Una implementación *OpenCCG* de la generalización de Perlmutter sobre el orden de palabras de los clíticos del español

An OpenCCG Implementation of Perlmutter's Generalization on Spanish Clitics Word Order

Víctor M. Castel
InCiHuSA, CONICET, and FFyL, UNCUYO
Mendoza, Argentina
vcastel@lab.cricyt.edu.ar

Resumen

Si bien los clíticos del español constituyen un dominio de fenómenos morfológicos, sintácticos y semánticos muy bien estudiados, tanto por enfoques formales como funcionales, quedan aún aspectos que merecen nuestra atención desde el punto de vista de la Lingüística Computacional. Así, por ejemplo, Castel (1994: Cap. V) es una formalización de la generalización *Se II I III* de Perlmutter (1971) sobre el orden de los clíticos del español en términos de la Gramática Categorial de Unificación de Zeevat *et al*. (1987). Sin embargo, debido a que la gramática resultante no está implementada computacionalmente, es muy difícil, si no imposible, evaluar con precisión las consecuencias de la propuesta. Esta limitación es remediable mediante una redefinición de los clíticos del español rioplatense en el marco de *OpenCCG* (Bozşahin *et al*. 2006), versión computarizada de código abierto de la Gramática Categorial Combinatoria de Steedman (2000). La secuencia *Se II I III* se descompone en configuraciones de rasgos que reflejan las restricciones de co-ocurrencia de los clíticos. Los clíticos, a su vez, son especificados en el léxico como functores de tipo S<1>$1/(S<2>$1|NP), donde la estructura de rasgos de S<2> contiene las restricciones de co-ocurrencia relevantes que definen el potencial combinatorio de un clítico dado.

Palabras claves: Análisis y generación de clíticos, Orden de clíticos del español, OpenCCG.

Abstract

Spanish unstressed pronouns, also known as *clitics* in the generative grammar literature, constitute a domain of morphological, syntactic and semantic phenomena thoroughly studied from the perspective of both formal and functional approaches to the study of language. Despite this coverage, there are still gaps which deserve our attention, particularly from the viewpoint of Computational Linguistics. Thus, for example, Castel (1994: Ch. V) is a formalization of Perlmutter (1971)'s generalization *Se II I III* on Spanish clitics word order in terms of a unification categorial grammar (Zeevat *et al*. 1987). However, since the resulting grammar is not implemented computationally, it is very difficult, if not impossible, to evaluate the consequences of the account. In this paper, I purport to remedy this limitation by redefining River Plate Spanish clitics in the framework of OpenCCG (Bozşahin *et al*. 2006), which is an (open source) computational implementation of Combinatory Categorial Grammar (Steedman 2000). The constraint *Se II I III* is here decomposed into feature configurations reflecting clitic co-ocurrence restrictions. Clitics, in turn, are lexically specified as functors of type S<1>$1/(S<2>$1|NP), where the feature structure of S<2> contains the relevant co-ocurrence restrictions which define the combinatory potential of a given clitic.

Keywords: Parsing and Generation of Clitics, Spanish Clitics Word Order, OpenCCG.

1. INTRODUCTION

Spanish clitics are a core area of research in hispanic linguistics from the perspective of both formal and functional approaches to the study of language. Despite this coverage, there are still gaps which deserve our attention, particularly from the viewpoint of Computational Linguistics. Thus, for example, [1:Ch.V] is a formalization of [6]'s generalization *Se II I III* on Spanish clitics word order in terms of a unification categorial grammar [9]. However, since the resulting grammar is not implemented computationally, it is very difficult, if not impossible, to evaluate the consequences of the account. In this paper, I purport to remedy this limitation by redefining River Plate Spanish clitics in the framework of OpenCCG [1], which is an (open source) computational implementation of Combinatory Categorial Grammar [7, 8]. The constraint *Se II I III* is here decomposed into feature configurations reflecting clitic co-ocurrence restrictions. Clitics, in turn, are lexically specified as functors of type S<1>$1/(S<2>$1|NP), where the feature structure of S<2> contains the relevant co-ocurrence restrictions which define the combinatory potential of a given clitic.

The paper is organized as follows. In section 2, the restrictions governing clitic co-occurrence properties are summarized. Section 3 conceptualizes clitics as functors which take verbs as arguments to output functors with the relevant arguments consumed. In section 4, an account of clitic sequences is presented within the framework of OpenCCG. Finally, the paper ends with a conclusion of the results attained and a reference to work in preparation.

2. SPANISH CLITICS WORD ORDER CONSTRAINTS

Perlmutter's generalization [6] on Spanish clitic word order is expressed by the following strictly monotonic sequence:

(1) *Se* II I III,

where the roman numbers stand for person values. This surface structure constraint correctly predicts that, independently of the position relative to the governing verb, the clitic sequences in (2), but not those in (3), are well-formed:

(2a) {se, te, os, me, nos, le(s), lo/a(s)}
(2b) {se te, se me, se le(s), se lo|a(s), te me, te le(s), te lo|a(s), me le(s), me lo|a(s), …}
(2c) {se te me, se te le(s), se te lo|(s), se me le(s), se me lo|(s), …}
(2d) {te me le(s), te me lo|a(s), …}
(2e) {se te me le(s), se te me lo|a(s), …}
(3a) *{se se, te te, os os, me me, nos nos, le(s) le(s), lo|a(s) lo|a(s), ...}
(3b) *{le(s) lo|a(s), lo|a(s) le(s), te se, me se, le(s) se, lo|a(s) se, …}
(3c) *{me te le(s), me te lo|a(s), se le(s) te, …}
(3d) *{te se me le(s), me te lo|a(s) se, …}

The examples in (4-7) illustrate a few of these grammaticality contrasts:

(4a) Me lo regalaron. Regalámelo. No es cuestión de andar por ahí regalándomelo.
(4b) *Lo me regalaron. *Regalálome. *… regalándolome.
(5a) Te le acercaron. Acercátele. … acercándotele.
(5b) *Le te acercaron. *Acercálete. *… acercándolete.
(6a) Se los enviaron. Enviáselos. … enviándoselos.
(6b) *Los se enviaron. *Enviálosse. *… enviándolosse.
(7a) Te me le acercaron. Acercátemele. … acercándotemele.
(7b) *Me te le acercaron. *Acercámetele. *… acercándometele.
(7c) *Te le me acercaron. *Acercáteleme. *… acercándoteleme.

These examples and the examples in (8-9) show that clitic sequences follow infinitives, imperatives and gerunds but precede all other verb forms:

(8) Me lo regalé/regalaste/regaló/regalaron.
(9) *Regalémelo/*regalástemelo/*regalómelo/*regaláronmelo.

3. CLITICS AS FUNCTORS

The data summarized in (2-9) are formally described in [2:282-339] by specifying a Unification Categorial Grammar of Spanish (UCGS) which draws on [9]. The distinctive trait of UCGS is the treatment of clitics as functors which take verbs as arguments to output functors with the relevant arguments saturated. The consequence of this approach is that clitic sequences are defined which allow for the expression in categorial terms of Perlmutter's surface structure filter (1). However, since UCGS is not implemented computationally, the task of evaluating it with paper and pencil is simply too difficult, given its high degree of complexity and abstraction. In the following section, I remedy this limitation by reformalizing the UCGS description in the framework of Combinatory Categorial Grammar (CCG, [7, 8]), more specifically within the open source implemented version of CCG (OpenCCG, [1]). I show that Pelmutter's generalization can be expressed straightforwardly in OpenCCG, so that, for example, sentence (10) is assigned a parse like (12) from a derivation like (13-17), but sentence (11) is not accepted at all:

(10) Me lo regala.

(11) *Lo me regala.

(12) $s_{<1>\{1stC=+,\ cltc=+\}}|.np_{<8>\{case=nom,\ index=X\_2:person,\ num=sg,\ pers=3rd\}}$: @r1:action(**regalar** ^ <tense>pres ^ <Least>(l1:sem-obj ^ **lo** ^ <num>sg) ^ <Less>(m1:person ^ **me** ^ <num>sg) ^ <Most>(x1:person ^ <num>sg ^ <pers>3rd))

(13) (lex) *me* :- $s_{<1>\{1stC=+,\ cltc=+\}}\$1/.(s_{<2>\{1stC=-,\ 2ndC=-,\ 4thC=-\}}\$1|.np_{<3>\{index=X\_0:person,\ num=sg,\ pers=1st\}})$: (@X_0:person(**me**) ^ @X_0:person(<num>sg))

(14) (lex) *lo* :- $s_{<4>\{3rdC=+,\ cltc=+\}}\$1/.(s_{<5>\{1stC=-,\ 2ndC=-,\ 3rdC=-,\ 4thC=-\}}\$1|.np_{<6>\{case=acc,\ gndr=msc,\ index=X\_1:sem-obj,\ num=sg,\ pers=3rd\}})$: (@X_1:sem-obj(**lo**) ^ @X_1:sem-obj(<num>sg))

(15) (lex) *regala* :- $s_{<7>\{index=E\_2:action\}}|.np_{<8>\{case=nom,\ index=X\_2:person,\ num=sg,\ pers=3rd\}}|.np_{<9>\{case=dat,\ index=Y\_2:person\}}|.np_{<10>\{case=acc,\ index=Z\_2:sem-obj\}}$: (@E_2:action(**regalar**) ^ @E_2:action(<tense>pres) ^ @E_2:action(<Least>Z_2:sem-obj) ^ @E_2:action(<Less>Y_2:person) ^ @E_2:action(<Most>X_2:person) ^ @X_2:person(<num>sg) ^ @X_2:person(<pers>3rd))

(16) (>) *lo regala* :- $s_{<4>\{3rdC=+,\ cltc=+\}}|.np_{<8>\{case=nom,\ index=X\_2:person,\ num=sg,\ pers=3rd\}}|.np_{<9>\{case=dat,\ index=Y\_2:person\}}$: (@E_2:action(**regalar**) ^ @E_2:action(<tense>pres) ^ @E_2:action(<Least>X_1:sem-obj) ^ @E_2:action(<Less>Y_2:person) ^ @E_2:action(<Most>X_2:person) ^ @X_1:sem-obj(**lo**) ^ @X_1:sem-obj(<num>sg) ^ @X_2:person(<num>sg) ^ @X_2:person(<pers>3rd))

(17) (>) *me lo regala* :- $s<_{1>\{1stC=+,\ cltc=+\}}|.np_{<8>\{case=nom,\ index=X\_2:person,\ num=sg,\ pers=3rd\}}$: (@E_2:action(**regalar**) ^ @E_2:action(<tense>pres) ^ @E_2:action(<Least>X_1:sem-obj) ^ @E_2:action(<Less>X_0:person) ^ @E_2:action(<Most>X_2:person) ^ @X_0:person(**me**) ^ @X_0:person(<num>sg) ^ @X_1:sem-obj(**lo**) ^ @X_1:sem-obj(<num>sg) ^ @X_2:person(<num>sg) ^ @X_2:person(<pers>3rd))

One significant advantage of this reconceptualization of the UCGS sign is that the resulting OpenCCG Grammar of Spanish clitics can be tested thoroughly.

4. AN OPENCCG ACCOUNT OF CLITIC SEQUENCES

As stated in [1:3], "OpenCCG is an open source natural language processing library written in Java, which provides parsing and realization services based on Mark Steedman's Combinatory Categorial Grammar (CCG) formalism [7]". This paper takes full advantage of this facility and shows how word order restrictions of River Plate Spanish clitics can be accounted for. Here, I specifically address the clitic sequences in (2a-b) and (3a-b); an account of the clitic sequences (2c-e) and (3c-d) is given in [3].

A run-time grammar for OpenCCG [1:4] consists of five files: *grammar.xml*, *lexicon.xml*, *morph.xml*, *rules.xml*, and *types.xml*. The objective of this section is to define an OpenCCG grammar of River Plate Spanish clitics with the specific purpose of accounting for Perlmutter's generalization (1). Thus, the following sections are an XML specification of morpho-syntactic and semantic properties of the lexical items *lo*, *la*, *los*, *las*, *le*, *les*, *se*, *te* and *me*. An specification of the verb form *regala* is also included so that parsing and realization of sentences with clitic sequences can be illustrated.

4.1. Clitics (and verbs) in the file *lexicon.xml*

4.1.1. Accusative family

```
<family name="Spnsh_Acc_Cltcs" pos="NNP" closed="true">
   <entry name="Acc_Cltc">
      <complexcat>
         <atomcat type="s">
            <fs id="1">
               <feat attr="cltc" val="+"/>
               <feat attr="3rdC" val="+"/>
            </fs>
         </atomcat>
         <dollar name="1"/>
         <slash dir="/"/>
         <complexcat>
            <atomcat type="s">
               <fs id="2">
                  <feat attr="1stC" val="-"/>
                  <feat attr="2ndC" val="-"/>
                  <feat attr="3rdC" val="-"/>
                  <feat attr="4thC" val="-"/>
               </fs>
            </atomcat>
            <dollar name="1"/>
            <slash dir="|"/>
            <atomcat type="np">
               <fs id="3">
                  <feat attr="num">
                     <featvar name="NUM"/>
                  </feat>
                  <feat attr="gndr">
                     <featvar name="GNDR"/>
                  </feat>
```

```
                    <feat attr="case">
                        <featvar name="CASE"/>
                    </feat>
                    <feat attr="pers">
                        <featvar name="PERS:pers-vals"/>
                    </feat>
                    <feat attr="index">
                        <lf>
                            <nomvar name="X"/>
                        </lf>
                    </feat>
                </fs>
            </atomcat>
        </complexcat>
        <lf>
            <satop nomvar="X:sem-obj">
                <prop name="[*DEFAULT*]"/>
            </satop>
        </lf>
    </complexcat>
</entry>
<member stem="lo"/>
<member stem="los"/>
<member stem="la"/>
<member stem="las"/>
</family>
```

4.1.2. Dative family

```
    …
        <atomcat type="s">
            <fs id="1">
                <feat attr="cltc" val="+"/>
                <feat attr="3rdC" val="+"/>
            </fs>
        …
        <atomcat type="s">
            <fs id="2">
                <feat attr="1stC" val="-"/>
                <feat attr="2ndC" val="-"/>
                <feat attr="3rdC" val="-"/>
                <feat attr="4thC" val="-"/>
            </fs>
        </atomcat>
    …
<member stem="le"/>
<member stem="les"/>
</family>
```

4.1.3. First person family

```
        …
            <atomcat type="s">
                <fs id="1">
                    <feat attr="cltc" val="+"/>
                    <feat attr="1stC" val="+"/>
                </fs>
            …
            <atomcat type="s">
                <fs id="2">
                    <feat attr="1stC" val="-"/>
                    <feat attr="2ndC" val="-"/>
                    <feat attr="4thC" val="-"/>
                </fs>
            </atomcat>
        …
    <member stem="me"/>
</family>
```

4.1.4. Second person family

```
        …
            <atomcat type="s">
                <fs id="1">
                    <feat attr="cltc" val="+"/>
                    <feat attr="2ndC" val="+"/>
                </fs>
            …
            <atomcat type="s">
                <fs id="2">
                    <feat attr="2ndC" val="-"/>
                    <feat attr="4thC" val="-"/>
                </fs>
            </atomcat>
        …
    <member stem="te"/>
</family>
```

4.1.5. 'Se' family

```
        …
            <atomcat type="s">
                <fs id="1">
                    <feat attr="cltc" val="+"/>
                    <feat attr="4thC" val="+"/>
                </fs>
            …
            <atomcat type="s">
                <fs id="2">
                    <feat attr="4thC" val="-"/>
```

```
                </fs>
            </atomcat>
        …
    <member stem="se"/>
</family>
```

4.1.6. Ditransitive verb family

```
<family name="DitransitiveBeneficiaryVerbs" pos="V" closed="true">
    <entry name="DTV">
        <complexcat>
            <atomcat type="s">
                <fs id="1">
                    <feat attr="index">
                        <lf>
                            <nomvar name="E"/>
                        </lf>
                    </feat>
                </fs>
            </atomcat>
            <slash dir="|"/>
            <atomcat type="np">
                <fs id="2">
                    <feat attr="case" val="nom"/>
                    <feat attr="num">
                        <featvar name="NUM"/>
                    </feat>
                    <feat attr="pers">
                        <featvar name="PERS:pers-vals"/>
                    </feat>
                    <feat attr="index">
                        <lf>
                            <nomvar name="X"/>
                        </lf>
                    </feat>
                </fs>
            </atomcat>
            <slash dir="|"/>
            <atomcat type="np">
                <fs id="3">
                    <feat attr="case" val="dat"/>
                    <feat attr="index">
                        <lf>
                            <nomvar name="Y"/>
                        </lf>
                    </feat>
                </fs>
            </atomcat>
            <slash dir="|"/>
            <atomcat type="np">
                <fs id="4">
```

```
                <feat attr="case" val="acc"/>
                <feat attr="index">
                    <lf>
                        <nomvar name="Z"/>
                    </lf>
                </feat>
            </fs>
        </atomcat>
        <lf>
            <satop nomvar="E:action">
                <prop name="[*DEFAULT*]"/>
                <diamond mode="Most">
                    <nomvar name="X:person"/>
                </diamond>
                <diamond mode="Less">
                    <nomvar name="Y:sem-obj"/>
                </diamond>
                <diamond mode="Least">
                    <nomvar name="Z:sem-obj"/>
                </diamond>
            </satop>
        </lf>
    </complexcat>
  </entry>
  <member stem="regalar"/>
</family>
```

4.2. Clitics in the file *morph.xml*

4.2.1. Entries for clitics and verbs

```
<entry word="lo" pos="NNP" class="sem-obj" macros="@3rd @sg @msc @acc @sg-X"/>
<entry word="los" pos="NNP" class="sem-obj" macros="@3rd @pl @msc @acc @pl-X"/>
<entry word="la" pos="NNP" class="sem-obj" macros="@3rd @sg @fmn @acc @sg-X"/>
<entry word="las" pos="NNP" class="sem-obj" macros="@3rd @pl @fmn @acc @pl-X"/>
<entry word="le" pos="NNP" class="sem-obj" macros="@3rd @sg @dat @sg-X"/>
<entry word="les" pos="NNP" class="sem-obj" macros="@3rd @pl @dat @pl-X"/>
<entry word="me" pos="NNP" class="person" macros="@1st @sg @sg-X"/>
<entry word="te" pos="NNP" class="person" macros="@2nd @sg @sg-X"/>
<entry word="se" pos="NNP" class="sem-obj" macros="@3rd"/>
<entry word="regala" pos="V" stem="regalar" macros="@pres @3rd @sg @sg-X @3rd-X"/>
```

4.2.2. Macros for person

```
<macro name="@1st">
  <fs id="2" attr="pers" val="1st"/>
</macro>
<macro name="@2nd">
  <fs id="2" attr="pers" val="2nd"/>
</macro>
<macro name="@3rd">
```

```
    <fs id="2" attr="pers" val="3rd"/>
</macro>
```

4.2.3. Macros for case

```
<macro name="@nom">
    <fs id="2" attr="case" val="nom"/>
</macro>
<macro name="@acc">
    <fs id="2" attr="case" val="acc"/>
</macro>
<macro name="@dat">
    <fs id="2" attr="case" val="dat"/>
</macro>
```

4.2.4. Macros for number

```
<macro name="@sg">
    <fs id="2">
        <feat attr="num" val="sg"/>
    </fs>
</macro>
<macro name="@pl">
    <fs id="2">
        <feat attr="num" val="pl"/>
    </fs>
</macro>
```

4.2.5. Macros for gender

```
<macro name="@fmn">
    <fs id="2">
        <feat attr="gndr" val="fmn"/>
    </fs>
</macro>
<macro name="@msc">
    <fs id="2">
        <feat attr="gndr" val="msc"/>
    </fs>
</macro>
```

4.2.6. Macros for semantic number

```
<macro name="@sg-X">
    <lf>
        <satop nomvar="X">
            <diamond mode="num">
                <prop name="sg"/>
            </diamond>
        </satop>
    </lf>
```

```
</macro>
<macro name="@pl-X">
    <lf>
        <satop nomvar="X">
            <diamond mode="num">
                <prop name="pl"/>
            </diamond>
        </satop>
    </lf>
</macro>
```

4.2.6. Macros for semantic person

```
<macro name="@3rd-X">
    <lf>
        <satop nomvar="X">
            <diamond mode="pers">
                <prop name="3rd"/>
            </diamond>
        </satop>
    </lf>
</macro>
```

4.3. Clitic instantiation

The following lexical entries are the result of an instantiation process on the basis of the families specified in the *lexicon.xml* file (cf. 4.1) and the entries specified in the *morph.xml* file (cf. 4.2).

la :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=acc, gndr=fmn, index=X_0:sem-obj, num=sg, pers=3rd}) : (@X_0:sem-obj(**la**) ^ @X_0:sem-obj(<num>sg))

las :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=acc, gndr=fmn, index=X_0:sem-obj, num=pl, pers=3rd}) : (@X_0:sem-obj(**las**) ^ @X_0:sem-obj(<num>pl))

lo :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=acc, gndr=msc, index=X_0:sem-obj, num=sg, pers=3rd}) : (@X_0:sem-obj(**lo**) ^ @X_0:sem-obj(<num>sg))

los :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=acc, gndr=msc, index=X_0:sem-obj, num=pl, pers=3rd}) : (@X_0:sem-obj(**los**) ^ @X_0:sem-obj(<num>pl))

le :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=dat, index=X_0:sem-obj, num=sg, pers=3rd}) : (@X_0:sem-obj(**le**) ^ @X_0:sem-obj(<num>sg))

les :- s<1>{3rdC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 3rdC=-, 4thC=-}$1|.np<3>{case=dat, index=X_0:sem-obj, num=pl, pers=3rd}) : (@X_0:sem-obj(**les**) ^ @X_0:sem-obj(<num>pl))

me :- s<1>{1stC=+, cltc=+}$1/.(s<2>{1stC=-, 2ndC=-, 4thC=-}$1|.np<3>{index=X_0:person, num=sg, pers=1st}) : (@X_0:person(**me**) ^ @X_0:person(<num>sg))

te :- s<1>{2ndC=+, cltc=+}$1/.(s<2>{2ndC=-, 4thC=-}$1|.np<3>{index=X_0:person, num=sg, pers=2nd}) : (@X_0:person(**te**) ^ @X_0:person(<num>sg))

se :- s<1>{4thC=+, cltc=+}$1/.(s<2>{4thC=-}$1|.np<3>{index=X_0:sem-obj, pers=3rd}) : @X_0:sem-obj(**se**)

4.4. Parsing and realization of clitic sequences

The OpenCCG library offers facilities for both parsing and realization.

4.4.1 Parsing

A sentence like (10) is assigned the parse (12) with the derivation (13-17); see section 3 above. In general, the proposed OpenCCG grammar of River Plate Spanish clitics correctly parses sentences containing the sequences in (2a-b), and excludes the sequences in (3a-b), by decomposing Perlmutter's generalization (1) into configurations of attribute-value pairs referring to person properties. Whenever a clitic concatenates with a verb, the resulting sign includes the feature(s) specified in the feature structure of $\mathbf{S_{<1>}}$ of the lexical entries in 4.3 above. Thus, the sign for *lo* (cf. (14)) concatenates with the sign *regala* (cf. (15)) to output the sign *lo regala* (cf. (16)). In turn, the sign for *me* (cf. (13)) concatenates with the sign *lo regala* to output the sign *me lo regala* (cf. (17)). Notice that the feature structure of $\mathbf{S_{<2>}}$ of the lexical entries in 4.3 contains the configuration of attribute-value pairs which guarantees that a clitic, say, *me* cannot combine with a verb whose feature structure specifies that it has already concatenated with *me* (**1stC=-**), or *te* (**2ndC=-**), or *se* (**4thC=-**). Since no value is assigned to the attribue **3rdC**, *me* can certainly concatenate with a verb which has already concatenated with *lo*, *la*, *los*, *las*, *le*, *les* so that (18) but not (19) are parsed (and realized):

(18) {me, me lo, me la, me los, me las, me le, me les} regala.

(19) *{me me, me te, me se} regala.

Mutatis mutandis, the proposed grammar provides analogous derivations for all the well-formed clitic sequences in (2a-b) but not for the clitic sequences in (3a-b).

4.4.1. Realization

A logical form like (20) is realized by the sign (21).

(20) @r1:action(**regalar** ^ <tense>pres ^ <Least>(l1:sem-obj ^ **lo** ^ <num>sg) ^ <Less>(m1:person ^ **me** ^ <num>sg) ^ <Most>(x1:person ^ <num>sg ^ <pers>3rd))

(21) *me lo regala* :- $s_{<4>\{1stC=+,\ cltc=+\}}|.np_{<8>\{case=nom,\ index=x1:person,\ num=sg,\ pers=3rd\}}$:
(@l1:sem-obj(**lo**) ^ @l1:sem-obj(<num>sg) ^ @m1:person(**me**) ^ @m1:person(<num>sg) ^ @r1:action(regalar) ^ @r1:action(<tense>pres) ^ @r1:action(<Least>l1:sem-obj) ^ @r1:action(<Less>m1:person) ^ @r1:action(<Most>x1:person) ^ @x1:person(<num>sg) ^ @x1:person(<pers>3rd))

Mutatis mutandis, given appropriate logical forms like (20), the proposed grammar allows for the generation of sentences containing the clitic sequences in (2a-b) but not in (3a-b).

5. CONCLUSIONS

I have shown that Perlmutter's generalization on Spanish clitics word order constraints [6], as interpreted and formalized in [2], can be implemented in the OpenCCG library. I have specifically shown how to construct well-formed (and only well-formed) sequences of one or two clitics occurring to the left of finite verbs by defining the relevant lexical entries (cf. (2a-b) and (3a-b)). I

have not shown, however, how to account for sequences of three or four clitics (cf. (2c-e) and (3c-d)), for these presuppose the treatment of "ethical" participants. Neither have I accounted for sequences of clitics occurring to the right of infinitives, imperatives and gerunds. These two pending problems are solved in a paper in preparation [3], which addresses the issue of semantic role assignments along the lines of [4], as interpreted and formalized in [2:239-279].

Acknowledgments

I am greatful to Leo Ferres for having encouraged me to come back to Categorial Grammar during a visit to the Human-Oriented Technology Lab at the University of Carleton, Ottawa, Canada.

References

[1] Bozşahin, Cem, G-J. Kruijff and M. White. Specifying Grammars for OpenCCG: A Rough Guide, 2006. Available from http://groups.inf.ed.ac.uk/ccg/index.html.

[2] Castel, Víctor M. Paquets de clitiques de l'espagnol. Calcul de la corrélation entre rôle sémantiques et proprietés morphologiques, et spécification dans une grammaire catégorielle d'unification. PhD thesis, Université Blaise Pascal,Clermont-Ferrand II,France,1994. Available from www.cricyt.edu.ar/institutos/incihusa/ul/WebHelpPhDThesis/Victor_M_Castel_PhD_Thesis.htm

[3] Castel, Víctor M. Spanish Clitics Revisited: An OpenCCG Account of Word Order and Semantic Role Assignments. Paper submitted to the *IV Encuentro de Gramática Generativa*, Mendoza, Argentina, July 2007.

[4] García, Erica. The Role of Theory in Linguistic Analysis: The Spanish Pronoun System. North-Holland Publishing Company, Amsterdam, 1975.

[5] Haddock, N., E. Klein and G. Morril (eds.) Categorial Grammar, Unification Grammar, and Parsing. Working Papers in Cognitive Science, Vol. 1, Centre for Cognitive Science, University of Edimburgh, 1987.

[6] Perlmutter, David. Deep and Surface Structure Constraints in Syntax. New York: Holt, Rinehart and Winston, New York, 1971.

[7] Steedman, Mark. The Syntactic Process. The MIT Press, Cambridge, Mass., 2000.

[8] Steedman, Mark and J. Baldridge. Combinatory Categorial Grammar. A Tutorial Paper, 2003. Available from http://homepages.inf.ed.ac.uk/jbaldrid/ccg.pdf.

[9] Zeevat, Henk, E. Klein and J. Calder. Unification Categorial Grammar. In Haddock et al. (1987).