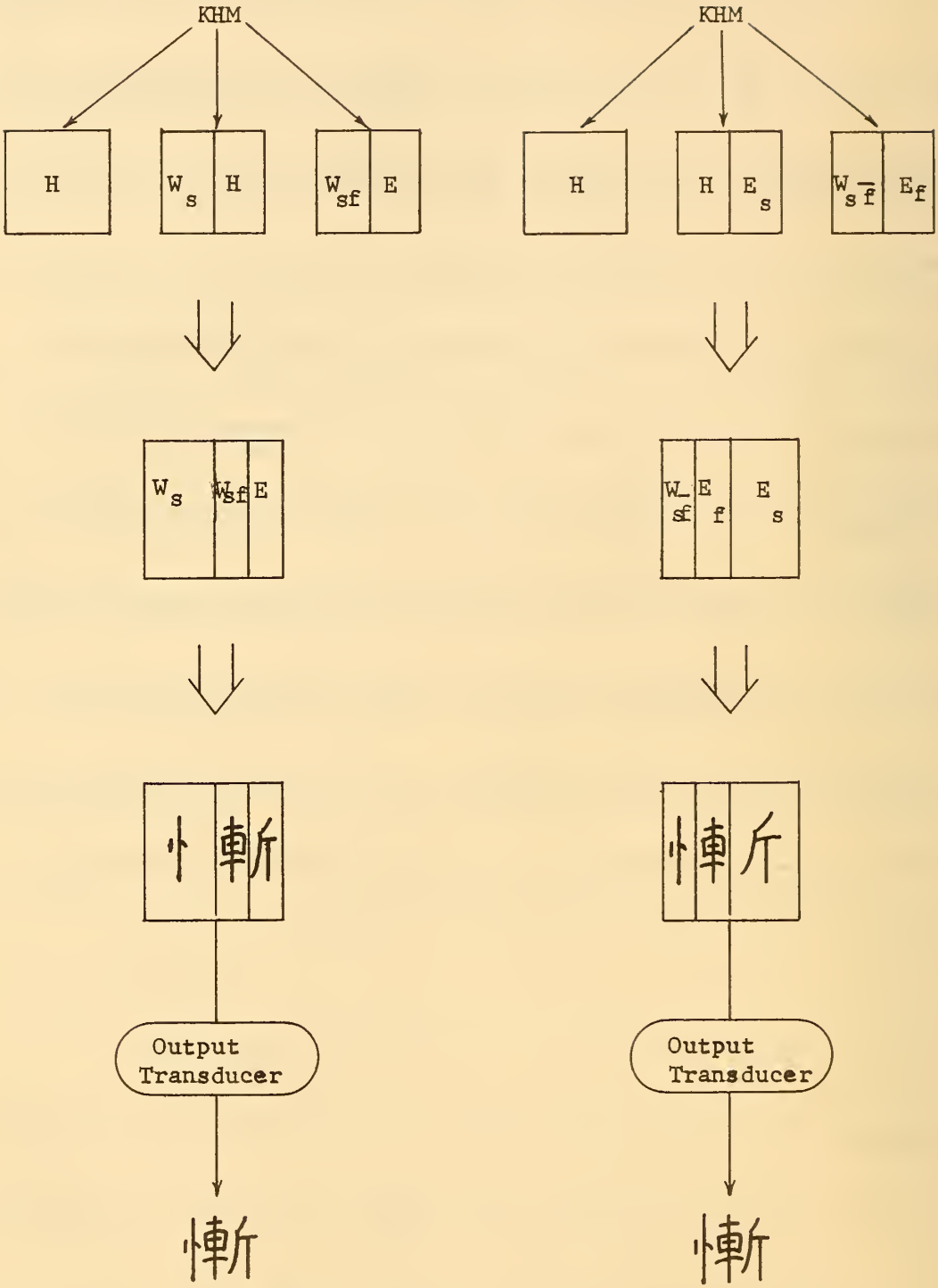
There are three types of ambiguous characters. Actually, ambiguity is not apparent at the output character level, but only at the post-transducer level. This is because our current output characters contain parts of their deriviations - their frames. For example, at the output character level, the following are distinct:  and  . However, at the post-transducer level both would be realized as 忻 , which is for that reason ambiguous. In this treatment, we will have to discuss ambiguity as if we already had the output transducer referred to on page iv and we will use the term "character" to mean "post-transducer character."

First, some characters which are (or contain) horizontal arrays of three or more components are ambiguous with respect to the grouping of components. For example, 忻 has two derivations, one of which imposes a 忄 plus 斤 grouping, the other of which imposes a

 plus  grouping. Then some of the characters which are (or contain) vertical arrays of three or more components are ambiguous. The sub-part  of the character has two derivations, one of which imposes a plus grouping, the other of which imposes a  plus  grouping.

This type of ambiguity is introduced in the first step of the process that generates Stage 2, that is, in the selection of frames from the table of correspondence. It is possible for two distinct sequences of frames (which are generated by the same output string) ultimately to determine the same character. This is shown in the following two derivations of  . In these, and the derivations to follow, the single arrow → is used for "directly determine" and the double arrow ⟹ for "ultimately determine".

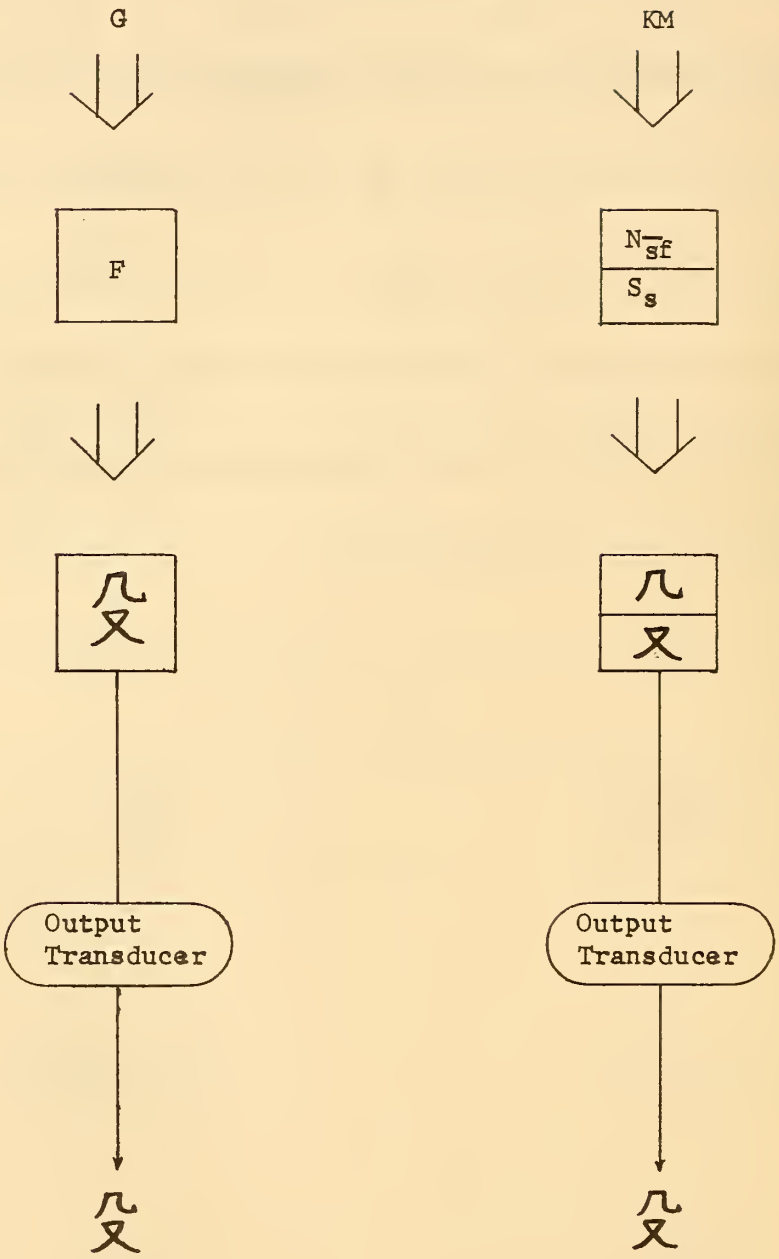
Two derivations of 慚:



Second, there are those characters which are themselves (or which contain components which are) further segmentable into components. For example,  is listed in the lexicon as a single component - because of its frequency of occurrence (specifically, it is a strong EAST). It is also the case that  can be generated by the grammar as  in construction with  .

This type of ambiguity is introduced at Stage 1, where it is possible for two distinct output strings from the state diagram ultimately to determine the same character.

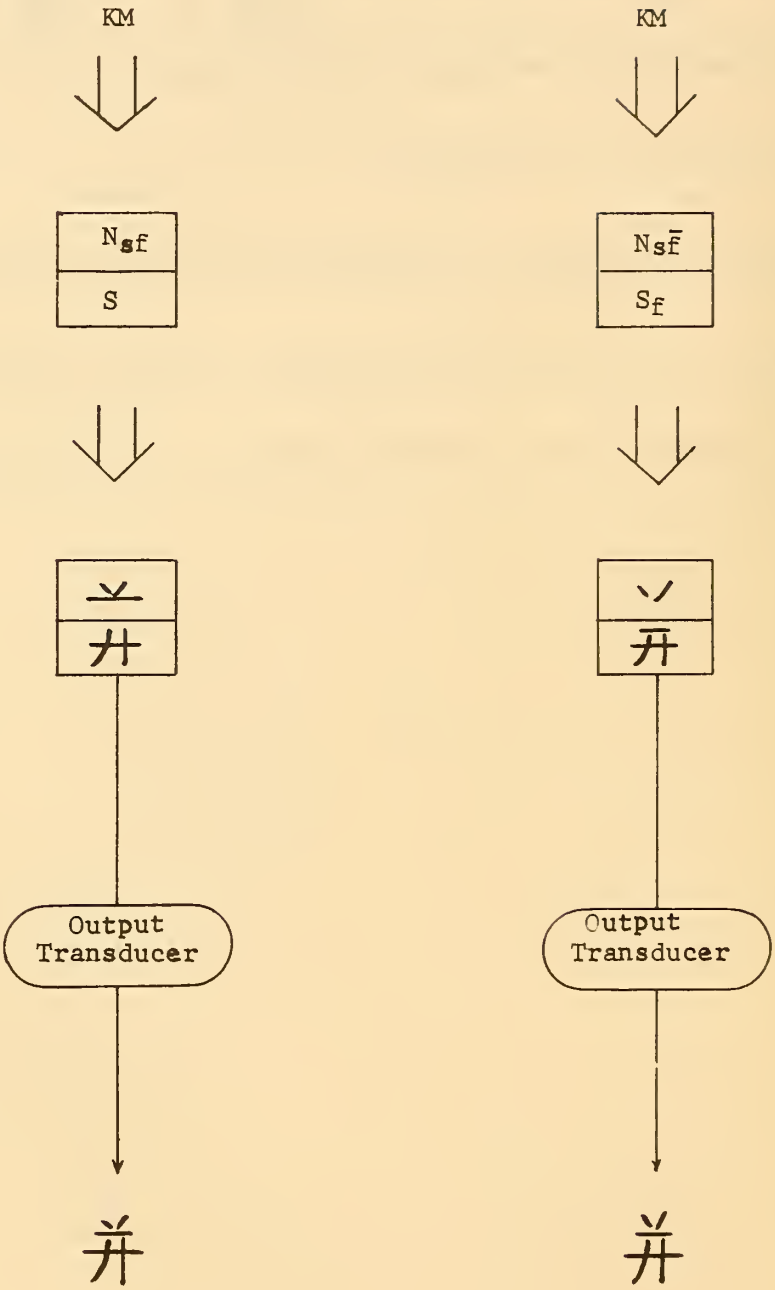
Two derivations of 𠬜 :



Finally, some characters are (or contain sub-parts which are) analyzable in more than one way-but into the same frame. An example is 并 which is analyzable as either  or  . So 并 in isolation is ambiguous, and any character which contains 并 as a sub-part is ambiguous.

This type of ambiguity is introduced (as in the first type discussed here) in the frame selection process. However, unlike the first type, these characters have identical frames at the output character level.

Two derivations of 并 :



ACKNOWLEDGEMENTS

The research summarized in this note was supported by Rome Air Development Center under RADC contract number (30-602)-66-4042. We are grateful for this support.

We wish to thank the following people for reading earlier versions of this note and for making valuable suggestions: Dr. Stephen J. Tauber, Mr. H. H. Ku, and Mr. David Rosenblatt of the National Bureau of Standards; Mr. W. C. Watt of Technical Operations Research; and Professor E. I. Burkart of American University.

Finally, we are indebted to Delores Allnutt of the National Bureau of Standards for typing the final version.

APPENDIX A

Classes of Mathews' entries which are deliberately excluded from the corpus (Mathews' sometimes lists several semantically related entries under one number. The number in parentheses indicates which of these entries is referred to.)

The following classification of excluded Mathews' entries arises out of a first approximation to a filtering process aimed at excluding entries which are not totally well-formed. We selected these particular ones in the following way. First, Mathews' lists all of these characters as secondary or tertiary variants. Second, the informant considers most of them to be less than acceptable. Third, they do not fit well into the current grammatical framework. This is by no means a definitive listing or classification.

1. "Radicals" which are cited in Mathews' as separate entries but which do not occur in isolation in everyday usage. 1/

240	(3)	亻	1439	(2)	巛
1282	(2)	辶	1650	(2)	扌
1373	(2)	𠂆	2735	(2)	忄
2735	(3)	忄	5788	(3)	礻
2989	(2)	礻	5838	(2)	扌
3037	(2)	阝	5922	(2)	氵
3097	(2)	亻	6124	(2)	刂
3153	(2)	月	6739	(3)	++
4737	(2)	牜	7666	(2)	王
5570	(2)	糸			

1/ A comparable example in English lexicography is the listing of bound morphemes (e.g., the prefix "pre-"), which do not occur as free words. Many Chinese dictionaries do not even list these radicals as lexical entries.

2. Printed forms whose hand-written variants are acceptable. 1/

297	(2)	眞	5283	(3)	氷
2309	(3)	囲	5788	(2)	永

3. Abbreviations whose unabbreviated forms are acceptable:

560	(2)	斉	3366	(3)	个
921	(2)	尓	3617	(2)	归
1082	(2)	尽	3953	(2)	两
1205	(2)	旧	6534	(2)	鬼
2222	(2)	画	7030	(2)	万
2753	(2)	与	7615	(2)	与

1/ This list includes only those printed forms for which Mathews' lists acceptable hand-written variants. In other cases, Mathews' lists printed forms only (without their hand-written variants). We account for these by copying into our lexicon only their hand-written variants. For example, we list in our lexicon 辶 for the Mathews' entry 辶, 子 for 子, 艹 for 艹, and 耂 for 耂.

4. Pictorially bizarre variants of acceptable characters:

666	(2)		3883	(3)	
1478	(2)		3953	(3)	
1517	(3)		3992	(2)	
2451	(2)		4083	(2)	
2896	(2)		4464	(2)	
3342	(2)		4725	(2)	
3406	(3)		5603	(2)	
5780	(2)		7044	(2)	
6209	(2)		7519	(2)	

APPENDIX B

1. Variation Processes and List

Component variation has been extensively studied, but a final statement on this phenomenon is not ready for presentation. What follows is a statement of four general processes and a list of rather more ad hoc variations. Component variation belongs properly in the rules of the output transducer, which has not been constructed, and is presented here mostly as an aid to human generation of output characters.

Further, some of the general variation processes may not be precisely stated here, and there are indications that some of the instances of ad hoc variation should be grouped together into general processes.

Variation Process 1, (BORDER)

For certain components whose base forms 1/ have ⌝ or ⌞

1/ A base form is that form of a component which can occur in isolation, those which belong to FREE.

in their "eastern" portions, this process causes the BORDER variant to assume a shape wherein 乙 becomes 乚 and 乚 becomes 乚 . Examples: 2.5.7: 九 → 九 , 12.8.1: 尭 → 尭 .

Variation Process 2, (WEST)

For certain components which have — in their "south central" portions, this process causes the WEST variant to assume a shape wherein — becomes ／ . Examples: 4.1.7: 止 → 止 , and 8.1.3: 金 → 金 .

Variation Process 3, (WEST)

For certain components which contain 木 or a 木 -like structure, this process causes the WEST variant to assume a shape wherein ＼ becomes 丶 . Examples: 4.4.21: 木 → 木 , and 7.4.6: 束 → 束 .

Variation Process 4, (BORDER)

For certain components which have ＼ or ＼ in their "south-eastern" portions, this process causes the BORDER variant to assume a shape wherein ＼ or ＼ become ＼ . Examples: 5.4.5: 瓜

瓜 → 瓜 , and 5.4.8: 永 → 永.

5. Ad hoc variations

Number	Base Form	Position	Variant Form
2.1.2	二	B	二
2.2.3	十	B	十
2.4.2	乂	N	乂
2.4.4	八	N,S	八
2.5.3	入	I (sometimes)	入
2.8.14	乃	B	乃
3.1.4	子	W	子
3.1.6	口	B	口
3.1.13	彐	sometimes	彐
3.2.2	千	B	千
3.2.3	干	B	干
3.2.7	丰	I (sometimes)	丰
3.2.8	艹	S	艹
3.3.1	弋	B	弋

Number	Base Form	Position	Variant Form
3.4.1	大	N (sometimes)	大
3.5.8	少	N (sometimes)	少
4.1.6	月	S	月
4.1.18	曰	sometimes	曰
4.2.9	牛	E	牛
4.2.16	丰	N,S	丰
4.3.11	戈	B	戈
4.4.4	攵	N	攵
		S	攵
		B	攵
4.4.16	攸	S	攸
4.4.21	木	S (sometimes)	木
4.5.9	歹	B	歹
4.5.11	少	B	少
4.7.5	手	B	手
4.8.10	气	B	气

Number	Base Form	Position	Variant Form
5.2.4	卯	B	
5.3.10	戌	B	
5.4.13	矢	W	
5.4.19	穴	N	
5.8.6	北	B	
6.1.5	舌	S	
6.1.18	耳	B	
		W	
6.2.3	羊	W	
6.3.2	共	N	
6.3.13	糸	W	
6.3.17	虫	W	
6.7.1	行	B	
8.1.6	尚	N	
8.2.7	耑	W	
8.3.6	其	N	

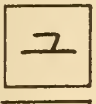
Number	Base Form	Position	Variant Form
9.3.3	頁	N	頁
9.4.4	食	W	食
11.3.1	麥	B	麥
11.8.1	鹿	B	鹿
12.3.2	絲	N	絲
		B	糸糸
13.3.1	與	N	與
13.7.2	鼠	B	鼠
14.1.1	齊	B	齊

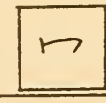
APPENDIX C

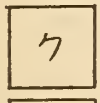
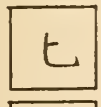
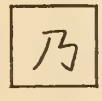
The Lexicon

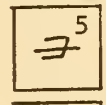
The lexicon is described on pages 34 through 39. There is, however, one extra-grammatical type of information included in the lexicon which must now be explained. There are superscripts (1-5) on some of the tallies, 1/ the significance of these superscripts is that the component on the row of this tally, when it occurs in the position class specified by the column heading of the tally, undergoes a shape variation. Superscripts 1-4 refer to variation processes 1-4 described in Appendix B; superscript 5 refers to the ad-hoc list of variants in Appendix B. The ad-hoc list follows the order of appearance of components in the lexicon.

1/ In two cases the component itself is superscripted. There are cases where there is more or less free variation among the component variants.

		W	E	N	S	B	I	F	V	C	T	D
1.1.1					s		x	x				
1.7.1								x				
1.8.1			s				x	x				
1.8.2					$\bar{s}$		x	x				
1.8.3						x						
1.8.4						x						
2.1.1				s								
2.1.2			$\bar{s}$	$\bar{s}$	$\bar{s}$	x^5	x	x				
2.1.3				s								
2.1.4				s								
2.2.1		$\bar{s}$	$\bar{s}$				x	x				
2.2.2			s		$\bar{s}$			x				
2.2.3		s	$\bar{s}$	$\bar{s}$	s	x^5		x			x	
2.2.4						x						
2.2.5		s										
2.3.1			$\bar{s}$	s	$\bar{s}$		x	x		x	x	

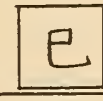
		W	E	N	S	B	I	F	V	C	T	D
2.3.2			s				x	x				
2.3.3				s								
2.4.1		s̄	s̄	s̄	s		x	x		x	x	
2.4.2		s̄		s[5]	s̄		x	x	x			
2.4.3			s̄	s	s̄		x	x		x		
2.4.4			s̄	s̄[5]	s[5]			x				
2.5.1			s	s̄	s			x				
2.5.2			s		s		x	x			x	
2.5.3							x[5]	x				
2.5.4						x						
2.5.5				s								
2.5.6						x						
2.5.7		s̄	s̄		s̄	x[1]	x	x				
2.6.1			s̄		s̄			x				
2.6.2		s										
2.7.1		s̄	s̄		s		x	x				
2.7.2				s								

		W	E	N	S	B	I	F	V	C	T	D
2.7.3			s									
2.7.4					$\bar{s}$			x				
2.8.1		$\bar{s}$	$\bar{s}$	$\bar{s}$	s		x	x				
2.8.2			$\bar{s}$	$\bar{s}$	s	x[1]	x	x				
2.8.3				$\bar{s}$		x		x				
2.8.4								x				
2.8.5					$\bar{s}$	x		x				
2.8.6					s		x	x				
2.8.7						x						
2.8.8		$\bar{s}$	$\bar{s}$		s		x	x				
2.8.9				s								
2.8.10					$\bar{s}$			x				
2.8.11		$\bar{s}$	$\bar{s}$	$\bar{s}$	s		x	x				x
2.8.12					$\bar{s}$		x	x				
2.8.13		s	$\bar{s}$	$\bar{s}$	$\bar{s}$		x	x				
2.8.14			$\bar{s}$	$\bar{s}$	$\bar{s}$	x[5]		x				

		W	E	N	S	B	I	F	V	C	T	D
3.1.1			s̄	s				x			x	
3.1.2		s^2	s̄	s	s		x	x	x		x	
3.1.3		s^2	s̄	s	s̄		x	x			x	
3.1.4		s^5	s̄	s	s		x	x		x	x	
3.1.5		s										
3.1.6		s	s̄	s	s	x^5	x	x			x	
3.1.7				s̄				x				
3.1.8				s	s̄			x				
3.1.9					s̄			x				
3.1.10							x	x				
3.1.11				s								
3.1.12		s	s̄	s̄	s		x	x	x	x	x	
3.1.13				s	s							
3.1.14						x						
3.2.1		s	s̄		s		x	x				
3.2.2		s̄	s̄		s̄	x^5		x		x	x	
3.2.3		s̄	s̄		s	x^5	x	x				

		W	E	N	S	B	I	F	V	C	T	D
3.2.4		s̄	s̄					x				
3.2.5		s	s									
3.2.6		s	s̄	s	s		x	x				
3.2.7					s̄		x[5]	x				
3.2.8					s[5]		x	x				
3.2.9								x				
3.2.10		s										
3.2.11		s										
3.2.12						x						
3.3.1			s̄	s̄		x[5]		x				
3.3.2			s̄					x				
3.3.3		s̄	s̄	s̄	s̄			x		x		x
3.3.4			s	s̄			x	x		x		
3.3.5			s	s̄			x	x				
3.3.6			s̄	s̄				x				
3.3.7			s		s		x	x				
3.3.8			s̄	s̄	s̄			x				

		W	E	N	S	B	I	F	V	C	T	D
3.3.9				s								
3.3.10			s̄		s̄			x				
3.3.11		s	s̄	s̄	s̄		x	x	x			
3.3.12		s										
3.3.13			s̄	s̄	s̄			x				
3.3.14			s̄	s̄				x				
3.3.15			s̄	s̄	s̄		x	x				
3.3.16						x						
3.3.17			s̄		s̄		x	x		x		
3.3.18		s										
3.4.1			s̄	s[5]	s		x	x			x	
3.4.2			s̄					x				
3.4.3			s̄		s̄		x	x				
3.4.4						x						
3.4.5								x				
3.5.1			s̄				x	x				
3.5.2		s̄				x		x				

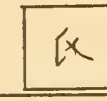
		W	E	N	S	B	I	F	V	C	T	D
3.5.3		s̄	s	s̄	s̄		x	x				
3.5.4		s										
3.5.5		s										
3.5.6			s̄					x				
3.5.7						x						
3.5.8				s̄[5]	s̄			x		x	x	
3.5.9		s										
3.6.1		s										
3.6.2		s										
3.6.3								x				
3.7.1			s̄	s̄	s̄		x	x				
3.7.2								x				
3.7.3				s								
3.8.1			s̄		s̄	x[1]		x				
3.8.2				s̄		x[1]		x				
3.8.3			s̄					x				
3.8.4			s	s̄			x	x		x		

		W	E	N	S	B	I	F	V	C	T	D
3.8.5	己	s̄	s̄	s̄				x				
3.8.6	乇		s̄		s̄		x	x			x	
3.8.7	亡	s̄	s̄	s̄	s̄		x	x				
3.8.8	弓	s	s̄		s			x				x
3.8.9	巛			s			x	x				
3.8.10	门					x						
3.8.11	凡		s̄					x				
3.8.12	亏		s̄	s̄	s̄			x				
3.8.13	亏		s̄		s̄			x				
3.8.14	儿				s							
3.8.15	也		s̄		s̄		x	x				
4.1.1	曰							x				
4.1.2	云			s̄			x	x				
4.1.3	夕			s	s̄			x				
4.1.4	月	s̄						x				
4.1.5	丑		s̄				x	x				

		W	E	N	S	B	I	F	V	C	T	D
4.1.6	月	s	s		s[5]		x	x		x	x	
4.1.7	止	s[2]	s̄	s	s		x	x		x	x	
4.1.8	王	s[2]	s̄	s̄	s		x	x		x		x
4.1.9	丱			s								
4.1.10	五		s̄	s̄				x				
4.1.11	生			s̄				x				
4.1.12	壬		s̄		s̄		x	x				
4.1.13	廿			s				x				
4.1.14	互		s̄					x				
4.1.15	丹	s̄	s̄				x	x				
4.1.16	毋			s̄			x	x				
4.1.17	白					x						
4.1.18	日[5]	s	s̄	s	s		x	x	x		x	
4.1.19	主	s̄		s̄				x		x		
4.1.20	毋							x				
4.2.1	午	s̄	s̄				x	x				
4.2.2	牛		s̄		s		x	x			x	

		W	E	N	S	B	I	F	V	C	T	D
4.2.3		s̄	s̄		s̄		x	x				
4.2.4			s̄		s̄		x	x				
4.2.5			s̄		s̄			x				
4.2.6			s̄	s̄			x	x				
4.2.7		s̄	s		s̄		x	x		x		
4.2.8		s̄	s̄					x				
4.2.9			s̄5		s̄			x				
4.2.10				s̄				x				
4.2.11				s̄	s̄			x				
4.2.12								x				
4.2.13			s̄		s̄		x	x				
4.2.14				s								
4.2.15			s̄		s̄		x	x				
4.2.16		s̄	s̄	s̄5	s̄5			x		x		
4.2.17			s̄		s̄		x	x				
4.2.18			s		s̄		x	x				
4.3.1			s̄					x				

		W	E	N	S	B	I	F	V	C	T	D
4.3.2	六			s̄	s̄			x				
4.3.3	云		s̄		s			x				
4.3.4	卞		s̄					x				
4.3.5	心		s̄	s̄	s		x	x			x	
4.3.6	内		s̄		s̄			x				
4.3.7	公	s̄	s̄	s̄	s̄		x	x				
4.3.8	太		s̄	s̄	s̄			x				
4.3.9	不		s̄	s̄	s̄		x	x				
4.3.10	歹	s			s̄			x				
4.3.11	戈	s̄	s		s	x[5]	x	x	x		x	
4.3.12	犬		s	s̄	s		x	x	x		x	
4.3.13	叉			s̄				x				
4.3.14	寸				s̄			x				
4.3.15	尤		s	s̄	s̄	x[1]	x	x				
4.3.16	夕	s̄		s̄				x				
4.3.17	小				s̄			x				
4.3.18	川				s							

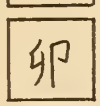
		W	E	N	S	B	I	F	V	C	T	D
4.3.19				s								
4.4.1		s̄	s			x[4]	x	x				
4.4.2			s		s̄			x				
4.4.3		s	s̄	s̄	s		x	x			x	
4.4.4			s	s[5]	s[5]	x[5]		x				
4.4.5			s		s̄		x	x				
4.4.6		s̄	s̄				x	x		x		
4.4.7		s̄	s̄	s̄				x				
4.4.8			s̄		s̄			x				
4.4.9			s̄					x				
4.4.10			s̄	s	s̄			x				
4.4.11			s̄	s̄	s̄		x	x				
4.4.12			s		s̄		x	x				
4.4.13				s	s̄			x				
4.4.14					s̄			x				
4.4.15			s̄	s̄	s		x	x				
4.4.16					s[5]		x	x				

		W	E	N	S	B	I	F	V	C	T	D
4.4.17					s̄			x				
4.4.18						x						
4.4.19								x				
4.4.20					s̄	x[4]	x	x				
4.4.21		s[3]	s̄	s	s[5]		x	x			x	x
4.4.22			s̄		s̄	x[4]		x				
4.4.23			s̄					x				
4.4.24		s	s̄	s̄	s		x	x	x	x	x	x
4.5.1			s̄		s̄	x		x				
4.5.2		s̄	s̄			x		x				
4.5.3			s̄					x				
4.5.4		s						x				
4.5.5		s̄	s̄	s̄	s̄		x	x				
4.5.6				s								
4.5.7		s̄	s̄		s		x	x				
4.5.8					s̄	x		x				
4.5.9						x[5]	x	x				

		W	E	N	S	B	I	F	V	C	T	D
4.5.10	夫					x						
4.5.11	少	$\bar{s}$	$\bar{s}$			x^5		x				
4.5.12	丰	s										
4.6.1	牙	s										
4.7.1	予	$\bar{s}$	$\bar{s}$	$\bar{s}$	$\bar{s}$		x	x				
4.7.2	氏		$\bar{s}$	$\bar{s}$				x				
4.7.3	屮			s								
4.7.4	屮			s								
4.7.5	手				s	x^5		x				
4.7.6	尹				$\bar{s}$			x				
4.8.1	元	$\bar{s}$	$\bar{s}$	$\bar{s}$	$\bar{s}$	x^1	x	x				
4.8.2	无		s	$\bar{s}$				x		x		
4.8.3	巳		$\bar{s}$		s		x	x				
4.8.4	屯	$\bar{s}$	$\bar{s}$		$\bar{s}$		x	x				
4.8.5	允	$\bar{s}$	$\bar{s}$	$\bar{s}$				x				
4.8.6	毛		s		s	x^1	x	x			x	
4.8.7	允					x						

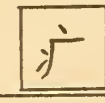
		W	E	N	S	B	I	F	V	C	T	D
4.8.8	无							x				
4.8.9	巳						x	x				
4.8.10	乞		s̄			x[5]		x				
4.8.11	比	s̄	s̄	s̄	s		x	x				
4.8.12	今	s̄	s̄	s̄	s̄			x				
4.8.13	方	s	s̄		s		x	x		x		
4.8.14	丙		s̄					x				
4.8.15	丐		s̄					x				
4.8.16	片	s						x				
4.8.17	幻		s̄					x				
4.8.18	匹	s̄						x				
5.1.1	屮			s̄				x				
5.1.2	丘	s̄	s̄	s̄				x				
5.1.3	丼			s̄	s̄			x				
5.1.4	目	s	s̄		s		x	x		x		
5.1.5	立	s[2]	s̄	s	s̄		x	x		x	x	

		W	E	N	S	B	I	F	V	C	T	D
5.1.6	且	s̄	s̄		s		x	x				
5.1.7	巨		s̄		s̄			x				
5.1.8	吕		s̄		s̄			x				
5.1.9	互		s̄					x				
5.1.10	册	s̄	s̄					x				
5.1.11	甘	s̄	s̄	s̄	s̄		x	x				
5.1.12	主		s̄					x				
5.1.13	正	s̄[2]	s̄		s		x	x				
5.1.14	止						x	x				
5.1.15	乍	s̄	s̄	s̄	s̄		x	x				
5.1.16	聿			s̄				x				
5.1.17	业			s	s̄		x	x				
5.1.18	石	s	s̄		s			x			x	
5.1.19	占	s̄	s	s̄	s̄		x	x				
5.1.20	加		s̄	s̄	s̄		x	x				
5.1.21	生	s̄	s̄	s̄	s̄		x	x		x		
5.1.22	北			s	s							

		W	E	N	S	B	I	F	V	C	T	D
5.1.23				$\bar{s}$				x				
5.1.24								x				
5.1.25			$\bar{s}$		$\bar{s}$			x				
5.1.26				$\bar{s}$			x	x				
5.1.27		$\overset{-2}{s}$						x				
5.1.28			$\bar{s}$		s			x				
5.1.29			$\bar{s}$	s	$\bar{s}$			x				
5.1.30					s			x				
5.1.31		s	$\bar{s}$	s	s		x	x		x		
5.1.32			$\bar{s}$	$\bar{s}$	$\bar{s}$		x	x				
5.1.33			$\bar{s}$	$\bar{s}$	$\bar{s}$			x				
5.1.34		s	$\bar{s}$	s	s		x	x			x	
5.2.1			$\bar{s}$					x				
5.2.2			$\bar{s}$	$\bar{s}$	$\bar{s}$		x	x				
5.2.3		$\bar{s}$	$\bar{s}$					x				
5.2.4			$\bar{s}$		$\bar{s}$	x^{5}		x				
5.2.5		$\bar{s}$	$\bar{s}$				x	x				

		W	E	N	S	B	I	F	V	C	T	D
5.2.6	出		s̄	s̄	s̄		x	x				
5.2.7	屮				s̄			x				
5.2.8	平		s̄		s̄		x	x				
5.3.1	术		s̄				x	x				
5.3.2	冬		s̄	s̄			x	x				
5.3.3	巳	s̄						x				
5.3.4	令	s̄	s̄		s̄		x	x				
5.3.5	氏	s̄	s̄				x	x				
5.3.6	去	s̄	s̄	s̄				x				
5.3.7	只	s̄	s̄		s̄		x	x		x		
5.3.8	水				s̄			x				
5.3.9	玉				s			x				
5.3.10	戊				s	x[5]		x				
5.3.11	瓦		s		s			x				
5.3.12	斥		s̄					x				
5.3.13	卡							x				
5.3.14	示		s̄		s			x		x		

		W	E	N	S	B	I	F	V	C	T	D
5.3.15		s̄	s̄	s̄	s̄		x	x				
5.3.16		s										
5.3.17		s̄						x				
5.3.18			s̄	s̄	s̄		x	x				
5.4.1		s̄	s		s̄		x	x				
5.4.2			s̄				x	x				
5.4.3			s̄		s̄			x				
5.4.4				s̄				x				
5.4.5		s̄	s		s̄	x[4]	x	x		x		
5.4.6			s̄		s̄			x				
5.4.7			s̄	s̄	s̄			x				
5.4.8			s̄	s̄		x[4]		x				
5.4.9			s̄		s̄		x	x				
5.4.10					s							
5.4.11				s̄			x	x				
5.4.12				s̄			x	x				
5.4.13		s[5]			s		x	x				

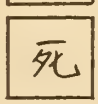
		W	E	N	S	B	I	F	V	C	T	D
5.4.14					$\bar{s}$			x				
5.4.15				$\bar{s}$			x	x				
5.4.16				$\bar{s}$	s		x	x				
5.4.17				s				x				
5.4.18			$\bar{s}$		$\bar{s}$			x				
5.4.19				s^5			x	x				
5.4.20			$\bar{s}$					x				
5.4.21		s^3	$\bar{s}$	s	s		x	x		x	x	
5.4.22					$\bar{s}$			x				
5.4.23				s								
5.5.1		s		$\bar{s}$			x	x				
5.5.2					$\bar{s}$			x				
5.5.3					$\bar{s}$			x				
5.5.4								x	x	x		
5.5.5								x		x		x
5.6.1		$\bar{s}$						x				
5.6.2						x						

		W	E	N	S	B	I	F	V	C	T	D
5.7.1	民	s̄	s̄	s̄				x				
5.7.2	乎		s̄				x	x				
5.7.3	可	s̄	s̄		s̄		x	x				
5.8.1	电				s̄			x				
5.8.2	包	s̄	s̄		s̄		x	x				
5.8.3	甲				s̄			x				
5.8.4	寿	s̄	s̄					x				
5.8.5	世		s̄	s̄			x	x				
5.8.6	北	s̄		s̄		x[5]		x				
5.8.7	甩							x				
6.1.1	百		s̄				x	x		x		
6.1.2	自		s̄	s	s̄			x				
6.1.3	臼		s̄	s̄	s		x	x				
6.1.4	血	s[2]	s̄	s̄				x				
6.1.5	舌	s	s̄		s̄[5]		x	x		x		
6.1.6	聿			s̄				x				

		W	E	N	S	B	I	F	V	C	T	D
6.1.7	回			s̄	s̄			x				
6.1.8	自	s	s̄	s̄	s̄		x	x				
6.1.9	亚			s̄	s̄			x				
6.1.10	羊			s	s̄			x				
6.1.11	吅			s				x				
6.1.12	西		s̄	s	s̄		x	x				
6.1.13	冉							x				
6.1.14	丙		s̄					x				
6.1.15	曲		s̄	s̄	s̄			x				
6.1.16	舟	s	s̄					x				
6.1.17	耳	s^5	s̄		s	x^5	x	x				
6.2.1	甲				s̄			x				
6.2.2	而	s̄	s̄	s̄	s̄			x				
6.2.3	羊	s^5	s̄		s		x	x			x	
6.2.4	州		s̄					x				
6.2.5	缶	s			s		x	x				
6.2.6	朱	s̄						x				

		W	E	N	S	B	I	F	V	C	T	D
6.2.7			s̄					x				
6.2.8								x				
6.2.9							x	x				
6.2.10			s		s		x	x				
6.2.11					s̄			x				
6.3.1		s										
6.3.2			s̄	s[5]	s		x	x				
6.3.3			s̄				x	x				
6.3.4					s̄			x				
6.3.5					s̄			x				
6.3.6			s̄					x				
6.3.7			s̄	s̄	s̄			x				
6.3.8						x						
6.3.9		s̄	s̄	s̄	s̄		x	x				
6.3.10				s̄			x	x				
6.3.11								x				
6.3.12								x				

		W	E	N	S	B	I	F	V	C	T	D
6.3.13	糸	s[5]			s		x	x		x		x
6.3.14	丟							x				
6.3.15	肉				s̄		x	x				
6.3.16	赤	s̄	s̄		s̄		x	x				
6.3.17	虫	s[5]	s̄		s		x	x			x	
6.3.18	氷						x	x				
6.3.19	良	s̄						x				
6.3.20	灬			s								
6.4.1	吏		s̄					x				
6.4.2	束	s̄						x	x	x		
6.4.3	艮		s	s̄			x	x				
6.4.4	朱	s̄	s̄	s̄	s̄			x				
6.4.5	耒	s	s̄					x				
6.4.6	亦	s̄	s̄		s̄		x	x				
6.4.7	夷	s̄	s̄				x	x				
6.4.8	歹			s								
6.4.9	米	s	s̄	s	s		x	x		x		

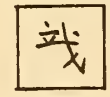
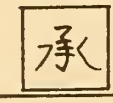
		W	E	N	S	B	I	F	V	C	T	D
6.4.10				s				x				
6.4.11			s̄	s̄	s		x	x				
6.4.12					s							
6.4.13				s̄				x				
6.5.1		s̄	s	s	s		x	x				
6.5.2			s̄					x				
6.5.3			s̄		s̄		x	x		x		x
6.5.4						x						
6.7.1			s̄		s̄	x[5]		x				
6.7.2				s								
6.7.3								x				
6.8.1		s	s̄		s̄			x				
6.8.2			s̄					x				
6.8.3		s̄	s̄				x	x		x		
6.8.4		s̄						x				
6.8.5					s̄			x				
6.8.6		s	s̄		s̄		x	x				

		W	E	N	S	B	I	F	V	C	T	D
6.8.7						x						
7.1.1		s	s̄					x				
7.1.2		s	s̄		s			x				
7.1.3		s^2	s̄	s̄	s		x	x				
7.1.4			s̄		s̄		x	x				
7.1.5		s	s̄		s		x	x		x		x
7.1.6		s̄						x				
7.1.7		s̄	s̄		s̄		x	x				
7.1.8		s	s		s̄			x				
7.1.9		s^2			s		x	x				
7.1.10		s̄	s̄		s̄			x				
7.1.11				s̄				x				
7.1.12							x	x				
7.1.13		s̄	s̄		s̄		x	x				
7.1.14			s̄		s̄		x	x				
7.1.15				s̄				x				

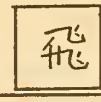
		W	E	N	S	B	I	F	V	C	T	D
7.1.16		s[2]	s̄		s		x	x				
7.2.1		s̄	s		s̄			x		x		x
7.2.2				s̄				x				
7.2.3		s	s̄		s		x	x		x	x	
7.3.1		s̄	s̄		s̄			x				
7.3.2			s	s̄	s̄		x	x				
7.3.3		s	s̄		s		x	x		x	x	
7.3.4		s						x		x		
7.3.5		s̄	s̄					x				
7.3.6		s̄	s̄		s̄			x				
7.3.7			s̄					x				
7.3.8			s̄				x	x				
7.3.9					s̄			x				
7.3.10								x				
7.3.11				s̄				x				
7.3.12					s̄			x				
7.3.13					s̄		x	x				

		W	E	N	S	B	I	F	V	C	T	D
7.3.14	丟	s̄	s̄		s̄			x				
7.4.1	夾	s̄	s̄	s̄	s̄		x	x				
7.4.2	采	s		s̄	s̄		x	x		x		
7.4.3	足		s̄		s	x[4]		x				
7.4.4	良	s̄	s̄		s̄		x	x				
7.4.5	更		s̄			x[4]		x				
7.4.6	束	s[3]	s̄				x	x				
7.4.7	走		s̄			x[4]		x				
7.4.8	豕	s	s̄		s	x[4]	x	x		x		
7.4.9	共						x	x				
7.5.1	步	s̄	s̄					x				
7.5.2	多	s										
7.5.3	羊					x						
7.5.4	身	s	s̄					x				
7.7.1	車			s								
7.7.2	宧			s̄				x				
7.8.1	見		s		s	x[1]		x				

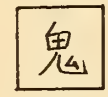
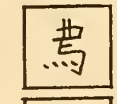
		W	E	N	S	B	I	F	V	C	T	D
7.8.2	鳥					x						
7.8.3	克					x[1]	x	x		x		
7.8.4	臣	s̄	s̄					x				
8.1.1	非		s̄	s̄	s		x	x				
8.1.2	隹		s	s̄	s		x	x		x		x
8.1.3	金	s[2]	s̄		s		x	x			x	
8.1.4	直		s̄	s̄	s̄		x	x		x	x	
8.1.5	亟		s̄					x				
8.1.6	尚	s̄	s̄	s̄[5]			x	x				
8.1.7	妻		s̄		s̄			x				
8.1.8	亞	s̄	s̄	s̄	s̄		x	x				
8.2.1	卑	s̄	s̄		s̄		x	x				
8.2.2	卓	s̄	s̄		s̄			x				
8.2.3	幸	s̄	s̄		s		x	x				
8.2.4	卓	s						x		x		
8.2.5	肅						x	x				

		W	E	N	S	B	I	F	V	C	T	D
8.2.6				s̄				x				
8.2.7		s̄⁵						x				
8.3.1			s̄				x	x				
8.3.2			s̄				x	x				
8.3.3			s̄				x	x				
8.3.4					s̄			x				
8.3.5		s̄		s̄				x				
8.3.6		s̄	s̄	s̄⁵	s̄		x	x			x	
8.3.7				s			x	x				
8.3.8		s̄	s̄					x				
8.3.9					s̄			x				
8.3.10			s̄		s̄		x	x				
8.3.11								x				
8.3.12						x						
8.4.1			s̄					x				
8.4.2		s̄	s̄		s̄			x			x	
8.4.3								x				

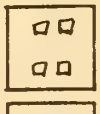
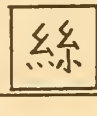
		W	E	N	S	B	I	F	V	C	T	D
8.4.4			s̄					x				
8.4.5			s̄		s̄		x	x				
8.4.6		s̄	s̄	s	s̄		x	x				
8.4.7			s̄					x				
8.4.8		s̄	s̄				x	x				
8.4.9			s̄					x				
8.4.10		s̄	s̄		s̄			x				
8.4.11		s̄	s̄	s̄				x				
8.4.12			s̄					x				
8.7.1				s								
8.7.2		s̄						x				
8.7.3				s								
8.7.4				s								
8.8.1			s̄		s̄	x[1]		x				
8.8.2								x				
8.8.3			s̄			x		x				

		W	E	N	S	B	I	F	V	C	T	D
9.1.1			s̄		s̄		x	x				
9.1.2		s̄	s̄		s̄		x	x				
9.1.3		s̄	s̄		s		x	x	x			
9.1.4		s̄	s̄		s̄		x	x				
9.1.5		s̄	s̄					x				
9.1.6				s	s̄			x				
9.1.7				s̄				x				
9.1.8		s̄[2]						x				
9.1.9		s̄	s̄		s̄			x				
9.1.10		s̄						x				
9.2.1			s̄					x				
9.2.2		s̄	s̄					x				
9.2.3		s	s̄	s̄	s			x				
9.2.4		s̄	s̄		s̄			x				
9.2.5			s̄					x				
9.3.1			s̄					x				
9.3.2					s̄			x				

		W	E	N	S	B	I	F	V	C	T	D
9.3.3			s	s̄[5]			x	x				
9.3.4			s̄					x				
9.3.5					s̄	x[1]	x	x				
9.3.6					s̄		x	x				
9.3.7			s̄	s̄	s̄		x	x				
9.3.8		s̄					x	x				
9.4.1			s̄				x	x				
9.4.2			s̄		s̄			x				
9.4.3						x[4]		x				
9.4.4		s[5]	s̄		s			x				
9.4.5			s̄		s̄		x	x				
9.4.6			s̄				x	x				
9.7.1				s								
9.8.1		s̄	s̄		s̄			x				
9.8.2		s̄	s̄	s̄	s̄		x	x				
10.1.1		s	s̄		s̄			x				

		W	E	N	S	B	I	F	V	C	T	D
10.1.2		s̄	s̄		s̄		x	x				
10.1.3		s̄	s̄					x				
10.2.1		s̄	s̄		s̄		x	x				
10.2.2					s̄			x				
10.2.3			s̄					x				
10.2.4		s̄	s̄				x	x				
10.2.5							x	x				
10.3.1			s		s	x[1]	x	x				
10.3.2		s̄	s̄					x				
10.3.3		s̄	s̄		s̄		x	x				
10.3.4								x				
10.3.5					s̄			x				
10.3.6			s̄					x				
10.3.7		s	s̄	s̄	s		x	x				
10.3.8								x				
10.3.9							x	x				
10.4.1		s̄	s̄					x				

		W	E	N	S	B	I	F	V	C	T	D
10.4.2		s̄	s̄				x	x				
10.5.1				s				x				
10.7.1				s								
10.7.2						x		x				
10.7.3		s̄	s̄					x				
10.8.1				s̄	s̄			x				
10.8.2								x		x		
11.1.1		s̄						x				
11.1.2		s	s̄		s̄			x				
11.1.3		s̄	s̄				x	x				
11.1.4								x				
11.1.5			s̄					x				
11.1.6					s̄			x				
11.1.7				s								
11.1.8		s̄	s̄		s̄		x	x				
11.1.9							x	x				

		W	E	N	S	B	I	F	V	C	T	D
11.2.1			s̄		s̄			x				
11.3.1		s				x		x				
11.3.2		s	s̄	s̄	s		x	x			x	
11.3.3		s̄	s	s̄	s		x	x				
11.3.4			s̄					x				
11.3.5		s̄	s̄		s̄		x	x				
11.3.6		s̄	s̄		s̄			x				
11.4.1		s̄						x				
11.4.2		s̄	s̄					x				
11.5.1								x				
11.8.1			s̄	s̄	s̄	x[5]		x			x	
12.1.1		s̄	s̄		s̄		x	x				
12.1.2						x						
12.2.1			s̄					x				
12.3.1			s̄		s̄		x	x				
12.3.2			s̄		s̄[5]		x[5]	x				

		W	E	N	S	B	I	F	V	C	T	D
12.3.3			s̄					x				
12.3.4		s̄	s̄		s̄		x	x				
12.3.5			s̄		s̄			x				
12.3.6					s			x				
12.3.7					s̄			x				
12.3.8		s		s̄	s			x				
12.4.1			s̄					x				
12.8.1			s̄		s̄	x[1]	x	x				
13.1.1			s̄					x				
13.1.2		s̄	s̄		s̄		x	x				
13.1.3			s̄		s̄		x	x				
13.1.4			s̄		s̄			x				
13.1.5			s̄		s			x				
13.2.1		s̄	s̄		s̄			x				
13.3.1		s̄	s̄	s[5]	s̄		x	x				
13.3.2			s̄					x				

		W	E	N	S	B	I	F	V	C	T	D
13.7.1				s								
13.7.2		s			$\bar{s}$	x^5		x				
13.8.1					$\bar{s}$			x				
13.8.2			$\bar{s}$		$\bar{s}$			x				
14.1.1		$\bar{s}$	$\bar{s}$		$\bar{s}$	x^5		x				
14.2.1								x				
14.3.1		$\bar{s}$	$\bar{s}$		$\bar{s}$		x	x				
14.3.2			$\bar{s}$		$\bar{s}$			x				
14.3.3			$\bar{s}$		$\bar{s}$			x				
14.3.4		$\bar{s}$	$\bar{s}$					x				
14.4.1		$\bar{s}$	$\bar{s}$	$\bar{s}$			x	x				
14.4.2					$\bar{s}$		x	x				
15.2.1		s			s			x				
15.4.1			$\bar{s}$					x				
15.7.1				s								

		W	E	N	S	B	I	F	V	C	T	D
15.7.2			s̄				x	x				
16.1.1			s̄					x				
16.1.2			s̄	s̄	s̄		x	x				
16.3.1								x				
16.4.1								x				
17.2.1		s	s̄		s̄			x				
17.3.1			s̄		s̄			x				
17.3.2								x				
17.3.3			s̄					x				
18.8.1								x				
19.8.1		s̄	s̄				x	x				

We now present a listing of components that occur in only one position.

WEST only:

2.2.5		3.3.18		4.5.12	
2.6.2		3.5.4		4.6.1	
3.1.5		3.5.5		5.3.16	
3.2.10		3.5.9		6.3.1	
3.2.11		3.6.1		7.5.2	
3.3.12		3.6.2			

EAST only:

2.7.3	

NORTH only:

2.1.1		2.7.2		4.1.9	
2.1.3		2.8.9		4.2.14	
2.1.4		3.1.11		4.3.19	
2.3.3		3.3.9		4.5.6	
2.5.5		3.7.3		4.7.3	

4.7.4		7.7.1		10.7.1	
5.4.23		8.7.1		11.1.7	
6.3.20		8.7.3		13.7.1	
6.4.8		8.7.4		15.7.1	
6.7.2		9.7.1			

SOUTH only:

3.8.14		4.3.18		6.4.12	

BORDER only:

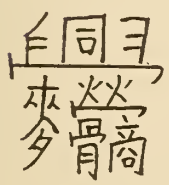
1.8.3		3.3.16		5.6.1	
1.8.4		3.4.4		6.3.8	
2.2.4		3.5.7		6.5.4	
2.5.4		3.8.10		6.8.7	
2.5.6		4.1.17		7.5.3	
2.8.7		4.4.18		7.8.2	
3.1.14		4.5.10		8.3.12	
3.2.12		4.8.7		12.1.12	

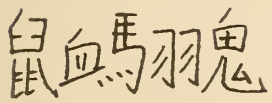
FREE only:

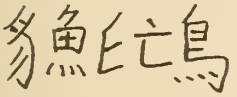
1.7.1	丿			7.3.10	卯
3.2.9	丫	5.1.24	凸	8.3.11	虱
3.4.5	孑	5.3.13	卡	8.4.3	承
3.6.3	孑	5.8.7	甩	8.8.2	乖
3.7.2	亍	6.1.13	再	10.3.4	豉
4.1.1	曰	6.2.8	年	11.1.4	雀
4.1.20	毋	6.3.11	乒	11.5.1	彪
4.2.12	卅	6.3.12	乓	14.2.1	舞
4.4.19	不	6.3.14	丢	16.3.1	興
4.8.8	无	6.7.3	竹	17.3.2	輿

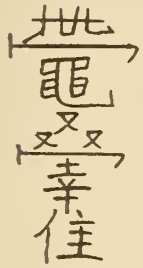
APPENDIX D

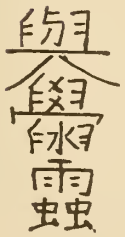
1. 

2. 

3. 

4. 

5. 

6. 

BIBLIOGRAPHY

1. Bach, E., An Introduction to Transformational Grammars, New York: Holt, Rinehart and Winston, 1964.

2. Chomsky, N., Syntactic Structure, The Hague: Mouton and Company, 1957.

3. Mathews, R.H., A Chinese-English Dictionary, Cambridge, Massachusetts: Harvard University Press, 1960.

4. Rankin, B.K. III, "A Linguistic Study of the Formation of Chinese Characters," University of Pennsylvania Dissertation, 1965.

5. Rankin, B.K. III, Sillars, W.A., and Hsu, R.W., "On the Pictorial Structure of Chinese Characters", National Bureau of Standards, Technical Note 254, Washington, 1965.

* U.S. GOVERNMENT PRINTING OFFICE: 1966—251-022/53

U.S. DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20230

OFFICIAL BUSINESS

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE