A MORPHOLOGICAL PROCESSOR FOR MODERN GREEK

Angela Ralli
- Université de Montréal, Montréal,
  Québec, Canada
- EUROTRA - GR,
  Athens, Greece

Eleni Galiotou
- National Documentation Center Prj.,
  National Hellenic Research Foundation,
  Athens, Greece
- EUROTRA - GR,
  Athens, Greece

ABSTRACT

In this paper, we present a morphological processor for Modern Greek.

From the linguistic point of view, we try to elucidate the complexity of the inflectional system using a lexical model which follows the recent work by Lieber, 1980, Selkirk 1982, Kiparsky 1982, and others.

The implementation is based on the concept of "validation grammars" (Courtin 1977).

The morphological processing is controlled by a finite automaton and it combines

a. a dictionary containing the stems for a representative fragment of Modern Greek and all the inflectional affixes with

b. a grammar which carries out the transmission of the linguistic information needed for the processing. The words are structured by concatenating a stem with an inflectional part. In certain cases, phonological rules are added to the grammar in order to capture lexical phonological phenomena.

1. Introduction-Overview

Our processor is intended to provide an analysis as well as a generation for every derived item of the greek lexicon. It covers both inflectional and derivational morphology but for the time being only inflection has been treated.

Greek is the only language tested so far. Nevertheless, we hope that our system is general enough to be of use to other languages since the formal and computational aspect of "validation grammars" and finite automata has already been used for French (c.f. Courtin et al. 1976, Galiotou 1983).

The system is built around the following data files:

1.A "dictionary" holding morphemes associated to morpho-syntactic information.

2.A "model" file containing items which act as reference to every morphematic entry in order to determine what kind of process the entry undergoes.

3.A word grammar which governs permissible word structures. The rules that can apply to an entry are divided in

a. a "basic initial rule" acting as a recogni-

tion process.

b. The validation rules that determine all possible combinations of the entry with other morphemes.

4. A list of phonemes described as sets of features. The same file contains also a set of phonological rules generating lexical phonological phenomena. These rules govern permissible correspondences between the form of entries listed in the dictionary and the form they develop when they are combined in sequences of morphemes.

These files are used both for analysis and generation. The process of the present morphological analysis consists of parsing an input of inflected words with respect to the word grammar. Stems associated to the appropriate morpho-syntactic information will be the output of the parsing.

The process of generation of a given inflected word consists of

a. determining its stem by a morphological analysis.

b. Generating all or a subset of the permissible word forms.

For the needs of this presentation, lexical items have been transcribed in a semi-phonological manner. According to this transcription, all greek vowels written as double character are kept as such:

(1)   Graphemes   Phonemes

εɩ → ei   [i]
οɩ → oi   [i]
αɩ → ai   [e]
ου → oy   [u]

Moreover, the sounds [i] and [o] written in Greek as η and ω respectively are transcribed as i: and o:. The transcription of the last two vowels reminds of their ancient greek status as long vowels.

As far as accent is concerned, we decided to exclude this aspect from the present form of the processor. Accentuation in Greek is a linguistic problem which has not been solved as yet. We are working on this matter and we hope to implement accent in the near future.

The morphological processing is controlled by a finite automaton[1] with the help of the dictio-

----
[1] For a detailed discussion on the control automaton, c.f. Courtin et al 1969.

nary and the word grammar which controls word formation and carries out the transmission of the linguistic information needed for the processing. In certain cases, the grammar makes use of phonological rules in order to capture lexical phonological phenomena such as insertion, deletion and change.

The processor is implemented in TURBO-PROLOG (version 1.0) running under MS-DOS (version 3.10) on an IBM-XT with 640 kB main memory. It consists of an analysis and a generation sub-module.

2. Linguistic assumptions

The theoretical framework underlying the linguistic aspects of the project is that of Generative Morphology, in particular the recent work by Lieber 1980, Selkirk 1982, Kiparsky 1982 and others.

In developing our system, we have adopted the proposals made in Ralli's study on Greek Morphology (Ph.D.diss., 1987). Therefore, we assume that the greek lexicon contains a list of entries (dictionary) and a grammar which combines morphology with phonology. The dictionary is morpheme based. It contains stems and affixes which are associated with the following information fields.

a. The string in its basic phonological form.

b. Reference to possible allomorphic variations of the string which are not productively generated by rule.

c. Specifications of grammatical category and other morpho-syntactic features that characterize the particular entries.

d. The meaning.

e. Diacritic marks which are integers permitting the correct mapping between the stem and the affix where this cannot be done by rule.

(1) Stem   Affix

   $vivli_3$ + o_3 "book" (neut, nom,sg)

   $krat_4$ + os_4 "state" (neut,nom,sg)

In our work, diacritic marks replace the traditional use of declensions and conjugations which fail to divide nouns and verbs in inflectional classes.

The inflectional structure of words is handled by a grammar which assigns a binary tree structure to the words in question. The rules are of the form

(2) Word → stem Infl,

where, Word and stem are lexical categories and Infl indicates the inflectional ending. For nominal stems, Infl corresponds to a single affix marked for number and case.

(3) Infl → affix

   Example: ∂romos → ∂rom-os (nom, sg)
                "street"

For verbs, the constituent Infl refers either to one or to two affixes. In the latter case, two

affixes belong to the endings of verbal types that are aspectually marked.

(4) Infl → affix Infl

Example: γrapsame → γrap   s   ame
         "we wrote" "write" [perf]   [1P
                                        pl
                                        past]

Note that the stem γrap is listed in the dictionary as γraf. The consonant [f] is changed to [p] because of the [s] that follows. The phonological rule in question is lexical and it applies to the morpheme boundary. As such, the rule is morphologically conditioned and it allows exceptions.[2]

When verbal types do not contain an aspectual marker, Infl refers to a single affix.

3.1 The dictionary structure

In our system, the dictionary consists of a sequence of entries each in the form of a Prolog term.

It has to be noted that no significant semantic information is present in our entries because that field is still unexploited. Similarly, the syntactic information concerning subcategorization properties of lexical entries is not taken into account.

The dictionary also contains information that permits the "linking" with the grammar. So, apart from the linguistic information mentioned in section 2, every entry of the dictionary contains also

a. a list of rules that permit the use of a particular entry (rules that have the entry as their terminal symbol).

b. a list of validation rules (rules that can be applied after each use of that entry).

As far as morphology is concerned, forms can be arranged into classes. We choose arbitrarily an element of this class called a "model" and every stem in the dictionary refers to a model. Morphological information is found at the model level. In this way, the size of the dictionary is significantly reduced.

The model file consists also of sequences of entries, each in the form of a Prolog term. Each model includes information concerning

a. the form of the string,

b. the "basic initial rule" which identifies the string,

c. the possible diacritic mark,

d. the set of morpho-syntactic features,

e. the validation rules which substitute word formation rules.

3.2 Examples from the dictionary

Example of a dictionary entry:
----
[2]For a detailed study of lexical phonological rules, c.f. Kiparsky 1982/83.

	Stem	Model	List of
			allomorphs

```
dict("paraθyr",  "vivli",    []).
      "window"    "book"
```

Model entry of the example above

Entry	B.In.R.	Diac.	Feat.	Valid.
stem ("vivli",	[init],	[3],	[n,neut],	[nl1,nl2]

We did not write separate dictionary entries for affixes because each affix is a model on its own. Therefore, information associated with an affix model must cover all unpredictable information listed within the corresponding dictionary entry. Instead of a "basic initial rule", every affix model refers to a set of rules that govern the combination of the affix with a particular stem. An affix that terminates a word is identified by an empty set of validation rules.

Example of an affix model

Entry	Rules	Diac.	Feat.	Val.
af("o",	[nl2,a4],	[3],	[nom,sg],	[])

4. The grammar

In order to carry out the processing we use a "validation grammar" as defined in Courtin 1977.

4.1 Review of validation grammars

A validation grammar GV is a 4-tuple
$GV=(V_{TV}, SV, \mathcal{G}V, E)$, where,

V_{TV} = a vocabulary of terminal symbols.

E=a subset of the set of integers.
$SV \in \mathcal{G}(E)$ and is called axiom

$\mathcal{G}V$=a finite set of production rules.

A production is an element of the application

$$E \rightarrow V_{TV} \times \mathcal{G}(E)$$

Productions are of the form

$i \rightarrow a[j_1,\ldots,j_q]$ or
$i \rightarrow a[\emptyset]$, where $i \in E$,
$\quad [j_1,j,\ldots,j_q] \in \mathcal{G}(E)$, $a \in V_{TV}$

Property 1

A validation grammar is equivalent to a regular grammar since they generate the same language. Consequently, there is a finite automaton that recognizes the strings generated by a validation grammar.

Property 2

The number of production rules of a validation grammar is less than or equal to the number of production rules of its equivalent regular grammar.

4.2 Control, transmission and phonological changes

Control is carried out with the help of valida-

tions which are redefined after the application of each rule. In our system, validation rules consist of a list of Prolog clauses.

Transmission concerns the grammatical category and other morpho-syntactic features.

Linguistically, we regard stems to be the head of inflected words. As such, they contribute to the categorial specifications of the words. Moreover, all morpho-syntactic features of inflectional affixes are also copied to the word. In word structures built in the form of a tree, features are percolated to the mother node according to the Percolation Principle as it was formulated by Selkirk.

(1) Percolation Principle (Selkirk 1982)

a. If a head has a feature specification $[aFi]$, $a\neq u$, its mother node must be specified $[aFi]$ and vice versa.

b. If a non head has a feature specification $[\beta fj]$ and the head has the feature specification $[uFj]$, then the mother node must have the feature specification $[\beta Fj]$. (page 76).

The principle in question is incorporated in our validation rules where, for each inflected word, it is determined which features are taken from the stem and which come from the affix.

(2) Example of a validation rule

```
rule(nl1,Stem,_,StFeat,_,
    Affix,[],AfFeat,AfVal
    Result,[],ResFeat,AfVal):-
    concat(Stem,Affix,Result),
    append_list(StFeat,AfFeat,ResFeat)
```

where, "concat" is a Prolog predicate performing the concatenation of two strings and "append list" is a Prolog predicate performing the concatenation of two lists.

However, according to Ralli's study, features are not only percolated to words from stems and affixes. Feature values may also be inserted to certain underspecified environments. For instance, when an inflected word fails to take certain features from both the stem and the ending, the rule then takes over the role of adding them. Consider the verbal form γrafo: "I write". It takes the category value from the stem (γraf-) and the features of person and number from the affix (-o:). It is clear that at this point, γrafo: is underspecified because besides the values of person and number, greek verbal forms must be characterized by aspect, tense and voice. Following this, we assume that specific values of the last three attributes are inserted by the rule governing the combination of the stem γraf- with the ending -o:.

(3) Rule generating γrafo:

```
rule(vl1,Stem,[],StFeat,_,
    Affix,[],AfFeat, AfVal,
    Result,[],ResFeat, AfVal):-
    Concat(Stem,Affix,Result),
    feat_ins(StFeat,[non_perf,present,
             active],AfFeat,ResFeat)
```

It is worth noting that a validation rule can also take into account instances of morpho-phonological phenomena.

4.2.1 Morpho-phonological insertion

In Greek, in several cases, transition elements appear at a morpheme boundary between two constituents (c.f.Ralli 1987). Both the insertion and the phonological form of the elements are always conditioned by the morphological environment.

Nominal as well as verbal inflection undergo morpho-phonological insertion depending on the kind of stem that is involved in the process. An example of morpho-phonological insertion is the verbal thematic vowel.

(1) Stem	Th.V.	Af	
γraf	o	mai	"I am written"
γraf	e	tai	"It is written"

Similarly, in certain nouns and adjectives, a vowel appears in singular, between the stem and the inflection.

(2) Stem	Th.V.	Af	
tami	a	s	"cashier"
foiti:t	i:	s	"univ. student"

Insertion is not the only morphophonological phenomenon.

4.2.2 Morpho-phonological change

As already mentioned in section 2, verbal inflection undergoes morphophonological changes on the stem and/or the affix during the construction of aspectually marked verbal types. Rules performing phonological changes are applied cyclically each time the appropriate lexical string is formed. Phonological rules take into account a list of phonemes described as sets of distinctive features. In our system, phonemes are listed as Prolog terms. Phonological rules are listed as Prolog clauses.

Take for example the form δe-s-ame "we tied". The stem δe- is listed in the dictionary as δen-. The validation rule authorizing the concatenation of δen- and -s- demands the application of a lexical phonological rule responsible for the deletion of the final |n|.

4.2.3 The augment rule

It is generally accepted that augment in Modern Greek must be considered as a phonological element introduced in the appropriate morphological environment. That is, an e- is prefixed to forms marked for past in which it is always accentuated. Given the fact that accentuation is not treated here, we decided to divide verbal stems in marked and unmarked for augment. Once a verbal item is built, the e- is added at the beginning of the form in singular and third person plural only if the stem carries the feature [aug].

In our system, the augment rule, listed also as a Prolog clause, is activated by validation rules authorizing the concatenation of a verbal stem and a verbal affix marked for past. The same rules

insert the feature value "active".

In this way, we obtain:

(1)	e-γraf-a	"I was writing"
	γraf-ame	"We were writing"

but not *e-γraf-ame

5. The Process

The analysis of a word form is carried out independently of its syntactic environment. Consequently, the analyzer will provide the set of all possible analyses.

In order to program and store the automaton,we perform a splitting of its transitions and each transition is represented by a rule.

(1) avli: "yard" (nom/acc singular)

dictionary entries

dict("avl","avl",[])

model entries

stem("avl",[init],[1],
[n,fem],[n11,n12,n21,n22,n23])

af(" ",[n21,n23,n32,n33,a21,a23],
[],[],[])

Transitions

Rule	String	Resulting string	Feat., Val.
init	"avl"	"avl"	cat=n gd=fem diac=[1] val=[n11,n12,n21, n22,n23]
n21	" "	"avli:"	cat=n gd=fem num=sg case=nom
n23	" "	"avli:"	cat=n gd=fem num=sg case=acc

The rule init starts the analysis by taking every information from the dictionary level. The stem "avl" is validated by rules n21 and n23, among others, which will also authorize the use of a ∅-affix. Moreover, they perform morpho-phonological insertion of the transition element -i: during the concatenation of "avl" and " ". The resulting string is avli: in both cases. These rules also perform feature insertions. Rule n21 inserts feature values [nominative] and [singular] while n23 inserts feature values [accusative] and [singular].

The analysis of the form avli: is completed in 27 hundredths of a second (cpu time).

As already mentioned the system is reversible. In order to generate all possible forms of avli: we apply all validation rules of the stem "avl" and thus we obtain:

```
"avl"   init                                              cat=n
                                                          gd=fem
                                                          case=nom
                                                          num=sg
                                                          string="avli:"

                                                                    Final
                                                                    state
        cat≠n
        gd=fem
        diac=[1]
     val=[n11,n12,n21,n22,n23]
        string="avl"

                           "  "  n23
                                        cat≠n
                                        gd=fem
                                        case≠acc
                                        num=sg
                                        string="avli:"
```
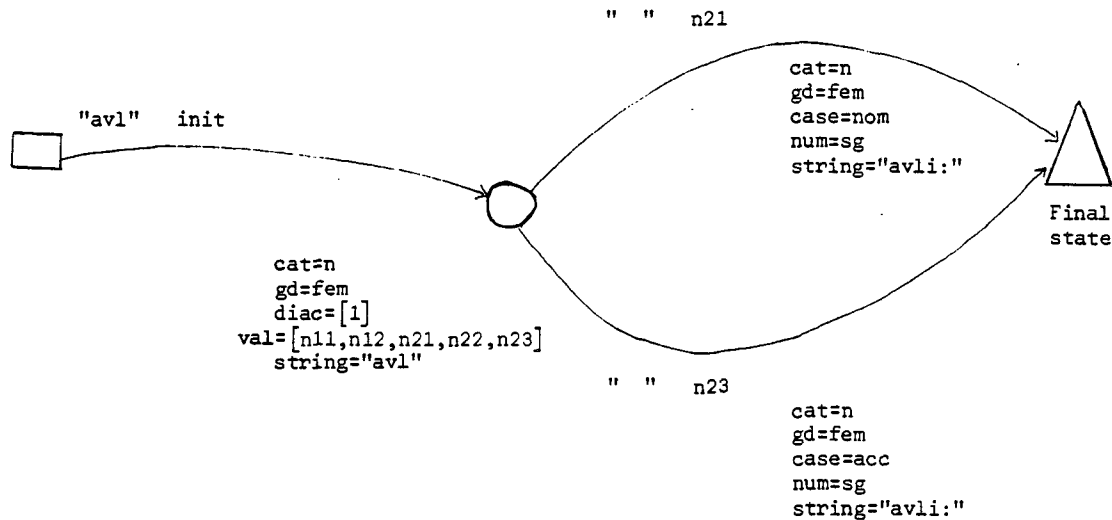
" " n21

Figure 1: Transition graph of the automaton

(2) avli: (fem,nom,sg)
     avli:s   (fem,gen,sg)
     avli: (fem,acc,sg)
     avles   (fem,nom,pl)
     avlo:n   (fem,gen,pl)
     avles   (fem,acc,pl)

The generation of all possible forms of avl-(i:)
is completed in 43 hundredths of a second (cpu
time).

As an example of processing of a verbal form
we mention the analysis of δe-s-ame "we tied"
discussed in section 4.2.2 which is completed in
60 hundredths of a second (cpu time), while the
generation of all possible forms of δen-(o:) "to
tie" is completed in 1 second and 59 hundredths
(cpu time).

5. Conclusion

In this paper, a morphological processor has
been presented that is capable of handling lexical
phonological phenomena. Future developments aim at
implementing a friendly user language and comple-
ting the user interface. We also plan to produce
an implementation under UNIX, probably in C,which
will hopefully become a component of an integrated
natural language processing system for Greek.

REFERENCES

Aronoff, M. 1976 Word Formation in Generative
Grammar, Linguistic Inquiry, Monograph 1., M.I.T.
Press

Babiniotis, G. 1972 The Greek Verb, Athens,
Greece

Chomsky, N. and M. Halle 1968 The Sound Pattern
of English, Harper and Row, New York

Courtin, J. 1977 Algrorithmes pour le traite-
ment interactif des langues naturelles, Thèse d'
Etat, Université de Grenoble I, Grenoble, France.

Courtin, J., Dujardin D. and Grandjean E. 1976
Editeur lexicographique pour les langues naturel-
les, Document Interne, I.R.M.A, Grenoble, France.

Courtin, J., Rieu J.L. and Szgall P. 1969 Un
métalangage pour l'analyse morphologique, Docu-
ment interne, C.E.T.A, Grenoble, France

Galiotou E. 1983 Construction d'un Analyseur
Morphologique du Français en Foll-Prolog, Mémoire
D.E.A., Université de Grenoble II, Grenoble,
France.

Kiparsky, P. 1982 Lexical Morphology and Pho-
nology, in Linguistic Society of Korea (Ed.),
Linguistics in the Morning Calm, Hanshin Publish-
ing Co, Seoul.
Kiparsky, P. 1983 Word Formation and the Lexi-
con, in F. Ingemann (ed.) Proceedings of the 1982
Mid-America Linguistics Conference, Univ. of Kan-
sas, Lawrence

Koutsoudas, A. 1962 Verb Morphology of Modern
Greek: a descriptive analysis, The Hague

Lieber, R. 1980 On the Organization of the Le-
xicon Ph. D. dissertation, M.I.T.

Malikouti-Drachman, A. 1970 Transformational
Morphology of the Greek Noun, Athens, Greece

Mohanan, K.P. 1982 Lexical Phonology, Ph. D.
dissertation, M.I.T.

30

Ralli, A. 1984 Verbal Morphology and the Theory
of Lexicon **Proceedings of the 5th meeting of Lin-
guistics,** Univ. of Thessaloniki, Greece (in Greek)

Ralli, A. 1986 Derivation vs Inflection **Pro-
ceedings of the 7th meeting of Linguistics,** Univ.
of Thessaloniki, Greece (in Greek)

Ralli, A. 1987? La morphologie verbale grecque,
Ph. D. dissertation Université de Montréal, Mont-
réal, Québec, Canada

Selkirk, E. 1982 **The Syntax of Words,** Linguis-
tic Inquiry Monograph, M.I.T-Press

Williams, E. 1981 On the notions "lexically
related" and "head of the word", Linguistic In-
quiry, 12(2).

Warburton, I. 1970 **On the Verb in Modern Greek**
Language Science Monographs, Volume 4 The Hague:
Mouton, Bloomington, Indiana University.