

CURRENT STATE AND FUTURE OUTLOOK OF THE RESEARCH AT GETA

Ch. Boitet, GETA
Groupe d'Etudes pour la Traduction Automatique
USTMG et CNRS, BP 68, F-38402 Saint-Martin-d'Hères

Introduction

GETA (Study Group for Machine Translation) has been founded in 1971 by Prof. B.Vauquois. It is a research group of USTMG (Scientific, Technological and Medical University of Grenoble), associated with the CNRS (Centre for National Research). Previously, from 1961 to 1971, Prof. Vauquois led CETA (Study Centre for MT), a laboratory depending only on the CNRS, towards the design and experimentation of the first complete second generation system.

The tradition of the laboratory is to pursue at the same time fundamental and applied research. From the theoretical as well as the practical point of view, for example, the Russian-French system obtained in 1970 was of quite remarkable coverage and quality. It embodied the (hybrid) "pivot" approach (translation through an interlingua). The system was developed using a corpus of more than one million running words, and was evaluated on more than 400,000 running words. The current Russian-French system (Boitet & Nedobekine 1981, 1983) embodies all features of the current approach at GETA, and has been tested as an operational prototype on about the same quantity of texts.

Main features of the current approach at GETA

Using the past experience of CETA, GETA began a new cycle of research and development, founded on new theoretical bases:

- the "transfer" strategy replaced the "pivot" (interlingua) approach ;
- heuristic (and procedural) techniques replaced purely combinatorial techniques (with "filters") for linguistic programming ;
- "fail-soft" techniques were introduced, due partly to the invention (by B. Vauquois) of "multilevel" structural descriptors for the units of translation, and partly to the replacement of analysers by transducers as basic algorithmic models of linguistic computation ;
- units of translations were extended to several sentences or paragraphs, in order to permit the use of a wider context for the treatment of phenomena such as anaphora, tense/mode/aspect agreement, etc.

The ideal (idealistic) goal of perfect machine translation (good enough not to need any revision) was abandoned for the goal of a "human aided machine translation" (HAMT), where the "raw" translation produced by the computer must be good enough to obtain a very high quality translation with a revision effort equal - or only slightly superior - to the one demanded by an average human translation.

Before going on to present the current state and future outlook of the research at GETA, it is useful to say a few words about the different kinds of techniques used in the HAMT approach.

Second generation versus first, third and fourth generations

However, the fundamental strategy of the (machine) translation remained that of the "second generation", as in the former CETA systems. Contrary to "first generation" techniques,

- analysis, transfer and generation steps are clearly separated, analysis and generation being strictly monolingual;
- linguists use Specialized Languages for Linguistic Programming (SLLPs) and not in usual algorithmic languages like PL/I, Fortran or even macro-assembler (Systran);
- the decisions concerning ambiguities are taken as late as possible, and can be reversed if necessary (by local backtracking or correction);
- in analysis, the grammatical work is very important, and partly replaces the effort of developing enormous dictionaries of individual cooccurrences, characteristic of first generation systems, where the basic idea is to translate word for word or string by string, with some local rearrangements ;
- in generation, the organization of the lexicon in derivational families called "lexical units" (*to control, the control, controller, controllable, controllability*) allows the generation step to produce paraphrases, by recomputing the lower levels of linguistic interpretation (syntactic functions, syntagmatic and syntactic classes) from the upper levels (semantic and logical relations).

The process of translation is of the "implicit understanding" type, that is, it uses only linguistic knowledge (kernel grammar and core dictionary) and paralinguistic knowledge (grammar specific of a typology and dictionary specific of a domain), but no extralinguistic knowledge in the form of an explicit representation of the domain of interpretation (the subject of the texts to be translated), and, even more so, no "pragmatic" knowledge. By definition, a "third generation" MT system should contain an explicit "expertise" of the domain in question. This principle has been strongly advocated, in particular by the Yale school (Shank & al.), but no translating system of this system has yet been produced. A "fourth generation" MT system should embody dynamic representations of the situations described in the text and perceived by its actors.

Technological transfer and current research

Since 1977, GETA has recognized the necessity to transfer its techniques to industry. From 1982 until now, it has participated in the ESOPE project of ADI, and then in the French National MT Project (CALLIOPE). But the development effort has not been allowed to swallow all resources of the laboratory, a common danger in such situations, and several themes of research have been and are pursued independently. The following division will be used for this presentation:

amelioration of second generation MT systems:

design of a linguistic workstation:

study of new types of MT systems.

1. Amelioration of second generation MT systems

Second generation MT systems can be improved directly in many ways (we will see later how to improve them indirectly), at the three levels of lingware engineering, software tools and linguistic content.

1.1 Lingware engineering

First, current techniques for writing analysers, transfers and generators are further investigated, in the framework of the available programming tools. For example, a better methodology to write generators in ROBRA is currently being developed. As a result, new generators should be more robust, and, which is very important in the framework of multilingual MT systems, should allow for a new type of *composition of translations*.

Reorganizing the generation step to improve the transfer approach

Let us explain this in more detail. The input to a generator is a *target interface structure* which is not in general the same as the *source interface structure* produced by an analyser. This is because the final form of the text is not yet fixed (paraphrases are possible), because polysemies not reduced by the transfer may appear as a special type of enumeration, and because the transfer may transmit to the generator some advices or orders (relative to the possible paraphrases), by encoding them in the structure.

Instead of producing directly the surface tree (to be passed to the morphological generator), the new technique consists in *producing a source interface structure for the target language* as an intermediate result. The idea is that, in the absence of a satisfactory pivot language, equipped with an unambiguous grammar (an illusion) and a complete vocabulary (an impossibility), the number of transfer steps in a multilingual transfer-based system can still be reduced.

For instance, consider the 9 languages of the European Community. They may be divided in three groups: 4 romance languages (French, Italian, Spanish, Portuguese), 4 germanic languages (English, German, Danish, Dutch), and Greek. Instead of constructing 72 transfers, it might be enough, for the beginning, to construct only 14 transfers, 6 between the groups, for example French <-> English, Greek <-> Italian, German -> Greek and Greek -> English, and 4 in each group, for example Portuguese -> Spanish -> French -> Italian -> Portuguese and Danish -> German -> English -> Dutch -> Danish (or any pragmatically better arrangement). To translate from Spanish to Dutch, one would then use the Spanish -> French -> English -> Dutch route.

A new generator of French is now currently being built. It should replace the versions used in Russian -> French, German -> French and English -> French, which have started from the same core, but evolved differently, because of the differing aims of the developers of those three MT systems (preoperational translation unit and linguistic studies at GETA, CALLIOPE project).

Treating new languages or language pairs

MT is about the actual translation of languages, and not only abstract speculation. It must be demonstrated that the linguistic techniques used are as universal as we claim them to be. Although it is quite impossible to construct significant systems for all languages, it is useful to test the methods on as many types of languages as possible.

In recent years, various cooperations have made it possible to develop prototypes for English -> Malay (Tong 1986), English -> Thai, English -> Chinese (Yang 1981), and a mockup from Chinese into several languages (Feng 1981).

In 1986, the effort on Chinese has started again, with Prof. X. Tcheou joining GETA, and the stay of Chinese colleagues. The immediate aim is to construct a large size French -> Chinese prototype.

During the last academic year, a student project, under the guidance of N. Nedobekine, has been devoted to the inversion of the Russian -> French to French -> Russian, the analysis of French being, of course, the same as that of the French -> English developed by the CALLIOPE project.

Also, the stay of M. Satoh from JICST has given the opportunity to start a study on the *junction between two different MT systems*. The idea is to construct a Japanese -> French MT system by starting from the source interface structures produced by the MU-project (some examples have kindly been made available for this purpose), interfacing them with the ARIANE system (GETA), constructing an adequate lexical and syntactic transfer, and then using an available generator of French.

1.2 Software tools

The ARIANE system is a complete and integrated software environment for preparing, testing and using MT systems (Boitet, Guillaume, Quézel-Ambrunaz 1982). It also includes a subenvironment for human revision (or translation, in case no MT system is available or adequate). Release 78.4 has been chosen at the beginning of the CALLIOPE project to support the industrial development. Release 85.2 (Boitet, Guillaume, Quézel-Ambrunaz 1985) will replace it after the formal end of the project. This system runs only on the IBM-370 architecture, under VM/CMS.

The compilers and interpreters of the SLLPs (ATEF for morphological analysis, TRANSF for lexical transfer, SYGMOR for morphological generation, and ROBRA for structural analysis, transfer and generation) are written in low level programming languages (macro-assembler, PL360), with a small part in PL/I. The monitor is written in EXEC/XEDIT, and some annex tools in Pascal (Bachut & Verastegui 1984).

The ARIANE system on a PC

Until recently, the system ran only on mainframes and minis. But the compactness and relative speed of the code have made it possible to adapt the complete ARIANE system to the IBM PC/AT-370 (under VMPC). Now, exactly the same programs run on the micro and on the mainframe. The minimal configuration uses a fixed 20Mb disk, the 370 kit (2 cards), and a 3278/79 emulation card. A memory extension of at least 1.5 Mb may be added and used for paging, which speeds it up considerably.

Of course, the speed of the 370 card (0.1 Mips) does not yet make it possible to consider such a configuration for the *production* of translations. But it is quite adequate to *build MT prototypes*, and could be used as low cost hardware for academic cooperation.

In order to use the PC for producing translations, IBM should modify VMPC in such a way that it would use *directly* the memory extension as real memory (as LOTUS does with the EEMS extension), instead of using only the basic 512 K as real memory. Available memory cards might then be used to get a real memory of 4 Mb (VMPCs limit for the virtual memory). Also, it is likely that faster 370 cards will be produced, and perhaps adapted to the faster hardware of the new PS series.

ARIANE -X: multilinguality and openness

In the framework of the National Project, it was decided to push ahead a project which had been prepared at GETA over recent years, but advancing slowly, due to the lack of resources. The idea is to construct a new basic software, in *Le_lisp* (a French dialect of LISP produced by INRIA and converging towards Common Lisp), with the following characteristics:

- there will be only one LSPL (TETHYS), integrating several rule engines;
- the system will be basically multilingual, down to the level of the characters;
- access to the implementation language will be possible from within the rules.

A very large set of characters, called CARIX, has been defined. Several properties are attached to a character. Among them, the natural language (neutral or math, French, English, Japanese,...), the name of its (usual) character set (keywords, special, greek, roman, Cyrillic, thai, hiragana, kanji, hanze,...), its case and diacritics, if any, and some information about its "stress" (italic, bold, underlined).

These properties have been selected because of their importance for linguistic processing. They are all important for defining the dictionary order of words. For example, the information on the natural language permits to consider "ch" as *one* letter in Spanish, and to display a keyword of TETHYS (considered as one character) in the appropriate language.

Two preliminary implementations have been prepared on a SM-90 under SMX (a version of UNIXTM), one in *Le_lisp*, using property lists, and one in Pascal, using a 32 bit representation, and imbedded in a version of Prolog-CRISS (CRISS is another research group of the University of Grenoble).

Also, a norm for representing multilingual documents has been defined. This is important, in order to be able to use information about the logical structure of a text, without bothering about its physical appearance. For example, a table with textual elements in the cells is not represented as a sequence of lines, with tabs between fragments of elements, but as a construct of matrix type, where each textual element is presented contiguously.

Of course, the set of rule engines contains engines which are upward compatible with those of ARIANE. But it is planned to integrate other engines, adapted from other previous or current systems (Q-systems, ATNs,...).

A simple, but powerful *language for writing transcriptors*, called LT, has been implemented (the compiler in Prolog-CRISS, the interpreter in Pascal/VS) by Y. Lepage (1986). A given transcriptor is expressed by a finite-state like deterministic automaton, with rules on the edges. There are two reading heads and one writing head. The states may be indexed by a set of attributes declared by the user (case, type of font,...). A more powerful model has been designed for use within TETHYS.

For example, our Russian texts are usually keyed in manually, or read by OCR. In the first case, we use a "minimal" transcription of the Cyrillic script, which is based only on uppercase latin letters, digits, and usual special signs, and is also used in ARIANE's grammars and dictionaries. In the second case, an "intermediate" transcription involving also lowercase letters is used. There are transcriptors to go from the second to the first, and from the first to that of a printer equipped with a Cyrillic font, so that texts may be printed in their original format.

A *tree transformational editor*, called TTEDIT, has been defined and implemented (in REXX/XEDIT) by J.C. Durand. Elementary commands are simplified versions of ROBRA's rules (for transformations) or tree patterns (for localisations). Commands may be grouped in packages analogous to usual editor macros.

Several uses are envisaged for TTEDIT. First, in the context of the construction of a MT system, it can be used for exploring and modifying intermediate structures produced by an existing analysis or transfer step, in order to prepare testbeds for subsequent steps. Second, it is a good tool to introduce students to more complex tree transformational systems. Last, but certainly not least, it could be extended to a tool for the *structural investigation of texts*, in the context of textual data bases. The source interface structures of the texts would be stored along with their source form. Questions too complex to be answered by string-matching mechanisms (working at the level of word forms or even at the higher levels of lemmas and grammatical categories) would be transformed into tree patterns and answered using the in-built tree-matching mechanism.

A *generator of analyzers*, called GA, has been designed and implemented (again in Prolog-CRISS) by Y. Lepage, and is used for prototyping. For example, the compiler of LT, of the MID items (see below), and of course of GA itself, are written in GA. Non LL or LR grammars may be handled, and there are several built-in facilities to construct an abstract tree and to handle error cases.

1.3 Linguistic content

Incorporating new ideas from modern linguistics

From the point of view of translation, not all studies in linguistics are relevant, because they often fail to give computable criteria to extract the useful information from a running text, or because they address rare cases, perhaps very interesting, but not as bothering as frequent "low level" ones. For example, no universal system for the tense/mode/aspect triad is yet really usable (computable) in second generation MT systems, and difficult problems of anaphoric reference are quite rare in texts which are candidates to MT, such as maintenance manuals.

On the other hand, some studies on formal criteria for computing the *determination* and the *thematization* in a text look quite promising and might considerably improve the average quality of raw MT. For example, in our Russian -> French system, we constantly encounter the problem of generating the correct article in French (definite, indefinite, or none). The current rate of success is only around 60%. This is extremely bothering, given the frequency of this problem. Hence, the ongoing study of determination in Russian is very important. In the German -> French system (Guilbaud 1984), it is necessary to correctly determinate the scope of negation, especially to avoid producing incorrect translations when paraphrasing in generation. The impressive work done by Prof. J.M. Zemb (Collège de France) is being studied intensively, to try to incorporate the *theme/rheme/pheme* distinction in the machine analysis of German. Also, contrastive rules for the transfer into French can be deduced from the contrastive grammar of the same author (Zemb 1978, 1984).

Better defining the source interface structures

Modern linguistics offers very interesting new views of the kinds of structures adequate to represent texts, together with justifications of formal, linguistic and philosophical character. This has prompted researchers at GETA to

compare the *multilevel interface structures*, introduced by B.Vauquois in 1974, with the proposals of GPSG and UFG, and to reconstruct the definition of these multilevel structures with the same kind of arguments (Zaharin 1987).

This line of work concerns not only the content of the linguistic description, but also the formal tools used to describe it. This is why an important effort is being made to give precise mathematical semantics to the formalism of SCSG (Static Correspondence Specification Grammar) used to specify and document big structural analysers or syntactic generators written in ROBRA (Zaharin 1986).

2. Design of a linguistic workstation

A linguistic workstation should be usable by linguists, lexicographers and translators. It would certainly help greatly, though indirectly, to build better MT systems.

2.1 Grammatical specification (SCSG)

Recently, a first environment for building SCSGs has been designed and implemented on a Macintosh + by Y.F. Yan (1987), under the direction of F. Peccoud. It incorporates a methodology for writing the different components of a SCSG (attributes, axioms, boards), while handling at the same time the corpus investigated and the examples from the corpus appearing in the boards.

There is still a lot of work to be done in this direction, before a complete environment can be offered to linguists. In particular, it would be useful to relate directly a dynamic grammar being executed on a text to the boards containing the specification of the partial correspondences computed by the executed dynamic modules or individual rules. However, the first version has already been tested by some users and found to be useful.

2.2 Lexical databases (FID / MID)

Ultimately, the cost of MT systems lies essentially in their dictionaries, which are quite difficult to construct and to maintain. Since 1982, we are working on *Fork Integrated Dictionaries* (Boitet & Nedobejkine 1986), now called *Multitarget Integrated Dictionaries*. They integrate the terminological and the grammatical aspects, as well as the "usual" and the "coded" information used in computer applications. A given dictionary contains terms in one language, with the information concerning that language, and the translations of the different meanings in one or several languages.

The structure of an item is a tree described by a grammar written in GA. From this grammar, an analyser has been automatically generated, as well as a DDL (data definition language) program defining the mapping into a commercial DBMS (now CLIO from SYSECA, but others could be added). From a given item, a DML (data manipulation language) program is automatically generated to load its image in the DBMS. There is an ongoing project to build a complete user interface, written in Prolog, and easy to interface with any reasonably powerful DBMS.

In order to get some practical experience, 3.000 terms in telecommunications have been written in the MID format in 1986, in three languages (French, Japanese, English), in cooperation with the French Telecommunications (DGT) and with KDD. The corresponding 9.000 entries have been keyed in on a Macintosh+ using Kanji-Talk™

The work to be done remains immense, in practice as well as in theory, and certainly calls for productive cooperation. One of the most difficult problems is the reuse of existing lexical information encoded in machine dictionaries developed for some applications. We have begun to study how to extract the lexical information from our Russian -> French system and put it in MID format. The preliminary results obtained show this to be far more difficult than we imagined at the beginning, because some information is implicit or absent from the codes, and has to be added in their comments in order to be accessible by a program.

2.3 Study of corpuses

This line of research has been hardly touched by GETA until now. Some elementary tools (concordances) have been constructed, and some methods used elsewhere have been experimented, in particular at the Bureau of Translation (Ottawa). However, we are trying to get researchers to explore this further, building and using such tools as TTEDIT. The study of *parallel corpuses* of texts and their translations into one or several languages should lead to interesting results, but they should be based (at least) on structured representations of the texts.

If I may be allowed a short digression at this point, I feel that computational linguistics, of which MT is a part, can only attain the status of an experimental science if *experiments* are really made, and their results used to prove or disprove scientific hypotheses, or to discover new phenomena, calling for new hypotheses, etc., very much like in physics. But things done nowadays (like MT systems or natural language interfaces) don't give any new insight into scientific questions, even if they can be shown to be usable, cost-effective and of "reasonable" quality.

For example, studies of existing parallel corpuses could lead to the *experimental* discovery of the distinction between what is really accepted as a translation by translators, at the level of structure and choice of word or word classes, and what is considered as paraphrase or reformulation.

2.4 Tools for the translator

The linguistic tools available should be put to use in order to offer really powerful linguistic aids for text processors. For example, the replacement of a verb by another should lead to the automatic regeneration of the parts concerned (valencies, word order, etc.).

In the same vein, we are investigating the nature of correspondences between trees (interface structures), in order to manipulate a text and its translation in a parallel way, at levels finer than paragraphs or sentences.

3. Study of new types of MT systems

3.1 Introduction of domain specific expertise

The linguistic and paralinguistic knowledge incorporated in modern second generation MT systems is quite enormous. Unless the knowledge of a given domain is extremely limited, it is not really feasible to put this information directly into such systems. We have proposed (Gerber & Boitet 1985) to *graft corrector expert systems* onto existing MT systems. The corrector system would detect "problem patterns" in the source interface structure, convert them into questions on the domain (represented as a knowledge base, an expert system, or as a simple data base), and modify the structure according to the answer.

A small prototype has been built by R. Gerber (1984) connecting a small English -> French system to such a corrector system, written in Prolog-CRISS. The problem here is that developers of MT systems have no time and no competence to build knowledge bases for any technical domain. But significant improvements in translations cannot be obtained if the knowledge base is too small. To continue this line of research, we are waiting for a situation where both an adequate knowledge base and texts to be translated are available.

A more fundamental research is pursued by M. Dymetman (1986), who is trying to represent commonsense knowledge in a consistent way. If one heaps axioms on axioms, the resulting model is almost guaranteed to be inconsistent. Dymetman's idea is to develop a formalism in which new knowledge can be incorporated by *definition*, that is by a sort of syntactic extension, without adding new axioms, as in usual mathematics.

3.2 Linguistic editor interacting with the author

Even if a big knowledge base is available, no machine analysis of a text can be 100% correct, because *new* knowledge is usually introduced by the translated text. But no adequate learning method is yet able to dynamically modify and enrich the knowledge base. Even if one would exist, the communicative character of the texts, their *pragmatic* aspect, would not be satisfactorily handled.

This is why we propose, as an alternative or complement to the above method, to return to the interactive approach. The essential difference with previous schemes like that of ITS (Provo, BYU) is to consider an *interaction with the author of a document*, and not with a specialist of both the domain and the MT system, without imposing a too restricted controlled language, as in the TITUS system (Ducrot 1982).

In his thesis, R. Zajac (1986) has proposed an organisation of the dialogues with the author, in an MT system using a traditional morphological analyser, but where the structural analyser would be replaced by a *syntactico-semantic editor* parametrized by the SCSG used to specify the usual purely automatic structural analyser. The structure of the

dialogues is partly an elaboration of Tomita's method for handling ambiguities (reduction of the number of questions asked, Tomita 1984), and partly original (paraphrases are proposed, and no questions are asked in terms of the grammar).

There are many situations where documents are produced within a computerized environment, and where the redactor must follow some norm of writing (e.g., the AECMA norm for airplane manuals written in English). Such an approach might lead to very good machine translations, transfer and generation being done automatically, as now. Of course, not all polysemies and ambiguities can be reduced in this way, because some are of a contrastive nature. Hence, revision would still be necessary to obtain "guaranteed" translations, as is the case with very good human translations of high importance (legal documents, maintenance manuals for potentially dangerous equipment, etc.).

3.3 Representation and treatment of ambiguities

The new types of MT systems just mentioned pertain respectively to the third and fourth generations. However, it is also interesting to investigate new approaches to the purely linguistic part of any MT process, even of second generation. What we have in mind is to *organize the linguistic knowledge as in an expert system*.

In all existing systems, much procedural knowledge is included in the rules and in the control of their application. The main part of this procedural knowledge is concerned with solving class and structure ambiguities. Usually, ambiguities are rather eliminated or ignored than handled.

In first generation systems, they are eliminated as soon as they are encountered. In second generation systems using the "filter" technique, their number is reduced by the combinatorial application of the rules, and one final solution is arbitrarily chosen (Veillon 1970). Sometimes (Slocum 1984), a weighting device is used to rank them, but it is extremely difficult to assign the weights in a meaningful way. In other systems (GETA, MU), heuristic programming is used to follow one or a few solutions at a time, with the possibility to backtrack (locally) or to "patch" later (Vauquois 1979, 1983). Programming in this way is quite delicate, although it allows to get some direct handling of the ambiguities.

What seems to be needed is a way to represent ambiguities such that the major part of the linguistic programming could be expressed without bothering about them, through rules of a declarative nature (like the "boards" of the SCSGs), and that ambiguities might appear as *problems* treated by a separate mechanism of *metarules* which would describe solutions to individual problems and be used by the linguist at appropriate points.

A first effort in this direction has been made by N. Verastegui (1982), before SCSGs were introduced. A promising line of research consists in using the boards of the SCSG as the declarative rules of an analyser, and to *precompute* as many ambiguity cases as possible, much in the same way this is done when testing that a context-free grammar is LL(1). Then, there would be some variant of the SLLP to describe how to solve those identified problems. Some default solution might be used in the absence of a good enough expert rule, and for the ambiguities which could not be precomputed. Note that the solution might involve only linguistic knowledge, or involve a call to some knowledge base or to a human being, so that this approach might also be good for the third and fourth generations of MT systems.

Acknowledgments

I would like to thank the organizers of this Machine Translation Summit for having given me the occasion to prepare this communication. Also, most of the research presented here has been funded by CNRS, by the University of Grenoble, and by several public agencies (Ministry of Defense, Ministry of Foreign Affairs, DGT and ADI). Finally, thanks to Elizabeth Guilbaud, who helped remove many grammatical errors from the first draft of this paper.

References

Bachut Daniel, Verastegui Nelson (1984)

Software tools for the environment of a computer-aided translation system. Proc. of COLING-84, ACL, 330-334, Stanford.

Boitet Christian (1976)

Un essai de réponse à quelques questions théoriques et pratiques liées à la traduction automatique. Définition d'un système prototype. Thèse d'Etat, Grenoble.

- Boitet Christian (1985)**
Traduction (assistée) par Ordinateur: ingénierie logicielle et linguistique. Colloque RF&IA, AFCET, Grenoble
- Boitet Christian (1986a)**
The French National MT-Project: technical organization and translation results of CALLIOPE-AERO. IBM Conf. on Translation Mechanization, Copenhagen.
- Boitet Christian (1987)**
Software and lingware engineering in modern MAT systems. To appear in the *Handbook for Computational Linguistics*, Batori, ed., Niemeyer, 1987.
- Boitet Christian, Gerber René (1984)**
Expert systems and other new techniques in MT. Proc. COLING-84, ACL, 468-471, Stanford.
- Boitet Christian, Guillaume Pierre, Quézel-Ambrunaz Maurice (1978)**
Manipulation d'arborescences et parallélisme: le système ROBRA. Proc. COLING-78. Bergen.
- Boitet Christian, Guillaume Pierre, Quézel-Ambrunaz Maurice (1982)**
ARIANE-78: an integrated environment for automated translation and human revision. Proc. COLING-82, North-Holland, Ling. series 47, 19-27. Prague.
- Boitet Christian, Guillaume Pierre, Quézel-Ambrunaz Maurice (1985)**
A case study in software evolution: from ARIANE-78 to ARIANE-85. Proc. of the Conf. on theoretical and methodological issues in Machine Translation of natural languages. 27-58. Colgate Univ., Hamilton, N.Y.
- Boitet Christian, Nedobejkine Nikolai (1981)**
Recent developments in Russian-French Machine Translation at Grenoble. Linguistics 19, 199-271.
- Boitet Christian, Nedobejkine Nikolai (1983)**
Illustration sur le développement d'un atelier de traduction automatisée. Colloque "l'informatique au service de la linguistique", Univ. de Metz.
- Boitet Christian, Nedobejkine Nikolai (1986)**
Toward integrated dictionaries for M(a)T: motivations and linguistic organization. Proc. COLING -86, IKS, 423-428, Bonn.
- Chappuy Sylviane (1983)**
Formalisation de la description des niveaux d'interprétation des langues naturelles. Thèse, Grenoble,
- Chauché Jacques (1975)**
Les langages ATEF et CETA. AJCL, microfiche 17, 21-39.
- Colmerauer Alain (1970)**
Les systèmes-Q, un formalisme pour analyser et synthétiser des phrases sur ordinateur. TAUM, Univ. de Montréal.
- Ducrot Jean-M. (1982)**
TITUS IV. In: *Information research in Europe.* Taylor, P.J., Cronin, P., eds., ASLIB, London. Proc. of the EURIM 5 conf., Versailles.
- Dymetman Marc (1986)**
Two approaches to Common Sense Inferencing for Discourse Analysis. Proc. COLING-86, IKS, 511-514, Bonn.
- Feng Zhi Wei (1981)**
Mémoire pour une tentative de traduction multilingue du chinois en français, anglais, japonais, russe et allemand. Doc. GETA, Grenoble, 40p. + annexes.
- Gerber René (1984)**
Etude des possibilités de coopération entre un système fondé sur des techniques de compréhension implicite (système logico-syntaxique) et un système fondé sur des techniques de compréhension explicite (système expert). Thèse, Grenoble.
- Gerber René, Boitet Christian (1985)**
On the design of expert systems grafted on MT systems. Proc. of the Conf. on theoretical and methodological issues in Machine Translation of natural languages, 116-134, Colgate Univ., Hamilton, N.Y.
- Guilbaud Jean-Philippe (1984)**
Principles and results of a German-French MT system. Lugano tutorial on Machine Translation.
- Guilbaud Jean-Philippe (1986)**
Variables et catégories grammaticales dans un modèle ARIANE. Proc. COLING-86, IKS, 405-407, Bonn.
- Harris Zellig (1962)**
String analysis of sentence structure. Mouton & Co, The Hague.
- Lepage Yves (1986)**
A language for transcriptions. Proc. COLING-86, IKS, 402-404, Bonn.

- Moneimne Walid (1986)**
La traduction assistée par ordinateur à Grenoble. Séminaire intern. d'Intelligence Artificielle, Tunis.
- Nakamura Junichi, Tsujii Junichi, Nagao Makato (1984)**
Grammar writing system (GRADE) of Mu-Machine Translation Project and its characteristics. Proc. COLING-84, ACL, Stanford.
- Nomura Hirosato, Naito Shozo, Katagiri Yasuhiro, Shimazu Akira (1986)**
Translation by understanding: a Machine Translation system LUTE. Proc. COLING-86, IKS, 621-626, Bonn.
- Sakamoto Yoshiyuki, Satoh Masayuki, Ishikawa Tetsuya (1984)**
Lexicon features for Japanese syntactic analysis in MU-project-JE. Proc. COLING-84, ACL, Stanford.
- Slocum Jonathan (1984)**
METAL: the LRC Machine Translation system. Lugano tutorial on Machine Translation.
- Tomita Masaru (1984)**
Disambiguating grammatically ambiguous sentences by asking. Proc. COLING-84, ACL, 476-480, Stanford.
- Tomita Masaru, Carbonell Jaime G. (1986)**
Another stride towards knowledge-based Machine Translation. Proc. COLING-86, IKS, 633-638, Bonn.
- Tong Loong Cheong (1986)**
English-Malay translation system: a laboratory prototype. Proc. COLING-86, IKS, 639-642, Bonn.
- Vauquois Bernard (1975)**
La traduction automatique à Grenoble. Document de linguistique quantitative n° 29, Dunod, Paris.
- Vauquois Bernard (1979)**
Aspects of automatic translation in 1979. IBM-Japan, scientific program.
- Vauquois Bernard (1983)**
Automatic translation. Proc. of the summer school "The computer and the arabic language", ch. 9, Rabat.
- Vauquois Bernard, Boitet Christian (1985)**
Automated translation at GETA (Grenoble University). Computational Linguistics, 11:1, 28-36.
- Vauquois Bernard, Chappuy Sylviane (1985)**
Static grammars: a formalism for the description of linguistic models. Proc. of the Conf. on theoretical and methodological issues in Machine Translation of natural languages 298-322, Colgate Univ., Hamilton, N.Y.
- Veillon Gérard (1970)**
Modèles et algorithmes pour la traduction automatique. Thèse d'Etat, Grenoble.
- Verastegui Nelson (1982)**
Utilisation du parallélisme en traduction automatisée par ordinateur. Proc. COLING-82, North-Holland, Ling. series 47, 397-405, Prague.
- Walker Donald E. (1985)**
Knowledge resource tools for accessing large text files. Proc. of the Conf. on theoretical and methodological issues in Machine Translation of natural languages, 335-347, Colgate Univ., Hamilton, N.Y.
- Woods William (1970)**
Transition network grammars for natural language analysis. CACM 13/10, 591-606.
- Zaharin (1986a)**
Strategics and heuristics in the analysis of a natural language in Machine Translation. Ph.D. thesis, Umversiti Sains Malaysia, Penang (research conducted in cooperation with GETA, Grenoble).
- Zaharin (1986b)**
Strategics and heuristics in the analysis of a natural language in Machine Translation. Proc. COLING-86, IKS, 136-139, Bonn.
- Zajac Rémi (1986a)**
Etude des possibilités d'interaction homme-machine dans un processus de traduction automatique. Spécification d'un système d'aide à la rédaction en vue d'une traduction par machine. Définition d'un langage de spécification linguistique. Thèse, Grenoble.
- Zajac Rémi (1986b)**
SCSL: a linguistic specification language for MT. Proc. COLING-86, IKS, 393-398, Bonn.
- Yang Ping (1981)**
Un essai sur la génération du chinois. Doc. GETA, Grenoble, 30p. + annexes.
- Yan Yong Feng (1987)**
Vers une ingénierie de la production de linguiciels. Spécification et réalisation d'un prototype de poste de travail linguistique pour la spécification de correspondances structurales. Thèse, Univ. de Grenoble.

-0-0-0-0-0-0-0-0-0-0-0-0-0-