A SURVEY OF FORMAL GRAMMARS AND ALGORITHMS FOR RECOGNITION AND TRANSFORMATION IN MACHINE TRANSLATION

Bernard Vauquois

A SURVEY OF FORMAL GRAMMARS AND ALGORITHMS FOR RECOGNITION AND TRANSFORMATION IN MACHINE TRANSLATION

(Edimburgh, August 1968, IFIP Congress-68, pp. 254-260)

B. VAUQUOIS

CETA, Grenoble - FRANCE

revised by Ch. Boitet, May 1988.

So much work has been done in the late years in the use of formal grammars in computational linguistics and so many processes have been explored that the author, rather than drawing up a list, has tried to bring out a general outline of this recent work. The grammars and algorithms for syntactical analysis on the one hand, and the transformational grammars on the other hand, have played a leading part in the development of the field. More recently, some attemps at semantic formalisation, extending previous work in the field of syntax, have made their appearance. The author tries to define the limits and the task of computational linguistics by showing where to situate formal grammars. An analysis is made of the various types of models that have been elaborated or that are under investigation. It should be a basic help to the use of the different grammars and their algorithms.

INTRODUCTION

The part plaid by formal grammars and algorithms for handling problems of recognition, transformation as well as generation, in machine translation, and more generally in computational linguistics, has increased considerably in the late years.

The task of making a survey is too difficult if no work in the field is to be overlooked. In this paper, we will therefore just try to bring out the current trends rather than to draw up the list of the published work.

The author expresses his gratitude to all researchers who have kindly helped him elaborate this paper, by forwarding an outline of their work and their results. The bibliography would be far too voluminous to keep within the bounds of this paper, consequently we shall only list at the end of this work the names of the main contributors and their institutes.

I - ABOUT THE PLACE FORMAL GRAMMARS AND ALGORITHMS HOLD WITHIN THE FIELD OF COMPUTATIONAL LINGUISTICS

The bustle that has been stirring so strongly the field of mechanical translation during the last few years might put the author in an awkward position. Fortunately, the actual state of things puts one better at ease.

The most overflowing enthusiasm over machine translation appeared at a time when the methods employed were still very elementary and it broke down at the very moment that much more subtle ways had been found, that set in action more and more elaborate linguistic analysis.

I do not think that full consciousness of the extent of the problem should lead to total discouragement. Anyhow, though machine translation has been supplemented by computational linguistics, which offer a field of research of far wider extent, a lot of work labelled computational linguistics contributes in some way or other to the advance of machine translation.

In fact, between these two widely different attitudes that have followed each other, it seems reasonable to choose a moderate attitude towards machine translation and to stress the various degrees of approximation that can be obtained.

To state precisely, on the one hand, the part taken by machine translation, and, on the other hand, the work on formal grammars and their algorithms, it seems fundamental to outline very accurately the exact scope of computational linguistics.

I-1. Aim and methods of computational linguistics

The heading "computational linguistics" covers such a variety of work that one instinctively looks for some architectural system in which to place the different stones belonging to the structures of the edifice. The pattern given below should by no means claim perfection or completeness, but is nevertheless an attempt to narrow the aim and methods of scientific research in this field.

First of all, computational linguistics should be considered as belonging to several subjects. It is a field where researchers of entirely different backgrounds meet, driven by a common purpose. The main subjects underlying computational linguistics are linguistics, mathematics and computer science.

Like any science, computational linguistics has two aspects :

a) a practical aspect with regard to other disciplines, namely what it gives in exchange for what it receives.
b) a proper aim and a system fitting this proper purpose.

Fig. 1 shows the exchanges between the three subjects mentioned above and computational linguistics.

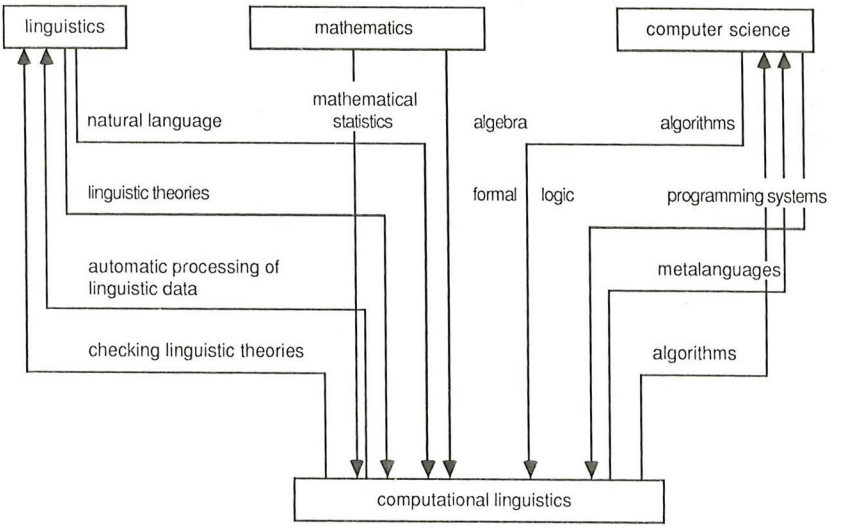

Figure 1

The raw material for computational linguistics, natural language, is supplied by linguistics. Natural language, after a more or less detailed study in the light of linguistic theories, provides more or less elaborate linguistic data. Furthermore, computational linguistics, calling upon the same source, takes advantage of some linguistic theories.

Attention should be drawn to the contribution – that could have been shown in the figure if it had been three-dimensional – of psychology, in particular psycholinguistics, for everything touching the cognitive level and its connection with semantics, and also to semiotics which plays a similar part.

If it is up to linguistics to provide the contents, mathematics provide the expression and orients to a great extent the systems used. Those expressions, presented as models, are mostly obtained from the formalisation methods of logic, of algebra (combinatorial systems, computable functions), of statistical mathematics and or random processes (behaviour models).

Finally, computer science provides the practical tools by means of algorithms elaborated for other purposes (for instance automatic programming) and software which allows easy communication with the computer. On the other hand, computational linguistics, in its practical aspect, may prove considerably useful for the work to be done in those fields. Thus, automatic processing of linguistic data may cover a wide area embodying simple wordcounting or concordances programs, as well as programs for automatic classification or structure detection. Moreover, one should take into consideration that computers may be used by linguists to check a theory, which might have no appropriate use in computational linguistics itself. Here again, it would be the practical aspect. The work carried on properly for the sake of computational linguistics can converge on metalanguages which improve man-machine cooperation.

The use of natural language in a limited form has already started. Anyway, if computational linguistics takes its algorithms somewhere else, it also happens that it defines new ones, which are then available for all computer scientists.

It seems more difficult to tell what good computational linguistics does in turn to mathematics ; may be it suggests new problems ?

The diagram on fig. 1 does not show, because of their being irrelevant, the connections that exist between the different standing subjects at stake, but which serve other purposes.

Once admitted that computational linguistics hold a central place at the junction of these three subjects and also that their development is complete enough to define their own aims, the following conclusions seem evident : an institute working on computational linguistics needs experts of all three sciences involved. Experience has proved that it is much easier to set up a joint team, open to an exchange of views between the various specialists, than to try to discover research workers mastering the three subjects at a time.

The conditions to be met for efficient work of such teams require, from each specialist, the goodwill to consider his subject as an amount of knowledge brought in for the common need, the ability to extract that part which is needed for the purpose of the computational linguistics, submissiveness to the constraints brought in by the two other sciences and enough understanding of these constraints to be able to do productive teamwork.

Consequently, a linguist who is involved in computational linguistics has an entirely different attitude from the attitude of a linguist dealing exclusively with linguistic theories, who may or may not appeal to computational linguistics to accelerate his work or check its qualities. The same goes of course for a mathematician and a computer scientist belonging to such a team.

As a matter of fact, there are two ways of considering the part played by the computer. Either the machine is only considered as a subsidiary tool asked to perform ancillary work, or the use of such a machine leads to new prospects to pursue as an aim and calls for an appropriate methodology.

It is doubtless this second point of view which has brought forward computational linguistics. Like any subject, it is the task of computational linguistics to elaborate models. These models should be characterized not only by their adequacy, which is a general requirement, but mainly by their computability and, what is more, by their suitability for efficient computing. This means that the model should be theoretically computable and moreover directed by sufficiently performing algorithms to ensure their practical computability. The applications of such models are partly found in some of the branches which go back from computational linguistics to the standing learnings that originate them ; for instance, man-machine communication in restricted natural language to computer science. Besides, computational linguistics generates its own applications in machine translation, automatic information retrieval, automatic indexing, some aspects of content analysis, learning, etc. All those applications are based on two types of models and we would like to go into the formal grammars which underly them, namely :

a) the text-meaning models ;
b) the meaning-text models.

This gives us the following diagram :

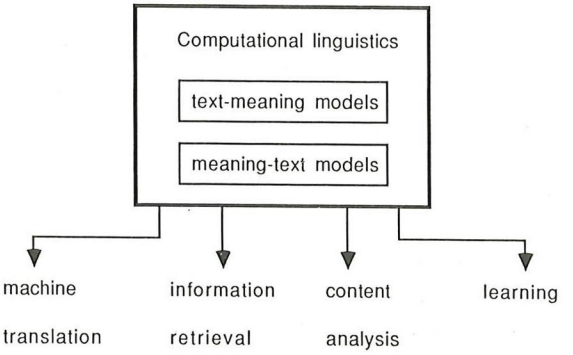

Figure 2

I-2. The constituents of a model

Most models are based on two levels : an input level and an output level. Thus, in a text-meaning model, the input level would be the representation of the text (for instance : a coded string of characters) and the output level would be the representation of the meaning (for instance : a formulation in semantic notation). Needless to say that for the time being there exists no such thing as a text-meaning or a meaning-text model. Up till now all energy has been devoted to the partitioning of the distance to cover by creating intermediate levels and elaborating models between these levels.

If we have a model M, such that the relation from the input level to the output level corresponds to a fragment of the meaning-text model, the model M is said to be functioning in generation, otherwise it is said to be functioning in recognition. A model which can be made to function both in generation and recognition is said to be reversible. Developing a model involves the elaboration of the following components :

a) determination of the contents of the input and output level ;
b) formalized description of both these levels ;
c) choice of the logical type of the model ;
d) creation of a metalanguage to be used for the notation of the grammar ;
e) determination of an algorithm which sets the grammar in action ;
f) writing of the grammar itself and indexing of a sample of the vocabulary ;
g) development of programming tools, including those to be used for setting the model in action, as well as those intended to help the linguist during the elaboration stage ;
h) perfecting of the model, which means testing its adequacy and modifying it accordingly ;
i) extension of the vocabulary when the grammar is supposed to have reached stability. Checking of the adequacy and the performance.

With regard to the one-level models, we can split them up into two families :

1) those having in view the transfer of the representation of a language into another without change of level ; for instance, the transfer of syntactical surface structures from Russian to German. Those models may be considered as particular cases of the preceding ones.

2) those dealing with the description of a level without having any transfer in view. In that case, either we have to do with a mere formalized description which is not a model, or this formalism has some means at its disposal to calculate equivalent expressions or/and to calculate distance or/and valuation processes, and in that case it really is a model. We will see an example of such a model later on.

I-3. Application to machine translation

Machine translation forms the most direct application of computational linguistics proceeding from the diagram on fig. 2. Indeed, if mechanical translation is no longer considered as a first aim, it still is a most valuable object for experiments, and will remain the really objective means to test the validity and the scope of many models.

If we consider that the ideal system of machine translation consists in passing through a text-meaning model, starting from the source language to extract a semantic formulation which in turn will be used as input to a meaning-text model for the target language, then one realizes that machine translation is the perfect checking tool for such models. The level of expressing meaning being still out of range, it is quite interesting to be able to situate, by means of machine translation experiments, the levels on the way from text to meaning which have been reached by the various models.

Fig. 3 shows where to situate the different types of models mentioned above.

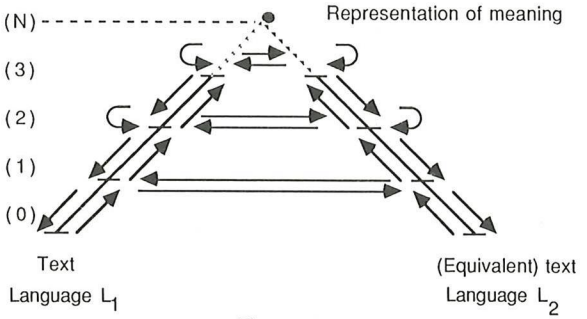

Figure 3

Levels (1), (2) and (3) represent stages on the text-meaning way. The arrows going up and those going down stand for two-level models functioning respectively in recognition and in generation. The horizontal arrows stand for the models operating a constant level transfer from one language to another. Finally, the circular arrows stand for the properly so-called one level models. The lower the level, and the greater the distance which separates the languages L_1 and L_2, the more complex the constant level transfer models are. The complexity of such a transfer model proves to be most indicative of the position of the level which has been reached, and consequently of the distance still to be covered in order to reach the integral semantic level.

The failure to construct a satisfactory transfer model at a certain level is particularly indicative of the need to carry on researches towards the "fulcrum", especially as the obstacles that stand in the way of the transfer model ought to guide this research.

II - GRAMMARS AND ALGORITHMS OF THE MODELS

The grammars and algorithms attached to a model are strongly connected with the formalized representations of the input and ouput levels. Up till now, these representations have mostly the shape of strings and tree structures, the use of networks is only very seldom to be seen. Grammars and algorithms are also conditioned by the way the model functions.

II-1. String-string models

In this category, the formalized representations of the input and output are strings. The most usual examples of such models are those of morphological analysis and synthesis, the simplest logical type being in that case the finite state model. The elaboration of a finite state grammar as well as the simulation of the corresponding automaton on a computer are particularly convenient and provide most efficient programs. Of course, this applications imply that the morphological structure of the natural language in question should be of the finite state type, in order to ensure the adequacy of the model.

The functioning of such a model in recognition is as follows.

a) the input level is made up of a string of constituents (word fragments) that have been found in a dictionary after the segmentation of the whole word occurrence. To be more exact, the dictionary substitutes, for the sequence of characters which constitutes this word fragment, the codes of an equivalence class in which all constituents having the same morphological pattern have been arranged. The whole set of these classes thus makes out the input terminal vocabulary for the grammar.

b) the output level might for instance be made up of a string of symbols which identifies the lexical item to which the occurrence belongs, and also the morphological derivations, the syntactical category, the values of the grammatical variables, etc. If the segmentation at the input was not a legal one, the model should reject it and should produce an empty string or a reject symbol instead.

II-2. String-tree structure model

These models are only used in systems for syntactical analysis, but numerous research teams have devoted their energy to them for years. It is worthwhile dwelling on these models (all functioning in recognition) because they show clearly what different components are needed for their elaboration. The determination of the contents of the input and output levels is not strictly the same in all these models, but whatever the choice may be, one always comes upon the major problem of syntactical analysis.

The input level, for instance, is, as the case may be, either a string of lexemes displaying the word segmentation, or a string of syntagmas comprising the words already recomposed.

At the output level, the differences appear either in the nature of the formalized description (dependency tree structures or tree structures of immediate constituents also called surface structures), or in the profusion of syntactical classes and in the number and combination of the constraints granting to this level a more or less refined power for selecting the "syntactically correct" sentences. Anyhow, it is the choice of the logical type of model that matters, and that is where the real difficulties arise. The choice of a context-free grammar or its equivalent in a dependency grammar comes up against the adequacy problem. The recognition power of such grammars is not strong enough to deal, for instance, with "discontinuous constituents" (or, using another terminology, with non-projectivity). It is noteworthy that solutions based upon exactly the same principle have shown up at different places in a totally independent way.

The reasoning is as follows : the context-free type provides an inadequate grammar even for determining the surface structure, hence there are only two attitudes left :

a) to use a logical type of higher rank (context-sensitive grammars) ;
b) to try to find, proceeding from context-free grammars, a formalism which allows the recognition of those particular cases.

The first way of solving the problem has been given up, because the recognition algorithms of context-sensitive languages are not efficient enough. At the present time, the question of practical systems for context-sensitive language is under investigation.

Consequently, preference has often been given to the second solution. This solution consists in assigning at first an incorrect structure to the sentence that should be analyzed, so as to resolve the decision problem (acceptance or rejection of the sentence) with the help of a context-free grammar algorithm. Owing to the systematic transformation of the achieved structure, the correct solution is reached subsequently. The difficulty of choosing a logical type having thus been overcome, the most trying obstacles have been of practical nature.

First of all, one should have clearly in mind that a formalism doing so well for the proof of a theorem is not necessarily suitable to the actual writing of a grammar dealing with the entirety of the phenomena to be recognized in the model. Consequently, it is essential to create a metalanguage which allows the writing of a complete grammar.

On the other hand, the efficiency constraints entail a most detailed study of the algorithms, in order to avoid as far as possible the combinatorial effects of computing structures from strings, if we have to do with ambiguous languages.

And last, but not least, the elaboration of a grammar satisfying over and over the adequacy tests needs endless dexterity ; up till now, systematic methods to this end seem to be non-existent. Syntactic analysis models satisfying all these constraints are therefore exceedingly scarce.

II-3. Tree structure - tree structure models

These are the models in fashion today. All types of models are used (two level models, transfer models and one level models). The problem they have in common is the application of a set of trees (often bi-ordered) into another set of trees. This application may be performed by means of composition of elementary transformations.

Thoroughly studied from the linguistical point of view, the transformational grammars, showing merely an aspect of the underlying mathematical problem, are still lacking for a satisfactory formalized theory. In a general way, the transformation problem can be stated as follows. Let us assume we operate on a set of trees (usually defined by a partial order relation Γ_1) for which we are given :

a) in addition to Γ_1, an order relation ("linear precedence") between the nodes $y_1,...,y_n$, which are the direct descendants of the same node x, this being for any node x in the graph.

b) an application which assigns to any such node a name in a given vocabulary. So, we have bi-ordered trees with nodes having a name. The transformation consists in producing other trees with nodes that may have other names. The transformation is performed by means of a sequence of tree re-writings, in a way similar to the derivation of a word in a combinatorial system by means of semi-Thue productions. As in the case of string-rewriting grammars, this leads to the construction of rules divided into two parts :

1) the left-hand part, which is used to recognize in the graph a set of nodes satisfying simultaneously structure relations and name restrictions. From that point of view, we have to find a formalism which is able to describe any subset of nodes in a bi-oriented tree structure.

2) the right-hand part, which performs the actual rewriting by means of a series of elementary transformations.

This kind of grammar should be thoroughly investigated from the theoretical point of view, because it is already in use in several types of models, as indicated below.

1. Two-level models : transformational grammars used to transform the deep structure of a sentence into its surface structure, as well as grammars allowing to go from a syntactical level to higher levels on the "text-meaning" way and vice versa.

2. Transfer models : going for instance from a surface structure of a sentence belonging to a language L_1 to the surface structure of an equivalent sentence of a language L_2.

3. One level models : generation of equivalent structures in a lexico-syntactical structure system, used in connection with a paraphrase generator.

There is no lack of entirely different examples : the way is open to develop considerably the researches in the field of these tree structure models.

II-4. Network models

The only example the author knows about is still at the stage of a first approach. It is a reversible model, between the morphemic and semantic levels, and has the outstanding quality of being, as to the size of the set of sentences it can handle, more powerful in recognition than in generation. The type of grammar used in such a model is worth a theoretical study, which is lacking up till now.

The other example is a formalized representation of a semantic level, for wich the calculating rules have not yet been laid down. To find what properties are interesting, the theoretical elaboration of such models will, in all likelihood, be carried out in the near future.

III - CONCLUSION

Actually, models are commonly used in most institutes working in computational linguistics or on machine translation. The problem of syntactic analysis, in spite of the numerous problems it leaves unsolved, has been so much developed that satisfactory though doubtlessly temporary solutions have been reached. In this field, the remaining difficulties, which are considerable, subsist mainly in the practical area and less in the theoretical area.

The study of levels higher than the syntactic level has already been taken up. It involves new types of models, hence other formal grammars which offer a wide field for researchers to investigate. It would be a good thing if research regarding the content (linguistic studies), and also the expression (logic studies) could be carried out simultaneously, and in close connection, in order to contribute to the fundamental purpose of computational linguistics.

The institutes having partially or entirely succeeded in elaborating such models are enumerated below. This list will give a clear idea of the actual importance of computational linguistics. The author apologizes for the omission of any institute due to an oversight or to the fact he was unaware of its existence, and hopes that this forgetfulness will be mentioned to him.

LIST OF INSTITUTES

AUSTIN (Texas, USA)
University of Texas, W. Lehman, W. Tosh.
TRACOR, E. Pendergraft, T. Ziehe.

BERLIN (German Democratic Republic)
Deutsche Akademie der Wissenschaften zu Berlin, Arbeitsstelle für Mathematische und Angewandte Linguistik und Automatische Uebersetzung, E. Agricola, J. Kunze.
Ostberliner Arbeitsstelle für Strukturelle Grammatik, M. Bierwirsch.

BONN (German Federal Republic)
Institut für Phonetik und Kommunikationsforschung der Universität Bonn, H. Schnelle.
Forschungsgruppe LIMAS. Linguistik und Maschinelle Sprachübersetzung, A. Hoppe.

BUCAREST (Roumania)
Institut de Matematica. S. Marcus.

BUDAPEST (Hungary)
Magyar Tudományos Akadémia Számitestechnikai Központje Gapi Nyelvesseti Csoport.
D. Varga, F. Kiefer.
Nyelvtudományi Inteszéte, Gy. Szépe

CAMBRIDGE (UK)
Cambridge Language Research Unit, University of Cambridge. M. Masterman.

CAMBRIDGE (Mas., USA)
The Computation Laboratory, Harvard University, A. Oettinger, S. Kuno.
Massachusetts Institute of Technology, N. Chomsky, G. Matthews.
IBM Federal Systems Division, Boston Programming Center, R. Tabory, P.S. Peters Jr.

CANOGA PARK (California, USA)
Bunker-Remo. P. Garvin.

CHICAGO (USA)
Graduate Library School, University of Chicago, V. Yngve.

EREVAN (USSR)
Computation Laboratory, Academy of Sciences. М. Тер-Микаэльян, В. Григорьян.

FUKUOKA (Japan)
Kyushu University, T. Tamati.

GRENOBLE (France)
Centre d'Etudes pour la Traduction Automatique.
CNES, B. Vauquois, G. Veillon, N. Nédobejkine.

ITHACA (N.Y., USA)
Cornell University, Department of Computer Science, G. Salton.

JERUSALEM (Israel)
The Hebrew University, Y. Bar-Hillel.

KIEV (USSR)
Institute of Cybernetics, Е.Ф. Скороходько.

KYOTO (Japan)
Kyoto University, Department of Electrical Engineering, M. Nagao, T. Sakai.

LENINGRAD (USSR)
University of Leningrad, Н. Андреев, Г. Цейтин, С. Фитиалов.

MARSEILLE (France)
CNRS, J.C. Gardin.

MONTREAL (Canada)
CETADOL, University of Montréal, G. Rondeau.

MOSCOW (USSR)
ВИНИТИ, Academy of Sciences USSR, Сектор Математической Лингвистики, Ю. Шрейдер, В. Борщёв.
Institute of foreign languages, В. Розенцвейг, Й. Мартемьянов, А. Жолковский.
Institute of linguistics, И. Мельчук
Institute of Mathematics, О. Кулагина.
Institute of Slavic Languages, И. Ревзин, А. Зализняк, И. Иванов.

NEW-HAVEN (Connecticut, USA)
Linguistic Automation. Yale University, S. Lamb.

NOVOSIBIRSK (USSR)
Academy of Sciences USSR, Institute of Mathematics, Siberian Section, Ляпунов, А. Гладкий, М. Рыбакова.

PHILADELPHIA (Pennsylvania, USA)
University of Pennsylvania. Z. Harris, A. Joshi, D. Hiz.

PARIS (France)
Université de Paris. ISUP, J.P. Benzécri.
Institut de Phonétique, A. Cullioli.
CNRS, Institut Blaise Pascal, M.P. Schutzenberger, M. Gross.

PRAGUE (Czechoslovakia)
Charles University, P. Sgall, P. Pit'ha.
Československa Akademie ved Matematicki Ustav, K. Čulik.

SANTA-MONICA (California, USA)
Rand Corporation, D. Hays, M. Kay.

SOFIA (Bulgaria)
Academy of Sciences, Institute of Mathematics, А. Лудсканов.

STOCKHOLM (Sweden)
Research Group for Quantitative Linguistics,
H. Karlgren, B. Brodda.

TEDDINGTON (UK)
National Physical Laboratory, Autonomic Division,
J. Mc Daniel, D. Yates.

TOKYO (Japan)
First Research Center, Defence Agency, I. Sakai.
Tokyo Electrical Engineering College, M. Nakano.

WARSAW (Poland)
Center of Applied Linguistics, University of Warsaw, I. Bellert.

YORKTOWN-HEIGHTS (N.Y., USA)
IBM Watson Research Center,
W. Plath, S. Petrick, J. Robinson, P. Rosenbaum.

-O-O-O-O-O-O-O-O-O-O-O-