A Study of some Lexical Differences between French and English Instructions in a Multilingual Generation Framework*

Farid Cerbah
Dassault Aviation
DGT/DTN/EL — 78, quai Marcel Dassault - cedex 300
92552 Saint-Cloud - FRANCE
e-mail: cerbah@dassault-avion.fr — Fax: 33 (1) 47-11-52-83

Abstract

This paper describes ongoing research on the lexicalisation problem in a multilingual generation framework. We will focus in particular on two major types of *verbal* differences observed in a corpus of bilingual (*French - English*) procedural texts extracted from aircraft maintenance manuals. To deal with these two types of differences, we propose lexicalisation mechanisms, which proceed from the same semantic representation for both French and English realisations. We will however discuss at the end of the paper other types of lexical differences which may require language-specific inputs.
keywords: Multilingual generation, lexical choice, controlled languages.

1 Introduction

Technical documentation appears as a promising application area for text generation. Several works ([18, 17, 6, 12, 7][1]) demonstrate that NLG techniques may contribute in the future to make technical documentation more reliable and maintainable. Many of these contributions are concerned with multilingual generation, which is often presented as an alternative to Machine Translation. The multilingual generation approach stipulates that technical documents, such as maintenance manuals, can be generated automatically in several languages from knowledge bases used in design processes or constructed for the purpose of automatic documentation production.

GhostWriter is a bilingual generation system under development at Dassault Aviation and British Aerospace. Our objective in this project is to show how French and English maintenance procedures can be generated from an abstract representation of underlying action plans expressed in a formalism inspired by AI planning models. The role of the text generator is to propose bilingual drafts of procedural texts intended to be integrated in maintenance manuals, and to perform rephrasing operations which may be requested by the technical author, for example grouping maintenance instructions at surface level or changing the specificity level of an instruction.

The design of a multilingual generation system, needless to say, requires a precise analysis of the linguistic means used by each language to express the same conceptual content. The aim of this paper is to describe the main *verbal* differences observed in a bilingual corpus of procedural texts and to analyse their impacts on the lexicalisation mechanisms of the sentence generation system GLOSE [4] used in GhostWriter. The structure of this paper is as follows. I give in section 2 an overview of GLOSE. Then, I discuss briefly in the next section the corpus analysis and its role in the design of the multilingual generation system. Sections 4 and 5 focus on specific types of lexical differences and the related lexicalisation mechanisms. Finally, the conclusion will describe some lexical divergences which may require the introduction of language-specific semantic representations.

*This paper partly covers a work made by the author at Dassault Aviation within a Technical cooperation between Dassault Aviation and British Aerospace - Military Aircraft Division. The University of Edinburgh was involved in this project as a sub-contractor of British Aerospace.

[1] This list is far from being exhaustive.

2 The sentence generator

Our sentence réaliser GLOSE is based on Meaning-Text Theory (MTT) [14]. This linguistic theory offers many potentialities for multilingual applications. In computational linguistics, it has been primarily used as a theoretical basis for language generation models (e.g. [2, 1, 16]). Recently, some works in the fields of machine translation and computational lexicography (e.g. [8], [9]) take advantage of lexicographic descriptive concepts offered by MTT, in particular the well-known notion of *lexical function*. In accordance with the stratified framework of MTT, the target representation of the lexicalisation process of GLOSE is a *Deep Syntactic representation* — mainly a dependency tree, whose nodes are labeled with *full lexemes* and lexical functions. The relations between nodes represent *deep syntactic relations* which are defined as abstractions over superficial syntactic relations. The dependency tree is enriched with communicative bipartitions such as *Theme/Rheme* and *Given/New*. We will ignore these communicative constraints in this paper because they are of minor importance for the linguistic phenomena considered here. Lexical functions are used to represent syntactico-semantic relations between lexemes, such as synonymy, hyperonymy, and various types of collocational relations.

GLOSE is composed of two MT-models[2], one for each of the two languages considered in our domain. It should be mentioned that only the grammatical realisation[3] component of GLOSE can be considered as an implementation of "*pure*" MT-models, since we do not use at the lexicalisation phase MTT-style semantic networks which represent in this theory a linguistically motivated semantic level, independent of the conceptual level. The integration of such semantic representations in a multilingual environment raises several theoretical and practical problems which will be the object of future investigations. We should note that these problems are studied by several NLG researchers (eg, [10, 11, 13]). At present, we consider the lexicalisation problem as a mapping process from conceptual representations to French and English lexemes. This process relies on *concept-lexeme mapping structures*, integrated in the lexicon, and which represent *elementary* transitions from conceptual structures to lexemes.

3 The contrastive analysis

The corpus is composed of about thirty bilingual pairs of extended procedural texts extracted from aircraft maintenance manuals. Our contrastive analysis concentrates on verbal expressions. Verbal differences between French and English instructions can be classified along three interrelated dimensions: (1) *lexical* — French and English versions diverge because of differences in the lexical resources available in both languages — (2) *syntactic* — equivalent verbs exist but the two versions cannot rely on *similar* syntactic constructions —, and (3) *stylistic* — lexically and syntactically equivalent versions may be obtained but one of them would be stylistically incorrect.

We should stress that, when designing the lexicalisation component of a multilingual generation system, one should be careful in deciding how much importance should be given to such a contrastive analysis. In the corpus, bilingual sentences expressing the same content may differ significantly, even though closely related and acceptable versions can be obtained. Hence, in such cases, it is difficult to know if the author(s) had good reasons to make the English and French versions so different and if the differences should be respected in the automatic generation process. For aeronautic maintenance procedures, controlled languages — in particular AECMA/AIA *Simplified English* and GIFAS *Rationalised French* — provide useful guidances, which help to identify the relevant differences for multilingual generation. The lexical differences reported in the next sections will be systematically evaluated from a controlled language perspective. This does not mean that controlled languages should be considered as "absolute" refrences. We will see that the writing rules defining these languages are sometimes too general.

[2] A Meaning-Text model consists of the grammar and the lexicon of a particular language.

[3] We mean by grammatical realisation the following (main) linguistic operations: (1) transition from deep syntactic representation to surface syntactic representation, (2) linearisation of the surface syntactic representation and (3) surface morphology.

4 Operator verbs

Our corpus analysis reveals that a precise account of *operator verbs* is required. This lexical class encloses semantically poor items like *do*, *carry out* in English and *effectuer*, *procéder* in French, which are combined with predicative nouns to form complex predicates. For example, in sentence (1F), the operator verb *procéder* takes as its direct object the predicative noun *remplissage* which, in some way, denotes the action to be performed:

(1F) ***Procéder au remplissage du réservoir hydraulique.***
(Lit. 'Proceeds with the filling of the hydraulic reservoir.')

Operator verb constructions have already been studied from a machine translation perspective [5]. Such constructions raise an interesting problem for MT because they cannot be translated in a purely compositional manner. For example, a compositional English translation of the sentence "*John a posé une question à Mary*" would lead to the incorrect sentence "*John put a question to Mary*", whereas the correct (or the more closely related) translation would be "*John asked Mary a question*". To make the appropriate translation, an MT system should be able to identify in the initial sentence the semi-idiomatic expression *poser une question* and consequently build a sentence based on the equivalent English expression *ask a question*. Besides, the equivalent expression in the target language does not always exist, which means that even more complex correspondences should be found. The literal translation associated to sentence (1F) illustrates this point. We can hardly get an acceptable English translation if we want to preserve the *structure* of the French instruction. The English equivalent of (1F) found in the corpus is based on the verb *fill* which takes as direct object the translation of the argument of the predicative noun *remplissage* in (1F):

(1E) ***Fill** the hydraulic reservoir.*

French and English instructions often diverge on this aspect. Operator verbs are exceedingly common in the French versions. We have found many pairs of bilingual instructions where the French instruction is based on an operator verb construction and the English instruction on a simple verb. Here are some excerpts which illustrate this regularity:

(2E) ***Bleed** suction lines.*
(2F) ***Effectuer la purge** du circuit d'aspiration.*
(Lit. 'Carry out the bleeding of suction lines.')

(3E) ***Change** the hydraulic fluid.*
(3F) ***Effectuer le renouvellement** du liquide hydraulique.*
(Lit. 'Carry out the renewal of hydraulic liquid.')

(4E) *Carefully **clean** the filter body.*
(4F) ***Effectuer un nettoyage** soigné du corps du filtre.*
(Lit. 'Carry out a careful cleaning of the filter body.')

It is important to note that, in many cases, these French instructions can be paraphrased by sentences based on simple verbs. For example, sentence (2F) can be paraphrased by the sentence based on the verb *purger*, directly related to the predicative noun used in (2F):

(2F') ***Purger** le circuit d'aspiration.*

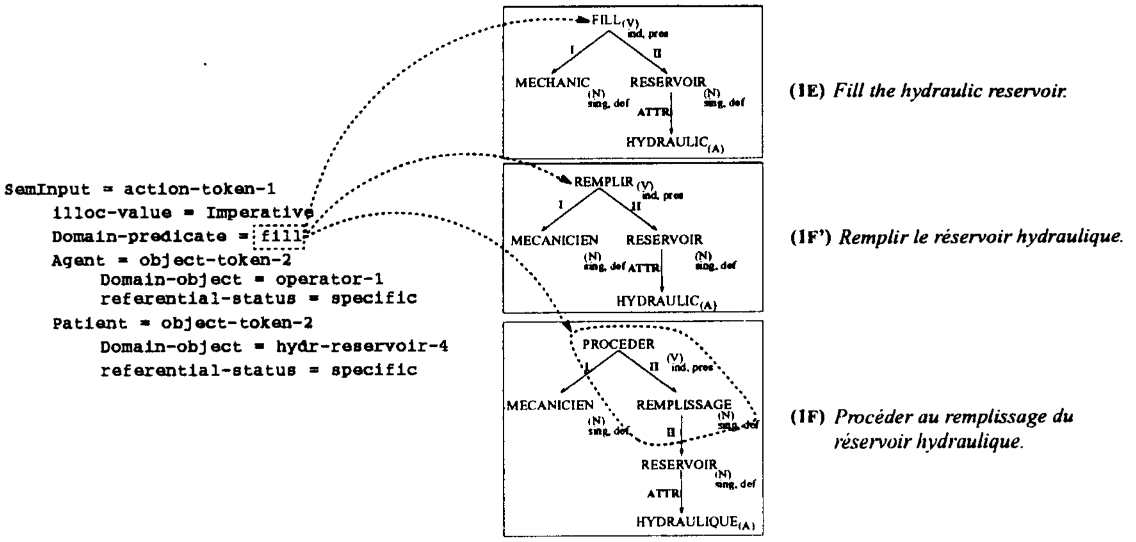
((2F') is the closest translation of the English version (2E))

This remark holds for all the examples given above. The choice of operator verbs is often a consequence of technical writers'stylistic preferences. However, as shown by the literal translations, stylistically inadequate sentences would result if this preference were equally applied for English.

Simplified English and Rationalised French suggest to restrict the use of operator verbs, assuming that verbs *that directly show the actions* make maintenance instructions clearer. However, operator verbs cannot always be avoided, even in English. Consider the following pair:

(5E) ***Gain access** to rear compartment.*
(5F) ***Accéder** à la soute arrière.*

We can hardly find an acceptable paraphrase of (5E) built on a simple verb. We will also show later that sometimes operator verbs cannot be avoided when some attributes of the action to be performed should be conveyed explicitly.



```
SemInput = action-token-1
  illoc-value = Imperative
  Domain-predicate = fill
  Agent = object-token-2
      Domain-object = operator-1
      referential-status = specific
  Patient = object-token-2
      Domain-object = hydr-reservoir-4
      referential-status = specific
```

Figure 1: An illustration of operator verb/simple verb selections.

4.1 Operator verb constructions in the lexicalisation process

The sentence generator should be able to generate multilingual pairs of instructions similar to the excerpts (2), (3) and (4), by selecting an operator verb construction for one element of the pair and a 'simple verb construction' for the other element. For this kind of differences, the French and English lexicalisations rely on the same basic mechanisms. However, the way these basic mechanisms are combined is language-specific.

Let us look more closely at the pair (1)[4] and at the lexicalisation process required to produce such sentences. Surface realisation starts with the following input representation:

```
SemInput = Action-token-1
  Illoc-value = Imperative
  Domain-predicate = fill
  Agent = object-token-2
     Domain-object = operator-1
     Referential-status = specific
  Patient = object-token-3
     Domain-object = hydr-reservoir-4
     Referential-status = specific
```

This structure represents an imperative illocutionary act. Its propositional content is an action of type **fill** which has two arguments **Agent** and **Patient**. The figure 1 illustrates potential correspondences between this input representation and the deep syntactic representations required to derive sentences (1E), (1F'), and (1F) after grammatical realisation. The dotted arrows indicate the possible lexical mappings of the conceptual predicate **fill**. The English realisation and the first French option (1F') rely on a simple correspondence between the predicate **fill** and corresponding verbs (*fill* and *remplir*). By contrast, the second French option is based on a complex correspondence between the predicate **fill** and a multi-lexemic structure *procéder* $\xrightarrow{II}$ *remplissage*.

To deal with this lexical phenomenon, two lexicalisation rules are involved. These rules may roughly be described as follows. Given the input representation[5]:

```
SemInput = action-token
         Illoc-value = Imperative
         Domain-predicate = P
         Agent = x₁
         Patient = x₂
         ...
         Roleₙ = xₙ
```

[4] (1E) *Fill the hydraulic reservoir.*
(1F) *Procéder au remplissage du réservoir hydraulique.*

[5] For sake of clarity, we consider that the illocutionary value is always imperative since we strictly focus in this paper on the *instructional* parts of the procedures. This illocutionary value does not affect the lexicalisation of the proposition, i.e the construction of the deep syntactic tree. However, it has an effect on grammatical realisation, such as erasing the subject during the transition to surface syntactic level.

r_1: Simple Verb Construction

1. Look in the *concept - lexeme mapping structures* for a correspondence P $\Longrightarrow$ V.
2. Lexicalise the arguments x_1, ..., x_n and link the resulting lexemic structures to V.

r_2: Operator Verb Construction

1. Look for a mapping structure P $\Longrightarrow$ N.
2. Look in the lexical entry of N for a verb V such that V = **Oper**$_1$(N).
3. lexicalise x_1 and link the resulting lexemic structure to V by means of a deep syntactic relation I.
4. Link N to V by means of a relation II.
5. Lexicalise the remaining arguments x_2, ..., x_n and link the resulting lexemic structures to N.

Several remarks should be made about these rules:

• To link predicative lexemes to their dependents (i.e. realisations of arguments), correspondences between conceptual roles and deep syntactic relations (I, II, ..., IV) are specified in the lexical entry of each verb and predicative noun. Hence, a conceptual-lexeme mapping structure indicates not only which lexeme(s) can be used to express a concept but also how the roles of the concept should be realised in terms of deep syntactic relations.

• In a MTT-like lexicon, predicative nouns are linked to their operator verbs by means of the lexical functions **Oper**$_1$, **Oper**$_2$, ...(for example, **Oper**$_1$(*remplissage*) = *procéder*). The number designates the actant of the predicative noun which is promoted as first actant (syntactic subject) of the operator verb. In the procedures we have analysed, only the **Oper**$_1$ function seems to be relevant.

• The rule r_2 maps a single concept P to a multi-lexemic structure composed of an operator verb governing a predicative noun. However, this correspondence is not given as such in the lexicon. It appears more natural to consider that the lexical realisation performed by rule r_2 relies primarily on a correspondence between the predicate P and the predicative noun. It should also be mentioned that such basic correspondences can also be exploited to generate similar phrases in other types of constructions. For example, the correspondence `fill` $\Longrightarrow$ *remplissage*, used by the rule r_2 when generating the sentence (1F) can also be used to construct the nominalisation *le remplissage de l'accumulateur* in the declarative sentence:

(6F) ***Le remplissage de l'accumulateur doit provoquer l'allumage du voyant sur le tableau hydraulique.***
(Lit. 'The replenish of the accumulator should cause the warning light to come on on the hydraulic panel.')

• The lexicalisation of arguments involves other mechanisms, which concern in particular the construction of referring expressions [3].

• An appropriate generation of multilingual instructions in accordance with these lexical differences can be achieved by assigning priorities to these rules. In English, r_1 should be privileged and r_2 applied only if r_1 fails. For example, this last case would occur when generating sentence (5E)[6]. r_1 would fail because the lexicon does not contain a mapping structure relating the atomic predicate **gain_access** and a simple verb. In French, it is, however, difficult to assign absolute priorities in the same way, since we can find both types of constructions in similar contexts. If stylistic preferences observed in the corpus have to be reflected in the automatically generated texts, a reasonable solution would be to select indifferently one of these rules. Notice that Rationalised French, which is not respected in the procedural texts we have analysed, will assign a higher priority to r_1, resulting in an identical parameterisation of the lexicalisation mechanisms for both languages.

4.2 The problem of complex actions

We have assumed so far that actions to be verbalised can be represented by simple predicate - argument structures. However, actions may have attributes (manner, temporal constraints,

[6] (5E) *Gain access to rear compartment.*

...) which should be conveyed explicitly. In general, the two types of constructions represented by rules r_1 and r_2 are possible, even when some attribute of the action should be realised at surface level. For example, in (4F)[7] the manner attribute of the *cleaning action* is expressed as an adjective since this action is nominalised. But if the same action were expressed as a verb the manner attribute would take the form of an adverbial modifier:

(4F') *Nettoyer soigneusement le corps du filtre.*
(Lit. 'Carefully clean the body of the filter.')

To deal with such modifiers, a minor extension of rules r_1 and r_2 is required. The rules should be able to introduce modifiers on the 'main' predicative element of the sentence, i.e. the main verb in r_1 and the direct object of the operator verb (the predicative noun) in r_2:

- In r_1: an attribute of the action will be realised as an adverb linked to the main verb V by means of an attributive deep syntactic relation (ATTR).
- In r_2: the attribute will be realised as an adjective which linked to the predicative noun N with an attributive relation.

The problem is that sometimes these attributes cannot take an adverbial form and in the analysed procedural texts, it seems that this limitation is an important motivation for using operator verbs. They provide the ability to introduce such attributes in an adjectival form. Consider the following pair:

(7E) *Carry out a dry ventilation of the reactor.*
(7F) *Effectuer une ventilation sèche du réacteur.*

From both English and French versions, we cannot derive in a simple way equivalent expressions based on a simple verb because of the adverbial modifiers:

(7E') **Ventilate drily the reactor.*
(7F') **Ventiler sèchement le réacteur.*

A key problem for text generation is to be able to avoid such incorrect sentences. This problem has already been tackled in [15]. Meteer proposes to express the input semantic content in terms of ***abstract linguistic resources***, i.e. semantic categories, which prevent incorrect combinations of ***concrete linguistic resources*** during surface realisation. Following Meteer's analysis, the lexeme ***dry*** in (7E) denotes a ***property*** which cannot be realised if an ***event perspective*** is taken on the predicate. This constraint enforces the nominalisation of the action. By contrast, an attribute of category *manner* can be combined with both event and object perspectives. This explains why (4F) and (4F') are both acceptable. In many cases, the characterisation of attributes along the semantic opposition *manner/property* explains the acceptability or inacceptability of the "adverbial forms". However, this characterisation is not always straightforward and it appears that more precise oppositions should be introduced.

5 Specificity level of verbal items

Another important lexical difference concerns the specificity level of each element of the bilingual pairs. A French instruction may be less specific because a conceptual argument has been left implicit while explicitly realised in the equivalent English instruction. However, even when both instructions are at the same specificity level, differences may appear in the way semantic content is spread over the lexical material. This is mainly due to the fact that verbs available in both languages do not necessarily cover the same part of the initial content.

We will focus on three types of lexical divergences which are frequent in the analysed procedures:

1. Domain-specific *vs* ordinary verb

The two verbs have *similar* argument structures but one of them belongs to the technical jargon of the domain.

(8E) ***Unlock valve clapper nut.***
(8F) ***Défreiner l'écrou du clapet de valve.***

The verbs *unlock* and *défreiner* have a very close meaning, but the second one is domain specific and imposes more constraints on its second argument (the direct object). For example, the English sentence *unlock the door* is acceptable but not the French one *Défreiner la porte.*

[7] (4F) *Effectuer un nettoyage soigné du corps du filtre.*

2. Specific *vs* general verb

One of the two verbs has a more specific meaning:

(9E) ***Charge the accumulator with nitrogen.***
(9F) ***Gonfler l'accumulateur à l'azote.***
(Lit. 'Inflate the accumulator with nitrogen.')

The choice of a more general verb for the English version is purely stylistic since a specific verb — *inflate* — exists, as shown in the literal translation of (9F). We have found several divergences of this kind, which seem to be stylistically motivated. [19] describes similar divergences between English and German instructions.

Notice that, with respect to Simplified English, sentence (9E) is not acceptable, since specific verbs have to be prefered when available.

We will see in section 5.1 that, interestingly, instructions can be made more precise with general verbs because of differences in argument structures: a general verb may have a more extended argument structure than a specific one.

3. Ordinary *vs* denominal verb

The two verbs have distinct argument structures. One of them, in general the English one, *incorporates* an argument which is expressed at surface level in the French version. Such argument incorporation is often realised through the use denominal verbs which are much more frequent in English procedures:

(10E) ***Jack up the aircraft.***
(10F) ***Mettre l'avion sur vérins.***
(Lit. 'Put the aircraft on jacks.')

The verb *jack up* has no direct equivalent in French. Hence, the French version has to rely on a general verb and the *locative* argument should be realised at surface level. In the corpus, denominal verbs are systematically used in the English versions (when they are available) even though this choice leads to bilingual pairs with quite different lexical structures. Such verbs ensure conciseness and, sometimes, the lack of denominal verbs in French makes the French version much longer. It should be stressed that, in general, both instructions are at the same specificity level, even though one of them appears more complex.

5.1 Consequences for the lexicalisation mechanisms

1. Let us start with the first type of differences, domain-specific *vs* ordinary verb. The corpus shows that domain-specific verbs are often prefered over ordinary verbs. A plausible motivation of this preference is that, as illustrated by example (8)[8], they impose precise selectional restrictions on the arguments. The important point for multilingual generation is that the absence of a domain specific verb in one language does not affect lexicalisation in the other one (i.e., a specific verb will be used if available).

2. The second type of differences is a more complex issue. Both Simplified English and Rationalised French include a writing rule which says that specific words should be prefered over general words. This rule can be used as a guiding principle in the verb selection mechanisms. However, it is not always sufficient in order to reach the appropriate specificity level required for the instruction. Selecting a more specific verb does not necessarily lead to a more specific instruction. A verb may have a precise meaning but a restricted argument structure which may force to leave implicit some part of the initial content. To illustrate this point, let us compare the following surface realisations of the same instruction:

(11E) ***Remove lockwire from filter bowl.***

(11E') ***Unlock the filter bowl.***

The verb *unlock* is more specific than *remove*, but the locking device to be removed is not specified as a surface argument of the verb. By contrast, this argument can be made explicit with the verb *remove*. Which of these two versions can be considered more specific? (11E) seems more specific, for the 'unlocking' action, though incompletely specified by the main verb *remove*, is somewhat suggested by the argument *lockwire* (since, obviously, the function of a lockwire is to lock). Besides, it brings another information — the nature of the locking device — which cannot be expressed in (11E'). The integration in a text generation system of such evaluations of instruction specificity level is not a straightforward issue. Complex world

[8] (8E) *Unlock valve clapper nut.*
(8F) *Défreiner l'écrou du clapet de valve.*

knowledge and lexical semantic inferences are involved in these evaluations, and they require a deeper model of domain knowledge and precise semantic definitions of lexical items. At present, our approach is less ambitious. We take advantage of the simple heuristic: "*the more arguments a verb has, the more specific the resulting instruction*" in order to detect potential conflicts. This ability of detecting lexical options may help to perform rephrasing operations. For example, if sentence (11E') is generated first, considering that more specific verbs should be privileged, a rephrasing request would cause the generator to propose an alternative realisation based on the general verb *remove* which allows to express at surface level the argument left implicit in the first proposal. According to our corpus, this kind of rephrasing operations will normally concerns only the English versions, since in the French procedures specific verbs are systematically prefered.

Let us now describe briefly how these functionalities are concretely integrated in the lexicalisation component. The generation of an instruction based on a specific verb involves the rules r_1 and r_2 (see section 4.1)[9]. These rules make correspondences between the conceptual predicate of the action and a specific lexical item. The choice of a more general verb relies on the same rules but the generation process will proceed from a transformed input representation built on a superordinate predicate.

For instance, to produce sentence (11E')[10], lexicalisation will proceed from the following representation, provided that the mapping structure `remove-locking-device` ⟹ *unlock* is given in the lexicon:

```
SemInput = Action-token-1
  Illoc-value = Imperative
  Domain-predicate = remove-locking-device
  Agent = object-token-2
     Domain-object = operator-1
     Referential-status = specific
  Patient = object-token-3
     Domain-object = lockwire-4
     Referential-status = specific
  Location = object-token-4
     Domain-object = filter-bowl-5
     Referential-status = specific
```

At the deep syntactic level, only arguments **Agent** and **Location** will be realised as actants of the verb *unlock* (**Agent** as actant I and **Location** as actant II). The generation of sentence (11E)[11] will proceed from an input representation based on the superordinate conceptual predicate **remove** with the same arguments. The predicate will be directly linked to the verb *remove* as specified in the lexicon and the three arguments will be realised at the deep syntactic level.

3. As we have already said, the use of denominal verbs often causes differences between the French and English versions of instructions, since they are usually not available in French. Besides, even when they are available they are not systematically used as in the English versions, as attested by the following example:

(12E) ***Pressurise the hydraulic system.***

(12F) ***Mettre le circuit hydraulique sous pression.***

(Lit. 'Put the hydraulic system under pressure.')

The sentence (12F') based on the denominal verb *préssuriser* and which is equivalent to (12F) is also present in the corpus:

(12F') ***Pressuriser le circuit hydraulique.***

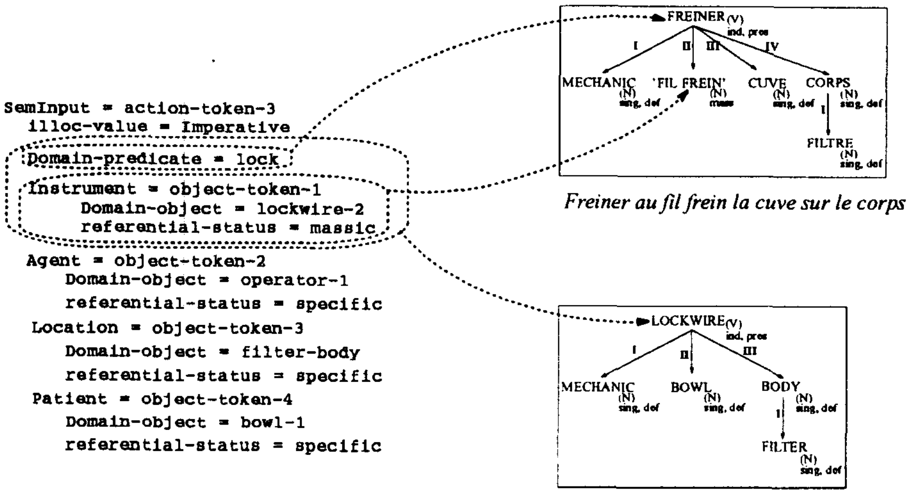
The lexicalisation rules defined so far perform mappings between a single concept (the predicate) and one or several lexemes. By contrast, the selection of denominal verbs involves mappings between several concepts and a single lexeme. A denominal verb covers not only the main predicate but also an argument of the predicate. In the example given in figure 2, the French and English versions are derived from the same conceptual representation. The French version results from a one to one mapping between concepts of the input representation and lexemes. In particular, the predicate `lock` is directly mapped to the verb *freiner* and the argument **Instrument** to the phraseme *'fil frein'*. The generation of such sentences relies on rules r_1 and r_2. However, in the English version, it is the combination of the predicate `lock` and the argument **Instrument** which is mapped to the main verb *lockwire*.

To ensure such correspondences, an additional

[9] And also the rule r_3 dedicated to the selection of denominal verbs and which will be defined later.

[10] (11E') *Unlock the filter bowl.*

[11] (11E) *Remove lockwire from filter bowl.*



Freiner au fil frein la cuve sur le corps du filtre.

Lockwire bowl on filter body.

Figure 2: In the English version, the predicate and the instrument argument are mapped to a denominal verb.

rule is required:

Given the input representation:

```
SemInput = action-token
          Illoc-value = Imperative
          Domain-predicate = P
          Agent = x₁
          Patient = x₂
          ...
          Roleₙ = xₙ
```

r_3: Argument Incorporation

1. Look in the concept – lexeme mapping structures for a correspondence $\mathtt{P} + \mathtt{x_i} \Longrightarrow \mathtt{V},\ \mathtt{i} \in \{1, ..., \mathtt{n}\}$.
2. Lexicalise the remaining arguments and link the resulting lexemic structures to V.

To be consistent with the lexical preferences observed in the corpus, this rule should have the highest priority.

The incorporated argument does not always hold the same semantic role. For example, it can be the *instrument* as in the verbs *lockwire, energise* and *pressurise* or a *locative* argument as in the verb *jack up*. It should also be mentioned that such *incorporations* are not restricted to arguments. [19] discusses closely related phenomena concerning German, English and French instructions. The authors provide in particular some examples where a manner attribute is realised as an adverb in English while incorporated in the verb in German and French[12].

6 Conclusion

We have focused in this paper on some frequent lexical differences between French and English instructions. We have also proposed a specification of lexicalisation mechanisms, without introducing distinct semantic representations for French and English lexicalisations. We do not claim however that distinct representations can always be avoided. Our corpus reveals the existence of *deeper* differences (though less frequent) which call for language-specific representations. For example, we have found several instructions where aspectual values are conveyed explicitly in French but not in English. Another interesting case concerns *agentivity values* assigned to the operator in the instructions. Consider the following example:

(13E) ***Allow hydraulic pressure to fall.***

(13F) ***Chuter la pression hydraulique.***

(Lit. 'Decrease hydraulic pressure.')

In (13E), the operator is presented as the *enabler* of a *physical process*, whereas in (13F), he

[12] For example:
(E) ***affect adversely*** - (G) ***beeinträchtigen*** - (F) ***amoindrir***

is presented as the *causer* of an *action*. It seems that the generation of such a bilingual pair requires language-specific semantic inputs built on distinct event categories. Interestingly, we have noticed that controlled languages will not, in most cases, allow these deeper differences to appear. One of the pair is often rejected by the corresponding controlled language. For example, (13E) does not comply with Simplified English, which would encourage the use of the more direct form: ***Decrease the hydraulic pressure.*** This last sentence is closer to (13F) and we can reasonably suppose that these two sentences can be generated from the same input.

Acknowledgments

I would like to thank Alexis Nasr, Corinne Fournier, and Owen Rambow for helpful comments on ealier versions of this paper.

References

[1] L. Bourbeau, D. Carcagno, E. Goldberg, R. Kittredge, and A. Polguère. Bilingual synthesis of weather forecasts in an operations environment. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING'90)*, Helsinki, 1990. COLING-90.

[2] M. Boyer and G. Lapalme. Generating paraphrases from meaning-text semantic networks. *Computational Linguistics*, 1:103–117, 1985.

[3] F. Cerbah. Referring Expressions in GhostWriter. Technical report, Dassault Aviation - British Aerospace, 1995.

[4] F. Cerbah and C. Fournier. The syntactic component of the GLOSE generation system. Technical report, Dassault Aviation, 1995.

[5] L. Danlos. Support verb constructions: linguistic properties, representation, translation. *French Language Studies*, (2):1–32, 1992.

[6] J. Delin, A. Hartley, C. Paris, D. Scott, and K. Van Linden. Expressing procedural relationships in multilingual instructions. In *Proceedings of the Seventh International Workshop on Natural Language Generation*, Kennebunkport, Maine, 1994.

[7] A. F. Hartley and C. L. Paris. Supporting Multilingual Document Production: Machine Translation or Multilingual Generation? In *IJCAI Workshop on Multilingual Text Generation*, pages 34–41, Montréal, 1995.

[8] U. Heid. Notes on the use of lexical functions for the description of collocations in an NLP lexicon. In *International Workshop on the Meaning-Text Theory*, pages 217–229, Darmstadt, 1992.

[9] D. Heylen, L. Humphreys, S. Warwick-Armstrong, N. Calzolari, and S. Murison-Bowie. Collocations and the lexicalisation of semantic operations — lexical functions for multilingual lexicons. In *International Workshop on the Meaning-Text Theory*, pages 173–183, Darmstadt, 1992.

[10] L. Iordanskaja, R. Kittredge, and A. Polguère. Lexical selection and paraphrase in a meaning-text generation model. In C. Paris, W. Swartout, and W. Mann, editors, *Natural Language Generation in Artifical Intelligence and Computational Linguistics*, pages 293–312. Kluwer Academic Publishers, 1991.

[11] R. Kittredge. Efficiency vs. Generality in Interlingual Design. In *IJCAI Workshop on Multilingual Text Generation*, pages 64–74, Montréal, 1995.

[12] L. Kosseim and G. Lapalme. Content and rhetorical status selection in instructional texts. In *Proceedings of the Seventh International Workshop on Natural Language Generation*, Kennebunkport, Maine, 1994.

[13] B. Lavoie. Interlingua for Bilingual Statistical Reports. In *IJCAI Workshop on Multilingual Text Generation*, pages 84–93, Montréal, 1995.

[14] I. A. Mel'čuk. *Dependency Syntax: Theory and Practice*. State University of New York Press, New York, 1988.

[15] M. W. Meteer. Bridging the generation gap between text planning and linguistic realization. *Computational Linguistics*, 7(4), 1991.

[16] O. Rambow and T. Korelsky. Applied Text Generation. In *Third Conference on Applied Natural Language Processing*, pages 40–47, Trento, Italy, 1992.

[17] E. Reiter, C. Mellish, and J. Levine. Automatic generation of on-line documentation in the IDAS project. In *Proceedings of the Third Conference on Applied Natural Language Processing (ANLP-1992)*, pages 64–71, Trento, Italy, 1992.

[18] D. Rösner and M. Stede. Customizing RST for the automatic production of technical manuals. In R. Dale, E. Hovy, D. Rösner, and O. Stock, editors, *Aspects of Automated Natural Language Generation*, Lecture notes in Artificial Intelligence 587, pages 199–214. Springer Verlag, Berlin, 1992.

[19] M. Stede and B. Grote. The lexicon: Bridge between language-neutral and language-specific representations. In *IJCAI Workshop on Multilingual Text Generation*, pages 129–135, Montréal, 1995.