MACHINE LANGUAGE TRANSLATION STUDY

Ninth quarterly progress report

1 May 1961 - 31 July 1961

Research on the machine translation of German

Contract No. DA 36-039 SC 78911
File No. 18678-PM-59-91-91 (6909)

●

W. P. LEHMANN, Chief Investigator

Prepared by
E. D. PENDERGRAFT

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

The University of Texas, Austin, Texas

MACHINE LANGUAGE TRANSLATION STUDY - W. P. Lehmann
Chief Investigator

Ninth Quarterly Progress Report, 1 May 1961 to 31 July 1961, 56 pp.
Signal Corps Contract DA36-039 SC 78911 File No. 19678-PM-59-91-91
(6909) Unclassified Report.

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is "generalized" in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

Unclassified

1. Machine Language Translation Study
2. Signal Corps Contract DA36-039 SC 78911

TABLE OF CONTENTS:

1. PURPOSE

The primary aim of this study is the development of a generalized computer system which will implement language translation by means of automatic programming techniques. Under sponsorship of U. S. Army Signal Research and Development Laboratory, the system is being programmed for the IBM 709 at U. S. Army Electronic Proving Ground, Fort Huachuca, Arizona.

The design of the computer system is 'generalized' in the sense that it is based on a mathematical model which formalizes a general theory of linguistic structure. Through operational interpretations, the model may be specialized to a definite translation process; these interpretations are the source of programming specifications for the system's generalized translation algorithm. Linguistic interpretations, on the other hand, can specialize the model to characteristic structures of specific languages. These linguistic data are then compiled by automatic programming techniques into machine-oriented formats used by the translation algorithm. Because of its generality, the computer system will have the capability of translating a wide variety of natural and artificial languages into one another through common interlingual symbols.

This report describes progress during the period from 1 May to 31 July 1961.

2. ABSTRACT

The developmental status of a generalized computer system for language translation is explained. A formal definition of synonymy in the mathematical model underlying the system and the function of synonymy in the translation algorithm are discussed. Methods are also described by which automatic programming techniques will use interlingual symbols to organize input and output grammars for the generalized translation algorithm. The system is 'generalized' in the sense that specific languages are not presupposed by the algorithm; it will be capable of translating a wide variety of artificial and natural languages.

3. PUBLICATIONS, LECTURES, REPORTS AND CONFERENCES

At the request of the sponsor, Mr. E. Pendergraft presented the mathematical theory underlying the computer system to members of the Programming Section of U. S. Army Signal Research and Development Laboratory, 6-7 June 1961.

On 13-15 June 1961, Mr. W. Tosh attended a conference at the RAND Corporation, Santa Monica, California. The conference was the second of a series of meetings exploring means of cooperation between government sponsored research in machine translation. It is believed that in the extended contract period, the preparation of a Russian grammar for the system could be expedited through such cooperation.

Dr. S. Lamb, Director of the machine translation research project at the University of California, visited the project 20-27 July 1961. The visit afforded many opportunities to explore the increasing number of similarities in approach that has been developing in the two projects.

Professor W. P. Lehmann addressed the 23rd Summer Meeting of the Linguistic Society of America at the University of Texas, 28 July 1961; his paper was entitled, "Linguistic Typology in Historical Linguistics".

4. WORK IN MATHEMATICS

Three mathematical models for translation were described in the immediately preceding quarter [1] . The first formalizes a simple 'expression-for-expression' translation process; the second and third, called 'rule-for-rule' and 'content-for-content' translation respectively, render formal schemes that suggest analogies to literal and free translation.

Theories underlying the last two models were also presented axiomatically. For convenience, the definitions and theorems which were discussed are contained in an appendix to this report. Because of the informality of the previous report, proofs are also given; these have been prepared by W. Estes. Occasional parenthesized references to the appendix occur throughout this section.

Mathematical research in the quarter extended the above theory in several ways:

It was shown, for instance, that the second and third models can be merged into a single scheme which combines advantages of each. Operational interpretations of the composite model provide heuristics to cope with semantic ambiguities, a critical problem cited in the fifth report [2] .

Close cooperation with work in linguistics during the quarter also provided convincing evidence that proposed axioms for the second level

of the grammar were not sufficiently general. The discovery entailed some disruption and replanning in all areas of the project, but it led to a theory which accounts for many aspects of transformational structure that had been anomalous under the original theory.

As an interesting by-product, the more general theory furnishes additional indications that automatic discovery techniques, such as those which Lamb [3] has applied to syntax, can also be effective in organizing semantic classifications. Further study is being given to the possibility, since it might considerably reduce the labor cost of introducing new languages into the system.

Another accomplishment in the quarter was an extension of the theory to include ' discontinuous constituents' similar to those postulated by Yngve [4]. This facet of the theory will not be studied extensively in the near future, since it is felt that the need for the more complicated model cannot be reasonably established before more experience is gained with the present system.

This report will explain only the first extension, by which the ' rule-for-rule' and ' content-for-content' models are to be combined. A preparatory digression, into the nature of structural abstraction in languages will be useful beforehand.

4.1 Structural Abstraction

It has already been pointed out that two basic types of phenomena

are available to the linguist who sets out to describe a language: Formational phenomena exhibit the presence or absence of given expressions in the language; transformational phenomena exhibit relations between expressions. The linguist might, for example, elicit the former by acquiring a corpus of material from a native speaker. To elicit the latter, he might determine from his informant which entities of the corpus could be substituted for one another. These fundamentally different questions relate to two distinct concepts of structural abstraction.

The first takes cognizance of possible levels of formational abstraction, such as those which, within a formation structure (Def 9), allow properties of an expression (Def 10) to be variously abstracted as a sequence of syntactic classes (Def 11: Theorem 9). The second recognizes levels of transformational abstraction, such as the relationship between an expression and its content (Def 26). Here, properties of the expression are abstracted as a sequence of semantic classes (Def 21). Notice that formational abstraction deals with replacement phenomena among expressions, while transformational abstraction is concerned with replacement among rules (Def 22).

Transformational abstraction is seen to occur only in the 'content-for-content' model, where formational abstraction is present as well. The less sophisticated 'rule-for-rule' model contains

formational but not transformational abstraction. Neither appears in the simple 'expression-for-expression' model. Each step in the progression to formational abstraction and, finally, to transformational abstraction augments the capacity of the computer system:

'Expression-for-expression' translation is so severely restricted in capacity that it can be applied with modest success only when formational and transformational phenomena in two languages are markedly similar. Even then results tend to be disappointing, and translation algorithms designed around this principle [5] must employ a great deal of ingenuity in trying to correct mistakes of an essentially unintelligent agent.

'Rule-for-rule' translation owes its greater scope to the presence of formational abstraction in the model. Because of formational abstraction, correspondences from one language to another can be established not only between expressions but also between abstracted formational structures; hence, structural orderings in one language can be translated into those of another. The translation process is literal in that it still presupposes parallelism of transformational phenomena in the two languages. The objectives of 'rule-for-rule' translation, therefore, have much in common with those generally prevalent in machine translation research at the present time.

To call the 'content-for-content' model a formalization of free

translation is perhaps misleading, because the objective of a free translation model should be to translate the abstracted transformational structures of one language into those of another. The 'second level', which will be added to the grammar at a later phase of the study, is in fact intended to make possible that particular function. It would be more accurate, then, to describe the 'content-for-content' model as a step in that direction, but one designed to compensate for the absence of second-level analysis at this earlier phase of the study. Instead of trying to resolve semantic ambiguities, which will be another function of second-level analysis, the model merely makes available a set of well-defined alternatives to be considered heuristically as first-level choices in translation. A rationale for these heuristics shall now be developed.

4.2 Vagueness and Nuance

That vagueness is often deliberately introduced into communication will perhaps be granted; neglecting some obvious but homely examples such as persuasion, deception and the like, one finds vagueness even in scientific and technical writings, particularly when the latter are exploratory. Deliberate use of nuance is similarly widespread, especially among politicians, advertising executives, and socialites.

Since authors cannot necessarily be depended upon to say what

they mean or, worse still, to mean what they say, a translation which fails to preserve vagueness and nuance may miss the mark entirely. This is a difficult problem, but one needing a solution if mechanically prepared translations are to reach acceptable levels of quality.

An hypothesis for this purpose was developed in the quarter. It extends the formal linguistic theory to include a concept of 'content field', which appears to have useful operational consequences.

4.3 Content Field

Suppose that T is an equivalence transformation structure in some formation structure F. Then an analysis in F shall be any sequence $X_1, X_2, \ldots, X_p$ of minimal rules in F such that $X_1X_2 \ldots X_p = Z$ are applications and Z is a maximal rule in F. If E is an expression in F then an analysis $X_1, X_2, \ldots, X_p$ in F shall be called an analysis of E whenever, for some syntactic class c in F, $X_1X_2 \ldots X_p = \langle c, E, \Lambda \rangle$.

Clearly each expression in F has at least one analysis, but may have more than one. An expression E having more than one analysis in F is said to be syntactically ambiguous in F. Choice heuristics for syntactic ambiguities will not be discussed in this report, since the stochastic model proposed by Solomonoff [6] is believed to be satisfactory for this purpose. Details of its application to the computer system will be explained at a future date. To attack the remaining problem, choice heuristics for semantic ambiguities, these

additional assumptions are necessary:

Let S be a collection of explicit relations $G_1, G_2, \ldots, G_s$ in F, and suppose that T is derived from the union $G=G_1 \cup G_2 \cup \ldots \cup G_s$. Then each G_i (i=1, 2, . . . , s) must be a semantic equivalence relation (Def 25) in F which defines the semantic equivalence class $[G_i]$ of minimal rules of F. If E is an expression in F then E must have at least one content (Theorem 27); the latter is represented in the theory as a finite sequence $R_1, R_2, \ldots, R_p$ of elements of S. Whenever E has more than one content, the expression is said to be semantically ambiguous. The total set of contents of E shall be called the content field of E. In linguistic interpretations, the content field of E should contain every content which E could have in some context.

The limitations of first-level analysis as a means to resolution of semantic ambiguities can now be made explicit. Corresponding to each analysis $X_1, X_2, \ldots, X_p$ of E there must exist at least one content $R_1, R_2, \ldots, R_p$ of E such that $X_i \in [R_i]$ for i=1, 2, . . . , p. The total set of such contents shall be called the content subfield of E corresponding to the analysis $X_1, X_2, \ldots, X_p$. The first-level analysis process can find each analysis of E (Theorem 14) and, consequently, can locate each subfield of the content field of E. In the stochastic model, each analysis has an associated probability.

Thus, if each content subfield of E contains just one content, the semantic ambiguity can be resolved by choosing that single content corresponding to the most probable analysis.

This is a popular strategy in current machine translation research, but one which seems quite unjustified by results. Linguistic research has forced this project to abandon it. Close examination of natural languages reveals too many semantic ambiguities to squeeze into such a narrow mold.

It is more often the case, in linguistic interpretations, that content subfields contain more than one content. When an expression is in a particular context, its content subfield is imagined to be partitioned into two components, one denotative and the other connotative. The intuitive assumptions underlying this dichotomy are, roughly, that denotative contents carry the primary message in communication, and that the hearer selects the denotative component of the content subfield by means of both linguistic and extralinguistic context. The connotative component, which owes its existence to multiple usage of linguistic entities, contains those contents which would be selected as denotative in some other context than the one before the hearer.

If, by clever manipulation on the part of the speaker, the denotative component is made to carry one message and the connotative component another, then a wide range of clandestine communication, from

humor to derision, can occur in the secondary channel. Vagueness may be associated with an abnormally large denotative component, since the larger number of possibilities presented to the hearer would increase his uncertainty about what was intended in the primary message. As implied above, some vagueness is normal in natural language communication, and, contrary to common opinion, it is probably desirable for utilitarian and aesthetic reasons. Nuance, on the other hand, may be associated with the connotative component; the number of possibilities that remain unselected by the hearer is normally large and would usually leave him uncertain about the full intentions of the speaker. However, even when the secondary message lacks clarity, it can be an important factor in certain kinds of communication, such as propaganda.

The desideratum of translation should evidently be minimum deviation in denotative content, fidelity in the connotative component being the less important consideration. Yet the reasoning above would imply that the best translation should preserve not only denotative content but, if possible, the entire content subfield.

A second, more practical argument for preserving as much as possible of the content subfield is imposed by the absence of second-level analysis in the present computer system. Without this process, whose function would select denotative content from the subfield, the

heuristic process most likely to preserve denotative content should be one that would attempt to translate the entire content subfield. A heuristic process of this kind can be based on a formal measure of synonymy between rules in F.

4.4 Synonymy

If T is an equivalence transformation structure in F and $X, Y \in F$ then let C_x and C_y signify, respectively, the set of semantic equivalence classes in F containing X and the set of those containing Y. The synonymy between X and Y shall then be defined as

$$|X\text{-}Y| = \frac{N(C_x \cap C_y)}{N(C_x \cup C_y)} .$$

Here, the numerator represents the number of equivalence classes in the intersection of C_x and C_y, the denominator the number of classes in their union. Thus $0 \leq |X\text{-}Y| \leq 1$, with zero synonymy unless X and Y are together in some semantic equivalence class.

Whenever two expressions, E_1 and E_2, have some content in common, there must exist an analysis $X_1, X_2, \ldots, X_p$ of E_1 and an analysis $Y_1, Y_2, \ldots, Y_p$ of E_2 such that

$$|X_1X_2\ldots X_p\text{-}Y_1Y_2\ldots Y_p| = \frac{\prod_{k=1}^{p} N(C_{x_k} \cap C_{y_k})}{\prod_{k=1}^{p} N(C_{x_k}) + \prod_{k=1}^{p} N(C_{y_k}) - \prod_{k=1}^{p} N(C_{x_k} \cap C_{y_k})} > 0.$$

The synonymy between the analyses of E_1 and E_2 is seen to be

identically the number of contents in the intersection of the content subfields of E_1 and E_2 divided by the number of contents in the union of the two subfields. A convenient measure of synonymy between expressions in F has therefore been obtained.

4.5 First-level Heuristics

Automatic programming techniques can then be applied to find, for each rule R in the grammar for F, those minimal rules which are the closest synonyms of R, and to arrange them in an order of decreasing synonymy to R. With this information, the research cycle will translate expressions in F into their nearest synonyms, as a means to verification of semantic classifications in the grammar.

Similarly, each semantic equivalence class of each language in the computer system will correspond to at least one interlingual symbol, and each interlingual symbol will correspond to at least one semantic equivalence class in each language. These two sets of correspondences will be maintained on the interlingual master tape. Through the interlingual symbols, the set C_r of semantic equivalence classes which contain the minimal rule R of one language can be translated into a set C'_r of semantic equivalence classes of any other language. Automatic programming can then find those rules in the grammar for the other language which are closest synonyms of the rule R relative to the translated set C'_r, and can order them in terms

of decreasing synonymy for use by the translation algorithm.

The 'content-for-content' model may, therefore, be used to organize 'rule-for-rule' translation through computations of synonymy. It is emphasized that these computations are all performed in the application function of the computer system, prior to production.

5. WORK IN LINGUISTICS

Another instance of the interplay between theoretical and applied linguistic research occurred in the quarter. As mentioned above, the previously proposed hypothesis for second-level structure gave way before accumulated weight of disconfirming evidence, and a more general hypothesis took its place. Work in linguistics was disrupted during the period of replanning; furthermore, some recoding of existent rules became necessary.

In net result, however, coding of the grammars will be simplified through the discovery or, more accurately, through the clarification of coding procedures which it entails. Implications of the hypothesis are, indeed, so explicit regarding the requirements of grammatical coding that the way appears to be open for a partially self-organizing system which would 'learn' syntactic and semantic classifications from samples of text. A program to take advantage of these possibilities could not be developed rapidly enough to be helpful in the current phase of the study, but it shall be approached seriously during the contract extension.

The immediate objective of linguistic work continues to be a demonstration of German-to-English translation before the end of the original contract period, 30 April 1962. Both grammars are being coded for the composite, one-level model explained in

this report. At present, there is no apparent deterrent to the attainment of this goal.

It has been found, in some instances, that first-level rules should not be coded, for the composite model, in the same way that they would be coded when the model is extended to include second-level analysis. These differences are the result of compromises to get as much as possible out of the less comprehensive model. For example, correspondences between German and English can sometimes be established only by means of a rather gross unit, such as an entire phrase. Rules of this kind are tentative, in that they will be replaced later by coding on the second level. At this phase of the study, they serve a dual purpose: to permit translation with the less elegant model, and to catalog areas in the languages which must be bridged by means of transformational rather than formational abstraction.

Linguistic interpretation of semantic equivalence classes on the first level will be an analogue of morphology, but this comparison is necessarily crude, since many of the rules are positional and do not contain symbols that would represent graphemic segments of text. Nevertheless, rules representing graphemic units, like APPLY and APPLI-, which would have the same morphemic classification, will appear in the same semantic equivalence class. The

class that would contain a rule representing the verb SEE, moreover, will contain those for other verbal units, such as OBSERVE, WATCH, etc., as well as UNDERSTAND, COMPREHEND, etc.

Those rules which are contained in the same set of semantic equivalence classes form a synonymy class in the grammar. By the formal definition of synonymy given in the mathematics section, it is evident that the measure of synonymy between two rules in the same synonymy class must be one; intuitively, the two rules occupy the same point in the semantic space. The rules, for example, which represent APPLY and APPLI- will be in the same equivalence class as well as the same synonymy class; they will be analogues of morphs. Those representing SEE and UNDERSTAND will appear in the same equivalence class, but not in the same synonymy class; here, relational analogies to polysemy and homonymy will be represented.

The ability of the computer system in producing synonymous expressions from these data will be tested against a special corpus prepared in the quarter. Members of the linguistic staff acted as informants to record expressions which they considered to be synonymous to those found in Corpus 05 of the master corpus tape. The special corpus will also be used, at a later date, as an input to experimental self-organizing programs.

6. WORK IN PROGRAMMING

System programming continued approximately on schedule during the quarter. Some delays in checkout schedules were occasioned by tape and machine malfunctions; however, these were counterbalanced by extra work accomplished during trips to Ft. Huachuca. Difficulties with tapes were mollified by the acquisition of new tapes to be used exclusively in the translation system.

Progress toward current programming goals is summarized below:

6.1 Corpus Revision

Major additions to the corpus revision program were completed. The program should soon be declared operational and added to the system library tape.

A minor addition to this program, not previously reported, extends the Insert Corpus (IC) system-request function to make it more convenient to insert lines below the last line of a corpus sample. To be specific, if parameter Z of the IC request is coded with a number that is greater or equal to the last non-blank line of the sample, then the insertion is made following that last line.

6.2 Grammar Revision

Coding was completed on all of those parts of the grammar revision function that handle first-level rules, and check out began

during a trip to Ft. Huachuca. A change in the grammar tape format, made necessary by other contingencies of system planning, delayed this program somewhat through recoding and assembly of new test data for the affected segments.

6.3 Grammar Display

Segments of the grammar display program were checked out individually. Sort segments were also created by selection of parameters for use in the general sort routine; some additional coding was put into the sort segments for purposes of selecting input.

All of the segments were assembled and prepared for testing late in the quarter. This program was also affected by the change in grammar format.

6.4 Analysis

Detailed data formats and tape flow were worked out for the analysis program. The basic pattern will be a system of lists, each list being identified with a definite position in the text and each item in a list being chained to a succeeding list. The items will represent syntactic classifications of segments of the text. Chain sequences through the system of lists represent concatenation possibilities among the segments.

The analysis process will not be sequential; it will work in parallel with as many lists as possible within the physical limits of the

computer. By referring to the grammar of the language being analyzed, the process will add items to the list structure as opportunities for formational abstraction develop among the syntactic classifications. Many disadvantages of sequential analysis can be avoided by this strategy.

Individual rules are not used in the analysis process, since this would result in redundant matching among items. The rules will be compiled into a very compact, machine-oriented format to facilitate matching. In the format, the separate rules will be combined into a single structure such that any two rules beginning with the same items will form a single path through the structure to their point of divergence. An input grammar compiler in the application section will perform the compacting function.

6.5 Synthesis

The synthesis program entered the planning stage, following completion of planning for first-level heuristics. Translations will be produced in order of decreasing synonymy to the input expression; parameters in the program will specify the number of alternatives to be produced. An output grammar compiler in the application section will perform the organizational functions described as 'first-level heuristics', by which output language rules are ordered in decreasing synonymy for the synthesis function.

7. CONCLUSIONS

Mathematical work for the first phase of the study is essentially completed. When remaining documentation has been concluded, members of the mathematical staff will concentrate on formalization of the principles of second-level analysis that are to be tested in the second phase of the study. Other members of the staff will continue to perfect operational and linguistic interpretations of the current model, so that evaluation of the principles of first-level analysis can begin within the original three-year contract period. System programming and compilation of German and English grammars are well advanced toward this goal.

The current model is not regarded as the ultimate solution to mechanical translation. Indeed, if such an ideal solution were to exist, it could be approached only through a series of increasingly costly approximations. The model now being implemented is thought to be a significant second approximation. It incorporates those improvements to expression-for-expression translation which can be made through systematic, rather than ad hoc considerations of syntax.

Semantic considerations in the model are rudimentary, but they have the advantage of being specific. As a consequence, even before the present configuration of the computer system is operational, much has been learned about its advantages and limitations. By present

standards, a high quality output is expected from the system. Translations will be generally more readable than those hitherto produced. This improvement will result from the system's capacity to perform formational abstraction. However, certain known exceptions will cause decidedly awkward constructions and mistranslations. Discontinuous constituents, for example, will not be handled conveniently in the model. A satisfactory attack on these problems must be deferred until transformational abstraction is a system capability.

The computer system is, therefore, carefully designed to be extended rather than replaced in successive phases of the study. This seeming indulgence is quite necessary, since no important benefit would result from fragmentary nostrums in such a complex field of research.

8. PLANNING FOR THE NEXT THREE MONTHS

Documentation of results of mathematical research will continue well into the next quarter. The optimization of synthesis will be studied, as well as the search problem in the output grammar compiler. Further work with second-level axioms will then be pursued.

In programming, corpus revision should be operational. Modifications of the grammar revision function will be completed. Checkout of those portions of the program that handle first-level rules should near completion. Parts of analysis will be coded. Planning for synthesis will begin, as well as for the input grammar and output grammar compilers.

Linguistic work will resume the coding of German and English grammars according to procedures that are now being specified. The work is expected to go much faster than in past quarters.

9. IDENTIFICATION OF PERSONNEL:

Chief Investigator:

Dr. W. P. Lehmann
(10 hrs. per week, 1 May 1961 to 31 July 1961)

Associate Investigator:

Mr. E. D. Pendergraft
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Linguistics:

Mr. L. W. Tosh
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mrs. B. Efrat
(40 hrs. per week, 1 May 1961 to 30 June 1961)
(25 hrs. per week, 1 July 1961 to 31 July 1961)

Mr. G. R. Lewis
(40 hrs. per week, 1 June 1961 to 31 July 1961)

Mrs. N. Orme-Johnson
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Dr. I. R. Shaw
(30 hrs. per week, 1 May 1961 to 31 July 1961)

Programming:

Mrs. B. Foster
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mrs. M. L. Hagemeyer
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mr. R. W. Jonas
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mathematics:

Mr. W. A. Holley
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mr. W. B. Estes
(40 hrs. per week, 1 May 1961 to 31 July 1961)

System Operations

Mr. C. A. Bramblett
(24 hrs. per week, 1 May 1961 to 31 July 1961)

Mrs. M. P. Burkland
(40 hrs. per week, 1 May 1961 to 31 July 1961)

Mr. G. R. Lewis
(24 hrs. per week, 1 May 1961 to 31 May 1961)

Mr. J. McLeroy
(15 hrs. per week, 1 May 1961 to 15 May 1961)

Mrs. N. Tosh
(20 hrs. per week, 1 May 1961 to 31 May 1961)

Auxiliary Studies:

Dr. E. Bach
(10 hrs. per week, 1 May 1961 to 31 July 1961)

Mrs. D. E. Casey
(30 hrs. per week, 1 May 1961 to 31 July 1961)

Miss Y. Takahashi
(14 hrs. per week, 1 May 1961 to 31 May 1961)

REFERENCES:

[1] E. Pendergraft, Machine Language Translation Study, Eighth Quarterly Progress Report, 1 February 1961 - 30 April 1961.

[2] E. Pendergraft, Machine Language Translation Study, Fifth Quarterly Progress Report, 1 May 1960 - 31 July 1960.

[3] S. Lamb, "Can Syntactic Analysis be Mechanized?" presented at the meeting of the Linguistic Society of America, Austin, Texas, July 1961.

[4] V. Yngve, "A Model and an Hypothesis for Language Structure", Proceedings of the American Philosophical Society, vol. 104, no. 5.

[5] Machine Translation Studies of Semantic Techniques, Final Report of Contract AF 30(602) - 2036, Ramo-Wooldridge, February 1961.

[6] R. Solomonoff, A Progress Report on Machines to Learn to Translate Languages and Retrieve Information, Zator Company, October 1959.

APPENDIX

FORMATION AND TRANSFORMATION STRUCTURES

Symbol is undefined in the hypothesis.

Definition 1. X is a rule means that X is an ordered triple,

$$X = \langle u, v, w \rangle,$$

where the first member u=a is a symbol; the second member $v = \langle b_1, b_2, \ldots, b_n \rangle$ is, for some positive integer n, an ordered n-tuple of symbols; the third member either is empty, denoted by $w=\Lambda$, or is a simple ordering of m ($1 \leq m \leq n$) of the n terms of v. The ordering is represented by an m-tuple $w = \langle b_{i_1}, b_{i_2}, \ldots, b_{i_m} \rangle$ in which i denotes a one-one mapping from the integers $1, 2, \ldots, m$ into m subscripts of the terms of v.

Definition 2. If X and Y are the rules,

$$X = \langle a, \langle b_1, b_2, \ldots, b_n \rangle, \langle b_{i_1}, b_{i_2}, \ldots, b_{i_m} \rangle \rangle,$$
$$Y = \langle c, \langle d_1, d_2, \ldots, d_p \rangle, \langle d_{j_1}, d_{j_2}, \ldots, d_{j_q} \rangle \rangle$$

then the application XY is the rule Z,

$$Z = \langle a, \langle b_1, b_2, \ldots, b_{k-1}, d_1, d_2, \ldots, d_p, b_{k+1}, \ldots, b_n \rangle, \langle d_{j_1}, d_{j_2}, \ldots, d_{j_q}, b_{i_2}, \ldots, b_{i_m} \rangle \rangle$$

if and only if $i_1 = k$ and the symbols denoted by b_k and c are identical. We write XY = Z. If $X_1, X_2, \ldots, X_p$ are rules then

$X_1X_2X_3=(X_1X_2)X_3$ and, in general, $X_1X_2\ldots X_p=(X_1X_2\ldots X_{p-1})X_p$ signify successive applications.

Let H be a set of rules:

Definition 3. For all $X \in H$ and $Y \in H$, $X \leq Y$ in H means that there exists a finite sequence $W_1, W_2, \ldots, W_p$ of rules of H such that $W_1W_2\ldots W_p = Y$ and $W_1 = X$.

Definition 4. $X < Y$ in H if and only if $X \leq Y$ in H and $X \neq Y$.

Definition 5. A rule $X \in H$ is maximal in H if and only if there exists no rule $Y \in H$ such that $X < Y$ in H.

Definition 6. A rule $X \in H$ is minimal in H if and only if there exists no rule $Y \in H$ such that $Y < X$ in H.

Definition 7. A class symbol in H is a symbol which occurs as the first member of some rule of H.

Definition 8. An alphabetic symbol in H is a symbol which occurs in the second member of some rule of H but is not a class symbol in H.

Definition 9. A formation structure is a set F of rules having the following properties:

Axiom 1: if $X \in F$ and $Y \in F$ and $XY = Z$, then $Z \in F$,

Axiom 2: if $Y \in F$, then a minimal rule $X \in F$ and a maximal rule $Z \in F$ exist such that $X \leq Y \leq Z$ in F,

Axiom 3: the set of minimal rules in F is finite,

Axiom 4: no $X \in F$ and $Y \in F$ exist such that $XY = X$ or $XY = Y$, and

Axiom 5: if $X = \langle u, v, w \rangle \in F$ then w orders just those terms of v which are class symbols in F.

In the following, F denotes a formation structure.

Theorem 1. There is no rule $X \in F$ having the form $X = \langle a, a, \gamma \rangle$.

If there exists a rule $X = \langle a, a, \gamma \rangle \in F$, then by Def 2, $XX = X$, contradicting Axiom 4, Def 9.

Theorem 2. If $X = \langle a, b, \gamma \rangle \in F$ then the rule $Y = \langle b, a, \delta \rangle \notin F$.

If there exist such rules X and Y in F then XY is an application (Def 2), and $XY \in F$ (Axiom 1, Def 9), but $XY = \langle a, a, \lambda \rangle$ and cannot be in F (Theorem 1).

Theorem 3. A rule of a formation structure F is maximal in F if and only if every symbol in its second member is an alphabetic symbol in F.

If $X = \langle u, v, w \rangle \in F$ and v contains no class symbol in F, then there exists no rule $Y \in F$ such that XY is an application (Defs 2, 7). Consequently, there exists no rule $Z \in F$ such that $X < Z$ (Defs 4, 3) and X is maximal in F (Def 5).

Conversely, if X is maximal in F, the second member of X contains no class symbols, for, if it is assumed that there is a term of the second member which is a class symbol, there

will be one, say c, appearing first in the ordering of the third member of X. There exists a rule Y of F having c as its first member (Def 7). XY = Z is an application (Def 2) implying X < Z (Defs 4, 3 and Axiom 4, Def 9) which contradicts the hypothesis.

Definition 10. E is said to be an expression in the formation structure F if and only if some maximal rule in F has the form $\langle c, E, \Lambda \rangle$.

Definition 11. If c is a class symbol in the formation structure F then the syntactic class [c] in F is the set such that $E \in [c]$ if and only if $\langle c, E, \Lambda \rangle$ is a maximal rule in F.

Definition 12. If d is an alphabetic symbol in F, then [d] denotes the set containing only symbol d.

Definition 13. If each of $b_1, b_2, \ldots, b_n$ is an alphabetic or class symbol in the formation structure F, then let $[b_1 \frown b_2 \frown \ldots \frown b_n]$ represent the set containing exactly those sequences of alphabetic symbols which may be formed by successive concatenations of just one member from each of the sets $[b_1], [b_2], \ldots, [b_n]$; i.e., $E_1 \frown E_2 \frown \ldots \frown E_n \in [b_1 \frown b_2 \frown \ldots \frown b_n]$ if and only if $E_1 \in [b_1]$, $E_2 \in [b_2], \ldots, E_n \in [b_n]$.

Theorem 4. For each rule $X = \langle a, \langle b_1, b_2, \ldots, b_n \rangle, \gamma \rangle \in F$, $[a] \supseteq [b_1 \frown b_2 \frown \ldots \frown b_n]$.

If $E \in [b_1^\frown b_2^\frown \ldots^\frown b_n]$ then there exists a sequence $E_1, E_2, \ldots, E_n$ such that $E_i \in [b_i]$ and $E = E_1^\frown E_2^\frown \ldots^\frown E_n$ (Def 13). If b_i is a class symbol, then $\langle b_i, E_i, \Lambda\rangle$ is a maximal rule in F (Def 11). Let $R_i = \langle b_i, E_i, \Lambda\rangle$. If γ, the third member of rule X, is $\langle b_{k_1}, b_{k_2}, \ldots, b_{k_p}\rangle$ then $XR_{k_1}R_{k_2}\ldots R_{k_p} = \langle a, E, \Lambda\rangle$, (Def 2). Thus, $E \in [a]$, (Def 11).

Theorem 5. If (XY)Z and X(YZ) are applications then

$(XY)Z = X(YZ)$.

Let $X = \langle A, \langle a_1, a_2, \ldots, a_r\rangle, \langle a_{n_1}, \ldots, a_{n_k}\rangle\rangle$,

$Y = \langle B, \langle b_1, b_2, \ldots, b_s\rangle, \langle b_{m_1}, \ldots, b_{m_e}\rangle\rangle$,

$Z = \langle C, \langle c_1, c_2, \ldots, c_t\rangle, \langle c_{p_1}, \ldots, c_{p_i}\rangle\rangle$.

Assuming $B = a_q = a_{n_1}$, by Def 2

i) $XY = \langle A, \langle a_1, \ldots, a_{q-1}, b_1, \ldots, b_s, a_{q+1}, \ldots, a_r\rangle,$

$\langle b_{m_1}, \ldots, b_{m_e}, a_{n_2}, \ldots, a_{n_k}\rangle\rangle$

and if $C = b_j = b_{m_1}$

ii) $(XY)Z = \langle A, \langle a_1, \ldots, a_{q-1}, b_1, \ldots, b_{j-1}, c_1, \ldots, c_t, b_{j+1}, \ldots$

$\ldots, b_s, a_{q+1}, \ldots, a_r\rangle,$

$\langle c_{p_1}, \ldots, c_{p_i}, b_{m_2}, \ldots, b_{m_e}, a_{n_2}, \ldots, a_{n_k}\rangle\rangle$.

iii) $YZ = \langle B, \langle b_1, \ldots, b_{j-1}, c_1, \ldots, c_t, b_{j+1}, \ldots, b_s\rangle,$

$\langle c_{p_1}, \ldots, c_{p_i}, b_{m_2}, \ldots, b_{m_e}\rangle\rangle$.

From iii) it is evident that the application X(YZ) results in ii).

Theorem 6. If $A_1A_2\ldots A_n$ are applications and $A_i = PQ$, then

$$A_1 \ldots A_i \ldots A_n = A_1 \ldots A_{i-1} PQ A_{i+1} \ldots A_n .$$

From Theorem 5, $\{(A_1 \ldots A_{i-1})P\}Q = (A_1 \ldots A_{i-1})\{PQ\}$

$$= (A_1 \ldots A_{i-1})A_i .$$

Theorem 7. If $A_1 A_2 \ldots A_n$ are applications and $A_i = P_1 P_2 \ldots P_h$ then $A_1 \ldots A_n = A_1 \ldots A_{i-1} P_1 \ldots P_h A_{i+1} \ldots A_n$.

Clearly, this result follows from a succession of applications of Theorem 6.

Theorem 8. If Z is a maximal rule which is not also minimal, then there is a sequence $X_1, X_2, \ldots, X_n$ $(n > 1)$ of rules such that $X_1 X_2 \ldots X_n = Z$, X_1 is a minimal rule and, if $j > 1$, X_j is a maximal rule.

There exists a minimal rule X_1 and an integer i $(i > 1)$ such that $X_1 X_2 \ldots X_i = Z$ (Def 9 and Def 3). Let n be the least such integer. Suppose that for some $j > 1$, X_j is not a maximal rule. Then, the second member of X_j contains a class symbol and $X_j X_{j+1}$ is an application. By Theorem 6, if $A = X_j X_{j+1}$ then $X_1 \ldots X_{j-1} A X_{j+2} \ldots X_n = Z$. But the sequence $X_1, \ldots, X_{j-1}, AX_{j+2}, \ldots, X_n$ contains n-1 rules, which is contrary to the definition of n. Therefore, if $j > 1$, X_j is a maximal rule.

Theorem 9. If c is a class symbol in F then $E \in [c]$ if and only if there exists a minimal rule $X = \langle c, \langle d_1, d_2, \ldots, d_p \rangle, \delta \rangle \in F$ such that $E \in [d_1 \frown d_2 \frown \ldots \frown d_p]$.

If $E \in [c]$ there exists a maximal rule of the form $\langle c, E, \Lambda \rangle$, (Def 11) and a sequence $X_1, X_2, \ldots, X_n$ such that $X_1 X_2 \ldots X_n = \langle c, E, \Lambda \rangle$ where X_1 is a minimal rule and X_i is a maximal rule if $1 < i \leq n$, (Theorem 8). Let $X_1 = \langle c, \langle a_1, a_2, \ldots, a_p \rangle, \langle a_{j_1}, a_{j_2}, \ldots, a_{j_q} \rangle \rangle$ and $X_i = \langle b_i, E_i, \Lambda \rangle$. Then $b_i = a_{j_{i-1}}$ and $q = n-1$. For $k = 1, 2, \ldots, p$ let $D_k = a_k$ if a_k is an alphabetic symbol. If a_k is a class symbol, there exists just one integer m such that $a_k = a_{j_m}$ (Def 1), so that $a_k = b_{m+1}$. In this case let $D_k = E_{m+1}$. Thus $D_k \in [a_k]$ and $E = D_1 \frown D_2 \frown \ldots \frown D_p$, so that $E \in [a_1 \frown a_2 \frown \ldots \frown a_p]$, (Defs 12, 13).

The converse is established immediately by Theorem 4.

Theorem 10. If $X = \langle A, a, \gamma \rangle$, let $n(X)$ denote the number of terms in the sequence a. Then, if XY is an application, $n(X) + n(Y) = n(XY) + 1$.

Let $X = \langle A, a, \gamma \rangle$, $Y = \langle B, b, \delta \rangle$ and $XY = \langle A, c, \lambda \rangle$. A particular term of the sequence a is replaced by the sequence b to obtain the sequence c, (Def 2). Sequence c contains all of the terms of b and all but one of the terms of a; consequently, $n(XY) = n(X) + n(Y) - 1$.

Theorem 11. If $A_1 A_2 \ldots A_q$ is a rule and $n(X)$ the function defined in Theorem 10, then $n(A_1) + \ldots + n(A_q) = n(A_1 A_2 \ldots A_q) + q-1$.

Let B_j denote $A_1 A_2 \ldots A_j$ for $j = 1, 2, \ldots, q$. Then, $B_j A_{j+1} = B_{j+1}$ for $j < q$, and by Theorem 10,

$$n(B_j) + n(A_{j+1}) = n(B_j A_{j+1}) + 1$$
$$= n(B_{j+1}) + 1$$

$n(A_{j+1}) = n(B_{j+1}) - n(B_j) + 1$. Summing,

$$\sum_{j=1}^{q-1} n(A_{j+1}) = \sum_{j=1}^{q-1} \{n(B_{j+1}) - n(B_j) + 1\}$$

$$= n(B_q) - n(B_1) + q-1 = n(A_1A_2 \ldots A_q) - n(A_1) + q-1.$$

Theorem 12. Let n(X) be defined as in Theorem 10. There exists an integer k such that if $X_1X_2 \ldots X_q$ are applications and $n(X_i) = 1$, $1 \leq i \leq q$, then $q \leq k$.

If c is a class symbol in F, then there exists a rule X (Def 7) and a minimal rule Y, $Y \leq X$ (Def 9, Axiom 2), each having c as its first member (Def 2). Hence, there are a finite number, k, (Def 9, Axiom 3) of class symbols in F.

Suppose that there is a sequence of k+1 rules in F such that $X_1X_2 \ldots X_{k+1}$ is a rule, and $n(X_i) = 1$ for each rule in the sequence. Then the second member of each rule of the sequence preceding X_{k+1} is a class symbol. Consequently, if $j < k+1$, then X_jX_{j+1} is an application. If A_i is the first member of rule X_i, then the class symbol A_i does not occur as the first or second member of any rule following X_i. For, suppose A_i is the first or second member of some rule X_{i+s}. Either A_i is the second member of rule X_{i+s} or A_i is the second member of rule $X_{i+(s-1)}$. Therefore, either $X_i \ldots X_{i+s}$ or $X_i \ldots X_{i+(s-1)}$ is a rule of the form $\langle A_i, A_i, \gamma \rangle$,

contrary to Theorem 1. It follows then that no two rules of the sequence $X_1, \ldots, X_{k+1}$ have the same first member. But this is impossible since there are only k class symbols and there are k+1 rules in the sequence.

Definition 14. The applications $X_1X_2 \ldots X_n$ are called a derivation of Z from Y whenever $X_1 = Y$ and $X_1X_2 \ldots X_n = Z$.

Theorem 13. If $X \leq Z$ in F, there exists an integer M such that every derivation of Z from X contains less than M rules.

Let $X_1X_2 \ldots X_p = Z$ and let $B_1 = X_1$ and $B_{i+1} = B_iX_{i+1}$ for $1 \leq i < p$. Then, by Theorem 11,

i) $$n(B_i) + n(X_{i+1}) = n(B_{i+1}) + 1$$

so that

ii) $$n(X_{i+1}) - 1 = n(B_{i+1}) - n(B_i).$$

Summing,

iii) $$\sum_{i=1}^{p-1} \{n(X_{i+1}) - 1\} = \sum_{i=1}^{p-1} \{n(B_{i+1}) - n(B_i)\}$$

$$= n(B_p) - n(B_1)$$

$$= n(Z) - n(X_1).$$

If c_i is the integer $n(B_{i+1}) - n(B_i)$, then $c_i \geq 0$. If $c_i = 0$, it follows from ii), above, that $n(X_{i+1}) = 1$. Furthermore, from iii), $c_1 + \ldots + c_{p-1} < n(Z)$. Consider the set of all terms of the sequence $c_1, \ldots, c_{p-1}$ which are not zero. Let $c_{k_1}, \ldots, c_{k_r}$ denote this sub-

sequence. The number of terms in this sequence is less than $n(Z)$. Consider the set S of maximal consecutive subsequences of $c_1, \ldots, c_{p-1}$ whose terms are all zero. The set S has, at most, $n(Z)$ members. Each sequence in S corresponds to a consecutive subsequence of $X_1, \ldots, X_p$ each rule of which has the property that $n(X) = 1$. By Theorem 12, the length of such a sequence is not greater than K, the number of class symbols in F. The length p-1 of the sequence $c_1, \ldots, c_{p-1}$ is therefore less than $[n(Z) + 1]K + n(Z) = M$. No derivation of Z, therefore, contains more than M rules.

Theorem 14. If Z is a rule of F then there exists a finite sequence $X_1, X_2, \ldots, X_p$ of minimal rules in F such that $X_1X_2 \ldots X_p = Z$.

If Z is a rule of F there exists a minimal rule X in F such that $X \leq Z$ in F (Def 9). Let S denote the set of derivations of Z from X (Def 14). S is not empty (Def 3). Since, for some M, the number of rules in each derivation in S is less than M (Theorem 13), there must be at least one having the greatest number of rules. Let $X_1X_2 \ldots X_p = Z$ where $X_1 = X$ be one such derivation. Then, each rule in the sequence must be minimal, for, if there is some rule X_i in the sequence which is not minimal, then $X_i = P_1P_2 \ldots P_q$, $q > 1$, (Defs 3, 4, 9). But, by Theorem 11, $X_1X_2 \ldots X_{i-1}P_1 \ldots P_qX_{i+1} \ldots X_p = Z$, yielding a

derivation of Z containing more than p rules.

Definition 15. The set of minimal rules in the formation structure F is said to be the grammar for F.

Theorem 15. The set of alphabetic symbols in a formation structure F is finite.

By Def 8, if a is an alphabetic symbol in F, it occurs in the second term of a rule R of F. By Theorem 14, there exist minimal rules $X_1, X_2, \ldots, X_n$ in F such that $X_1 X_2 \ldots X_n = R$. Symbol a, therefore, occurs in the second term of a minimal rule in F. The set of minimal rules is finite (Def 9) and, consequently, the set of alphabetic symbols in F is finite.

Theorem 16. The set of all expressions in a formation structure F is at most denumerably infinite.

The set of alphabetic symbols in F is finite (Theorem 15). Every expression in F is a finite sequence of alphabetic symbols. If M is a finite set, the collection of finite sequences of elements of M is a denumerable collection.

Definition 16. A symbol c in the formation structure F is recursive in F if and only if there exists a rule in F having c as its first member and as at least one term of its second member.

Theorem 17. If c is a recursive symbol in F, then the syntactic class [c] contains a denumerably infinite subset.

Let C be the set of all rules in F whose first member is the class symbol c, and whose second member has c as at least one term (Def 16). If X is a rule, let n(X) denote the number of terms in the second member of X.

If X is a rule in C then there is a rule Y in C such that $n(X) < n(Y)$. To prove this, consider a maximal rule Z, such that $X \leq Z$ (Def 9). There is a sequence $X_1, X_2, \ldots, X_n$ of rules in F such that $X_1 = X$ and $X_1X_2 \ldots X_n = Z$ (Def 3). Let p be the largest integer i such that the rule $X_1 \ldots X_i$ belongs to C but $X_1 \ldots X_{i+1}$ does not. Let $A = X_1X_2 \ldots X_p$. The class symbol c occurs as a term of the second member of A, and this term occurs first in the third member of A; otherwise, $X_1 \ldots X_{p+1}$ would belong to the collection C. It follows that AX_1 is defined (Def 2) and is a rule of C. By Theorem 10, $n(AX_1) + 1 = n(A) + n(X_1)$. If R is a rule in C, $n(R) > 1$ (Def 16, Theorem 1). Therefore, $n(AX_1) > n(X_1)$ and there is a rule Y, namely AX_1, such that $n(Y) > n(X)$.

If X is a rule in C, then there is a maximal rule Z in F such that $n(X) \leq n(Z)$, (Def 9, Theorem 11). The second member of Z is an expression in the syntactic class [c], (Def 11).

The conclusion to the theorem follows from the two results above. For, if [c] does not contain a denumerably infinite

subset, it is finite (Theorem 16), and contains an expression E whose length k is maximal in [c]. By the first result, there is a rule Y in C such that $n(Y) > k$. By the second, there is a maximal rule Z such that $n(Z) \geq n(Y) > k$, but the second member of Z is an expression in [c] of length greater than k.

Definition 17. G is said to be an explicit relation in the formation structure F if and only if G is a binary relation consisting of ordered couples $\langle X, Y \rangle$ of minimal rules in F and, for each $\langle X, Y \rangle \in G$, the third members of X and Y are either both empty or they order the same number of symbols.

Definition 18. A binary relation T is called an implicit relation in F whenever, for some explicit relation G in F, $\langle U, V \rangle \in T$ if and only if there exists a sequence $\langle X_1, Y_1 \rangle, \langle X_2, Y_2 \rangle, \ldots, \langle X_p, Y_p \rangle$ of elements of G such that $X_1 X_2 \ldots X_p = U$ and $Y_1 Y_2 \ldots Y_p = V$. T is said to be derived from G.

Definition 19. T is a transformation structure in F means that:

(i) T is an implicit relation in F derived from an explicit relation G in F,

(ii) the domain and converse domain of T are formation structures, and

(iii) the field of T is F.

Theorem 18. If G is an explicit relation in F such that the trans-

formation structure T is derived from G, then the field of G is the grammar for F.

The field of T is F (Def 19), so that if X is a minimal rule in F then, for some $Y \in F$, either $\langle X, Y\rangle \in T$ or $\langle Y, X\rangle \in T$. If $\langle X, Y\rangle \in T$ then (Def 18) there exists a sequence $\langle X_1, Y_1\rangle$, $\langle X_2, Y_2\rangle, \ldots, \langle X_p, Y_p\rangle$ of elements of G such that $X_1X_2 \ldots X_p = X$. But since X is minimal, the sequence has only one term, whose first member is X. Thus X is in the field of G. Conversely, if X is in the field of G, then (Def 17) X is minimal in F and (Def 15) is in the grammar for F.

Theorem 19. If the rules X_1 and Y_1 have the same number of class symbols and rules X_2 and Y_2 have the same number of class symbols, and $X_1X_2 = U$ and $Y_1Y_2 = V$ are applications, then the rules U and V have the same number of class symbols.

This property follows directly from the definition of application (Def 2 and Axiom 5, Def 9).

Theorem 20. If T is a transformation structure in F and $\langle U, V\rangle \in T$ then U is maximal in F if and only if V is maximal in F.

If $\langle U, V\rangle \in T$ then there exists a sequence $\langle X_1, Y_1\rangle, \ldots, \langle X_p, Y_p\rangle$ of elements of some explicit relation G in F such that $X_1X_2 \ldots X_p = U$ and $Y_1Y_2 \ldots Y_p = V$ (Def 18). Each pair of rules X_i and Y_i have the same number of class symbols (Def 17). Let $A_j = X_1 \ldots X_j$ and

$B_j = Y_1 \ldots Y_j$. By Theorem 19, $A_j X_{j+1} = A_{j+1}$ and $B_j Y_{j+1} = B_{j+1}$ have the same number of class symbols, $j = 1, 2, \ldots, p-1$.

Theorem 21. If formation structures F_1 and F_2 have no class symbol in common, then their union $F_1 \cup F_2$ is a formation structure.

To prove this, it is sufficient to observe that the axioms of Def 9 are satisfied in $F_1 \cup F_2$. By the hypothesis and Def 2, there exist no rules $X \in F_1$ and $Y \in F_2$ such that XY is an application. Hence if $X, Y \in F_1 \cup F_2$ and $XY = Z$ is an application then either $X, Y, Z \in F_1$ or $X, Y, Z \in F_2$, but not both. Thus Axioms 1 and 4 are satisfied. Since no rule is in $F_1 \cup F_2$ which is not in F_1 or F_2, Axioms 2 and 5 are satisfied. The set of minimal rules in $F_1 \cup F_2$ is finite, as required by Axiom 3, since this is the union of two finite sets.

Theorem 22. If F_1 and F_2 are formation structures having no class symbols in common, any implicit relation T in $F_1 \cup F_2$, whose domain and converse domain are F_1 and F_2, respectively, is a transformation structure derived from the grammar for $F_1 \cup F_2$.

$F_1 \cup F_2$ is a formation structure (Theorem 21) and, by the hypothesis, the three properties of Def 19 are satisfied. The field of the explicit relation in $F_1 \cup F_2$ from which the relation T is derived is the grammar for $F_1 \cup F_2$ (Theorem 18).

Definition 20. Let S denote a set of explicit relations $G_1, G_2, \ldots, G_s$ in F. Let T be an implicit relation in F derived from the union $G = G_1 \cup G_2 \cup \ldots \cup G_s$ of elements of S. Then, the semantic relation $R = R_1 R_2 \ldots R_p$ in F is the relation such that $\langle U, V \rangle \in R$ if and only if each term of $R_1, R_2, \ldots, R_p \in S$, and there exists a sequence $\langle X_1, Y_1 \rangle, \langle X_2, Y_2 \rangle, \ldots, \langle X_p, Y_p \rangle$ such that $\langle X_i, Y_i \rangle \in R_i$ for $i = 1, 2, \ldots, p$ and $X_1 X_2 \ldots X_p = U$ and $Y_1 Y_2 \ldots Y_p = V$.

Definition 21. If U is in the domain of the semantic relation R then let $[U, R]$ denote the set such that $V \in [U, R]$ whenever $\langle U, V \rangle \in R$. $[U, R]$ is called the R-semantic class in F generated by U.

Definition 22. The semantic relation $R = R_1 R_2 \ldots R_p$ is said to be a semantic equivalence relation in F whenever it is true that $\langle U, V \rangle \in R$ if and only if $\langle U, U \rangle \in R$ and $\langle V, V \rangle \in R$.

Theorem 23. If R is a semantic equivalence relation in F and U and W are elements of the domain of R, then the R-semantic classes gererated by U and W are identical.

Assume that there exists a rule $V \in [U, R]$ but $V \notin [W, R]$. But $\langle U, V \rangle \in R$ implies that $\langle V, V \rangle \in R$ and since $\langle W, W \rangle \in R$, $\langle W, V \rangle \in R$ (Def 22). Then, $V \in [W, R]$ (Def 21), contrary to the hypothesis.

Definition 23. If R is a semantic equivalence relation in F then the unique R-semantic class in F generated by every rule in the

domain of R is called the R-semantic equivalence class in F, and is denoted by [R] .

Theorem 24. If $U \in [R]$ and $W \in [R]$ then the third members of U and W are either both empty or both order the same number of terms.

Every semantic relation is a subrelation of some implicit relation (Def 18, 20). The conclusion follows from Theorem 19.

Definition 24. The degree of the semantic equivalence class [R] is a non-negative integer specifying the number of symbols ordered by the third member of rules of [R] .

Theorem 25. If each of $R_1, R_2, \ldots, R_p$ is a semantic equivalence relation in F then the semantic relation $R = R_1 R_2 \ldots R_p$ is a semantic equivalence relation in F.

From Def 20, $\langle U, V \rangle \in R$ if and only if $X_1 X_2 \ldots X_p = U$ and $Y_1 Y_2 \ldots Y_p = V$ and $\langle X_i, Y_i \rangle \in R_i$, $i = 1, 2, \ldots, p$. If each of $R_1, R_2, \ldots, R_p$ is a semantic equivalence relation in F, then $\langle X_i, Y_i \rangle \in R_i$ if and only if $\langle X_i, X_i \rangle$ and $\langle Y_i, Y_i \rangle$ are in R_i. Consequently, $\langle U, U \rangle$ and $\langle V, V \rangle$ are in R if $\langle U, V \rangle \in R$ and, conversely, $\langle U, V \rangle \in R$ if $\langle U, U \rangle$ and $\langle V, V \rangle$ are in R. R is therefore a semantic equivalence relation (Def 22).

Definition 25. Let T be called an equivalence transformation structure in F whenever T is a transformation structure in F derived from

$G = G_1 \cup G_2 \cup \ldots \cup G_s$, and each of $G_1, G_2, \ldots, G_s$ is a semantic equivalence relation in F.

Theorem 26. If T is an equivalence transformation structure in F, then every rule in F must be contained in some semantic equivalence class [R] in F.

By Def 25, T is a transformation structure derived from an explicit relation $G = G_1 \cup G_2 \cup \ldots \cup G_s$ and each G_i is a semantic equivalence relation in F. If X is a rule in F, let $X_1, X_2, \ldots, X_n$ be the sequence of minimal rules such that $X_1 X_2 \ldots X_n = X$ (Theorem 14). The field of G is the grammar for F (Theorem 18) so that, for each minimal rule X_j in the sequence, $\langle X_j, X_j \rangle \in R_j$ (Def 22), where R_j is some member of the collection $G_1, \ldots, G_s$. Then, $\langle X, X \rangle \in R = R_1 R_2 \ldots R_n$ (Def 20), and R is a semantic equivalence relation (Theorem 25). Therefore, $X \in [R]$, (Def 23).

Definition 26. If E is an expression in F then the sequence $R_1, R_2, \ldots, R_p$ is to be called a content of E whenever, for some class symbol c in F, $R = R_1 R_2 \ldots R_p$ is a semantic equivalence relation in F and $\langle c, E, \Lambda \rangle \in [R]$.

Theorem 27. Every expression E in the formation structure F has at least one content.

E is the second member of some maximal rule $\langle c, E, \Lambda \rangle$ in

F (Def 10). By Theorem 26, there exists a semantic equivalence relation $R = R_1 R_2 \ldots R_p$ in F such that $\langle c, E, \Lambda \rangle \in [R]$. The sequence $R_1, R_2, \ldots, R_p$ is therefore a content of E, (Def 26).