Generation of Vietnamese for French-Vietnamese and English-Vietnamese Machine Translation

DOAN-NGUYEN Hai
Groupe de recherche sur l'asymétrie des langues naturelles,
Université du Québec à Montréal, H3C-3P8, Canada.
E-mail: c1322@er.uqam.ca
and
Laboratoire d'Analyse et de Technologie du Langage (LATL)
Faculté des Lettres, Université de Genève
2, rue de Candolle, CH-1211 Genève 4, Switzerland

Abstract

This paper presents the implementation of the Vietnamese generation module in ITS3, a multilingual machine translation (MT) system based on the Government & Binding (GB) theory. Despite well-designed generic mechanisms of the system, it turned out that the task of generating Vietnamese posed non-trivial problems. We therefore had to deviate from the generic code and make new design and implementation in many important cases. By developing corresponding bilingual lexicons, we obtained prototypes of French-Vietnamese and English-Vietnamese MT, the former being the first known prototype of this kind. Our experience suggests that in a principle-based generation system, the parameterized modules, which contain language-specific and lexicalized properties, deserve more attention, and the generic mechanisms should be flexible enough to facilitate the integration of these modules.

1 Introduction

Although Vietnamese is now spoken by about 80 millions people in the world, there has not been much work on machine translation (MT) from and to this language, except some English-Vietnamese MT implementations (eg. Doan-Nguyen, 1994) of minor success. As far as we know, there has been yet no similar implementation for French-Vietnamese MT. This paper presents the implementation of the generation module for Vietnamese in *ITS3*, a multilingual MT system developed at the *Laboratoire d'Analyse et de Technologie du Langage* (LATL), University of Geneva. Together with the generation module, we construct bilingual lexicons, and thus obtain prototypes of French-Vietnamese and English-Vietnamese MT.

As Vietnamese is very different from European languages, the implementation of the generation module for Vietnamese based on the generic mechanisms of ITS3 poses non-trivial problems. We present here some main problems and their solutions, such as the construction of Vietnamese noun phrases (NPs), verb phrases (VPs), adverbial phrases (AdvPs), relative clauses, etc.

2 Brief description of ITS3

ITS3 (Wehrli, 1992; Etchegoyhen & Wehrli, 1998; L'haire & al, 2000) can now translate from French to English and vice versa. Modules for other languages such as German, Italian, are under development. ITS3 is a principle-based system, linguistically inspired by the Government & Binding (GB) theory. (See eg. Haegeman (1994) for an introduction to GB, Berwick & al (1991) for principle-based systems). The system chooses the classical analysis-transfer-generation approach of MT (see Hutchins & Sommers, 1992). ITS3 works on single isolated sentences. A sentence in the source language is analyzed into a logico-linguistic structure, called *pseudo-semantic structure (PSS)*. After a lexical transfer phase, this PSS is passed to the generation phase, which finally produces the sentence in the target language. By default, ITS3 gives a unique solution, the best one.

Let's take an example of French-English translation to illustrate the process. The analysis phase consists of two steps: GB-based syntax analysis and PSS construction. Syntax analysis is carried out by the *IPS* parser (Wehrli, 1992), which builds the X-bar structure of the sentence, using many filtering constraints (on thematic roles, on cases, etc.) to reduce overgeneration.

(1) La maison a été vendue.

(2) [TP [DP la [NP maison]]i [T' a [VP été [VP vendue [DP ei]]]]]

A PSS is then derived from the syntax analysis results (Etchegoyhen & Wehrli, 1998). Components of the sentence are represented in corresponding frame-liked structures. For example, a clause gives rise to a PSS of type CLS, which contains the main verb or adjective (the `Predicate` slot) and other information on tense, mood, voice, etc., as well as the PSS's of its arguments and adjuncts (the `Satellites`). Similarly, a noun phrase gives rise to a PSS of type DPS, which contains, besides the main noun (the `Property` slot), its number, gender, referential index for binding resolution, etc. A PSS thus contains abstract linguistic values for "closed" features (tense, mood, voice, number, gender, etc.), and lexical values for "open" features (CLS `Predicate`, DPS `Property`, etc.).

```
PSS[{ }
   CLS[
      Mood           : real
      Tense          : E = S
      InfoFunction   : categorical
      Modality       : undefined
      Aspect                     :  (non
progressive, perfective)
      Voice          : passive
      Causative      : not causative
      Negation       : not negated
      Utterance type : declaration
      Predicate      : vendre
    ]CLS
   Satellites  {
      PSS[{ }
         Theta role       : theme
         DPS[
            Property         : maison
            Operator         : the
            Number           : singular
            Gender           : feminine
            Ref. index       : i
          ]DPS
       ]PSS
     }
 ]PSS
```

In the lexical transfer phase, the lexical units in the PSS are replaced by those in the target language, using frequency data for translation selection. In the generation phase, a generic engine called *GBGEN* (Etchegoyhen & Wehrle, 1998; Etchegoyhen & al, 1999) cooperates with language-specific modules to construct the output from the PSS in three steps. First, D-structure generation maps the PSS into an X-bar structure in a top-down fashion (see 3a). Next, S-structure generation carries out movements and bindings (3b). Finally, morphological realization is done (3c), and the result is output, as in (3d).

(3) (a) [CP [TP [VP aux [VP aux [VP sell [DP the [NP house]]i]]]]]

(b) [CP [TP [DP the [NP house]]i [T' [VP aux [VP aux [VP sell [DP ei]]]]]]]

(c) [CP [TP [DP the [NP house]]i [T' [VP has [VP been [VP sold [DP ei]]]]]]]

(d) The house has been sold.

Note that ITS3 does only lexical, and not structural, transfer. This approach can therefore be considered as half transfer half interlingual. It is not the purpose of this paper to discuss the pros and cons of the transfer and interlingual approaches in MT. See eg. Gdaniec (1998) for discussions about advantages of a particular transfer-based MT system, and Dorr (1993) for an interlingual one. The latter, also based on GB, concentrates on treating mismatches across languages, an issue less considered in ITS3. It needs however to use very complex representations for its interlingual approach, hence is not likely to become a practical system.

As for the specification issue, ITS3 chooses to be purely procedural. All generic engines and language-specific modules are written in Modula-2. Procedure templates are designed so that one can fill in language-specific parameters when adding a new language. However, this is not always straightforward, as one will see in the integration of Vietnamese below. In general, any development requires to read, understand, and often modify some parts of the huge code. This is an important reason why a declarative approach would be preferred (see eg. Emele & al, 1992; Nicolov & Mellish, 2000). Unfortunately, we do not have at our disposal any declarative system with high-quality French analysis. Also, as best as we know, there are no parallel French-Vietnamese or English-Vietnamese corpora built so far to think of statistical or example-based MT approaches. ITS3 is one among few systems that can do French syntax analysis with large lexical and grammatical coverage. It can therefore serve our main purpose to develop a prototype of French-Vietnamese MT in a short term.

3 Generation of Vietnamese

In this section, we present the problems and our solutions for constructing Vietnamese NPs, VPs, AdvPs, relative clauses, etc. in ITS3. Below we will use generalized notions of NP and VP in GB, that of DP and TP, respectively.

3.1 DP construction

3.1.1 Vietnamese noun categorizers

Many Vietnamese nouns have to be preceded by a *"categorizer"* to form an NP. For example, knowing that *"a"="một", "cat"="mèo"*, we cannot translate *"a cat"* into **"một mèo"*, but *"một **con** mèo"*. Here *"mèo"* needs the categorizer *"con"*. A categorizer is also a noun, giving some vague idea on the semantic class of the noun which requires it. For example, almost every noun designating an animal needs *"con"*. However, there seems to be no general rule to determine the categorizer for a particular noun. We therefore specify the categorizer for each noun in the Vietnamese lexicon. This information helps to form Vietnamese NPs appropriately, eg. *"a cat"* gives rise to

(4) [DP một [NP con [NP mèo]]],

but *"a language"* to

(5) [DP một [NP ngôn ngữ]],

because *"ngôn ngữ"* needs no categorizer.

3.1.2 Plural DPs

One important task in DP construction for many languages is to assure agreement (on number, gender, etc.). Vietnamese words are morphologically invariant with respect to all these concepts. For plural DPs, we need to add an appropriate determiner: a quantifier if it is specified (*"two students" = [DP hai [NP sinh viên]], "some students" = [DP vài [NP sinh viên]]*), or *"những"* otherwise (*"(the) students" = [DP những [NP sinh viên]]*).

3.1.3 Determiners

GBGEN supposes a 1-1 mapping in which a determiner in a language corresponds to a universal *operator* and vice versa, eg.:

English	French	Operator
each	chaque	*every*
this, these	ce, cette, ces	*demonstrative*
no	aucun, aucune	*no*

"Ces chats", eg., is analyzed into a PSS like (note the `Operator` slot):

```
DPS[
      Property          : chat
      Operator          : demonstrative
      Number            : plural
      Ref. index        :
]DPS
```

After *"chat"* is replaced by *"cat"*, this gives *[DP these [NP cats]]*. This model does not apply totally to Vietnamese DPs. Some operators correspond to a determiner as prescribed by the model. Some do not, but require instead an adjective after the noun, and some others need both a determiner and an adjective.

Operator	English/ French	Vietnamese
every	each cat/ chaque chat	[DP **mỗi** [NP con [NP mèo]]]
demonstrative (singular)	this cat/ ce chat	[DP [NP con [NP mèo [AP **này**]]]]
demonstrative (plural)	these cats/ ces chats	[DP **những** [NP con [NP mèo [AP **này**]]]]
no	no cat/ aucun chat	[DP **không** [NP con [NP mèo [AP **nào**]]]]

3.1.4 Strategy for Vietnamese DP construction

It turns out to be somewhat problematic to construct Vietnamese DPs in the generic model of GBGEN. First, the procedure template for deriving the determiner from the DPS `Operator` slot does not expect that there may be an adjective after the noun. Modifying this procedure template would lead to many obligatory changes in modules for all other languages of the system. Moreover, this would not mean that the template be generic enough for every human language. Second, the generic model does not evidently foresee a facility for treating Vietnamese categorizers. We therefore found more convenient to develop a *specialized*[1] procedure for Vietnamese DP construction. This allows a safe treatment of Vietnamese DPs while still respecting the available system.

This procedure computes the determiner and post-nominal adjective from the `Operator` and `Number` slots of the DPS. A DP is then projected from the determiner. Its NP complement is built from the main noun (the `Property` slot in the DPS). If the noun needs a categorizer, which is given in its lexical entry, the NP will be of structure *[NP Categorizer [NP Main]]*, otherwise it will be only *[NP Main]*. Finally, the post-nominal adjective is added as a complement of the NP.

3.2 TP construction

The principal strategy of GBGEN for TP construction is to create the following general frame, and attempt to fill it gradually with appropriate elements:

[TP [T' Modal [VP Perfective [VP Passive [VP Progressive [VP Main]]]]]]

where *Modal*, *Perfective*, *Passive*, and *Progressive* stand for auxiliary verbs representing respectively the modal, perfective, passive, and progressive aspects of the TP, and *Main* is the main verb. See example (3) above. This model seems to work at least with French and English. However, Vietnamese has many differences from these languages on verbal notions and on VP formation, as will be presented in the following.

3.2.1 Tenses and aspects

In Vietnamese, verbs are not conjugated, and tense and aspect are generally understood in context. *"He sleeps"*, *"He slept"*, *"He is sleeping"* eg., can all be translated in suitable contexts into *"Anh ta ngủ"*. To explicit the tense and aspect, Vietnamese uses some adverbs as shown below.

He sleeps	[TP [NP Anh ta] [T' [VP ngủ]]]
He slept	[TP [NP Anh ta] [T' **đã** [VP ngủ]]]
He will sleep	[TP [NP Anh ta] [T' **sẽ** [VP ngủ]]]
He is sleeping	[TP [NP Anh ta] [T' **đang** [VP ngủ]]]
He has slept	[TP [NP Anh ta] [T' **đã** [VP ngủ]]]

There are some cases where it is difficult to have a concise translation in Vietnamese, eg. *"He has been sleeping"* may be translated into *"Anh ta đã ngủ"* (past tense emphasized) or *"Anh ta đang ngủ"* (progressive aspect emphasized)[2]. We choose the one that seems preferable, eg. the second sentence for this example.

[1] As understood in object-oriented paradigm.

[2] It is impossible to say **"Anh ta đã đang ngủ"* or **"Anh ta đang đã ngủ"*.

3.2.2 Negation and modality

The `Negation` slot of a CLS specifies whether it is in negative form or not. The `Modality` slot contains an abstract value for the modality of the verb, eg. `possibility` corresponds to English *"can"* and French *"pouvoir"*, `obligation` to *"must"* and *"devoir"*. GBGEN foresees an orthogonal combination of negation and modality; it inserts *"not"* after the modal verb for English, or *"ne"* and *"pas"* around it for French. In Vietnamese, one generally adds the adverb *"không"* before the verb to form a negation.

(6) Tôi chạy. (I run.)

(7) Tôi ***không*** *chạy. (I do not run.)*

(8) Tôi có thể chạy. (I can run.)

(9) Tôi ***không*** *có thể chạy. (I cannot run.)*

Evidently, this orthogonal model will have trouble in translation, because a modal verb in negative form may have different logical interpretations from one language to another. For example, *"must"* = *"phải"*, *"I must run"* = *"Tôi phải chạy"*, but

(10) I must not run.

cannot be translated into *"Tôi* ***không*** *phải chạy"*, which means *"I don't have to run"*. The right translation should be

(11) Tôi ***không*** *được chạy.*

using another modal, *"được"*.

At the moment, the specifications in the PSS does not allow to determine the logical interpretation of a negated modal verb. In waiting for an improvement of GBGEN on this issue, we implement a temporary solution which helps to translate negative modal verbs from English and French, *specifically*, to Vietnamese. The appropriate Vietnamese negative modal verb form is derived not only from the `Modality` slot of the interested CLS, as done in GBGEN, but also by examining its `Negation` slot.

3.2.3 Passive

Passivization is realized in Vietnamese by adding *"được"* or *"bị"* before the verb. *"Bị"* is used when the subject suffers a bad effect from the action, otherwise *"được"* is used. We put *"được"* or *"bị"* in the specifier component of the VP, ie. [Spec, VP]. The choice of *"được"* or *"bị"* for a verb is considered as a lexical one, and stored in the Vietnamese lexicon.

(12) Le chat a été tué. (The cat was killed.)

(13) [TP [DP Con mèo]i [T' đã [VP ***bị*** *[V' giết ei]]]]*

(14) Le livre a été écrit par John. (The book was written by John.)

(15) [TP [DP Quyển sách]i [T' đã [VP ***được*** *[V' viết ei [PP bởi John]]]]]*

3.2.4 Translations of **be/être**

The lexical transfer procedure in ITS3 does not take into account the interaction between the components of the sentence when it translates the lexical units in the PSS. In particular, the English *"be"* is always translated into the French *"être"*, and vice versa. However, to translate *be/être* into Vietnamese, one has to distinguish between at least three cases[3].

He is a student	Anh ta [T' [VP **là** [DP một sinh viên]]]
He is intelligent	Anh ta [T' [VP (**thì**) [AP thông minh]]] (*thì* is optional.)
This flag is of this country	Lá cờ này [T' [VP (**thì / là / thì là**) [PP của nước này]]] (Here *thì*, *là* or even *thì là* are all possible and optional.)

For the first case, it suffices to test the theta role of the complement of the verb in the PSS, which should be *THEME*, to have the right translation *"là"*. In the last two cases, whether using *"thì"* or *"là"* or neither is too delicate to explain, as it concerns pragmatic issues. We decide to put

[3] We ignored to treat, eg., the case of *be* + infinitive (*"He is to do it"*, *"Anh ta* ***phải*** *làm việc đó"*).

"(thì)" for the second case where the complement is an AP, and *"(thì/là)"* for all other cases.

3.2.5 Strategy for Vietnamese TP construction

From the discussion above, it seems not very natural to follow the construction order of GBGEN in building Vietnamese TPs, neither to reuse some of its pre-designed procedure templates, such as selecting auxiliary verbs. We need rather to implement a different strategy. At first, a simple frame *[TP [T' [VP ...]]]* is built as D-structure. Verbal information, such as tense, aspect, modality, negation, is gathered from the PSS as much as possible. The complete TP is then constructed based on the combination of gathered information, and in an order particular to Vietnamese. The adverb representing the tense/aspect of the clause, if exists, will occupy the head position of the TP. The modal, passive, and main verb make up layers of VPs in the TP. Values of negation and modal are computed together. The maximal frame looks like:

[TP [T' Tense [VP Negation [V' Modal [VP Passive [V' Main]]]]]]

For example, for the sentence

(16) Il n'a pas pu être tué. (He could not be killed.)

the past tense gives *"đã"*, the negation and the modality combine and give *"không thể"*[4], and the passive gives *"bị"* by consulting the lexical entry of the verb *"giết"*:

(17) [TP [DP Anh ta]i [T' đã [VP [V' không thể [VP bị [V' giết ei]]]]]]

In particular, if the main verb is a translation of *be/être* (checked with a bit in the lexical entry), its complements will be examined to give the right translation.

3.3 Other constructions

3.3.1 AdvP location

In ITS3, a large set of adverbs and, more generally, adverbial phrases (AdvPs) are classified into semantic groups, specified by a *value*. For example, English *"much"* and French *"beaucoup"* are assigned the abstract value *degree*. GBGEN uses this information to locate the generated AdvP in an appropriate position.

This generic approach is not perfect. For example, the equivalent adverbs *"where"* (English), *"où"* (French), and *"ở đâu"* (Vietnamese) all have the *where* value, and would be moved to [Spec, CP] of the subordinate clause. This would give a bad Vietnamese sentence (20). The correct one is (21).

(18) I know [CP [AdvP ***where****]i [C' [TP he [T' [VP sleeps [AdvP ei]]]]]].*

(19) Je sais [CP [AdvP ***où****]i [C' [TP il [T' [VP dort [AdvP ei]]]]]].*

*(20) *Tôi biết [CP [AdvP* ***ở đâu****]i [C' [TP anh ta [T' [VP ngủ [AdvP ei]]]]]].*

(21) Tôi biết [TP anh ta [T' [VP ngủ [AdvP ***ở đâu****]]]].*

This example shows that AdvP location should be language-specific and lexicalized. The generic procedure is in fact just a specialized one valid for some class of languages. It is not difficult here to imitate it for a treatment of AdvP location specific to Vietnamese.

3.3.2 Negative words

Translating structures with negative words, such as *"jamais"* = *"never"* = *"không bao giờ"*, *"rien"* = *"nothing"* = *"không cái gì cả"*, etc. into Vietnamese is problematic. A straightforward application of the generic engine might yield exactly the opposite meaning, eg.:

(22) Je / ne dors ***jamais****. (I never sleep.)*

*(23) *Tôi / không*[5] ***không bao giờ*** *ngủ. (It is not that I never sleep.)*

[4] *"không thể"* is a concise and more frequent form of *"không có thể"* (see example (9)).

[5] We recall that in Vietnamese the adverb *"không"* is inserted before the verb to form a negation.

The right sentence should be

*(24) Tôi **không bao giờ** ngủ.*

The same problem was known in French-English translation, and cured in GBGEN by realizing the English sentence not in negative but in affirmative form. This solution does not work for Vietnamese:

*(25) Je / n'écris **rien**. (I write nothing.)*

*(26) *Tôi / viết **không cái gì cả**.*

*(27) Tu / ne dois **jamais** / courir trop vite. (You must never run too fast.)*

*(28) *Anh / phải **không bao giờ** / chạy quá nhanh.*

The right translations for (25) and (27) should be, respectively:

*(29) Tôi / **không** viết **cái gì cả**. (I do not write anything.)*

*(30) Anh / **không bao giờ** được / chạy quá nhanh. ("được" is used instead of "phải". See 3.2.2)*

Our solution here is to keep the verb in the negative form, and use the "indefinite" counterparts *"bao giờ"*, *"cái gì cả"*, etc. of the expressions *"không bao giờ"*, *"không cái gì cả"*, etc[6]. The structure of eg. the translation (24) is thus

*(31) [TP Tôi [T' [VP **không** [VP **bao giờ** [V' ngủ]]]]],*

where *"không"* and *"bao giờ"* are two different constituents. Note however that this solution gives a less good but still acceptable translation of (27), that of *"Anh không được **bao giờ** chạy quá nhanh"*. We could have done better, but at the cost of much more complicated programming.

3.3.3 Wh-movements

Vietnamese wh-questions do not need a wh-movement as in English:

*(32) **Whom** have you seen ?*

*(33) Anh đã thấy **ai** ? ("whom"="ai")*

We therefore block the wh-movement procedure in GBGEN in constructing wh-questions. However, there is a case where a movement is preferred and realized, that of *why*[7].

*(34) **Pourquoi** il ne dort pas ? (Why doesn't he sleep?)*

*(35) Anh ta không ngủ **tại sao** ? (Acceptable)*

*(36) **Tại sao** anh ta không ngủ ? (Preferred)*

3.3.4 Relative clauses

To form a relative clause in Vietnamese, one can generally add an optional complementizer *"mà"* before the clause. We decide to put *"(mà)"* for subject relative clauses, and *"mà"* for object relative clauses, as it is more acceptable to drop *"mà"* in the former case than in the latter.

(37) The student / who has seen the cat / is John.

(38) Người sinh viên / [CP [C' (mà) [TP đã thấy con mèo]]] / là John.[8]

(39) The student / whom you see / is John.

(40) Người sinh viên / [CP [C' mà [TP anh thấy]]] / là John.

The translation of adjunct relative clauses which begin with a preposition from French or English into Vietnamese is difficult. In general, we need to keep the preposition at the end of the relative clause, rather than move it to the beginning as GBGEN proposes:

*(41) La fille / **avec** qui John parle / est Mary.*

*(42) The girl / **with** whom John talks / is Mary.*

[6] Just as *"anything"* to *"nothing"* in English.

[7] This is done by the AdvP location procedure (see section 3.3.1).

[8] If *"mà"* is dropped, it is a sort of garden-path sentence. But this is common in Vietnamese, and may be an interesting subject to study.

(43) Cô gái / mà John nói chuyện ***với*** */ là Mary. ("avec"="with"="với"; "parler"="talk"="nói chuyện".)*

At the moment, we cannot deal with cases where a paraphrase is needed for a correct translation. Knowing that *"without"="không có"*,

(44) The girl / ***without*** *whom John cannot work / is Mary.*

*(45) *Cô gái / mà John không thể làm việc* ***không có*** */ là Mary.*

(46) Cô gái / mà nếu không có (cô ấy) John không thể làm việc / là Mary.

(The girl / that if she is not there, John cannot work / is Mary.)

4 Results

The implemented generation module for Vietnamese can realize almost all structures that can be generated from the intermediate PSSs. Many of them are of course not yet perfect, but a French-Vietnamese translation test on a sample of French sentences of many different syntactic structures gave encouraging results. We did not consider tests on English-Vietnamese translation, because the English analysis module in ITS3 has not yet been well developed.

We have not been able to do a large-scale test on real corpora yet, because our lexicons are still small (about 400 entries for each bilingual lexicon, among them many functional words (prepositions, adverbs, pronouns, conjunctions)). However, tests are not necessarily restricted by the size of the lexicons, because if a source language word is not found in the bilingual lexicon, it is still retained in the PSS during the lexical transfer phase. This word will then appear in the target language sentence exactly at the position of its supposed translation.

As it is well known, lexicon building requires huge investments on human work and time. One can use methods of (semi-)automatic acquisition of dictionary resources (see eg., Doan-Nguyen, 1998) to obtain quickly a large draft of necessary lexicons, provided that such resources (eg. a French-Vietnamese dictionary text file) exist. In the worst case, a human will verify and complete this draft, but in general this is still much cheaper than developing a lexicon from scratch. We did not, unfortunately, have any of these resources. Nevertheless, we profited much from a French-English lexicon draft extracted from ITS3's lexicons: much lexical information in its entries can be reused in the corresponding Vietnamese entries (eg. the part-of-speech, the verb theta grid). Moreover, English translations of a French word, as well as French translations of an English word, help to choose correct corresponding Vietnamese translations.

5 Discussion

Although not totally perfect, ITS3, and in particular GBGEN, show to be good systems for multilingual MT. They have a solid linguistic theoretical base, a modular computational design, and a surprising performance. Besides the problems presented in this paper, we find convenient to use many available procedure templates, such as PP construction, movements and bindings. In particular, ITS3 is able to do robust, high-quality, and broad-coverage syntactic analysis for French. Our experience can be seen as a test on integrating an "exotic" language into the sytem.

As we have shown above, many difficulties in implementing the generation module for Vietnamese stem from "mismatches" between Vietnamese grammatical notions and the model of the generic engine GBGEN. It is largely agreed that designing a generic, flexible, and efficient system for pratical applications of multilingual generation and MT is a very difficult problem. Our experience suggests that in a principle-based generation system such as GBGEN, the parameterized modules, which contain language-specific and lexicalized properties, should be of more importance. The flexibility of a generic system consists in designing good "slots" so that modules for a new language can be plugged in systematically and conveniently.

As discussed in section 2, a declarative approach may be very beneficial for system development, including genericity and flexibility. The programming paradigm is also an important factor. The LATL has recently begun to reengineer ITS3 in an object-oriented language, which facilitates the development of the system while still guanratees its performance[9].

Apart from the generation phase, the quality of an MT system depends heavily on the analysis modules. The construction of the PSS from the syntactic analysis of the input sentence is of crucial importance. We find that this is a real bottleneck in ITS3: in many cases, despite a good syntactic analysis, the translation fails because of a bad PSS construction. PSS construction is obviously a very difficult task, as it is in fact a kind of translation, that goes from a syntactic structure into a logical formalism. See eg. Alshawi (1992) for a similar task, ie. translating English sentences into a logical representation.

6 Conclusions

With the Vietnamese generation module and the lexicons developed, we have implemented first prototypes of French-Vietnamese and English-Vietnamese MT. As we know best, this is the first time a French-Vietnamese MT prototype is realized.

Our future work is to develop the lexicons, improve the implemented module, and test it on real corpora for a more precise evaluation. We also envisage doing Vietnamese GB-based analysis in the framework of ITS3.

Acknowledgements

I am grateful to the *Agence Universitaire de la Francophonie* whose scholarship allowed me to carry out this project. This research has also been supported by the Social Sciences and Humanities Research Council of Canada (grant # 412-97-0016), for the "Asymmetry and Natural Language Processing Project", awarded to professor Anne-Marie Di Sciullo, in the *Département de Linguistique at Université du Québec à Montréal (UQAM).* Eric Wehrli, Thierry Etchegoyhen, Luka Nerima, Anne Vandeventer, and all members of LATL, reserved for me precious help and friendship during my work in Geneva. I would like also to thank Eric Wehrli, Nicolas Nicolov, Nikolay Vazov, and the EWNLG-01 reviewers for helpful comments on earlier versions of this paper, and finally, Anne-Marie Di Sciullo for her support to this research.

References

Berwick R., Abney S., & Tenny C., editors (1991) *Principle-Based Parsing: Computation and Psycholinguistics*. Kluwer Academic Publishers.

Alshawi, H. (1992) *The Core Language Engine*. MIT Press.

Doan-Nguyen H. (1993) *The English-Vietnamese Translation Machine-88*. Proceedings of HoChiMinh City Mathematics Consortium -1993, HoChiMinh City, pp. 217-222.

Doan-Nguyen H. (1998) *Accumulation of Lexical Sets: Acquisition of Dictionary Resources and Production of New Lexical Sets.* Proceedings of the 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics, COLING-ACL '98, Montreal, pp 330-335.

Dorr B. (1993) *Interlingual Machine Translation: A Parameterized Approach.* Artificial Intelligence, Vol. 63, N. 1-2, pp. 429-492.

Emele M., Heid U., Moma S. & Zajac R. (1992) *Interactions between Linguistic Constraints: Procedural vs. Declarative Approaches*. Machine Translation, Vol. 7, N. 1-2, pp. 61-98.

Etchegoyhen T. & Wehrli E. (1998) *Traduction automatique et structures d'interface*. Actes de la Conférence sur le Traitement Automatique du Langage Naturel, TALN '98, Paris.

Etchegoyhen T. & Wehrle, T. (1998) *Overview of GBGen*. Proceedings of the 9th International Workshop on Natural Language Generation, Niagara-on-the-lake, Canada.

Etchegoyhen T., Wehrle T., Mengon J. & Vandeventer A. (1999) *Une approche efficace à la génération syntaxique. Le système GBGen*. Actes du 2ème colloque francophone sur la Génération Automatique de Textes, GAT '99, Grenoble.

Gdaniec C. (1998) *Lexical Choice and Syntactic Generation in a Transfer System: Transformations in the New LMT English-German System*. In Farwell D. & al (ed.) Machine Translation and the Information Soup, Third Conference of the Association for Machine Translation in the Americas, AMTA '98, Langhorne, PA, USA, pp. 408-420.

[9] Eric Wehrli, personal communication.

Haegeman L. (1994) *Introduction to Government & Binding Theory, 2nd Edition*. Blackwell, Oxford (UK) and Cambridge (USA), 701 p.

Hutchins J. & Sommers L. (1992) *An Introduction to Machine Translation*. Academic Press, London.

L'haire S., Mengon J. & Laenzlinger C. (2000) *Outils génériques et transfert hybride pour la traduction automatique sur Internet*. Actes de la Conférence sur le Traitement Automatique du Langage Naturel, TALN '2000, Lausanne.

Nicolov N. & Mellish C. (2000) *PROTECTOR: Efficient Generation with Lexicalized Grammars*. Recent Advances in Natural Language Processing, John Benjamins, pp. 221-243.

Wehrli, E. (1992) *The IPS system*. Proceedings of the 14th International Conference on Computational Linguistics, COLING '92, Nantes, pp. 870-874.