Newcastle Working Papers
in Linguistics
Volume 15 (2009)

The Person Case Constraint in Modern Greek: A Unified Dynamic Syntax Analysis

STERGIOS CHATZIKYRIAKIDIS

NWPL ISSN 2041-1057

http://www.ncl.ac.uk/linguistics/research/workingpapers/index.htm

The Person Case Constraint in Modern Greek: A Unified Dynamic Syntax Analysis*

STERGIOS CHATZIKYRIAKIDIS

Abstract

In this paper I look at the so-called Person Case Constraint (PCC) in Modern Greek, arguing that once a shift into a dynamic view of natural language syntax is taken, a straightforward account of the PCC can be given. In specific, I will present a unified analysis based on two analyses both proposed for the same phenomenon within the Dynamic Syntax framework (Kempson et al. 2001, Cann et al. 2005). The one based on Chatzikyriakidis (forthcoming) argues that the PCC arises out of the ordering restrictions on clitics, encoded in the entries of clitics as restrictions on the current parse state. This first analysis will propose a competition-like account of the PCC, where the members of the illicit clitic combinations found in the PCC compete for the first fixed position in the tree. The second analysis based on Kempson et al. (2008) and Kempson et al. (forthcoming) will argue that the PCC is in fact a hard structural constraint, in effect a result of the tree logic of the system, namely the impossibility of having two unfixed nodes at the same time in the sense of Kempson et al. (forthcoming). I will then provide a unified analysis that avoids the main drawbacks of each of the two analyses.

1. Introduction

The PCC is a cover term referring to a number of restrictions that ban some combinations of clitics from co-occurring. There are a number of different versions of the PCC depending on the language. The strong version, formulated in Bonet (1991), is the following:

(1) **Person Case Constraint***:* If DAT then ACC(ABS) = 3rd (Bonet, 1991).

Modern Greek (MG) exhibits the strong version of the PCC as the examples below show[1]:

---

* This paper has benefited from various discussions with my supervisor Ruth Kempson. She is gratefully thanked for providing stimulating suggestions and invaluable intellectual help. Ronnie Cann and Eleni Gregoromichelaki are also thanked for useful suggestions on various parts of this paper. Wilfried Meyer-Viol is also thanked for clearing out a number of formal issues for me. The reviewers and the editing team of the Newcastle Working Papers in Linguistics are also thanked for useful comments and invaluable editing help. The AHRC, the Leventis Foundation and the Foteini Androni Board for providing partial funding to research related to work presented in this paper. Any inconsistencies remain my own.

[1] Modern Greek does not have a distinctive dative case. The genitive case is used to denote the dative function. In this paper, we will use the two terms interchangeably to denote the dative function.

(2) *Tu me eδose
him-CL_{GEN} me-CL_{ACC} gave-IND
'He/She/It gave me to him.'

(3) *Su me eδose
you-CL_{GEN} me-CL_{ACC} gave-IND
'He/She/It gave me to you.'

(4) *Mu se eδose
you-CL_{GEN} me-CL_{ACC} gave-IND
'He/She/It gave you to me.'

The same facts hold for enclitic clitics in imperative environments:

(5) *Δose tis me
give-IMP her-CL_{GEN} me-CL_{ACC}
'Give me to her.'

(6) *Δose su me
give-IMP you-CL_{GEN} me-CL_{ACC}
'Give me to you.'

(7) *Δose mu se
give-IMP me-CL_{GEN} you-CL_{ACC}
'Give you to me.'

The Minimalist literature on the PCC is vast (see Anagnostopoulou 2003, Rezac 2003, 2008, Adger & Harbour 2007, Nevins & Săvescu 2007 among others). However, there are some general guidelines these analyses follow. The crucial feature in all these is that the PCC arises when a number of person or number features on the clitics are not checked, thus remaining uninterpretable. For example, according to Anagnostopoulou (2003), the PCC is in effect the result of checking two clitics against one functional head. The relevant functional head, say F, has person and number features. If the dative clitic checks its person feature, which is what is assumed to happen, then the accusative clitic can no longer check the same feature, thus the PCC. Third person accusative clitics can however participate in such constructions since they are assumed to bear only number and not person features (Anagnostopoulou 2003, 2005). Third person accusative clitics being licit in such constructions further assumes that dative clitics do not bear any number features. Adger & Harbour (2007) on the other hand propose that the participant features of the first/second person clitics are responsible for the PCC[2]. Without getting into the specifics of the analysis, Adger & Harbour (2007) argue that the PCC arises from the inability of

[2] Participant features can roughly be seen as animacy features. See Adger & Harbour (2007) for more on the notion of participant features.

first/second person clitics to check their participant features. The problem with all the above analyses lies in the decision one has to make regarding the features each clitic is assumed to have. There is no principled explanation of why, for example, third person dative clitics have to be specified as having a [-person] feature, whereas third person accusative clitics have to be specified as bearing no person feature at all, an assumption made in Anagnostopoulou (2003, 2005)[3]. The same holds for the assumption made in Adger & Harbour (2007), i.e. that all datives must be animate. It seems that such a strong generalization cannot be sustained given the data presented in Bonet (2007) and Chatzikyriakidis (forthcoming) for Catalan and Greek respectively. The interested reader is directed to Chatzikyriakidis (forthcoming) for an extensive review of these analyses.

Besides purely syntactic analyses like the ones mentioned, morphological approaches also exist, notably Bonet (1991, 1997). According to these approaches, the PCC is dealt with within the morphological component, arguing that such a constraint is not syntactic at all. The characteristic of these approaches is the stipulation of morphological clitic templates that derive the licit clitic combinations. However, there are good arguments that favour a syntactic rather than a morphological analysis. I cannot go into these arguments in this paper for reasons of space. The interested reader is however directed to Rezac (2009) for a detailed discussion.

Furthermore, there are usage-based analyses like Haspelmath (2004) arguing that the reason behind the unavailability of the illicit clitic clusters can be attributed to infrequent usage of these clusters compared to the licit forms. However, there is no principled way of expressing such a claim within a formal framework, and thus such an assumption is doomed to remain a potential informal functional explanation.

Abstracting away from all the above analyses, I will propose that the PCC can receive a straightforward explanation once we shift into a dynamic syntactic model. Before I move on to propose the analysis, let me first briefly and rather informally introduce the DS framework.

2. The DS framework

In this section I briefly introduce the reader to the framework of Dynamic Syntax (Kempson et al. 2001, Cann et al. 2005). The introduction will be fairly informal and aims at giving a basic understanding of the formalism to someone with no affiliation with the model.

2.1. Basic assumptions behind DS

DS is a parsing-oriented framework. Parsing in DS is seen as the basic syntactic activity in natural languages (Kempson et al. 2001, Cann et al. 2005). Under that assumption, natural languages are modelled as the accumulation of semantic representations during the parsing process. This accumulation is assumed to be strictly incremental, proceeding in a linear, word to word manner. The upper goal in every parsing

[3] The assumption that dative clitics do not bear number features is also not uncontroversial. See Bonet (2007: footnote 14) for a discussion.

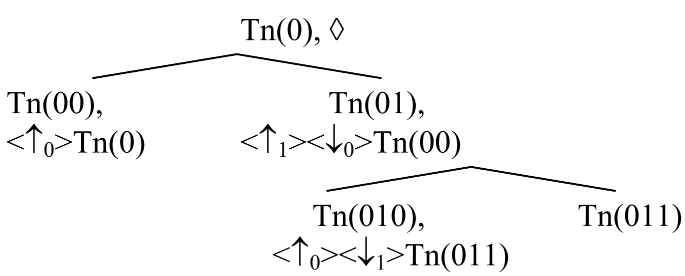
process is the construction of a logical formula of type t, obtaining a proposition in other words. Binary trees are employed to represent this process. However, these tree structures are considerably different from those found in derivational or declarative frameworks like Minimalism or HPSG respectively in at least two respects. Firstly, trees in DS are not inhabited by words as such but rather by the representations that these words provide, following Fodor (1975) in that respect. Furthermore, the tree structure corresponding to the end result of parsing a natural language string is a semantic representation assigned to this natural language string with respect to some context. Word order is not encoded in the tree but can however be obtained by tracing the different steps of the parsing process. The partial tree structures in conjunction with the history of the parsing steps can provide an account of word order phenomena. This is a natural way to achieve it in such a framework since the whole parsing process is dependent on the way words are ordered. In what follows, I will present the formal tools DS uses in order to carry out these assumptions.

2.2. An informal view of the formal framework

2.2.1. Logic of finite trees – Blackburn & Meyer-Viol (1994)

As already mentioned, DS uses binary trees in order to represent the parsing process. What is rather novel in these tree structures, however, is a logical language underpinning the whole tree system called LOFT (Logic of Finite Trees – Blackburn & Meyer-Viol 1994). LOFT is a modal language to describe trees. It is an extremely flexible and simple language of talking about any node in a tree from the perspective of any node in that tree. The basic tool behind this modal tree system is a mechanism labelling the nodes according to their position in the tree. This is done by using a predicate that gives every node a unique address. This predicate is Tn (standing for treenode) followed by the actual tree address in parentheses, e.g. Tn(01). Nodes on the left are conventionally addressed by adding 0 to the Tn value, while nodes on the right (the functor nodes) by adding 1. The different 0s and 1s found in a given treenode encode the route that someone needs to traverse in order to get to that node beginning from the topmost treenode addressed as Tn(0). The mother and daughter relation modalities, $\langle\uparrow\rangle$ and $\langle\downarrow\rangle$ respectively, allow the system to express relations between the different nodes:

(8)



In the above tree every treenode has a unique address given by the procedure I have just sketched. For example, Tn(01) states that the node in question can be reached if starting from the initial node, one goes down the 1 node. The same reasoning applies to all the other addresses found in the tree. Notice that some of the nodes bear statements involving the modal language LOFT. For example the statement ‘$\langle\uparrow_1\rangle\langle\downarrow_0\rangle$Tn(00)’ reads as follows: You will find Tn(00) if you first go up the 1 mother relation and then go down the 0 daughter relation. We will see later on the relevance of using such a flexible modal language in the tree system. Lastly, the ◊ symbol, called the pointer, is used to denote which node is the relevant node any time during the parsing procedure.

2.2.2. Decorations and Requirements

Nodes are decorated with a number of elements called ‘Treenode decorations’. The basic elements of this set are the Formula and Type value decorations. The former are represented by the predicate ‘Fo’ followed by the value of the concept denoted by the word in question while the latter are represented by the predicate ‘Ty’ and the type value in brackets, e.g. Fo(John') and Ty(e) respectively[4].

Requirements, on the other hand, are essentially goals to be achieved. They have the basic form ?La(i) (e.g. ?Fo(x))[5]. Requirements must be satisfied in order for a given parse to be successful. In this respect, requirements can be also seen as a device explaining ungrammaticality, since a successful parse must not contain any outstanding requirements. The result of a successful parse is shown below. Notice that no outstanding requirements exist:

(9) The result of successfully parsing *John hit Bill*

Ty(t),(Hit’(Bill’(John’)))

Ty(e),Fo(John’) Ty(e→t),Fo(hit’(Bill’))

Ty(e),Fo(Bill’) Ty(e→(e→t)),Fo(Hit’)

2.2.3. Computational – Lexical – Pragmatic Actions

Three types of actions, computational, lexical and pragmatic drive the parsing process. Computational actions comprise the basic tree growth machinery in DS. These involve an input and an output description. The input description has information about the node the pointer is on, whereas the output description shows a transformation of the input. This transformation might involve a number of different actions such as adding nodes, eliminating or creating requirements, pointer movement etc. An example of such an action, the rule of COMPLETION is shown below:

---

[4] The prime indicates that the concept rather than the word is represented. Furthermore, DS does not assume any recursion on types. Types are a closed set in DS. See Kempson et al. (2001) and Cann et al. (2005) for more information on semantic typing in DS.

[5] Where La stands for label and i≥1. For a detailed discussion of declarative units in DS see Kempson et al. (2001: chapter 7).

(10) COMPLETION

$$\frac{\{\ldots\{Tn(n),\ldots\},\{\langle\uparrow_i\rangle,Tn(n),\ldots,Ty(X),\ldots,\diamond\}\ldots\}}{\{\ldots\{Tn(n),\ldots,\langle\downarrow_i\rangle Ty(X),\ldots,\diamond\},\{\langle\uparrow_i,Tn(n),\ldots,Ty(X),\ldots\}\ldots\}}$$
$$i=\{0,1,*\}$$

The upper part of the rule presents a situation in which two nodes exist. The Tn(n) node is the mother of the other node; As can be seen from the modal statement the latter node has ($\langle\uparrow_i\rangle$ Tn(n)). This daughter node bears a type value and the pointer. The transformation presented in the second line of the rule presents a situation where the pointer has moved to the mother node by virtue of the type value in the daughter node. The information that the daughter node has a type value is encoded at the mother node ($\langle\downarrow_i\rangle$Ty(X)). A number of other computational actions exist in DS which cannot be presented here due to space limitations. Additional rules will be introduced when needed. The reader is referred to Cann et al. (2005) for the complete set of computational actions.

Lexical actions on the other hand, are the actions induced by the words themselves, roughly corresponding to lexical entries in other frameworks. These actions are basically instructions on how the parse should or should not continue. Lexical entries are represented using a standardised algorithmic schema. This schema plus a sample lexical entry is shown below:

(11) Generalised algorithmic schema for lexical entries in DS

IF	Trigger
THEN	Actions
ELSE	Elsewhere statement

(12) Sample lexical entry for *Bill*

IF	?Ty(e)
THEN	put(Ty(e), Fo(Bill’)), $[\downarrow]\bot$
ELSE	Abort

The IF part acts as the triggering point. It denotes a condition that should be satisfied in order for the ACTION part to be induced. The THEN part denotes the actions induced by the word in case the trigger is satisfied. The ELSE part provides an alternative in case the IF part is not satisfied. The lexical entry for *Bill* above reads as follows: If you are in a node that has a type e requirement then decorate this node with a type value and a formula value standing or the representation of a contextually given individual named Bill.

The [↓]⊥ restriction ensures that no nodes below the current one can be built[6], in effect something like a terminal node restriction.

Lastly, pragmatic actions involve contextual information and inferences used to help the parsing process. An example of such an action is the rule of SUBSTITUTION. The latter rule effectively updates a formula metavariable, by giving it a proper value from context[7].

2.2.4. The notion of Underspecification

Central within the DS framework is the notion of underspecification, i.e. the claim that natural languages are to a high extent underspecified. DS uses both semantic and structural underspecification as being vital parts of the system itself. As regards semantic underspecification, DS employs the use of metavariables acting as content placeholders to be substituted later on from context or from the natural language string itself. The whole idea is that some linguistic elements provide only partial semantic information as regards the entity they refer to. Full information is provided later on in the parsing process. An example of this sort is pronouns. Pronouns in DS are taken to project a type value and a formula metavariable, the latter to be substituted later on from context or from the natural language string itself. The metavariable carries restrictions on the potential substituent value. These are denoted as subscripts. Notice that the triggering point has an extra restriction, i.e. $?\langle\uparrow_0\rangle Ty(t)$. This will capture the fact that a pronoun like *he* cannot be parsed as an object. Such a requirement will be satisfied only if *he* is parsed in the subject node, i.e. the 00 node. The lexical entry for the pronoun *he* is shown below:

(13) Lexical entry for pronoun *he*

IF	$?Ty(e), \langle\uparrow_0\rangle?Ty(t)$
THEN	put($Ty(e), Fo(U_{male'}), ?\exists\mathbf{x}.Fo(\mathbf{x})$), $[\downarrow]\bot$
ELSE	Abort

Structural underspecification on the other hand is expressed by means of unfixed nodes. These are nodes that have not yet found their position in the tree. These are introduced using a number of computational rules, known as the ADJUNCTION rules. One such rule, the *ADJUNCTION rule, effectively introduces such an unfixed node from the type t requiring node, and decorates the unfixed node with a type e requirement plus a requirement for a proper treenode address to be found ($?\exists\mathbf{x}.Tn(\mathbf{x})$). The only requirement as regards the fixing site of such a node is that this site must be somewhere below the type t requiring node:

[6] The ⊥ symbol is the well known 'falsum' symbol of classical logic. Formally it reads: For all nodes below me nothing holds.
[7] See Kemspon et al. (2001) and Cann et al. (2005) for the formal definition of the rule.

(14) The *ADJUNCTION rule

$$\frac{\{\ldots\{Tn(a),?Ty(t),\ldots,\Diamond\}\}}{\{\ldots\{Tn(a),?Ty(t),\ldots\}\},\ \{\langle\uparrow *\rangle Tn(a),?\exists x.Tn(x),\ldots,?Ty(e),\Diamond\}\}}$$

In tree notation, the effect of the rule is the following:

(15)

```
                 Tn(a), ?Ty(t),◊
                        /
                       /
                      /
                     /
    <↑*>Tn(a),?∃x.Tn(x),?Ty(e),◊
```

With the pointer at the unfixed node, an accusative marked NP can be parsed satisfying the type requirement:

(16)

```
                 Tn(a), ?Ty(t),◊
                        /
                       /
                      /
                     /
    <↑*>Tn(a),?∃x.Tn(x),Ty(e),Fo(x),◊
```

The latter rule will effectively capture the fact that an accusative marked NP can receive multiple structural positions, i.e. direct object, indirect object, direct or indirect object outside the local domain. The underspecified modality, denoted by the kleene star operator (*), will effectively capture all these cases, since fixing of the node is possible in all positions below the type t requiring node[8]. A number of different rules do exist in the family of ADJUNCTION rules depending on the nature of the phenomena they deal with. For example, in order to deal with local scrambling cases, i.e. cases where the scrambled NP must be updated inside the local propositional domain, DS employs another rule, the LOCAL *ADJUNCTION rule, which basically restricts the fixing sites of the unfixed node

[8]Some people may object to this, by saying that the rule will vastly overgenerate assuming that fixing of the node can be done anywhere below the current node. However, this is not true, since the possibilities of fixing the node considerably reduce as the parsing process proceeds. For example, after a verb has been parsed, all the functor nodes will bear predicate type requirements rendering fixing of the unfixed node in any of these nodes impossible. Furthermore, case filters (Cann et al. 2005 and Chatzikyriakidis in preparation) are used in case one wants to avoid fixing of the node in specific structural positions.

to the minimal propositional domain by employing a more specified underspecified modality[9]:

(17) The LOCAL *ADJUNCTION rule

$$\frac{\{\ldots\{Tn(a),\ldots,?Ty(t),\Diamond\}\ldots\}}{\{\ldots\{Tn(a),?Ty(t),\ldots\}\ldots\{\langle\uparrow_0\uparrow_1^{*}\rangle\, Tn(a),\ ?Ty(e),\exists x.Tn(x),\Diamond\}\ldots\}}$$

In tree notation:

(18)

Tn(a), ?Ty(t),◊
/
/
/
/
$\langle\uparrow_0\uparrow_1^{*}\rangle Tn(a),?\exists \mathbf{x}.Tn(\mathbf{x}),Ty(e),Fo(x),\Diamond$

Notice that the modality has changed from $\langle\uparrow^{*}\rangle$ to $\langle\uparrow_0\uparrow_1^{*}\rangle$. This will ensure that the NP in question is parsed in the local propositional domain[10]. The two rules are used for long and short distance scrambling effects respectively. We will see later on the relevance of these rules with respect to clitics.

2.2.5. LINK

The family of LINK rules is another collection of rules which however are fairly different from the ADJUNCTION rules presented above. The difference lies in the fact that LINK rules involve two separate trees. A LINK relation is imposed between the two trees, with the first tree imposing a requirement on the second that some of its formulae should be found in the LINKed tree. We define two new modal operators, <L> and $\langle L^{-1}\rangle$, to refer to the LINKed tree and the tree of which the LINK starts respectively. I will use the rules of TOPIC STRUCTURE INTRODUCTION and TOPIC STRUCTURE REQUIREMENT to exemplify how LINK structures work. As you may have inferred, these two rules are used for topic constructions, specifically for Hanging Topic Left Dislocation (HTLD) constructions such as the one shown below:

[9]See Cann et al. (2005) for a detailed discussion on the usage of the two ADJUNCTION rules and their relevance for deriving scrambling effects.

[10]Assuming that all argument nodes are the 0 nodes and an additional propositional domain will involve a type t in one of the argument nodes, this rule will exclude cases where the NP is associated with an argument in an embedded propositional domain. See Cann et al. (2005) for a definition of locality in DS.

(19) As for Mary, I met the man who married her

The first of them effectively creates a LINK transition from the initial node, to a type- requiring node:

(20) TOPIC STRUCTURE INTRODUCTION rule

{{Tn(0),?Ty(t),◊}}

{{Tn(0),?Ty(t)}},{<L>Tn(0),?Ty(e),◊}

As soon as the dislocated topic is parsed in the type e requiring node the rule of TOPIC STRUCTURE REQUIREMENT adds the requirement for a shared term in the initial node:

(21) TOPIC STRUCTURE REQUIREMENT rule

{{Tn(0),?Ty(t)}},{<L>Tn(0),Fo(a),Ty(e),◊}

{{Tn(0),?Ty(t),?<D>Fo(a),◊}},{Tn(0),?Ty(t)},{<L>Tn(0),Fo(a),Ty(e)}

After the introduction of these two previous rules we get the following tree structure:

(22) After both rules have applied and *as for Mary* is parsed as the topic:

<L>Tn(0),Fo(a),Ty(Mary')

<L^{-1}>Tn(X),?Ty(t),?<↓ $_*$>Fo(Mary')

The dislocated NP *as for Mary* is parsed in the node where the LINK begins. After the TOPIC STRUCTURE REQUIREMENT rule has applied, a requirement for the same formula value provided in the first tree is put on the LINKed tree (?<↓ $_*$>Fo(Mary')). This will ensure that a copy of the formula *Mary* will also be provided in the LINKed tree by the linguistic string, for example a resumptive pronoun in English. There are a number of important things with respect to the general LINK rule, but I will not discuss them here since these are not relevant to the scope of the paper. The interested reader is however referred to Kempson et al. (2001) and Cann et al. (2005) for more information on the various LINK rules.

3. A DS analysis
3.1. General assumptions with respect to clitics

Clitics in DS are taken to involve their own lexical entries. However, this is not something that forces us to enter into the affix-word debate (Monachesi 1993, Miller & Sag 1997, Philippaki & Spyropoulos 1999, Condoravdi & Kiparsky 2002 among others), since for DS every linguistic element, whether it is a word or an affix, can have its own lexical entry if this element provides distinct procedural information on how the parse should or should not continue (see for example the case of Japanese case suffixes in Cann et al. 2005). Within this line of reasoning, we do not have to pre-theoretically decide on the status of clitics, since such a decision will have no significance for the analysis proposed. Lexical entries for clitics in DS capture clitic positioning by defining restrictions on the current parse state. For example the lexical entry for the third person accusative clitic *to* 'it' in SMG based on Chatzikyriakidis (2006) defines two different parsing triggers, effectively capturing proclitic and enclitic positioning:

(23) Lexical entry for *to* 'it'

IF		?Ty(t)
THEN	IF	$[\downarrow_1^+]$?Ty(x)
	OR	
	IF	Mood(Imp)
	THEN	makego($<\downarrow_1>$), makego($<\downarrow_0>$); put(Ty(e), Fo(V_{neut}), ?∃**x**.Fo(**x**)); gofirst(?Ty(t))
	ELSE	Abort
ELSE	Abort	

If the initial ?Ty(t) trigger is satisfied then the parser checks if all the functor nodes bear type requirements ($[\downarrow_1^+]$?Ty(x)) and proceeds to the actions if the latter fact is true. Such a thing will be true only in case a verb has not been parsed, thus ensuring proclisis. This is because if a verb is parsed at least one functor node will bear a type value standing for the verb's semantic type, i.e. a type dependent on the verb's addicity. On the other hand, the second trigger (Mood (Imp)) searches for an imperative feature in the ?Ty(t) node. If such a feature is present, then the clitic can be also parsed, thus the enclisis with imperatives. The actual actions induced by the clitic, if the triggering points are satisfied, include building of the direct object node and the decoration of this node with a type e and a formula metavariable carrying gender presuppositions along with a requirement for a proper Formula value (?∃**x**.Fo(**x**)) to be found on that node. The latter will be true either by context substitution or via substitution from the natural string itself. The latter option will be an instance of clitic doubling[11]. For more information of how this entry works and detailed examples see Chatzikyriakidis (2006, forthcoming).

Having set out the basic assumptions with respect to clitics, let us now see how the PCC can be captured in DS.

3.2. The PCC in DS
3.2.1. The competition analysis

The first analysis we are going to propose is based on the analysis Chatzikyriakidis (forthcoming) proposed for Grecia Salentina Greek (GSG)[12]. According to this analysis, the PCC arises when competition for the first fixed node in the tree occurs. We assume that all preverbal objects are parsed either via *ADJUNCTION or via LINK structures in MG. In that respect, when clitics come into parse a number of elements might already have been parsed. The latter fact might lead us to an analysis of the PCC in the following way: since genitive clitics must always be first in a sequence of clitics and first person accusative clitics cannot occur at all in sequences of clitics, we can posit a triggering restriction for these two types of clitics specifying that the clitic in question can be parsed only if no fixed nodes exist in the tree, in other words the trigger will abort in case any fixed structure has been projected before the clitic comes into parse. Such a move will give us the correct ordering facts plus the PCC. The entries for genitive and first/second person accusative clitics are shown below:

(24) General entry for genitive clitics

IF		?Ty(t)
THEN	IF	$[\downarrow^{+}]?\exists\mathbf{x}.Tn(\mathbf{x})$
	THEN	makego($<\downarrow_1>$), makego($<\downarrow_1^{*}>$), makego($<\downarrow_0>$); put(Ty(e), Fo(V), ?$\exists\mathbf{x}$.Fo($\mathbf{x}$)); gofirst(?Ty(t))
	ELSE	Abort
ELSE	Abort	

[11] See Cann et al. (2005), Gregoromichelaki (forthcoming) and Chatzikyriakidis (in preparation) for a DS treatment of Clitic Left Dislocation (CLLD) and Clitic Doubling in MG.
[12] Grecia Salentina Greek is one of the main varieties of the Grico dialect, a Modern Greek dialect spoken in some parts of Southern Italy.

(25) Entry for first/second person accusative clitics

IF		?Ty(t)
THEN	IF	$[\downarrow^{+}]?\exists\mathbf{x}.Tn(\mathbf{x})$
	THEN	$makego(<\downarrow_1>), makego(<\downarrow_0>);$ put(Ty(e), Fo(V), $?\exists\mathbf{x}.Fo(\mathbf{x})$); gofirst(?Ty(t))
	ELSE	Abort
ELSE	Abort	

The only difference between the two entries is that genitive clitics are underspecified with respect to their position in the tree, since genitive clitics can occupy different positions in the clause, i.e. they can function as direct or indirect object:

(26) Mu tilefonise
me-CL$_{GEN}$ telephoned
'He telephoned me'

(27) Tu milisa
him-CL$_{GEN}$ talked
'I talked to him'

(28) Tu to eδosa (SMG)
him-CL$_{GEN}$ it-CL$_{ACC}$ gave-IND
'I gave it to him'

The latter claim might be argued to hold for accusatives as well, since there exist a number of double accusative verbs in SMG. However the data are quite vague as regards these verbs. For example, when we try to substitute the full NPs with clitics in these double accusative constructions what we do get is one of the following: a) When there is only one clitic which is the indirect argument, the clitic can optionally be genitive marked (example 30) or b) When we substitute both NPs with clitics, the indirect argument must be genitive marked, i.e. a situation where two accusative clitics are present is not possible (examples 31 and 32):

(29) O Jorγos kerase ton maθiti to paγoto tu
The George bought the-ACC student-ACC the-ACC ice-cream-ACC his
'George bought the student his ice-cream.'

(30) O Jorγos ton kerase to paγoto tu
The George him-CL$_{ACC}$ bought the-ACC ice-cream-ACC his
'George bought him his ice-cream.'

(31) O Jorγos tu kerase to paγoto tu
The George him-CL_{GEN} bought the-ACC ice-cream-ACC his
'George bought him his ice-cream.'

(32) *O Jorγos ton to kerase
The George him-CL_{ACC} it-CL_{ACC} bought
'George bought it for him.'

(33) O Jorγos tu to kerase
The George him-CL_{GEN} it-CL_{ACC} bought
'George bought it for him.'

The situation is quite vague as can be seen from the above examples. For that reason, I will assume that accusative clitics, unlike accusative NPs, are always fully specified with respect to the position they occupy in the tree. Apparent counterexamples like example (29) will not be dealt with here, since a number of things need to be accounted for before we attempt an analysis of clitic behaviour with these verbs, e.g. the actual argument structure involved in these constructions. The interested reader is directed to Anagnostopoulou (2001) for an extensive discussion of these types of double object constructions. Leaving aside the complications presented by double accusative constructions, I will assume that accusative clitics are associated with the direct object node, directing however the interested reader to Chatzikyriakidis (in preparation) for a discussion on how double accusative constructions can be accommodated within such an analysis.

Returning to the entries we have given in (23) and (24), we observe that both the entries have a second triggering restriction, i.e. $[\downarrow^{+}]?\exists\mathbf{x}.Tn(\mathbf{x})$. This statement reads as follows: All nodes below the type t requiring node have a requirement for a proper treenode address to be found, that is only unfixed nodes exist in the tree. Note that such a statement would be true even if no nodes are present in the tree, since the universal modality [] will be true in the presence of an empty set, following standard assumptions in Classical Logic[13]. Once we posit such a triggering restriction, the PCC follows directly since assuming that one of the two clitics has been parsed, some fixed structure will have been built. This will automatically exclude any occurrence of genitive with first/second person accusative clitics. On the other hand, third person accusative clitics will be able to be parsed after an accusative clitic has already done so, since no such restriction is imposed by third person accusative clitics. In order to get the PCC with imperatives, I posit a similar more specified restriction that no fixed object must exist by the time the clitic is parsed in an imperative construction. The entries for genitive and first/second person accusative clitics including imperatives are shown below:

[13] See any introductory book on First Order Logic. We suggest Enderton (1972) for a more mathematical approach to logic and Gamut (1991) for a more linguist-oriented introduction to the subject.

(34) Lexical entry for genitives including imperatives

IF		$?Ty(t)$
THEN	IF	$[\downarrow_1^+]?\exists \mathbf{x}.Tn(\mathbf{x})$
	OR	
	IF	$Mood(Imp), [\downarrow_1^+][\downarrow_0]?Ty(x)$
	THEN	$makego(<\downarrow_1>), makego(<\downarrow_1^*>), makego(<\downarrow_0>);$
		$put(Ty(e), Fo(V),$
		$?\exists \mathbf{x}.Fo(\mathbf{x})); gofirst(?Ty(t))$
	ELSE	Abort
ELSE	Abort	

(35) Lexical entry for the first person accusative clitic including imperatives

IF		$?Ty(t)$
THEN	IF	$[\downarrow_1^+]?\exists \mathbf{x}.Tn(\mathbf{x})$
	OR	
	IF	$Mood(Imp), [\downarrow_1^+][\downarrow_0]?Ty(x)$
	THEN	$makego(<\downarrow_1>), makego(<\downarrow_1^*>), makego(<\downarrow_0>);$
		$put(Ty(e), Fo(V),$
		$?\exists \mathbf{x}.Fo(\mathbf{x})); gofirst(?Ty(t))$
	ELSE	Abort

The analysis correctly captures the PCC in both environments. The analysis is in effect an analysis of competition for the first fixed position in the tree in indicatives and a competition for the first fixed object position in imperatives. The posited restrictions are specific to the strong PCC version, and can arguably be altered to accommodate weak PCC effects. Such an analysis does not assume a hard structural constraint being involved as regards the PCC. Such an analysis is thus against the traditional view that assumes the PCC to be a universal structural constraint (Anagnostopoulou 2003, Adger & Harbour 2007, Kempson & Cann 2008) but it is much closer to analyses like Haspelmath (2004) where the PCC receives a usage-based explanation and is not in that respect associated with a hard wired structural constraint, claiming however that the phenomenon is indeed a syntactic phenomenon. Assuming that parsing triggers are something like heuristics or facilitators of the parsing process, it is quite natural to assume such restrictions in the entries for clitics. For more details on how these triggers might have developed diachronically see Chatzikyriakidis (forthcoming).

The next analysis I am going to look at assumes that the PCC derives from a much more general structural constraint as regards underspecification of structure.

3.2.2. The structural constraint analysis

The second analysis we are going to look at, is based on Kempson & Cann (2008) and Kempson et al. (forthcoming). According to such an analysis, the PCC is as a reflex of the tree logic under-pinning the DS system in which no more than one unfixed node with the same underspecified address can be present in a tree structure. As we have already seen, DS makes extensive use of the notion of structural underspecification by employing unfixed nodes. Kempson et al. (forthcoming) assume that the actions of genitive clitics and first/second person accusative clitics in MG project a locally unfixed node. The original analysis of Kempson et al. (forthcoming) is based on Romance clitics in which first/second person clitics are syncretised, which explains the underspecification involved in the entries. However, such an account is argued to be extendable to MG where case syncretism is not present, assuming that the clitics in question in MG also project a locally unfixed node but they further posit restrictions on the possible update on the tree, in effect something like a case filter. The reason for choosing to encode the rule of *LOCAL ADJUNCTION rather than the rule of *ADJUNCTION is the locality properties of clitics, namely the fact that clitics must always be interpreted within the local domain[14]. Assuming the latter, treating the clitics as projecting an unfixed and not a locally unfixed node, will fallaciously predict that clitics can be also updated outside the local domain. Assuming that both genitive and first/second person accusative clitics project locally unfixed nodes, if any of these clitics co-occur, two nodes with the same underspecified modality will be present in the tree. Then by definition, these two nodes will collapse into being the same node since there will be nothing to distinguish between the two nodes. However, the two nodes cannot merge into one node since they carry incompatible formula values. For example the presuppositions of the formula metavariable of a first person singular clitic will be specified as $Fo(U_{Speaker'})$ and for a second person singular clitic as $Fo(U_{Hearer'})$. If both these clitics co-occur in the same node, their metavariables will never be able to unify into one compatible formula value. The entries based on such an analysis will be the following:

[14] There is of course the phenomenon of Clitic Climbing (Rizzi 1977, 1982, Kayne 1989, Cardinaletti & Shlonsky 2004, Cinque 2006 among others) where the clitic seems to be interpreted outside its local domain. However, the standard assumption in the literature is that no such domain violation occurs and in fact a monoclausal structure is involved in all these cases (see all the above references). Furthermore, there is no Clitic Climbing in Modern Greek.

(36) Lexical entry for genitives including imperatives

IF		?Ty(t)
THEN	IF	$[\downarrow_1^+]$?Ty(x)
	OR	
	IF	Mood(Imp)
	THEN	makego($<\downarrow_1>$), makego($<\downarrow_1^*>$), makego($<\downarrow_0>$); put(Ty(e), Fo(V), ?$\exists$**x**.Fo(**x**),gofirst(?Ty(t))
	ELSE	Abort
ELSE	Abort	

(37) Lexical entry for the first person accusative clitic including imperatives

IF		?Ty(t)
THEN	IF	$[\downarrow_1^+]$?Ty(x)
	OR	
	IF	Mood(Imp)
	THEN	makego($<\downarrow_1>$), makego($<\downarrow_1^*>$), makego($<\downarrow_0>$); put(Ty(e), Fo(V), ?$\exists$**x**.Fo(**x**),?$<\uparrow_0>$Ty(e$\rightarrow$t)); gofirst(?Ty(t))
ELSE		Abort

For MG, as mentioned earlier, where case syncretism is not present, Kempson et al. (forthcoming) assume that non-syncretised clitics impose a filter on output, in effect something like a restriction on tree update. For example in the case of an accusative clitic, the restriction will be '?$<\uparrow_0>$Ty(e$\rightarrow$t)' shown in the entry in (35). This restriction posits a requirement that the immediate mother of the node must be the predicate. The latter statement can only be true if the clitic is the direct object, thus the restriction is something like a case marker. A similar filter on output restriction is assumed to be present in genitive clitics. However, such a restriction is not needed for genitive clitics, since as we discussed earlier genitive clitics in MG, even though non-syncretised, can appear in different structural positions (see examples 25, 27).

Let me illustrate in more detail how the account just proposed works. Let us say we want to parse the ungrammatical sentence shown below:

(38) *Mu se eδose
you-CL$_{\text{GEN}}$ me-CL$_{\text{ACC}}$ gave-IND
'He/She/It gave you to me.'

Both of the clitics are assumed to project a locally unfixed node. Parsing a genitive clitic will give us the following structure:

(39)

$$Tn(a),\ ?Ty(t),\lozenge$$
$$/$$
$$\langle\uparrow_0\rangle\langle\uparrow_1\rangle\langle{}^{*}\uparrow_1\rangle Tn(a),?\exists\mathbf{x}.Tn(\mathbf{x}),Ty(e),Fo(U_{Speaker'}),\lozenge$$

The second person accusative is parsed after a genitive clitic. In that respect, another unfixed node with the same address will be projected. These two nodes will collapse into the same node by means of sameness of address. This will result in the following node:

(40)

$$Tn(a),\ ?Ty(t),\lozenge$$
$$/$$
$$\langle\uparrow_0\rangle\langle\uparrow_1\rangle\langle{}^{*}\uparrow_1\rangle Tn(a),?\exists\mathbf{x}.Tn(\mathbf{x}),Ty(e),Fo(U_{Speaker'}),Fo(V_{Hearer'}),\lozenge$$

In the above structure both clitics occupy the same node. The metavariables posited cannot be updated into the same formula value since they carry incompatible presuppositions. In that respect, the parse will never be completed. Thus, constructions where a first person genitive clitic is followed by a second person accusative clitic are ruled out. The same reasoning applies to all the other illicit combinations[15]. An apparent drawback for such an analysis is how this analysis will work with respect to the PCC in imperatives. Remember that the PCC is active in imperatives as well:

(41) *Δose tis me
give-IMP her-CL$_{GEN}$ me-CL$_{ACC}$
'He gave me to her.'

(42) *Δose su me
give-IMP her-CL$_{GEN}$ me-CL$_{ACC}$
'He gave me to you.'

(43) *Δose mu se
give-IMP me-CL$_{GEN}$ you-CL$_{ACC}$
'He gave you to me.'

[15] We should note that the metavariable presuppositions will have to involve more information than what we tentatively represent as Speaker' and Hearer' in the above examples. More information will be needed to distinguish two first person clitics for example. We will not go into details on the nature of metavariable presuppositions in this paper.

Assuming that the imperative verb will project the whole propositional spine, the first unfixed clitic will manage to fix its address in one of the argument positions projected by the verb before the next clitic is parsed. The latter fact will then predict contrary to fact that the very same combinations should be licit in postverbal position.

3.4. Unifying the two analyses

The difference between the two analyses I have presented lies in the theoretical status these attribute to a constraint like the PCC. According to the first analysis, the impossibility of some clitic combinations co-occurring is due to a competition between two clitics for the first fixed position in the tree. In effect, the 'first fixed node restriction entails an 'appear first in a clitic sequence' restriction, even though it is more than that. It can be argued to be of stipulatory nature assuming that the PCC is a generalised constraint. However, some kind of stipulation will be needed to cover the ordering facts with respect to clitics. Even adopting the second analysis we proposed, where the PCC is taken to involve a general structural constraint, we need a similar trigger to the 'first fixed node restriction' trigger, in order to get the ordering facts right. If such a restriction is needed anyway for genitive clitics, then it does not seem implausible to extend the same restriction to first/second person accusative clitics and get the PCC facts right.

On the other hand, the second analysis assumes a generalisation inherent to the tree logic of the system to be responsible for the PCC. Within such an approach, the PCC is a general constraint on tree unfolding, a general constraint on underspecified modalities. However as noted above, even under such an analysis, we need a restriction similar to the 'first fixed node' restriction to get the facts right, namely dative-accusative ordering in pre-verbal cases. In that respect, we can reformulate a third analysis based on the two analyses discussed, in effect combining the analyses found in Chatzikyriakidis (forthcoming) and Kempson et al. (forthcoming) into a single analysis that attributes the PCC to a general tree logic constraint as in Kempson et al. (forthcoming), while retaining the 'first fixed node' restriction for genitives to capture dative-accusative ordering found in preverbal licit clitic sequences in the same sense as Chatzikyriakidis (2008). The prediction that the PCC must not be active with imperatives that such an analysis seems to make is only apparent. Assuming that even though parsing proceeds word by word a number of words can be parsed as clusters, and assuming that the two clitics form such a cluster, as is standardly assumed in the literature for clitics (see practically any analysis of clitic sequences in Greek and Romance), no problem arises for such an analysis, since the general computational rules that would fix the first clitic's address won't be operative unless the whole cluster has been parsed first. Under such an assumption, the PCC is predicted to be active in imperative environments by using the same reasoning we have used for the pre-verbal cases.

Within this combined analysis the lexical entries for genitive, first/second person accusative and 3rd person accusative clitics are shown below:

(44) Lexical entry for genitives including imperatives

IF $?Ty(t)$
THEN IF $[\downarrow^{+}]?\exists \mathbf{x}.Tn(\mathbf{x})$

OR
IF Mood(Imp)
THEN makego($<\downarrow_1>$), makego($<\downarrow_1^{*}>$), makego($<\downarrow_0>$);
put(Ty(e), Fo(V_x),
$?\exists \mathbf{x}.Fo(\mathbf{x})$); gofirst(?Ty(t))
ELSE Abort
ELSE Abort

(45) Lexical entry for the first-second person accusative clitics including imperatives

IF $?Ty(t)$
THEN IF $[\downarrow_1^{+}]?Ty(x)$
OR
IF Mood(Imp)
THEN makego($<\downarrow_1>$), makego($<\downarrow_1^{*}>$), makego($<\downarrow_0>$);
put(Ty(e), Fo(V_x),
$?\exists \mathbf{x}.Fo(\mathbf{x})$, $?<\uparrow_0>Ty(e\rightarrow t)$); gofirst(?Ty(t))
ELSE Abort

(46) Lexical entry for 3rd person accusative clitics including imperatives

IF $?Ty(t)$
THEN IF $[\downarrow_1^{+}]?Ty(x)$
OR
IF Mood(Imp)
THEN makego($<\downarrow_1>$), makego($<\downarrow_0>$);
put(Ty(e), Fo(V_x),
$?\exists \mathbf{x}.Fo(\mathbf{x})$); gofirst(?Ty(t))

ELSE Abort
ELSE Abort

The above entries capture both the ordering facts plus the PCC. The above analysis implies that ordering restrictions are highly language dependent and that is why they derive from a number of restrictions in the lexical entries of various clitic forms and not from general computational actions of the system. This is a reasonable stance to take since the lexicon seems to be the domain par excellence of variation in natural languages. It would be a mistake to attribute clitic ordering facts to a universal generalisation or schema, since we cannot really say how to accommodate the diversity of clitic ordering in various languages (see for example the case of French). On the other hand the PCC seems to be part of a stronger generalization, exhibited in a large number of languages unrelated to each other (Bantu, Romance, Greek, Georgian, Kiowa, Basque to name a few). Whether this fact is a strong universal tendency, a strong but not inviolable constraint or a hard-wired natural language constraint is something that cannot be decided at the moment, since facts from more clitic languages need to be collected. Even with the data I have at the moment it is very difficult to attribute the PCC to a hard- wired constraint since clitic languages exhibiting no PCC effects do exist (e.g. Polish - Haspelmath 2004). However, the account proposed does not predict that every clitic language will obligatorily exhibit the PCC, but rather that if two linguistic elements are introduced having the same underspecified address, then they will collapse into decorating the same node, thus rendering the sentence ungrammatical. So, the explanation proposed is not specific to clitics but is a general account of how underspecification works in DS and arguably in natural languages. The proposed unified analysis avoids explaining the PCC as a language specific phenomenon like the analysis put forth in Chatzikyriakidis (forthcoming) while on the other hand it retains the restrictions that capture clitic ordering. Furthermore, the main drawback of Kempson & Cann (2008) and Kempson et al. (forthcoming), i.e. the fact that such an analysis does not seem to work for imperatives is easily accommodated as we assume that general computational actions cannot operate before the whole clitic cluster is parsed.

The advantage of this analysis, compared to recent minimalist analyses (Anagnostopoulou 2003, 2005, Adger & Harbour 2007) is that a number of stipulations are avoided in providing an account of the PCC. In specific, the choice of a set of features for each clitic, which in a number of cases seems unmotivated (see the discussion in section 1), is side-stepped since it is irrelevant to the actual account. Dubious descriptive constraint generalisations like the ‘animacy constraint’ (see section 1) in Adger & Harbour (2007) are also irrelevant to such an account. The PCC under the account presented stems from a general restriction on underspecification, the fact that no more than one underspecified modality can exist. In that respect, both the minimalist analyses and ours attribute the PCC to general mechanisms of the system (Agree and underspecification respectively). In the present analysis however, no added stipulations (stipulatory feature assignments, dubious descriptive constraints) are needed in order for the analysis to work.

Conclusion

In this paper I examined the PCC in Greek from a dynamic perspective. I have presented two existing accounts of the phenomenon within DS and proposed a unified version which avoids the drawbacks of the previous two analyses. Specifically, I take the PCC in Greek to arise from a general constraint on underspecification which bans two unfixed nodes with the same underspecified address from co-occurring in the tree structure. Assuming that genitive and first/person accusative clitics project unfixed nodes which have the same underspecified address, the PCC follows directly.

References

Adger, D. & Harbour, D. (2007). Syntax and Syncretisms of the Person Case Constraint. *Syntax* 10, 2-37.

Anagnostopoulou, E. (2001). Two Classes of Double Object Verbs: The Role of Zero Morphology. In van Oostendorp, M. & Anagnostopoulou, E. (eds.), *Progress in Grammar: 20 Years of Grammaticamodellen*, Meertens Institute, electronic book.

——— (2003). *The Syntax of Ditransitives: Evidence from Clitics*. Berlin: Mouton de Gruyter.

——— (2005). Strong and weak person restrictions: A feature checking analysis. In L. Heggie and F. Ordonez (eds.), *Clitics and Affixation*. Amsterdam: Benjamins, 199-235.

Blackburn, P. & Meyer-Viol, W. (1994). Linguistics, Logic and Finite Trees. *Bulletin of Interest Group of Pure and Applied Logics* 2, 2-39.

Bonet, E. (1991). *Morphology after Syntax: Pronominal Clitics in Romance*. Doctoral dissertation, MIT.

——— (1994). The Person Case Constraint: A Morphological Approach. In H. Harley and C. Philips (eds.), *The morphology Syntax Connection*. Cambridge, MA: MIT, 33-52.

——— (2007). The Person Case Constraint and Repair Strategies. To appear in Alessasndro, R., Fischer, S., Hrafnbjanson, G. H. (eds.), *Person Restrictions*. Berlin: Mouton de Gruyter.

Bouzouita, M. (2008). At the Syntax-Pragmatics Interface: Clitics in the History of Spanish. In Cooper, R., & Kempson, R. (eds.), *Language in Flux: Variation, Change and Evolution*. London: King's College Press, 221-263.

Cann, R., Kempson, R., Marten, L. (2005). *The Dynamics of Language*. Oxford: Elsevier.

Cardinaletti, A. & Shlonsky, U. (2004). Clitic Positions and Restructuring in Italian. *Linguistic Inquiry* 35, 519-557.

Chatzikyriakidis, S. (2006). *Clitics in Modern Greek: A Dynamic Account*. MSc thesis, King's College London.

——— (2008). *Clitics in Grecia Salentina Greek: A Dynamic Account*. To be published in *Lingua*.

——— (in preparation). *Clitics in Four Dialects of Modern Greek: A dynamic account*. PhD thesis, King's College, London.

Cinque, G. (2006). Restructuring and Functional Heads. *Oxford Studies in Comparative Syntax*. Oxford: Oxford University Press.

Condoravdi, C. & Kiparksy, P. (2002). Clitics and Clause Structure. *Journal of Greek Linguistics* 2: 1-39.

Gregoromichelaki, E. (forthcoming). A Dynamic Perspective on Left-Right Asymmetries: CLLD and Clitic Doubling in Greek. In *Rightward Movement in a Comparative Perspective: Selected Papers from the DGFS-AG Rightward Movement workshop*. Forthcoming from John Benjamins.

Enderton, H. (1972). *A Mathematical Introduction to Logic*. Academic Press. New York-London

Fodor, J. (1975). *Language of Thought*. Cambridge: Harvard University Press.

Gamut, L.T.F. (1991). *Logic, Language and Meaning. Volume I: Introduction to Logic.* Chicago: University of Chicago Press.

Haspelmath, M. (2004). Explaining the Ditransitive Person - Role Constraint. Electronic form article. Available from: <http://email.eva.mpg.de/~haspelmt/2004dpr.pdf.>. Accessed on 21/05/2009.

Kayne, R. (1989). Null Subjects and Clitic Climbing. In Jeggli, O., Safir, K.J. (eds), *The Null Subject Parameter*. Dordrecht: Kluwer Academic Publishers, 239-261.

Kempson, R., Meyer-viol, W., Gabbay, D. (2001). *Dynamic Syntax: The Flow of Language Understanding*. Oxford: Blackwell Publishing.

Kempson R., Cann R. (2008). *Production pressures, Syntactic Change and the Emergence of Clitic Pronouns.* In Cooper R., and Kempson R. (eds.), *Language in Flux: Variation, Change and Evolution*. London: King's College Publications, 179-220.

Kempson R., Cann R., Marten, L. (forthcoming). Treegrowth Dynamics. In C. Chesi (ed.), *Directions in Derivations*. Amsterdam: Elsevier.

Monachesi, P. (1993). Object Clitics and Clitic Climbing in Italian HPSG Grammar. In *Proceedings of the Sixth Conference on European Chapter on the Association for Computational Linguistics.* Utrecht: Association for Computational Linguistics, 437-442

Nevins, P. & Săvescu O. (2008). An Apparent Number Constraint in Romanian. In Silvina Montrul et al. (eds.), *Romance Linguistics 2008* (to appear 2009).

Ormazabal, J. & Romero, R. (2007). The Object Agreement Constraint. *Natural language and Linguistic Theory* 25, 315-347.

Philippaki-Warburton, I. & Spyropoulos, V. (1999). On the Boundaries of Inflection and Syntax. *Yearbook of morphology* 1998, 45-72.

Rezac, M. (2008). The syntax of eccentric agreement: The Person Case Constraint and Absolutive Displacement in Basque. *Natural Language and Linguistic Theory* 26 (1), 61-106.

Rezac, M. & Bejar, S. (2003). Person licensing and the derivation of PCC effects. In Ana Teresa Pérez-Leroux and Yves Roberge (eds.), *Romance Linguistics: Theory and Acquisition*. Amsterdam: John Benjamins, 49-62.

Rizzi, L. (1982). Issues in Italian Syntax. *Studies in Generative Grammar* 11. Dordrecht: Foris Publications.

Săvescu, O. (2007). Challenging the Person Case Constraint: Evidence from Romanian. In Camacho, J., Flores-Ferran, N., Sanchez, L., Deprez, V., Jose Cabrera, M. (eds.), *Romance Linguistics 2006. Selected Papers from the 36th Linguistic Symposium on Romance Languages* (LSRL), March-April 2006, 261-274. New Brunswick: John Benjamins.

Stergios Chatzikyriakidis
Philosophy Department, King's College London
20 Thornaby House, Canrobert Street
E20BE, London, UK

kafouroutsos@hotmail.com