II

THE DERIVATION

OF SYNTACTIC RELATIONS

FROM A LATTICE MODEL

A. F. PARKER-RHODES

M. MASTERMAN

See

A New Model of Syntactic Description
A.F. Parker-Rhodes and M. Masterman
Cambridge Language Research Unit, M.L. 142

CAMBRIDGE LANGUAGE RESEARCH UNIT
20 MILLINGTON ROAD
CAMBRIDGE, ENGLAND

ABSTRACT PART II

In this part the argument is reversed: using logical rather than linguistic considerations a general model of syntactic structure is set up. In the model a subject-predicate distinction is developed, and different syntactic functions are defined and related, by being placed in a simple lattice. A classification of groups into endocentric and exocentric is constructed, and components of groups are qualified as governors or dependents. Algorithms arising out of the lattice model for finding the functions of compound groups from those of their components are given.

PART II. THEORY OF THE GENERAL SYNTAX LATTICE

The Primary Syntax Lattice

In Part I it was shown that it is possible to construct an argument in which, starting from an asymmetrical relation of 'replaceability' (defined more or less as a structural linguist would define it), and deriving from it a symmetrical relation of 'equipollence' (similarly defined), a system of _syntactic functions_, with the mathematical properties of a _lattice_ can be obtained (1).

When, however, we ask "_what_ lattice, as opposed to _some_ lattice, does this system form?" we are faced with a difficulty. For to find this out we should have to carry out more replacement than is humanly, or even mechanically, possible. That is, for each language to which the general system of syntactic functions is to be applied, we should have to take an indefinitely large number of indefinitely long texts, list the words in each, and, for each word in each text in turn, consider whether it could be replaced by each member of the list. Of course we can always say that, if we assume that this has been done, the replacement patterns holding between the different words in the language, - any language, - would indicate a very simple pattern of complementarities, and that this system would in turn dictate the form of the general lattice of syntactic functions which is constant for different languages. The point is, that since the enterprise is infinite, we can never complete it. We must therefore derive our general syntax lattice another way.

Two alternative procedures immediately suggest themselves :
(a) A closed model of what has been shown, in Part I, to be an open state

of affairs could be obtained if a single text was taken and its replaceability examined; research along these lines, using Richards and Gibson's "English through Pictures" (2) as a text, will in due course be carried out. This approach has the disadvantage, however, that it is too small in scale to give more than a hint of the information required in order to decide what lattice, or at least what type of lattice, the general syntax lattice for all languages must be.

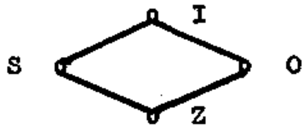
(b) Alternatively, we may try to utilise the intuition on which philosophical thinking has been based for 2500 years, namely that the fundamental sentence patterning is on the subject-predicate principle. * To proceed solely from Western European philosophical thinking and to look for the concrete exemplification of this principle in any language is, on the other hand, open to question: for other equally sophisticated logicians, whose thinking is based on Far-Eastern cultures, do not seem to feel any need for the principle (3). What we in fact want, therefore, is a schema of syntactic functions from which the subject-predicate sentence pattern can, though it need not, be derived. That is, to speak in terms of propositional logic, we want as the units of our system not x or y and P, but weaker units from which combinations of the subject-predicate form can be built up. This paper attempts to provide such a schema.

Bearing in mind, as our starting point, both the generality of the subject-predicate principle and the formal argument mentioned above to the effect that a system based on replacement will lead to a lattice, we can now consider the elementary lattice generated by two elements given below. We will

* The syntactic analysis for machine translation initiated by Ida Rhodes and developed at Harvard is explicitly built round the principle of finding the subject-predicate pattern in a sentence.

take this as the syntactic model of an Imaginary language; i.e. we will assume that all the substituents in the language, as 'substituent' is defined in Part I, can be handled by this classification.

Fig. 1. 4-element Boolean lattice representing the simplest imaginable type of language



All the points on this lattice are substituent functions:

S = substantive function
O = operative function
I = indeterminate, i.e. can function either as a substantive or as an operative
Z = what is in common between substantive and operative, i.e. the function of being a substituent in language.

Even in this simple schema the following lattice properties are used:

i) the 'side-to-side' symmetry of the lattice, i.e. its complementarity;
ii) the 'top-to-bottom' asymetry of the lattice, i.e. its inclusion relation;
iii) the 'join' relation '$\cup$' and
iv) the 'meet' relation '$\cap$' , i.e. that in a lattice any pair of elements a,b, has a unique upper bound, their join, $a\cup b$, and a unique lower bound, their meet, $a\cap b$.

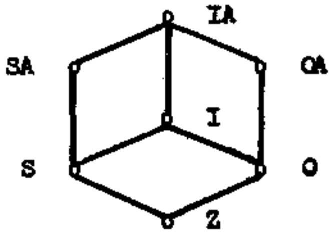
1) In giving this mathematical system its syntactical interpretation the symmetrical complementarity represented by the sides of the lattice, S and O, will be taken, a priori, as indicating the substantive-operative dichotomy: for we can say, logically, that the substantive element S = ~ O, and the operative element O = ~ S.

ii) The asymmetry determined by the inclusion relation will be interpreted, again a priori, in relation to what will be called the governor-dependent relation. (As in this simple lattice this interpretation can only be trivial, the point will be discussed more fully below).

iii) & iv) In the linguistic interpretation the join element $a \cup b$ is a unit of language which can, according to its positional collocation be either a or b', and the meet element $a \cap b$ is a (governor-dependent-wise qualified) group of the two units a and b which we will here call 'a clause'. The fact that the lattice axioms give us these two relations is for our linguistic purposes important: for the former takes account of the indeterminate elements in language, that is, of units which can vary in syntactic function according to their position, and the latter gives us the possibility of grouping units of language together (logically speaking, the possibility of constructing propositions of the form $S \cap O$).

Thus the lattice just considered, with the two primary functions S and O, gives us our basic schema. But it is clear that it is far too primitive as it stands for its use to be extended from our imaginary language to any real one. In order to obtain a more elaborate system the basic schema is therefore enlarged by the addition of points representing new functions which are included by the initial polar elements S and O respectively.

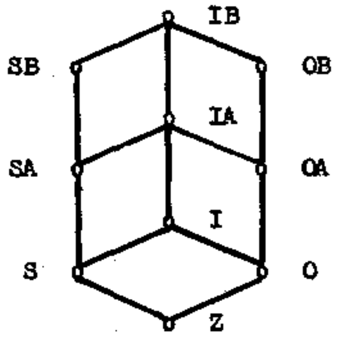
Fig. 2. Extension of the 4-element lattice



It is obvious that if we are to retain the substantive-operative distinction represented by the complementarity of the two sides of the lattice we cannot extend the system in any other way: for the more refined classification that we require must represent a sophistication of this fundamental division. The two new side points SA and OA are therefore interpreted as substantive and operative adjuncts, i.e., very roughly, as adjective and adverb; their join gives us the new indeterminate adjunct IA which also includes the initial indeterminate element I.

This extended lattice is, however, still judged inadequate, and two more functions representing 'secondary' adjuncts are therefore added.

Fig. 3. The 4-element lattice further extended: the primary syntax lattice

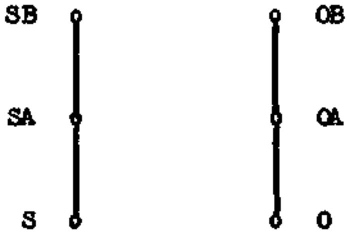


Preliminary empirical investigation has shown that, to a surprising extent, this system contains the 'hard core' of the general syntactic classification that we require *, and we shall therefore call it the primary syntax lattice. We shall also say that the function defining a substituent is the lattice position indicator + of the substituent.

* A more pessimistic way of putting this fact is to say that this lattice, weak as it is, represents all that we find we can get when the notion of syntactic classification, which is essentially unilingual, is stretched so as to make it apply to all languages.

+ This expression was originally used in a more elaborate sense; the use given above has, however, been found more satisfactory.

Fig. 4.

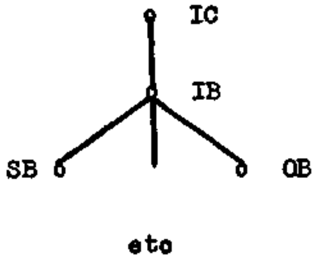


These by themselves are inadequately structured: but by the rules of lattice theory we can automatically generate from them meet and join elements and obtain the more realistic and fruitful classification represented by the primary schema. In choosing this particular mathematical model, therefore, we can start with an apparently simple schema and nevertheless derive from it, by purely 'mechanical' means, a much richer structure.

The points IC (C) and ZA

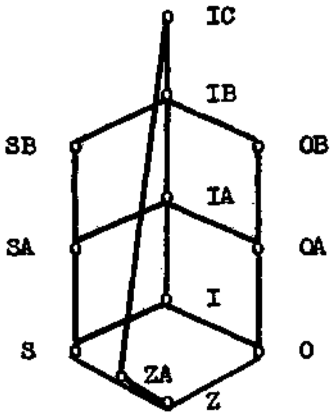
While the primary lattice as it stands has functions representing, very roughly, nouns and adjectives and verbs and adverbs, it has none for prepositions or conjunctions, i.e. for the primary syntactic auxiliaries of language. By treating prepositions as post-verbs and giving them the function OB we can incorporate them in the schema in a not wholly arbitrary way. Conjunctions, or connectives as the logician understands them, are, however, still not accounted for. On the other hand there is a sense in which they are built into the system: for we have interpreted the join relation as an alternative, i.e. as the basic logical connective 'and-or'. Using this fact we can create a new point at the top of the lattice to define the conjunctive function.

Fig. 5. The point IC (C)



There is a further distinction, however, which must be made. With the point IC alone we cannot distinguish conjunctions which, to put it crudely, connect words from those which connect clauses. We therefore introduce a new point to define the latter.

Fig. 6. The point ZA



The introduction of these points completes the primary lattice.

Sophistications of the primary lattice

The meet algorithm

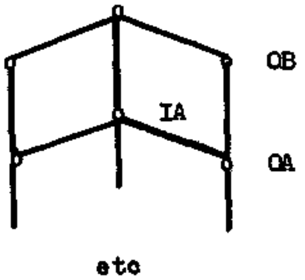
II. 8

Having set up the classification represented by the primary lattice, we can now use it in determining the function of groups of substituents. Here we introduce the first algorithm derivable from the theory, which we will call the meet algorithm:
the function of a group of substituents is the meet of the functions of its components.

If we look at the nomenclature of the primary lattice, bearing in mind the fact that the meet of a pair of elements is 'what is in common' between them, we can see how the algorithm works. The meet of the points S and SA., for example, is S; and indeed the notion of substantive is what is in common between the notions of substantive adjunct. Similarly, the meet of OA and O is O, the latter representing the common notion of operative. (The relation is transitive, so the meet of SB, SA and S is S). For either side of the lattice, therefore, the algorithm works in a straightforward way and gives linguistically satisfactory results: for it is from a linguistic point of view clear that if we group substituents with a common character, the resulting group should also have this character.

The algorithm fails, harmlessly, where elements with indeterminate functions are concerned. Consider, for example, the meet *of* the *two* points IA and OB as shown in the following diagram:

Fig. 7.



The meet is OA which does not represent what is common to both OB and IA. The failure is, however, logically harmless, once it is remembered that the points in the middle of the lattice define substituents with alternative functions: thus IA is 'can be either OA or SA'. When such a substituent is to be grouped with one with an O-type function, therefore, the O-alternative is selected; and when such a substituent is to be grouped with one with an S-type function, the S-alternative is taken. The meet algorithm can in this case be amplified without destroying the consistency of interpretation of the system.

The algorithm fails logically, however, and therefore vitally, when we consider the meet of any two points on opposite sides of the lattice, i.e. when the meet of any of the substituents concerned is Z. (The meet Z of two substituents one of which already has the function Z is of course dealt with by the algorithm as described above). For here we are faced with the fact that this point in the lattice can be interpreted in different ways: thus we have interpreted 'meet' as 'what is in common' between the two elements generating the meet, and Z should therefore represent what is in common between operative and substantive; and given the complementarity of the two sides of the lattice we can only describe this, very weakly, as 'the property of being a substituent in language'. This is serious, since we also interpreted Z, much more strongly, as representing the syntactic function of 'clause'.

Moreover, a further difficulty arises in this case: the meet of any pair of points on opposite sides of the lattice is Z; and while it is linguistically plausible to treat a substantive-operative group as a clause, we cannot plausibly treat a substantive-subadjunct-operative-subadjunct group as a full clause. We require, therefore, not only an explanation, within the theory, of the groups where the meet algorithm as it stands breaks down, but also a way of handling these cases which takes account of the particular, and not merely the general, character of their components.

Endocentric and exocentric groups

In order to deal with the situation just described, we now introduce a distinction between endocentric and exocentric groups. This can be defined, very crudely, along lines familiar to linguists, as follows :

an endocentric group of substituents is a group which has the same function as one of its components;

an exocentric group is a group which has a function different from that of any of its components.

It is clear, in terms of our previous discussion, that the meet algorithm holds for endocentric groups and fails for exocentric groups. We must now, therefore, develop our lattice schema to take account of exocentric groups; and we can only do this if we give a positive, and not merely a negative, definition of exocentric group. In fact, the strength or weakness of this syntactic theory as a whole is primarily determined by its success or failure in giving an adequate and precise account of the notion of the character of an exocentric group in language.

To develop the lattice structure we introduce the notion of convergences. This must be distinguished from that of complement which we discussed in connection with the initial 4-element 'diamond' lattice. It was there pointed out that the two sides of the lattice S and O were complements of each other, the definition of complement being that for any pair of elements, a,b in a lattice, $a \cup b = I$ and $a \cap b = O$. We interpreted this complementarity as representing the substantive-operative dichotomy. In building up the system we retained the dichotomy, but if we examine the primary lattice it is apparent that it is not fully complemented. We cannot, therefore, connect the notions of exocentric group and complement in any very obvious way.

Moreover, as just described, the notion of complement applies to _points_ in a lattice. In the meet algorithm, however, we make use of the _relation between points_ in a lattice, i.e. the inclusion relation. As, in order to account for subject-predicate groups, we want a notion of contrast in some form, we must try to give this notion a meaning in terms of the inclusion relation. And in fact we can find in lattice algebra a notion of contrast with respect to the inclusion relation which is analogous to that of complementarity between points. This is the notion of converseness:

the converse of a relation r is the relation r' such that a r' b if and only if b r a.

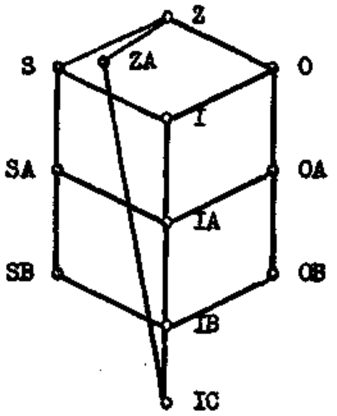
This gives us, as required, a top-to-bottom contrast for a pair of points, as opposed to the side-to-side contrast of complementarity. We derive from it, moreover, the notion of _dual lattice_, which gives us, for any lattice, the contrasting system:

the dual L' of a lattice L is the lattice defined by the converse relation on the same set of points.

The secondary lattice

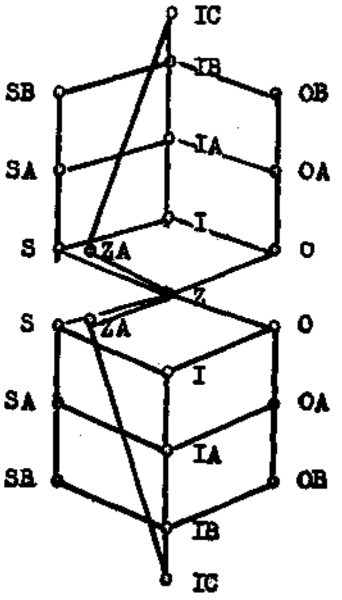
In terms of our syntactic schema, the dual of the primary lattice, i.e. the lattice with the inclusion relation working in the opposite way, is the following system which we will call the _secondary syntax lattice_.

Fig. 8. The secondary syntax lattice



To bring out the 'mirror-image' character of the dual we give the two together.

Fig. 9. The primary and secondary syntax lattices



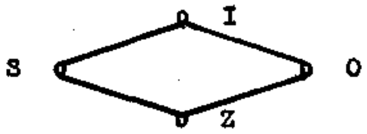
This figure brings out the fact that the dual lattice depends on the primary lattice by the point Z, i.e. by the point representing the clause. Two consequences should be noticed: that Z, as the connecting point between the two lattices, is unique, and that all the points included by Z, i.e. all the points in the dual lattice, must represent groups. In a first attempt to deal with our problem, therefore, we can say that we have a system (for the two lattices together constitute a lattice) divided into two 'fields': an endocentric field represented by the primary lattice, and an exocentric field represented by its dual.

This extension of our original schema would appear to give us what we need: for we have obtained a system in which the notion of exocentric group is properly represented, and further, in which we allow different kinds of exocentric group. Closer examination reveals, however, that we cannot use the new exocentric part of the system: for the meet of any pair of points in the primary lattice is still Z, and we never enter the dual lattice. Our new system as it stands is thus only a halfway measure: we have constructed, in creating the dual, a system for dealing with exocentric groups; we must now make it possible to obtain an exocentric group as the meet of every pair of complementary points in the primary lattice, i.e. to enter the dual from every pair of points in the primary lattice. Put crudely, this means that we have to connect each point in the primary lattice with each point in the dual; in lattice terms it means that what we want is the cardinal product of the primary lattice and its dual.

The full (product) lattice

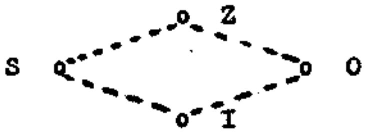
That this is so is best brought out if we consider a very simple product lattice. We will therefore take as our 'primary' lattice the 4-element lattice which is the simplest form of our syntactic lattice.

Fig. 10. The simple 'primary' lattice



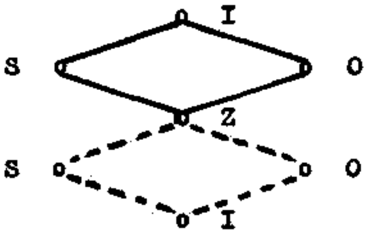
Its dual we will represent thus:

Fig. 11. The dual of the simple 'primary' lattice



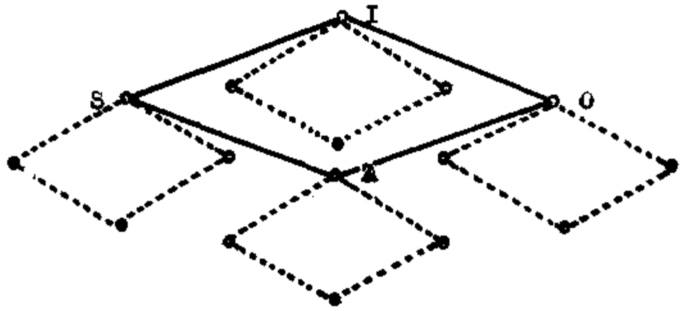
In constructing the product lattice we start with the 'primary' lattice and its dual as follows.

Fig. 12. The simple 'primary' lattice and its dual



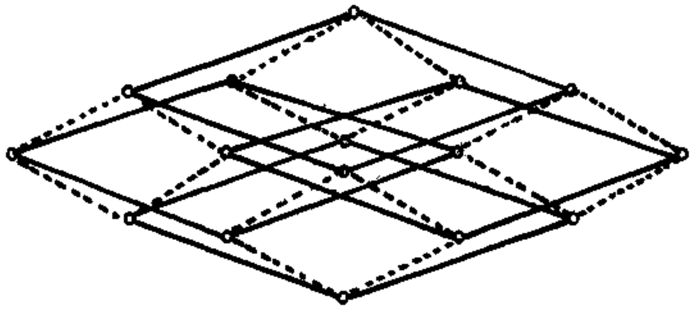
The next stage is to 'hang' the dual from the other points in the 'primary' lattice: this connects each point in the 'primary' lattice with each point in the dual.

Fig. 13. <u>Constructing the product lattice</u>



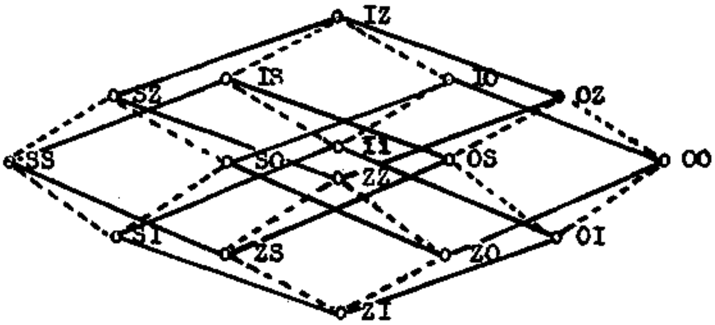
We now, turning the lattice upside down in our minds, 'hang' the 'primary' lattice from each point in the dual, to connect each point in the latter with each point in the former. The resulting system, which is the full product lattice, is as follows:

Fig. 14. <u>The product lattice</u>



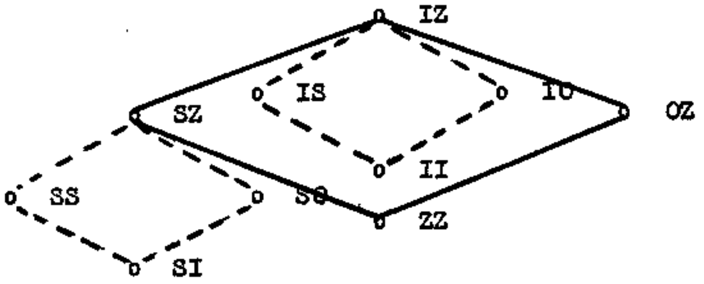
We must now show what we are doing in terms of our syntactic interpretation. We start by labelling all the points in the product lattice. The convention for doing this is to give the letter representing the 'primary' lattice point first, followed by the letter for the 'secondary' lattice. Each point in the product lattice will therefore be labelled by a pair of letters representing the relevant functions in the 'primary' and dual lattices respectively.

Fig. 15. The product lattice labelled



As the points in the 'primary' lattice define unit substituents, and as the points in the dual lattice define groups, we are giving, in our 'double' labelling, firstly the specification of the function which a substituent itself has, and secondly the function of a group in which it can figure; in fact, for any particular substituent function defined by the 'primary' lattice we give, as groups in which it may in principle figure, all the groups defined by functions in the dual lattice. This is best brought out if we 'abstract' the 'primary' lattice and two of its dependent dual lattices as follows.

Fig. 16. 'Abstract' from the produot lattice



All the points in the dual lattice dependent from S represent substantive substituents which are distinguished by the kinds of group in which they figure. Similarly, all the elements in the dual lattice dependent from I represent substituents with an indeterminate function and the kinds of group of which they can be a member. In creating this extended system, therefore, we are introducing a more refined classification of substituents: for we take into account not only the functions which they themselves have, but also the functions of the groups in which they can appear.

We must now see how the meet algorithm works in the extended lattice, But before we can do this we must discuss one feature of the lattice which we have not so far emphasised.

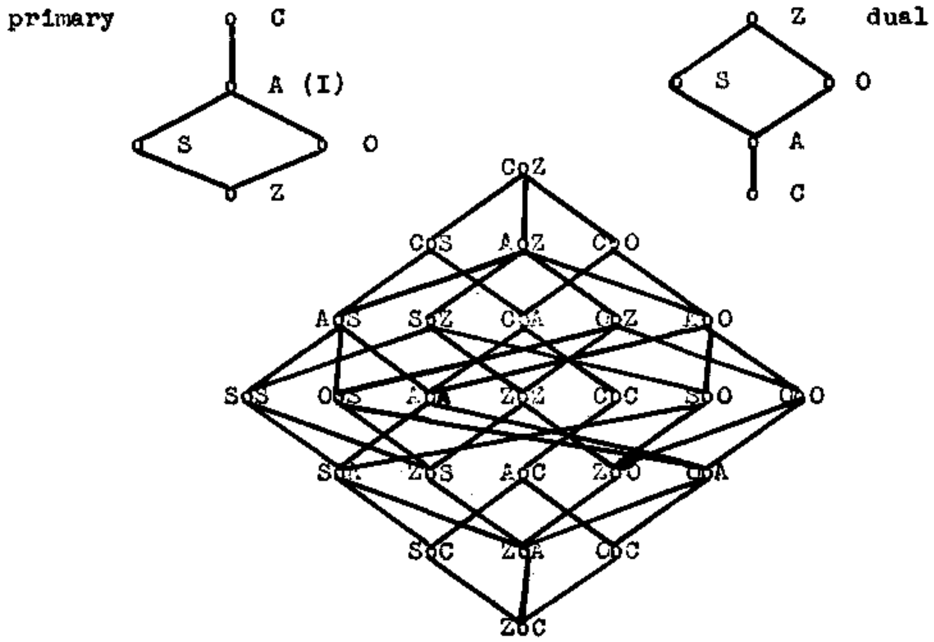
Although the product lattice as a whole appears to be what we may describe as 'homogeneous', it is derived from the 'primary' lattice and its dual with Z as their connecting point. Within the product lattice there **is thus a** sub-system with a special character which we can define, in **lattice** terms, by introducing the notion of centrality.

Conventionally, following Birkhoff, in a product lattice of the kind we are considering we find four <u>central elements</u> derived from the top and bottom elements of the two initial lattices; these are the two 'bounds' IZ and ZI, and the two 'centrals' II and ZZ. Both lattice-wise and from the syntactic point of view ZZ is the most important of the two centrals, and we shall therefore call it the positive central, and II the negative central.

We can now distinguish areas of the lattice with respect to the centrals, and in particular, with respect to ZZ. To do this we introduce the notion of ideal; put crudely, the <u>ideal</u> of an element in a lattice is the set of elements which it includes. When we take ZZ as our starting point, therefore, it is clear that its ideal is the initial, or primary exponent of, the dual lattice. More particularly, we shall say that the initial dual lattice is the principal ideal of ZZ, and that the initial 'primary' lattice is the dual principal ideal of ZZ. Thus by considering them as ideals of ZZ, we can separate the primary exponents of the 'primary' lattice and its dual from the system represented by the full product lattice.

Bearing these points in mind we must now return to the genuine syntax lattice. It is clear that if we start with the genuine primary syntax lattice and its dual as shown in Fig. 9, we can obtain a product lattice in the way that the illustrative product lattice was obtained. We shall call this product lattice the <u>full syntax lattice</u>. It is too complicated for visual diagramming, but one point should be noticed. In contrast to our illustrative 'primary' lattice, it is not self-dual. As the product lattice obtained from a lattice which is not self-dual differs slightly in general character from that obtained with a self-dual lattice, the product lattice derived from such a lattice is given below. This lattice of five elements is again a simplification of the primary lattice, being the smallest modular non-self-dual lattice.

Fig. 17. Product obtained with the 5-element non-self-dual lattice



Given the full syntax lattice we can now see how the meet algorithm works in it. For endocentric groups this is quite straightforward: as the members of the group may not have functions represented by points in the same exponent of the primary lattice, their meet will obviously not be a point in a particular exponent; but it will clearly be a point in some exponent of the primary lattice. We have thus, in the full lattice, merely obtained a more refined version of the algorithm as it worked for the primary lattice alone.

For exocentric groups the situation is more complicated. In most cases the algorithm gives an appropriate answer: in particular, we can, through having a more elaborate classification, separate exocentric groups which have functions other than Z (more strictly, than ZZ) from those which are genuine full clauses. In some cases however, the meet is a point in the principal, or 'lower', ideal of ZZ. This in itself is not unsatisfactory: for the groups will indeed be clauses; the difficulty arises when any attempt is made to treat a group of this kind as a member of another group. For the meet of any point in the lower ideal with any other point whatever will itself be in the lower ideal. As we defined ZZ as full clause, or sentence, this means that we can never, once we have a substituent defined by a point in the lower ideal, reach the stop-point in grouping represented by the full sentence. This last is unsatisfactory in all respects, and in order to deal with the problem we make use of the converse relation introduced above. What we need, in lattice terms, is to be able to get to a point above, i.e. including, ZZ; but as the inclusion relation is asymmetrical, there will be no other points in the system having the same function as one obtained as a meet in the lower ideal. We can, however, find one which is mathematically closely related: for the dual lattice is defined as the reverse of the primary lattice, i.e. as representing the converse relation on the same set of points. There is a sense, therefore, in which ZN in the dual lattice corresponds to NZ in the primary lattice; and it has been found experimentally that if, whenever we have a meet in the dual, we replace it by its converse point, linguistically satisfactory results will be obtained. We have thus a refinement of the meet algorithm which we will call the polar algorithm:

when the meet algorithm gives a point in the lower ideal of ZZ, replace this point by the corresponding point, or polar, in the upper ideal of ZZ.

We have thus divided exocentric groups into three classes: these have the common property that they do not have the function of any of their components, and differ according to whether they have the function ZZ, a function defined by a point in the lower ideal of ZZ, or any other function. Linguistically these three classes represent different degrees of 'completeness': those defined by ZZ are complete clauses; those defined by points in the lower ideal of ZZ are clauses, and those defined by any other point represent groups which are too incomplete to be called clauses.

The governor-dependent relation

In the system as now elaborated we can at last interpret the inclusion relation in a linguistically non-trivial way. To do this we correlate it with what we shall call the governor-dependent relation; and, from the linguistic point of view, and following Hays (4) we shall call this M T model of syntactic description the governor dependent, or GD model. We find in language that in any grouping of substituents one 'colours', or 'lends its tone to', the group as a whole, and that the rest 'are coloured', or 'have their tone lent to them'. We can say that the substituent which colours the group is the governor of the group, and that the other substituents are dependents. In a group, therefore, we have an asymmetrical relation which we will call the governor-dependent relation. The way in which the relation works in the lattice is best shown if we consider endocentric groups in the primary lattice. Suppose we have an endocentric group of two members with functions A and B respectively, and that in the lattice the point corresponding to A includes that corresponding to B. By the meet algorithm the group has the function B, and we can clearly say, therefore, that substituent with function B is the governor of the group. In such groups the governor-dependent relation in reverse, i.e. the dependent-governor relation is thus identified with the inclusion relation. This interpretation will not, however, hold for exocentric groups where the points defining the functions cannot include one another. In these cases we adopt the convention that the substituent with the O-type function is the governor.

In the primary lattice the situation is thus relatively straightforward. In the full lattice, however, we find exocentric groups which are not combinations of substituents with S and O-type functions, though they are diverse in their composition. For endocentric groups the existing interpretation of the governor-dependent relation is still satisfactory. Again, when the primary functions of two substituents represent S and O-type functions respectively, the substituent with the O-type primary function will be the governor. (This is clearly merely a refinement of the rule for exocentric groups in the primary lattice). The two new cases to be dealt with are as follows: firstly, when the primary functions are different, but one includes the other, i.e. when one of them has an I or Z function; and secondly, when the two primary functions are the same, but neither includes the other. In a sense these are strictly exocentric, but have endocentric features. In the first case we say that the substituent with the lower function, lower being defined by reference to the primary lattice, is the governor. In the second we say that the substituent with the O-type secondary function is the governor. We are thus treating groups of the first kind as if they were endocentric, and groups of the second kind as if they were exocentric.

The conjunction device

According to the theory, the function IC, being at the top of the lattice, can never be a governor: for by the meet algorithm the meet, for example, of IC and S will be S, and so on throughout. This result in fact correctly represents what actually happens in language: for a conjunctive substantive group like "boys and girls" does function as a substantive group. However, as it is necessary for practical purposes to distinguish conjunctive groups, which essentially have three elements, from other groups, which have two, the following conjunction device has been adopted:
when any point which is a join in the lattice is interpreted as the join-relation itself, it will be the governor of any substituent group in which it occurs.

The system described above may be compared with that of Lambek (5), although the latter was designed principally for the analysis of formal rather than natural languages. If Lambek's calculus is applied to the words of a natural language we find that they have to be given a large variety of different 'types' to correspond to their different syntactic uses. Consider, for example, the two uses of the preposition "for" as exemplified in the two sentences "John works for Jane" and "John's work for Jane is tedious". In the first the prepositional phrase "for Jane" functions as an adverb, in the second as an adjective; in Lambek's system it has to be given separate types to allow for this ambiguity. In contrast the function O.IB which we would give it in our system covers both these cases and avoids the multiplication of alternatives. A more fundamental difference concerns the number of possible types: for Lambek this is infinite, but for us it is 144. Our meet and polar algorithms allow an indefinite complexity of sentence patterns, whereas Lambek's simple replacement algorithm demands ever more elaborate types.

Conclusion

In linguistic terms we can now say that we have been building up different 'strengths' of exocentricity in groups of substituents, the strength of any group being gauged by the extent of the possibility of completing it. Those groups representing a meet outside the lower ideal of ZZ do not lead to the polar algorithm and are thus, exocentric-wise, not completeable. Those groups which represent a meet in the lower ideal of ZZ are completeable, in that by applying the polar algorithm and then the meet algorithm we can reach ZZ. (Linguistically interpreted such groups represent subordinate clauses). The last class contains those completed exocentric groups which, through the application of the polar and meet algorithms, have the function ZZ. (These, and these alone, are subject-predicate groups).

Thus we see that this theory of syntax has, by interpreting its function point as the centre (the positive central) of the lattice, constructed the subject-predicate pattern in language. In the theory the functional property of being a subject-predicate group, as opposed to that of being a subordinate clause, is the result of applying a stop-rule: it is the function which occupies the central point in the relation-schema of functions, at which the syntactical relation-building enterprise comes to final rest. This interpretation does not seem, at first sight, to have much to do with the subject-predicate form, for we have not, so far, thought of the subject-predicate relation in these terms. It has been thought of either grammatically, as 'sentence with main verb', or, since Russell, as a formula of the type xP which can be extended either by extending the relational subscript to a sequence x,y,z...n forming monadic, dyadic, tetradic ... polyadic relational predicates respectively, or by adding quantifying restrictions to the meaning of the x-sequence. (Note that P remains unchanged throughout).

Now seen grammatically, the theory gives an explanation not so much of the nature of 'verb-ness' as of the nature of 'main-ness'; that is, of how to build up the logical difference between main verb and subordinate verbs (all verbs being characterised alike at the start as operatives). Seen logically the theory gives not a definition of the relation underlying predicative logic (though the differences between monadic and dyadic relations, and so on, could be built up in terms of the theory by counting and relating the subordinate groups in a sentence), but a definition of the fact that, however far predicative logic is developed, the essential symbol P remains unique and unchanged. P (i.e., ZZ) is the centre point of the lattice: you can 'hang' the whole lattice from it and still get a lattice; analogously, in predicate logic, P is the distinguishing mark of a sentence, no matter how far the x-sequence is extended or restricted or absent: P is the "point" in the predicate-logic system of relations from which the whole system of relations hangs. It is in this sense - which, in our view, though new, is a real sense, - that it is possible to say that, starting with, and defining more exactly, acknowledged linguistic notions, we end with the logically central subject-predicate form.

The derivation of the subject-predicate construction completes the exposition of the theory in its primitive form.

It is evident that, to provide further possibilities of linguistic distinction, the theory can be developed, both theoretically and analytically. Theoretically, a third factor can be introduced, making a 3-factor self-dual product lattice, which would permit the introduction into the theory of Tertiary syntactic functions; and so on, if required, up to N orders of function. Alternatively, it is possible to subdivide exocentric groups according as to whether the governor, or the dependent, or both, or neither, falls within the upper ideal on the positive central.

Analytically, the theory could be further developed by investigating the possibility in interpreting linguistically the lattice-distinction between decomposable and non-decomposable factors, and the properties of the Boolean lattice composed of the centre of any product-lattice.

The next part does not, however, undertake either of these developments; but explains the method being used in the Cambridge Language Research Unit of applying the theory in its primitive form so as to give an actual computer program.

References

1. G. Birkhoff : "Lattice Theory", American Mathematical Society Colloquium Publications, 2nd Edn., 1948.

2. I.A. Richards and C.M. Gibson : "English through Pictures", Pocket Books, Cardinal Edn., New York, 1952.

3. R.C. Yuen : "Mandarin Primer", Harvard University Press, Cambridge, Mass., 1947. (See especially p. 214).

4. D.G. Hays : "Basic Principles and Technical Variations in Sentence Structure Determination", The Rand Corporation, Santa Monica, Calif., 1960.

5. J. Lambek : "The Mathematics of Sentence Structure", American Mathematical Monthly, March 1958.