A Dynamic Account of Clitic Positioning in Cypriot Greek

Stergios Chatzikyriakidis

Royal Holloway, University of London; Open University of Cyprus

abstract

In this paper, a full account of clitic positioning in Cypriot Greek is attempted within the framework of Dynamic Syntax. Firstly, it is shown that the existing approaches dealing with CG clitic positioning are inadequate to deal with the full range of clitic positioning phenomena as these are described by Pappas (2010) and Chatzikyriakidis (2010). Then it is argued that this complex system can be effectively captured assuming a lexical entry where three generalized parsing strategies, i.e. ways of structure building, function as lexical triggers for parsing CG clitics. Variation in positioning with the non-factive complementizer *oti* as well as the causal subordinator epidi are accounted for assuming that these elements can be parsed as either subordinators or coordinators. Furthermore, the challenge of providing an account for complex markers/subordinators formed with the coordinator *tze* is provided, arguing that the unexpected enclisis caused in these cases is due to the fact that these elements provide two separate linked domains where the first acts as the context in which the second is parsed (e.g. a negative context). Lastly, the account proposed will be shown to be grounded in historical considerations as well, arguing that the transition from a descriptively simpler system (that of Medieval Cypriot Greek, MCG) to a more complex one (CG) is only epiphenomenal, showing that the transition from MCG to CG involves simplification of the lexical entry for clitics.

1. Introducing the data

Taking a look at the positioning system of CG, the major difference one notes between systems of the Standard Modern Greek (SMG) type and of the CG type is that in the latter clitic positioning is not defined solely with respect to the form of the verb (imperative-gerund vs indicative) but a number of non-verbal elements can affect clitic positioning as well. Specifically, CG exhibits three different positioning environments: a) enclitic b) proclitic and c) variation environments. Clitics in CG are in general enclitic in indicative and non-indicative contexts:

(1)

a.	*Poli*	*anthropi*	*kamnun*	*to*	*sosta.*
	many	people	do.3PL	it.CL-ACC	right
b.	**Poli*	*anthropi*	*to*	*kamnun*	*sosta.*
	many	people	it.CL-ACC	do.3PL	right

‘Many people people do it right.’

(2)

Email address: Stergios.Chatzikyriakidis@rhul.ac.uk (Stergios Chatzikyriakidis)

a.

Kamne	*to.*
do.IMP	it.CL-ACC

b.

**To*	*kamne.*
it.CL-ACC	do.IMP

'Do it.'

This is in contrast to what we find in SMG, where positioning is regulated by the form of the verb: indicative verbs are associated with proclisis (3) while imperatives and gerunds with enclisis (4 and 5 respectively):

(3)

a.

**Poli*	*anthropi*	*kanun*	*to*	*sosta.*
many	people	do.3PL	it.CL-ACC	right

b.

Poli	*anthropi*	*to*	*kanun*	*sosta.*
many	people	it.CL-ACC	do.3PL	right

'Many people people do it right.' [SMG]

(4)

a.

Kane	*to.*
do.IMP	it.CL-ACC

b.

**To*	*kane.*
it.CL-ACC	do.IMP

'Do it.'

(5)

a.

Kanontas	*to...*
do.GER	it.CL-ACC

b.

**To*	*kanontas...*
it.CL-ACC	do.GER

'By doing it.'

However proclisis obtains in indicative environments in CG in case a number of elements appear at the left-periphery. These elements include Wh elements and fronted constituents, NEG, modal/tense particles and subordinating conjunctions. These proclisis-inducing elements are further subcategorized to those that obligatorily induce proclisis and to those that optionally do so. In the former category, one finds: a)Wh-elements:

(6)

Pios	*ton*	*ide*	*(*ton)?*
who.NOM	him.CL-ACC	saw.3SG	him.CL-ACC

'Who saw him? '

b) Modality/Tense/Mood/Negation markers:

(7)

En	*ton*	*iksero*	*(*ton).*
NEG	him.CL-ACC	know.1SG	him.CL-ACC

'I do not know him.'

(8) *Min ton dis (*ton).*
NEG him.CL-ACC see.2SG.PNP him.CL-ACC
'Do not see him.'

(9) *Thelo na ton do (*ton).*
want.1SG SUBJ him.CL-ACC see.1SG him.CL-ACC
'I want to see him.'

(10) *Enna ton do (*ton) apopse.*
FUT him.CL-ACC see.1SG him.CL-ACC tonight
'I'm going to see him tonight.'

b) A number of subordinating conjunctions , notably conditional and temporal ones (11 and 12-13 respectively):

(11) *An ton dite (*ton),...*
if him.CL-ACC saw.2PL him.CL-ACC
'If you see him...'

(12) *Pu ton ida (*ton), milisa tu.*
when him.CL-ACC saw.1SG him.CL-ACC spoke.1SG him.CL-GEN
'When I saw him, I spoke to him.'

(13) *Aman ton ipopsiazete (*ton), gorazi tis lulufkja/fjora.*
if/when him.CL-ACC is-suspicious him.CL-ACC buy.3SG her.CL-GEN flowers
'If/When she is suspicious of him, he buys her flowers.'

c) The factive complementizer *pu*:[1]

(14) *Lipume pu ton ides (*ton).*
be-sorry.1SG COMP him.CL-ACC saw.2SG him.CL-ACC
'I'm sorry that you saw him.'

(15) *Thimume pu ton ides (*ton).*
remember.1SG COMP him.CL-ACC saw.2SG him.CL-ACC
'I remember that you saw him.'

The second category involves elements that induce proclisis optionally. In this category, we firstly find fronted objects, fronted subjects and fronted adverbs. It is worth noting that in the existing literature, these elements are categorized as belonging to the first category (see e.g. Agouraki, 2001), i.e. elements that obligatorily trigger proclisis. The reasoning behind this is based on the claim that fronted elements are always associated with proclisis when these are focused. In that respect, the generalization in these analyses is that focused elements and not fronted elements trigger proclisis. However, Pappas (2010) has disputed the claim that focused elements are always associated with proclitic positioning, arguing that focus is not a prerequisite of proclisis with fronted elements. Following this observation, I believe that the correct description is that fronted elements sometimes trigger proclisis and sometimes do not. The examples below are illustrative of this variation in positioning with fronted elements. Examples 16 to 19 seem to show that focus is what triggers proclisis. However, examples (20) and (21) show that focus does not always induce proclisis and, vice versa, that non-focused fronted elements can also trigger proclisis (21):

[1]Note that given the subjunctive is expressed via the proclisis inducing subjunctive marker *na*, subjunctives are always associated with proclisis in CG.

(16) *TUTO tu edokes.*
this him.CL-GEN gave.1SG him.CL-GEN not the other
'You gave him this one.'

(17) *Tuto edokes tu.*
this him.CL-GEN gave.1SG him.CL-GEN not the other
'You gave him this one.'

(18) *I MARIA ton ikseri oi i Ioana.*
the.NOM Mary.NOM him.CL-ACC know.3SG not the.NOM Ioana.NOM
'Mary knows him, not Ioanna.'

(19) *I Maria ikseri ton.*
the.NOM Mary.NOM him.CL-ACC know.3SG
'Mary knows him.'

(20) *Etsi/ETSI ton ida.*
that-way him.CL-ACC saw.1SG
'I saw him like this/It was like this that I saw him.'

(21) *Sinexia/SINEXIA eksetimaze me.*
continuously was-swearing.3SG me.CL-ACC
'He was continuously swearing at me.'

In Pappas' (2010) corpus, 15% of fronted coreferential OV constructions are associated with preverbal placement. The standard assumption in the literature is that these constructions should be illicit given that doubling constructions are topics, and thus non-focused. This is disputed by Pappas (2010) who gives the following Clitic Left Dislocated (CLLD) example from his own corpus where proclisis is attested:

(22) *Ki emena m aresi.*
and me.DAT me.CL-DAT like.SG
'I like this.'

It is certainly true that focus plays a role in CG clitic positioning and potentially proclisis is preferred when a fronted element is focused. However, the data in Pappas (2010) show that this should not be taken as a strict constraint but rather as a preference. In the approach taken in this paper, Pappas' (2010) claim as regards the association of focus with proclisis will be neatly explained by the dynamics of the framework involved, namely the fact that more than one strategy might be possible for parsing one and the same word. Given that these are parsing strategies that do not involve features like e.g. focus or topic features, the account predicts that variation with fronted elements should be possible no matter the pragmatic import in each case.

Besides variation found with fronted elements, which is also found in languages like Portuguese (Crysmann, 1997; Luis & Otoguro, 2004 inter alia) or other dialects of Modern Greek, like e.g. Cretan Greek, there are a number of environments which are unambiguous cases of variation no matter the descriptive or theoretical standpoint one takes towards these elements. These concern subordinating conjunctions and complementizers that freely allow proclisis and enclisis without any interpretational difference whatsoever. The first such element is the non-factive complementizer *oti*, which was firstly reported to trigger both proclisis and enclisis in Revithiadou (2006):[2]

[2]One of the reviewers asks whether optionality means preference or chance. This is something that is not yet elucidated. Some speakers indicate that there is preference towards one of the two options in a specific construction (e.g. a preference for enclisis in the case of factive complementizer *oti*) while the rest indicate that there is no preference for one structure against the other. While this is an important issue, I believe what matters the most in these cases is that the speakers note that both proclitic and enclitic versions of the same sentence are perfectly licit CG sentences.

(23)

a.

Ipen	*oti*	*ethkiavasen*	*to.*
said.3SG	COMP	read.3SG	it.CL-ACC

b.

Ipen	*oti*	*to*	*ethkiavasen.*
said.3SG	COMP	it.CL-ACC	read.3SG

‘S/He/It said that s/he read it.’

Pappas (2010) as well as Chatzikyriakidis (2010) independently reported that variation is also possible with subordinates of cause epiδi and γiati ‘why’, with Pappas further identifying variation with *afu* ‘given that’ as well:

(24)

a.

Epidi	*enevriases*	*me,*	*en*	*kamno*	*tipota.*
because	made-angry.2SG	me.CL-ACC	NEG	do.1SG	nothing

b.

Epidi	*me*	*enevriases,*	*en*	*kamno*	*tipota.*
because	me.CL-ACC	made-angry.2SG	NEG	do.1SG	nothing

‘I’m not doing anything because you made me angry.’

(25)

a.

Giati	*kamni*	*me*	*tze*	*niotho*	*orea.*
because	makes.3SG	me.CL-ACC	and	feel.1SG	nice

b.

Giati	*me*	*kamni*	*tze*	*niotho*	*orea.*
because	me.CL-ACC	makes.3SG	and	feel.1SG	nice

‘Because s/he/it makes me feel nice.’

(26)

a.

Afu	*edimiurgisan*	*tis*	*provlima.*
since	created.3PL	her.CL-DAT	problem

‘Since rthey caused trouble to her.’ [Pappas, 2010: ex.9]

b.

Afu	*me*	*kamni*	*tze*	*niotho*	*orea.*
because	me.CL-ACC	makes.3SG	and	feel.1SG	nice

‘Since s/he/it makes me feel nice.’

Lastly, it is worth looking at a number of environments that involve a negation-conjunction or a conjunction-conjunction complex, the second part of the complex always being the coordinator *tze* ‘and’. The first part of the complex is always a proclisis inducing element (the negation marker or a subordinate conjunction) whereas the second part of the complex is the coordinating conjunction *tze*. In all these cases, proclisis is ungrammatical with enclisis being the only grammatical option:

(27)

a.

En	*tze*	*(*ton)*	*ida*	*ton.*
NEG	and	him.CL-ACC	saw.1SG	him.CL-ACC

b.

An	*tze*	*(*ton)*	*ida*	*ton.*
if	and	him.CL-ACC	saw.1SG	him.CL-ACC

‘Even though I saw him...’

1.1. Existing analyses

A number of analyses regarding clitic positioning in CG have been put forth by the years, notably Agouraki (1997, 2001), Terzi (1999a, 1999b), Condoravdi & Kiparsky (2001) and Revithiadou (2006). With the exception of Revithiadou (2006), all the other accounts are inadequate on independent empirical grounds, since they deal with a rather incomplete set of data (e.g. no variation environments are discussed). Besides that, a number of technicalities of these accounts are problematic, like e.g. the motivation for V movement in Terzi (1999a,b) and Agouraki (2001), or the existence of two NegPs in Agouraki (2001). These approaches are discussed in detail in Revithiadou (2006), Chatzikyriakidis (2010) and Pappas (2010), showing the problems associated with these accounts in full detail and as such will not be reviewed here again. The interested reader is directed to Revithiadou (2006), Chatzikyriakidis (2010) and Pappas (2010) for more details on these approaches.

An interesting account of CG is that of Revithiadou (2006). Revithiadou (2006) departs from both 'all syntax' and 'all phonology' accounts of clitic positioning and provides an account where phonology has a filtering role on syntax, i.e. there is no direct intervention of phonology in the syntax like e.g. PF movement, but however phonology provides a number of well-formedness rules that apply to pairs of syntactic constructions. Then, given such a pair, these rules choose the construction that scores the less violations and discard the other. Revithiadou (2006) follows Bošković (1995, 2001) who argues that syntax assigns more than one copy of clitics and pre-specified phonological filters act as filters that eventually decide which of the two copies will be pronounced. Then, Revithiadou (2006) proposes the following prosodic ranking hierarchy for CG, i.e. a set of ranked well-formedness rules:

(28) Ranking of prosodic structure in CG

FAITH(acc), EXH, WCON(L), NONREC >> PCON,WCON(R)

FAITH(acc) is a constraint which posits that inherent accent of the input must be preserved in the output (Revithiadou 2006: 87, apud Revithiadou 1999). EXH and NONREC are prosodic domination constraints. The former specifies that no C^i immediately dominates C^j for $j<i-1$ while the latter that no C^i dominates C^j for $j=1$. The rest of the constraints are alignment constraints (see Revithiadou, 2006: 85 for the formal details). Revithiadou then proceeds and shows how these constraints work and give rise to the various patterns in CG. For example, in case only a verb is present, enclisis is the only option given that in case of proclisis, NONREC will be violated which is ranked higher than the PCON and WCON(R) in the enclitic case. In cases of proclisis, there are two options for the clitic to be phrased. In case of a stressed function word, the clitic appears as enclitic to that stressed element since this case incurs less violations than the respective construction where the clitic encliticizes to the verb. In case of an unstressed function word, the cluster function word+clitic appears proclitic to the verb, though the actual details are not given in the paper (it seems that both cases will incur a WCON(L) violation, but the enclitic case will incur two violations of PCON and one of WCON(R) while the proclitic case only one violation of PCON). The account works fine up to that point. However, the problem arises with variation constructions. Specifically, Revithiadou argues that variation attested with *oti*, which is a stressed complementizer, is explained given the existence of two alternative phrasing possibilities. It can be either phrased with the verb of the main clause or alternatively with the verb of the embedded clause. The two different phrasing possibilities cause the two alternative patterns of clitic positioning. In case *oti* is phrased along with the main verb, then enclisis obtains. However, if *oti* is phrased along with the embedded verb then the clitic is enclitic to the *oti* and surfaces preverbally. Revithiadou does not show how these variation data will be captured under her constraint ranking. But even if we assume that Revithiadou's account indeed captures variation with *oti*, there are a number of facts that would immediately falsify such an account. The first counter-evidence for such an account is constructions that are the same as the *oti* constructions, the only difference being a different complementizer, e.g. sentences introduced with the conditional conjunction *ean* (which is also stressed). In these constructions variation is not possible, the only possible construction being proclisis (or preverbal position with the clitic enclitisizing to the subordinate conjunction):

(29) *Ipen* *ean* *to* *ethkiavasen* *(*to).*
said.3SG COMP read.3SG him.CL-ACC
'S/He said/asked if s/he read it.'

It is not clear how the above will be handled assuming a phrasing account is used to explain variation with *oti*. Even worse, there are constructions in which *oti* has only one phrasing option, i.e. being phrased with the embedded verb only, and still both proclisis and enclisis are allowed. Such a situation obtains if the *oti* clause is left-dislocated, i.e. when *oti* appears as the first element of the clause:

(30)

a.	*Oti*	*ton*	*ikseri,*	*xtes*	*mu*	*to*	*ipe.*
	COMP	him.CL-ACC	know.3SG	yesterday	me	it	said

b.	*Oti*	*kseri*	*ton,*	*xtes*	*mu*	*to*	*ipe.*
	COMP	know.3SG	him.CL-ACC	yesterday	me	it	said

’Yesterday s/he told me that s/he knows him.’

It is not clear how cases like the above will be treated under an account which attributes variation to the different phrasing possibilities of *oti*. Furthermore, the proclitic account of elements that bear inherent is also not devoid of problems. Pappas (2010), in discussing Revithiadou (2006), argues that elements like *enke* (en tze) that carry inherent stress will be associated with preverbal positioning in Revithiadou’s system contrary to fact. Furthermore, the phonological data from s-voicing and e-deletion Revithiadou presents in order to support the claim that preverbal clitics can be either proclitic to the verb or enclitic to a stressed function word are disputed by my speakers. Revithiadou claims that e-deletion and s-voicing between the verb and the clitic is not possible when the clitic is enclitic to the function word and not proclitic to the verb. Thus Revithiadou says that in e.g. *enna tus maierepsi* ‘S/he will cook for them’ s-voicing does not apply and we thus get [enna tus mairepsi] and not [*enna tuz mairepsi]. However, my speakers indicated s-voicing to be the case here. Furthermore, as regards e-deletion, Revithiadou claims that this is obligatory when the function word is unstressed and does not occur when the function word is stressed, e.g *pos to ftkiavas*e vs *pjós to efkiavase* ‘that s/he read it’ vs ‘who read it .’ My speakers indicated optionality of e-deletion in both the stressed and the unstressed case, and thus noted that both *pos to ftkiavase/pos to efkiavase* and *pjós to efkiavase/pjós to fkiavase* are possible. If my data are correct, then the basic spine of the account put forth by Revithiadou (2006) needs to be completely rethought of. In case Revithiadou is correct and my data are just plainly wrong,[3] then still the account proposed cannot as already said account for variation environments and predicts proclisis instead of enclisis for *en tze*.

The task I’m going to take up in this paper, is to argue that a full account of CG can be given once a shift into a dynamic, parsing-oriented framework has been made. In section 2 an introduction to the framework of Dynamic Syntax is given and in section 3 a full account of the CG positioning system is given.

1.2. Interim note on data collection methodology

Before I proceed to the main part of this paper, I should make a note on data collection methodology, given that a number of the data presented in this paper is the result of 3 year data collection that took place from 2007 to 2010 in discontinuous time intervals. The participants were all native speakers of CG aged between 20-35, that have been raised and lived in Cyprus at least until the age of 18 (or still live there). All speakers were also speakers of SMG. Data elicitation was done by using pre-designed questionnaires which contained two different sections. The first section asked the participants to translate a number of SMG sentences into the equivalent CG ones, and indicate whether more than one structural possibility is possible. The second section asked the participants to posit grammaticality judgments as regards a number of pre-constructed CG sentences. It must be noted that CG, despite being the native tongue of possibly most (if not all) Cypriot Greeks in both Cyprus and diaspora, the official language of the Cypriot Greek state is SMG (Newton, 1972; Tsiplakou 2009, among others), and as such modern day CG stands in a diglossic relationship to SMG (Tsiplakou, 2009, in press). Tsiplakou (2009) showed that this diglossic relationship in modern day Cyprus is between SMG and an emergent pancypriot Koiné. At the same time, all

[3]Given the uniformity of judgments from all the speakers I consulted in this case, this is something that I doubt.

regional varieties contributing to the formation of this new form of Cypriot, i.e. the Koiné, are subject to levelling (Tsiplakou, 2009). I do not know at the present whether differences in clitic positioning are found among individual regional varieties or whether the data gathered as presented in this paper present a clitic positioning system which is influenced by SMG. Unfortunately, there is no way at present to test this. I believe that the data gathered, besides presenting some novel aspects on clitic positioning (e.g. variation with subordinates of cause reported in Chatzikyriakidis (2010) and independently by Pappas (2010), are a fairly accurate description of the CG positioning system, at least in the way it is spoken today by young CG speakers given that the data selected were extremely uniform across speakers (8) and variation between these was almost non-existent. It is however wise to keep in mind that this paper will not try to address regional differences with respect to clitic positioning (if any). Therefore, it should not be taken for granted that the clitic positioning system described in this thesis is uniform across all regional dialects of idioms of CG (which might be true but at the present it is just impossible to tell).

2. DS preliminaries

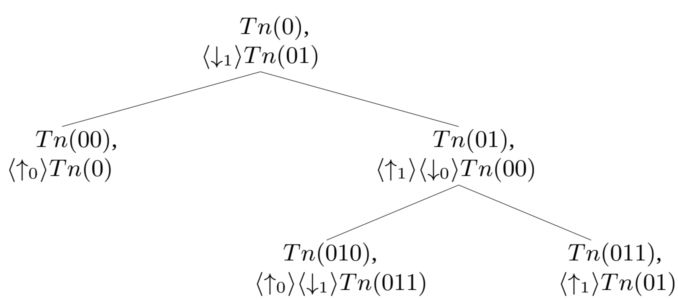
The Dynamic Syntax (DS) framework (Kempson et al. 2001; Cann et al. 2005) is a processing oriented framework. The basic assumption behind DS is that natural language syntax can be seen as the progressive accumulation of transparent semantic representations with the upper goal being the construction of a logical propositional formula (a formula of type *t*). Such a process is driven by means of monotonic tree growth, representing the attempt to model the way information is processed in a time-linear, incremental, word-to-word manner. DS is a goal driven framework driven by means of requirements. For example the starting point for every parse, as already mentioned, is a requirement to obtain a propositional formula of type t:

(31) The AXIOM

$$?Ty(t), \diamond$$

The parsing process is represented via means of binary trees, underpinned by the Logic of Finite Trees (LOFT, Blackburn & Meyer-Viol, 1994), an expressive modal language that allows one to talk about any treenode from the perspective of any treenode. LOFT uses two basic modalities, ↓ and ↑, corresponding to the daughter and mother relation respectively. Left nodes are addressed as 0 nodes, whereas right nodes as 1 nodes. Conventionally, nodes on the left correspond to the argument nodes, i.e. the nodes in which the arguments will be represented, whereas the 1 nodes correspond to the functor nodes, i.e. the nodes in which all the various types of predicates will be represented. The rootnode, defined as the sole node that does not have a mother node, is given the treenode address 0. The two basic LOFT modalities (↓ and ↑) are used in either their existential or universal version (represented as $\langle\downarrow\rangle$ and $[\downarrow]$ respectively). The example below illustrates the flexibility of LOFT by showing a binary tree where different nodes are addressed from the perspective of other nodes using treenode modalities:

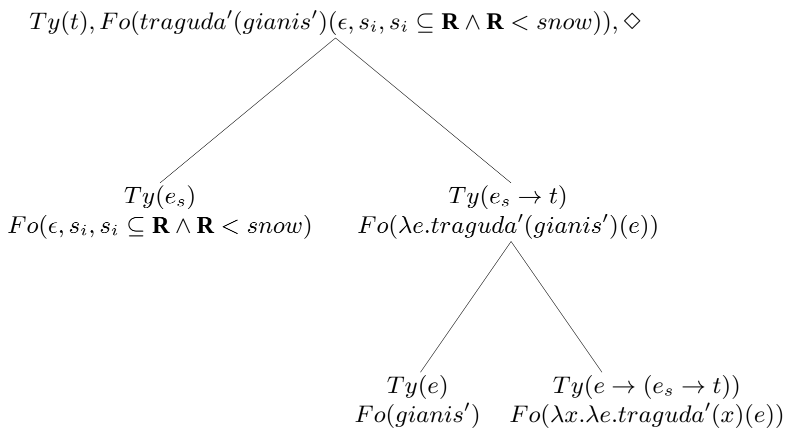
(32) The LOFT modalities in action



In the above tree, all nodes have a treenode address and a further statement identifying another node in the tree. For example, the statement $\langle\uparrow_0\rangle\langle\downarrow_1\rangle Tn(011)$ found in the 010 node reads as: you will find treenode 011 if you take a step across the 0 mother relation followed by a step across the 1 daughter relation. Furthermore, the two kleene operators * and + are used in combination with the basic tree modalities, denoting the reflexive transitive and the transitive closure of the modality in each case. Thus, $\langle \downarrow^{*} \rangle Tn(x)$ reads as $Tn(x)$ holds at the current node or at a node below the current one (of arbitrary depth), whereas $\langle \downarrow^{+} \rangle Tn(x)$ reads as $Tn(x)$ holds on a node below the current one.

Treenodes are inhabited by labels expressing Type/Formula value information, tree-logic modalities and unfulfilled requirements of all the latter sort. The end result of every successful parse of a given natural language string involves a binary tree, in which all the nodes have type and formula values and no outstanding requirements exist. Such a complete tree depicting the parse of the intransitive struture *O Gianis traguda* 'John sings' in CG is shown below:[4]

(33) A complete parse of *O Gianis traguda* 'John sings'



A complete DS tree is a binary tree where every node carries a Formula and a Type (Fo and Ty respectively) value (among potentially other information). In the above tree all nodes carry formula and type value information. Formula and Type values combine via functional application and modus ponens respectively. The lower functor node (011 node) represents the node where the Fo and Ty value of the word/concept of traγuδa is projected. In the same vein, the 010 node represents the node where the subject is projected whereas the 00 node and the nodes below it represent the situation argument node, i.e. the locus of tense/aspect information.[5] The formal details of the information on the situation argument node shall not bother us here. However, the interested reader is directed to Cann (2011) and Chatzikyriakidis (2010, 2011) for a detailed explanation.

2.1. Lexical entries and Computational rules

Treenode decorations and requirements comprise the declarative part of the DS system. However, DS further employs a number of procedural mechanisms that project the features of the declarative system. The basic devices behind the DS procedural system, are lexical and transition/computational rules (henceforth computational rules). The first are language-specific rules projected from the words of a given language. These are the lexical entries each word is assumed to involve. On the other hand, computational rules are language-general rules and, as such, are assumed to hold for every language. Lexical actions in DS are encoded in a simple algorithmic IF THEN ELSE format. The IF part contains a statement(s) that should be true or false with respect to the partial tree at the time the word comes into parse. If this statement is satisfied, the actions encoded in the THEN part of the entry are projected. There is an elsewhere statement (ELSE) that might contain alternative information in case the first triggering point is not satisfied. The examples below present a sample entry for the proper noun *Bill* in English:

[4]The situation argument node (the node with type e_s has complex structure which we do not show for ease of exposition. This structure will be made explicit later on if needed.

[5]This is based on the assumption that every sentence has an explicit situation argument. See Gregoromichelaki (2006), Cann (2011) and Tenny & Pustejovsky (2000) for the relevant argumentation.

(34) Lexical entry for the proper noun *Bill* in English

IF	$?Ty(e)$
THEN	$\texttt{put}((Ty(e), Fo(Bill'), [\downarrow]\bot)$
ELSE	abort

The above example reads as follows: if you are at a node that has a type *e* requirement, then decorate this node with a type *e* value and a formula value representing the concept *'Bill'*. In any other case abort. In that sense, a proper noun like *Bill* in English will be able to get parsed as soon as a node has a requirement for a type *e*. This will allow a word like *Bill* to be parsed either as a subject or as an object in English. Languages with overt NP case marking are assumed to involve further restrictions, encoded as output filters, that will ensure that an NP marked for structural accusative for example will end up on the direct object node. Such an entry is shown below:

(35) Lexical entry for an NP marked for structural accusative

IF	$?Ty(e)$
THEN	$\texttt{put}((Ty(e), Fo(x'), ?\langle\uparrow_0\rangle Ty(e \rightarrow t), [\downarrow]\bot)$
ELSE	abort

In the above lexical entry, the accusative marked NP posits a requirement that the predicate node ($Ty(e \rightarrow t)$) is found in case a step across the zero daughter relation is taken from the current node, in effect identifying the NP's position with the direct object node. Such an encoding of case has the advantage of not having to posit case features. Case under such an account is seen as a restriction on the place in the overall configuration at which the term will end up.[6] Besides projecting information on nodes already present in the partial tree, lexical entries can also construct nodes. For example, a lexical entry for a monotransitive verb in a typical pro-drop language like CG will involve unfolding of the whole propositional template. The verb starting from the type *t* node will build the whole propositional template and further decorate the 011 node with a type and formula value, representing the adicity and the LOT concept of the verb in question. The pointer is then assumed to be returned to the type *t* requiring node. This will allow parsing of both VSO and VOS structures, since the pointer can move to both the subject and the object node by ANTICIPATION as we shall see shortly. The lexical entry for the monotransitive verb *vlepi* 'sees' in CG is shown below:

(36) Lexical entry for a monotransitive verb in CG[7]

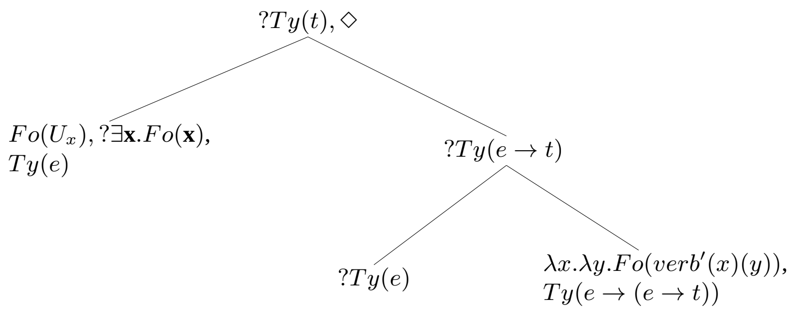
IF	$?Ty(t)$
THEN	$\texttt{make}(\langle\downarrow_1\rangle); \texttt{go}(\langle\downarrow_1\rangle); \texttt{put}(?Ty(e \rightarrow t);$
	$\texttt{make}(\langle\downarrow_1\rangle); \texttt{go}(\langle\downarrow_1\rangle);$
	$\texttt{put}(Fo(\lambda x.\lambda y.verb'(x)(y)), Ty(e \rightarrow e(\rightarrow t)), [\downarrow]\bot);$
	$\texttt{go}(\langle\uparrow_1\rangle); \texttt{make}(\langle\downarrow_0\rangle); \texttt{go}(\langle\downarrow_0\rangle); \texttt{put}(?Ty(e));$
	$\texttt{go}(\langle\uparrow_0\rangle\langle\uparrow_1\rangle); \texttt{make}(\langle\downarrow_0\rangle); \texttt{go}(\langle\downarrow_0\rangle);$
	$\texttt{put}(Ty(e), Fo(U_x)), gofirst(?Ty(t))$
ELSE	abort

The effect of parsing *vlepi* in tree notation is shown below:

(37) Parsing a monotransitive verb in CG

[6] If one wants to be accurate, such an account of case functioning as an output filter will not disallow an NP from being parsed in other nodes that do not satisfy the case restriction. However, if the NP is parsed on these nodes the case filter requirement will never be satisfied and thus the parse will never be successful. Note that no account of case of determiners or adjectives is going to be given or discussed in this paper.

[7] The subscript x on the U metavariable represents the restrictions that the subject metavariable will bear. I will not get into the details as regards these restrictions or subject agreement in general.



Verbs in CG, just like verbs in SMG, are assumed to build the whole propositional structure and decorate the subject node with a type *e* value and a formula metavariable. This is done in order to capture the subject pro-drop properties of the language. Update of metavariables into proper Fo values can be done either by the context or by the natural language string itself. Update by the context is done using the rule of SUBSTITUTION, which substitutes an *Fo* metavariable by a proper *Fo* value from the context in case this *Fo* value does not appear anywhere in the local domain of the tree under construction.[8] On the other hand, update by the natural language string itself is done when the lexical entry of a word provides an *Fo* value for the *Fo* metavariable. Given these two possibilities of metavariable update, the subject can be overt or covert in CG. Note that such an update from the context is not possible for the object node, since no metavariable and type value are projected in the object node but only a type requirement. In that sense, overt material is needed in order to fill the object slot, thus no object pro-drop being possible in CG. There is a great deal of parametric variation with respect to the entries of lexical verbs in different languages in DS. For example, non pro-drop languages like English will project a requirement for a type *e* in both the subject node and object nodes (see Cann et al., 2005: 48), while full pro-drop languages like Japanese or presumably Latin (on the assumption that Latin is a full pro-drop language) will project a type value and a formula metavariable in both the subject and the object nodes (Kempson & Kurosawa, forthcoming: 7). Lexical actions are thus the domain of parametric variation in DS. As such, syntactic differences across languages are assumed to derive from different lexical specifications, i.e. different lexicons.

The algorithmic format of lexical entries can be of arbitrary complexity. Embedded IF THEN schemas can appear within the initial IF THEN schema and so on. A lexical entry with the structure shown below is perfectly legitimate:

(38) An embedded IF THEN lexical entry format

IF	...	
THEN	IF	...
	THEN	...
		...
	ELSE	abort
ELSE	abort	

A number of partial recursions on specific parts of the algorithm may be used. For example, an entry containing a disjunctive IF part will be denoted by separating the two disjunctive parts with the symbol |:

(39) An embedded IF THEN with a disjunctive IF part

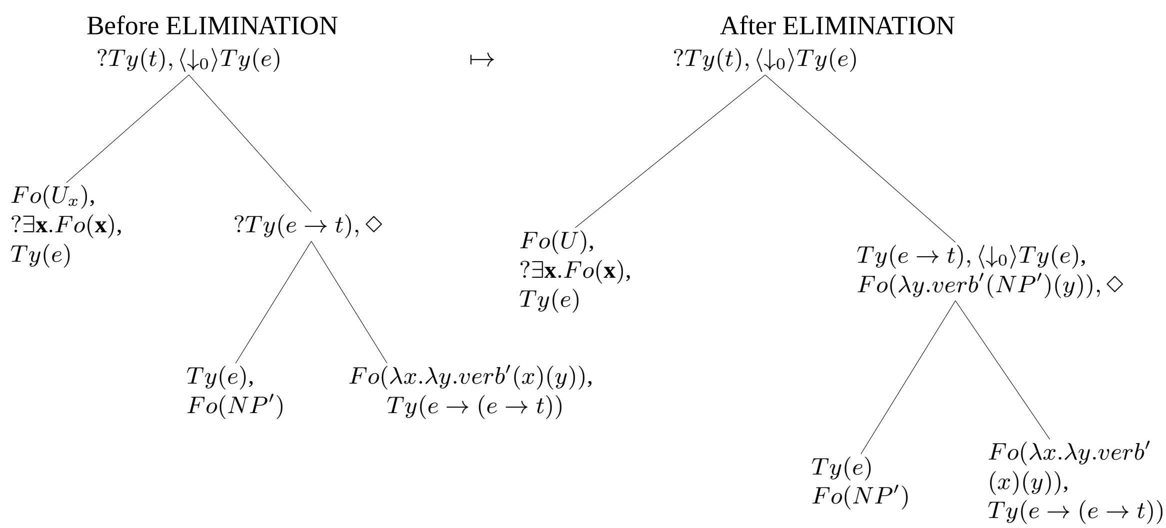
[8] See Cann et al., 2005: chapter 2 for the exact definition of SUBSTITUTION.

```
IF          ...
THEN        IF        ...|
                        ...
            THEN   ...
            ELSE   abort
ELSE abort
```

Computational rules are general computational devices, comprising the basic tree construction mechanism. Formally, computational rules involve an input and an output description. The former designates where the pointer must be along with information about the node that the pointer is on as well as information about other nodes with respect to the pointer node, while the latter shows the transformation of the input in terms of requirements, adding nodes, pointer movement and so on.

Computational rules are assumed to be a closed set of rules universally available to every language. This closed set of rules are basically rules that help the parsing process unfold. These involve pointer movement rules, rules that perform functional application and modus ponens (for formulas and types respectively) or rules that get rid of requirements as soon as these are satisfied. A characteristic case of a computational rule is the rule of ELIMINATION. This rule basically performs functional application on formulas and modus ponens on types, in case both the argument and the function nodes bear complete formula and type value:[9]

(40) ELIMINATION



Additional rules involve ANTICIPATION, which moves the pointer to a daughter node in case an unfulfilled requirement exists, COMPLETION, which moves the pointer to the mother node in case a type requirement is satisfied in the daughter node and THINNING, which eliminates a requirement as soon this is satisfied. The rules in their formal format are shown in the appendix.

2.2. Structural underspecification - Unfixed nodes

2.2.1. Structural Underspecification - The ADJUNCTION Rules

A central concept in DS is the assumption that natural languages are to a large extent underspecified regarding both content and structure. And while content underspecification has been largely employed within the formal semantics literature of the

[9] We show the effect of these rules in tree notation. See appendix A for the actual rules.

past 30 years, no attempts to move underspecification into the area of syntax have been made.[10] We have already seen how DS deals with semantic underspecification by employing metavariables as content placeholders. Structural underspecification on the other hand, is encoded in DS using a family of computational rules, the ADJUNCTION rules. The function of these rules is to introduce unfixed nodes, i.e. nodes that have not found their fixed position in the tree at the time of their introduction. The first of the ADJUNCTION rules we are going to see is the rule of *ADJUNCTION. According to this rule, an unfixed node marked with a type *e* requirement is projected from a type *t* requiring node. The effect of the above rule in tree notation is shown below:

(41) The effect of the *ADJUNCTION rule

$$Tn(n), ?Ty(t)$$

$$\langle\uparrow^*\rangle Tn(n),$$
$$?Ty(e),$$
$$?\exists\mathbf{x}.Tn(\mathbf{x}), \Diamond$$

The pointer is left at the type *e* requiring node. The node is structurally underspecified since it does not carry a fixed treenode address. The only thing the node "knows" as regards its treenode position is that somewhere up above or at the current node, *Tn(n)* must be found.[11] The pointer is left at the type *e* requiring unfixed node. At that point, given that a lexical entry for an NP will have a type *e* requiring trigger, it can be parsed on that unfixed node. This will be on a node whose structural position in the tree is not yet known. The *ADJUNCTION rule works neatly for left dislocated structures like OV focus structures.[12] In these cases, the preposed object is assumed to be parsed on an unfixed node:

(42) Parsing *tuto to vivlio* 'this book' in *tuto to vivlio agorasa* 'This is the book I bought' on an unfixed node[13]

$$?Ty(t), Tn(n)$$

$$\langle\uparrow^*\rangle Tn(n),$$
$$Ty(e), Fo(vivlio'),$$
$$?\exists\mathbf{x}.Tn(\mathbf{x}), \Diamond$$

The pointer moves up to the type *t* requiring node via COMPLETION and the verb can now be parsed. The structure we get after parsing the verb, given the entry for monotransitive verbs shown in (36), is the following:

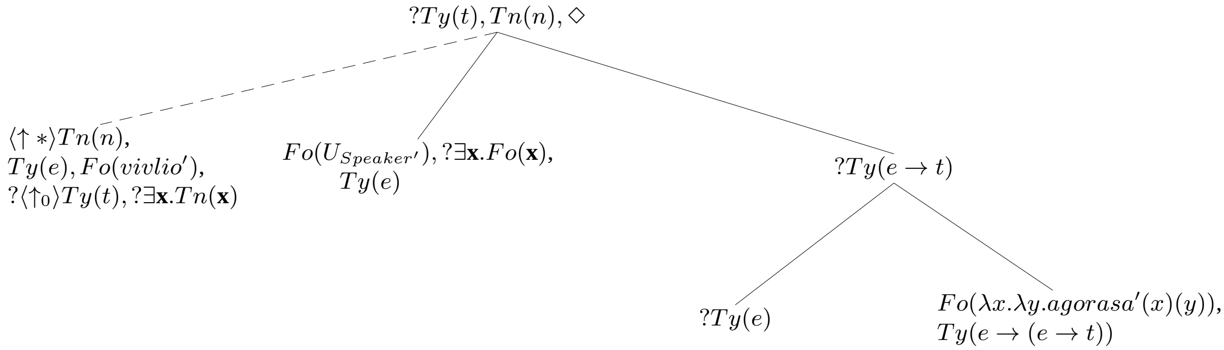
(43) Parsing *agorasa* 'hit' in *tuto to vivlio agorasa* 'This is the book I bought'

[10] With the exception possibly being the notion of functional uncertainty, formalized within Lexical-Functional Grammar (Bresnan, 2001).

[11] Notice that the reflexive satisfaction of the * in which the node unifies with its host trivially is not possible. *Tn(n)* is the treenode address of the type *t* requiring node. In that sense, the only way such a reflexive satisfaction will hold is in case the unfixed node unifies with the type t requiring node. However, such unification is impossible, given the incompatible specifications of the respective type requirements (*?e* and *?t*).

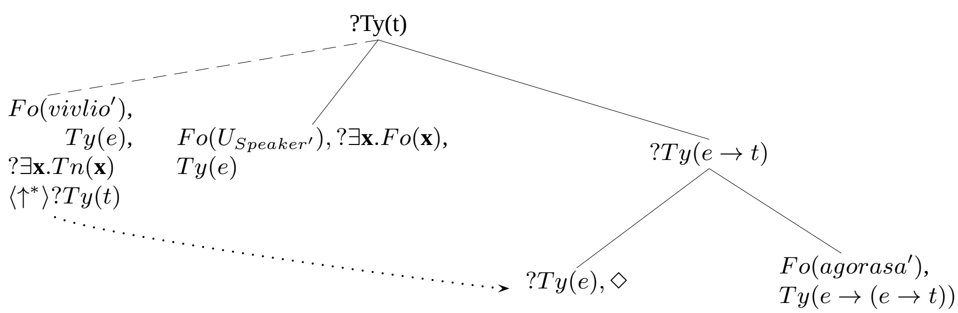
[12] Even though object FOCUS is not an option in CG in general but only with specific NPs like for example, deictic elements (see Tsiplakou et al., 2007).

[13] I ignore determiners for the moment.

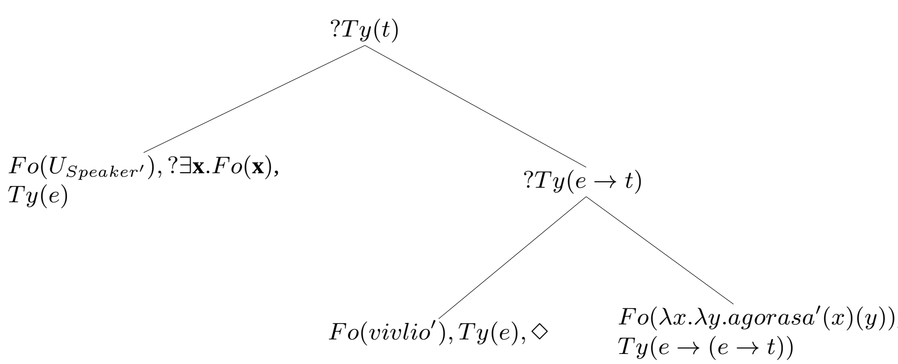


In the above structure there is an unfixed node with a type and a formula value and an open slot decorated with a type *e* requirement (the 010 node). It is at that point that a process of unification between the unfixed node and the fixed object node (010) can take place using MERGE. MERGE is a computational rule which unifies two nodes just in case one of the two updates the treenode address of the other.[14] The notion of update is defined by treenode address entailment. If a treenode address entails another treenode address, then the former can be seen as an update of the latter. In that sense an underspecified address like $\langle\downarrow^*\rangle$Tn(a) can be updated to a more specified address like $\langle\downarrow\rangle$Tn(a) but not vice versa. Furthermore, the two nodes must not bear any conflicting specifications. In example (43), the treenode address of the direct object node (010) can be a proper update of the underspecified address the unfixed node carries. Furthermore, the fact that the 010 node has a proper treenode address will eliminate the requirement of the unfixed node that a fixed treenode address should be found ($?\exists\mathbf{x}Tn(\mathbf{x})$). Note that no conflicting specifications exist, since the fixed node (DU') has a type *e* requirement and the unfixed node (*DU*) a type *e* value. Given that a type *e* value is an update of the type *e* requirement, no conflict arises. The trees below display the tree structure before and after MERGE has applied:[15]

(44) Before and after MERGE of the unfixed node



(45)



[14]See Appendix A for the actual rule.

[15]The steps of THINNING, eliminating the type *e* requirement as well as the requirement $?\exists\mathbf{x}Tn(\mathbf{x})$ are not shown.

Notice that the rule of *ADJUNCTION will also work for cases where the left dislocated element is embedded in a complement clause. The modality associated with the *ADJUNCTION rule makes MERGE of the unfixed node with a type *e* node that is deeply nested possible. The only requirement is that this node must be found within the same tree structure. There are a number of other ADJUNCTION rules but I will not present them here for reasons of both relevance and space. However, the interested reader is directed to Cann et al. (2005) and Chatzikyriakidis (2010) for more details on the ADJUNCTION family of rules.

2.3. *Parsing in Context - LINK Structures*

Besides the tree structures in which each sentence involves a single tree (regardless of tree embedding), DS also makes use of pairs of trees which are linked to each other via a relation called LINK. LINK structures involve two separate tree structures linked by means of an arrow relation (LINK), that share in most of the cases a term. The node from which the LINK starts can be seen as setting the context in which the LINKed tree is going to be parsed. Examples of LINK relations include relative clauses, in which case the relative clause is parsed within the context of the head or HTLD constructions in which case the HTLD sentence is parsed within the context of having parsed the left-dislocated element first. LINK structures have a variety of uses in DS. Let us illustrate LINK structures in more detail by looking at an HTLD example. In order to analyze HTLD constructions, Cann et al. (2005) define two rules which link a type *e* node where the dislocated element is parsed to a type t requiring node, where the rest of the HTLD structure is parsed. The first rule introduces a LINK transition from a type *e* requiring node to a type *t* requiring node, while leaving the pointer in the first of the two. The second rule introduces a requirement for a shared term as soon as the dislocated element is parsed. In parsing an HTLD sentence like the one shown in (46), we apply the first rule (TOPIC STRUCTURE INTRODUCTION, see Appendix A) that introduces the type *e* requiring node. The NP is parsed on that node and then the second rule (TOPIC STRUCTURE REQUIREMENT, see Appendix A) takes effect moving the pointer to the type *t* requiring node and positing (on the same node) a requirement that a copy of the formula found in the node where the LINK begins must be found somewhere in the LINKed tree or to a tree LINked to the LINKed tree ($?\langle D\rangle Fo(Giorgos')$):

(46) *O* *Giorgos,* *gnorizo* *ton.*
the.NOM George.NOM know.1SG him.CL-ACC
'I know George'

(47) $\langle L\rangle Tn(0)$, $Fo(giorgos'), [\downarrow]\top$, $Ty(e)$ → $\langle L^{-1}\rangle Tn(n), ?Ty(t), ?\langle D\rangle Fo(giorgos')$

3. A DS analysis

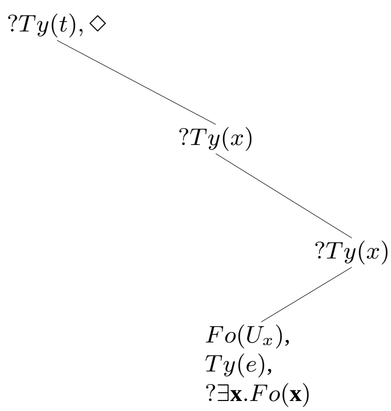
3.1. *Preliminaries on clitics in DS*

Being a lexicalized framework, DS assumes parametric variation in natural languages to reside in the lexicon. Therefore, one could plausibly point out that DS, like any other syntactic framework, is vulnerable to the affix-word debate, since a decision with respect to the status of elements like clitics will have further consequences on whether such elements will have their own lexical entry or rather be part of the lexical entry of a larger unit (the verbal complex in this case). However, this assumption is only superficially correct since in DS every element, no matter its traditional characterization (as a word or affix), can have its own lexical entry. The only requirement is that this element provides distinct procedural information on how the parse should/must or should not/must not proceed. This might sound controversial but becomes fairly straightforward assuming that for DS lexical entries are just pieces of information on how the parsing process proceeds. In that respect, every element providing such information can have its own lexical entry in DS. A classic illustration of that is the analysis of Japanese case suffixes put forth by Cann et al. (2005). In this analysis, Japanese case suffixes involve a separate lexical entry and are not part of the entry for the NP preceding these suffixes. Their role is to fix the locally unfixed node the preceding NP is parsed on, in order for multiple scrambling to be allowed (see Cann et al., 2005: 236-240 for details).[16] Given this view on lexical entries, the affix-word debate is largely irrelevant to DS, since any DS account can be compatible with any pre-theoretical decision on the affix-word debate. It is rather uncontroversial under this view that clitics will provide distinct procedural infromation with respect to the parsing process, thus involving their own lexical entries. This is the stance I am going to take in this paper, following earlier approaches within the same framework notably Bouzouita (2008a,b), Cann et al. (2005), Chatzikyriakidis (2009a,b, 2010, 2011), Cann & Kempson (2008) and Gregoromichelaki (2010) among others.

3.2. The enclitic environments

I will begin our discussion on CG clitic positioning by using the 3rd person accusative clitic *to* ‘it’ as the role model. One of the issues that need to be resolved as regards this type of clitic is the actions projected via its lexical entry. Following Chatzikyriakidis (2010), I take 3rd person accusative clitics to project fixed structure. This would basicaly mean that 3rd person accusative clitics are always interpreted as direct objects. This seems to be correct for CG, since no double accusative constructions exist in CG. In that sense in parsing a 3rd person clitic in CG, the actions found in the lexical entry for the clitic build the 01,011 and 0110 (the direct object node) nodes and decorate the latter node with a type value (type e) and a formula metavariable along with a proper formula value to be found later on and return the pointer to the initial type t node. The effect of parsing a 3rd person accusative clitic is shown below:

(48) The effect of parsing a 3rd person accusative clitic



With the actions projected by the clitic in line, the next step is to see how enclitic positioning is going to be captured. Hopefully, there is a straightforward way to formalize the entry in order for enclitic positioning to be captured. What we need to do is provide a triggering point which will basically abort in case a functor node bears a type value, i.e. in case any kind of verb has already been parsed. The lexical entry for CG clitics capturing enclitic positioning only is shown below:

(49) Lexical entry capturing enclitic positioning in CG:[17]

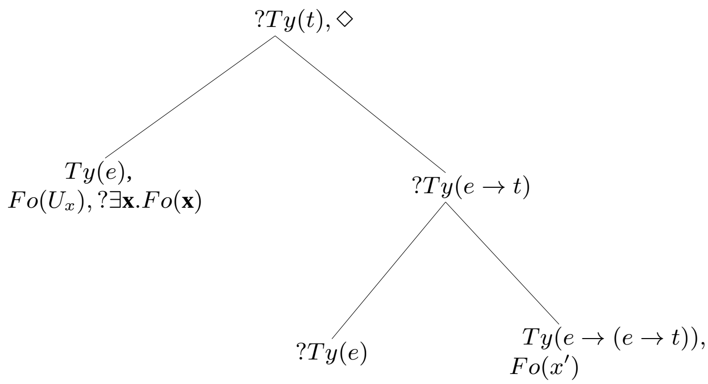
[16] This is due to the “no more than one unfixed node at a time constraint”, put forth in Cann & Kempson, 2008 and Chatzikyriakidis & Kempson, 2010.

[17] The bold parentheses indicate optionality of application depending on the assumption one makes as regards the presence or not of the higher situation node. In the absence of these higher nodes, application of the *make/go* functions inside the bold parentheses is not performed. On the contrary, in the presence of the higher situation nodes, application of the *make/go* functions proceeds normally.

IF $?Ty(t)$
THEN IF $\langle\downarrow_1^+\rangle Ty(x)$
THEN $\mathbf{(}\mathtt{make}((\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle)); \mathbf{)}$
$\mathtt{make}(\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle);$
$\mathtt{make}(\langle\downarrow_0\rangle); \mathtt{go}(\langle\downarrow_0\rangle)$
$\mathtt{put}(Ty(e), Fo(U_x), ?\exists\mathbf{x}.Fo(\mathbf{x});$
$gofirst(?Ty(t)))$
ELSE abort
ELSE abort

The above entry will correctly predict sentences like (1a.) and (2a.) to be grammatical according to fact. Assuming that a verb has been parsed, a verbal type will be projected in a functor node. Which functor node that will be depends on the verb's adicity. For example parsing of a transitive verb will result in the projection of a verbal type in the 011 node:

(50) Parsing a transitive verb (the higher situation nodes are omitted for ease of exposition)



At this point the clitic comes into parse. Both its triggers are satisfied, since the pointer is at the type *t* requiring node and a type value exists in one of the functor nodes, and as such the parsing process can continue.[18] The same mechanism is at play in parsing imperative verbs. Imperative verbs will also project a type value in a functor node. In that sense, enclitic positioning with imperatives is predicted in exactly the same way as with indicatives. Of course, such an entry will not suffice to capture the whole range of positioning since proclitic positioning is still left unaccounted for.[19] The next step is to look at the way proclitic positioning is going to be captured.

3.3. The Proclitic Environments

The set of proclitic triggers found in CG is anything but homogeneous, as it consists of a number of differently functioning elements: modality/tense markers, wh-elements, fronted constituents and subordinating conjunctions among others. However, a number of generalizations can be captured by looking into the nature of these triggers more carefully.

[18]The meticulous reader should have noted that the entry for the clitic has a mechanism for building nodes (make). However, in the case we are discussing, these nodes are already there by means of parsing the verb. The nodes built from the entry of the clitic will basically collapse with the already existing nodes, given that they do not carry any incompatible descriptions. This is a general feature of the underlying tree description language according to which any two nodes with compatible descriptions can harmlessly coalesce. An anonymous reviewer is thanked for pointing this out to me.

[19]Note that such an entry will suffice for systems like Pontic Greek (PG), where clitics appear postverbally in all environments (Drettas, 1997; Chatzikyriakidis, 2010 among others).

3.3.1. Unfixed Nodes as Proclitic Triggers

In the introduction to the DS framework, we have discussed the use of structural underspcification via means of the ADJUNCTION rules. Even though, there are a number of variants of the ADJUNCTION rule we only presented one variant, *ADJUNCTION. Surprisingly *ADJUNCTION can be used in order to capture a generalization across elements triggering proclisis in CG. Bouzouita (2008a,b) in discussing clitic positioning in MedSp, argued that one of proclitic triggers in MedSp is the existence of an unfixed node. This was based on standard assumptions in DS, according to which Wh-elements and fronted constituents are or can be parsed on an unfixed node (Kempson et al., 2001; Cann et al., 2005). This assumption extends straightforwardly to CG, since the similarity of clitic positioning in the two systems is striking. In that sense, the first generalization that can be drawn is that the existence of an unfixed node constitutes a trigger for proclisis. Note that this proclitic trigger does not refer to the actual content of the elements triggering proclisis but rather to the parsing strategy used (unfixed node). But let us see in detail the basis of this intuition. Let us start with Wh-elements:

(51) *Pios* *ton* *ikseri* *ton* *Giorko?*
who.NOM him.CL-ACC knows.3SG the.ACC George.ACC
'Who knows George?'

Following Kempson et al. (2001), I take wh-elements to decorate an unfixed node, marking the initial node with a Q feature, its role being the identification of the wh-element as an interrogative wh-element. The effect is that the unfixed node is decorated with a type value and a specialized formula metavariable WH. The entry below is the lexical entry for *pios* 'who.NOM' in CG:

(52) Lexical entry for pios in CG

IF	$?Ty(e)$	
THEN	IF	$\langle\uparrow_*\rangle ?Ty(t), ?\exists\mathbf{x}.Tn(\mathbf{x})$
	THEN	go($\langle\uparrow_*\rangle$); put($Cat(Q)$);
		go($\langle\downarrow_*\rangle$); put(**WH**, $Ty(e)$,
		$?\langle\uparrow_0\rangle Ty(t), [\downarrow]\bot$)
	ELSE	abort
ELSE	abort	

Notice the existence of the case filter $?\langle\uparrow_0\rangle Ty(t)$, identifying the unfixed node as immediately dominated by a type *t* node, i.e. as being the subject node (in effect a nominative case filter).

In the same vein, fronted constituents can be modelled as decorating unfixed nodes. A fronted NP at the left periphery can be parsed on an unfixed node, finding its position in the tree later on in the parse. The same reasoning is used for fronted adverbs or any other fronted element. Within this line of reasoning, we can formulate a trigger which will allow the clitic to be parsed in case an unfixed node is present in the tree structure. The entry below captures enclitic positioning plus proclitic positioning with elements parsed on an unfixed node:

(53) Lexical entry containing the enclitic and the unfixed node trigger:

IF	$?Ty(t)$	
THEN	IF	$\langle\downarrow_1^+\rangle Ty(x)\vert$
		$\langle\downarrow_*\rangle Ty(x), ?\exists\mathbf{x}.Tn(\mathbf{x})$
	THEN	**(**make(($\langle\downarrow_1\rangle$); go($\langle\downarrow_1\rangle$)); **)**
		make($\langle\downarrow_1\rangle$); go($\langle\downarrow_1\rangle$);
		make($\langle\downarrow_0\rangle$); go($\langle\downarrow_0\rangle$)
		put($Ty(e), Fo(U_x), ?\exists\mathbf{x}.Fo(\mathbf{x})$;
		$gofirst(?Ty(t)))$
	ELSE	abort
ELSE	abort	

Such a trigger will ensure that the clitic will appear before the verb in case an unfixed node is present in the tree structure. Note that the specification is not restricted to unfixed nodes that are type specific. Actually, the specification proposed (following the formulation found in Bouzouita, 2008a) does not refer to the type value of the unfixed node or to its formula value. Such a specification will allow us to account for cases where a strong pronoun is parsed on an unfixed node, in which case a metavariable instead of a proper formula value will have been projected and proclisis obtains:

(54) *Ego* *tu* *edoka* *lefta.*
I.NOMINATIVE him.GEN gave.1SG money.ACC
'It was me that I gave him money.'

However, given the nature of inclusive disjunction, a clitic can be parsed in case both of its triggers are satisfied, i.e. in case both a verbal type and an unfixed node exist. Such a fact will fallaciously predict that variation is possible with e.g. Wh-elements. In that sense, we will have to encode an exclusive version of disjunction according to which the parse cannot proceed in case both triggering points are satisfied. The revised lexical entry is shown below (exclusive disjunction is noted as ||):

(55) Updated lexical entry containing the enclitic and the unfixed node trigger:

IF	$?Ty(t)$		
THEN	IF		$\langle\downarrow_1^+\rangle Ty(x)\vert\vert$
			$\langle\downarrow_*\rangle Ty(x), ?\exists\mathbf{x}.Tn(\mathbf{x})$
	THEN		$\mathbf{(}\mathtt{make}((\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle)); \mathbf{)}$
			$\mathtt{make}(\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle);$
			$\mathtt{make}(\langle\downarrow_0\rangle); \mathtt{go}(\langle\downarrow_0\rangle)$
			$\mathtt{put}(Ty(e), Fo(U_x), ?\exists\mathbf{x}.Fo(\mathbf{x});$
			$gofirst(?Ty(t)))$
	ELSE	abort	
ELSE	abort		

Note that the use of unfixed nodes for the parse of fronted constituents is one of the options of parsing these elements in DS and as such will not predict categorical proclisis with fronted constituents but on the contrary will predict variation in this case. This is indeed the case with fronted subjects, adverbs and PPs, and has been shown to be the case for CLLD as well (Pappas, 2010). Such a situation is easily predicted given the availability of different processing strategies for the one and the same element, given that fronted elements can be parsed by making use of more than one alternative parsing strategies (unfixed node and LINK), and given that only the unfixed node strategy is associated with proclisis, we expect variation to be possible. Furthermore, the assumption that non-focused fronted elements can also trigger proclisis is also predicted given such an account, since the argument in the account given is that proclisis is associated with a processing strategy, i.e. the unfixed node strategy, and not with a some kind of pragmatic focus feature or a specific strategy for parsing focus structures. The fact that focused elements and proclisis-inducing may to a large extent overlap is a wholly different issue.[20]

3.4. Subordinating Conjunctions, Modality/Tense marker, Negation

As already mentioned, subordinating conjunctions, with the exception of causal subordinators, trigger proclisis in CG. For these specific both proclisis and enclisis are possible. The crucial question to ask here is what is the general characteristic that links all these conjunctions. In order to answer this, we need to see how subordination is analyzed in DS. Gregoromichelaki

[20]Note that the availability of different parsing strategies is one of the main assumptions made in DS and is argued to be one of the reasons of diachronic change in both Bouzouita (2008a) and Chatzikyriakidis (2010). The argument there is that this availability of alternative parsing options may lead to production/parsing mismatches that in effect can lead to syntactic change (see Bouzouita, 2008a and Chatzikyriakidis, 2010: chapter 8). See also section 4 in this paper.

(2006), in analyzing conditionals in DS, assumes that subordination includes pairs of LINKed trees sharing a situation argument. For example, in conditional structures, the IF clause is assumed to be linked with the THEN clause via means of a LINK relation. The LINK relation is built via the ACTIONS of the conditional conjunction *if*. The lexical entry for *if*, links two situation argproclisis/enclisisument nodes with the requirement that these two are shared. The assumption is that the situation argument node of the subordinate will act as the context in which the main sentence (or the THEN clause in conditionals) is going to be parsed. For example, in the case of conditionals the context in which the THEN clause is going to be true is the set of all situations of the type depicted in the IF clause.[21] In a similar way, Chatzikyriakidis (2010), argues that subordinating conjunctions LINK two independent trees and project a requirement for a situation argument node in the LINKed tree, i.e. the tree where the subordinate clause is going to be parsed. A way to look at this requirement, is to think of it as an action for the establishment of a new propositional domain. Parsing one of the subordinate conjunctions signals a new propositional domain and a way to signal this domain is by projecting a requirement for a situation argument node, i.e. the need for a new situation/event. There is a lot of formal detail behind this intuition, but this will not bother us here for reasons of relevance, since a DS internal discussion is highly irrelevant for the needs of this paper. However, the full formal details of such a proposal can be found in Chatzikyriakidis (2010). What is needed for the needs of the present analysis is the assumption that conditional/temporal/cause[22] and in general subordinating conjunctions build the internal type e_s requiring node of the LINKed tree and leave the pointer at the most local type *t* node. Given this property of subordinating conjunctions to build the type e_s requiring node, we can assume a lexical trigger which proceeds in case a type e_s requirement exists in the immediately dominated argument daughter. This lexical trigger is again an item neutral trigger that refers to a parsing strategy rather than to a specific lexical item. Given the existence of a situation argument node requirement, the clitic can be parsed. The lexical entry with the added trigger is shown below:

(56) Lexical entry containing the enclitic, the unfixed and the type e_s requiring node trigger:

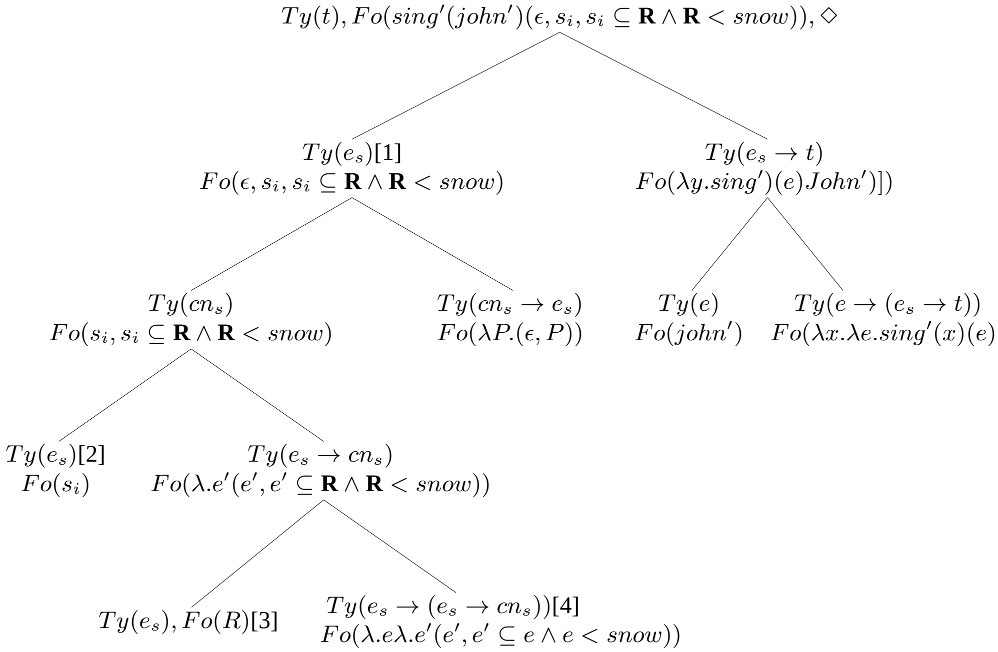
IF	$?Ty(t)$		
THEN	IF	$\langle\downarrow_1^+\rangle Ty(x)\|\|$	
		$\langle\downarrow_*\rangle Ty(x), ?\exists\mathbf{x}.Tn(\mathbf{x})\|$	
		$\langle\downarrow_0\rangle ?Ty(e_s)$	
	THEN	$\mathbf{(}\mathtt{make}((\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle)); \mathbf{)}$	
		$\mathtt{make}(\langle\downarrow_1\rangle); \mathtt{go}(\langle\downarrow_1\rangle);$	
		$\mathtt{make}(\langle\downarrow_0\rangle); \mathtt{go}(\langle\downarrow_0\rangle)$	
		$\mathtt{put}(Ty(e), Fo(U_x), ?\exists\mathbf{x}.Fo(\mathbf{x});$	
		$gofirst(?Ty(t)))$	
	ELSE	abort	
ELSE	abort		

The type e_s requiring trigger correctly captures proclisis found with subordinating conjunctions. Surprisingly, the same line of reasoning will turn out to be extremely relevant for modality/tense markers as well. Let us explain by reverting to the simplest type of English example: *John sang*. In DS, tense and aspect properties are projected inside a complex situation argument node. This node, following standard DS assumptions with respect to NP structure, is assumed to involve complex structure. In the example below, illustrating the complete parse of the sentence *John sang*, tense properties are introduced in the lower functor node (noted as [4] in the tree), specifying that the event described (the singing event) took place in the past:

(57) Parsing *John sang*

[21] See Gregoromichelaki, 2006 on how the quantification over situations is achieved in this case.

[22] Variation presented with this subordinator will be dealt with later on.



Given this analysis of tense and aspect, let us think how the lexical entries for modality/tense markers are going to look like. Without going into the exact details of what is the exact contribution of these elements in terms of tense or aspect, we can safely assume that all these elements will project this kind of information in the internal functor node (the node marked [4]). This means that all these elements will have to build this node as well, given that they appear before the verb and given that the internal functor node will not have been constructed by the time they will come into parse. This last fact has the further consequence that the higher argument node (noted with [1] in the tree structure) has to be built as well, decorated with an e_s requirement.[23] The final consequence of all these assumptions is that when any of these elements is parsed, and assuming that the pointer is returned to the initial type *t* requiring node after parsing any of these elements, the clitic can be parsed given the trigger shown in (55), i.e. in the presence of a type e_s requirement. As already mentioned, the exact details of the lexical entries of these elements are not going to be fleshed out here, since whatever these details will be, the clitic positioning analysis can effectively stay the same. The relevant part common to all these entries is the construction of the e_s [1] node and its decoration with a type e_s requirement.[24] Given these assumptions, proclisis is predicted to obtain in case any modal/tense marker is parsed.

3.5. The Complementizer pu

The factive complementizer *pu* is another environment where proclisis arises in CG.[25] The example below exemplifies the relevant facts:

[23]Note that this is requirement for the type and not the actual type itself. When the verb is going to be parsed the rest of the information needed to compile the [1] node into a node with a complete type will be provided. The stadnard assumption is that when a verb, let us say an indicative verb is parsed, it provides all the necessary information to compile the situation argument node. Given this, after the verb has been parsed and having acquired a type in the situation argument node, the requirement for this node is eliminated and thus cannot function as a trigger anymore.

[24]The interested reader is however directed to Giannakidou (2007) for a discussion involving the formal semantics of the subjunctive marker *na* plus the SMG future/mood particle *θa*.

[25]Terzi (1999a: fn 19) mentions that some of the Cypriot speakers consulted indicated that enclisis with *pu* is also possible. I got the same judgments from one of the speakers consulted. All the other speakers indicated that proclisis is the only option. In this paper, I will assume that clitic placement is strictly proclitic with the factive complementizer *pu*. It will become evident when the analysis for the non-factive complementizer *oti* will be given, how such an account can be expanded to cover variation with *pu*, in case such variation is proven to exist in CG. See the relevant discussion as regards variation with *oti*.

(58) *Lipume* *pu* *ton* *ides* *(*ton).*
be-sorry.1SG COMP him.CL-ACC saw.2SG him.CL-ACC
'I'm sorry that you saw him.'

Proclisis with *pu* will be captured in the same sense proclisis with subordinating conjunctions is.[26] Under this analysis, parsing of *pu* will build the situation argument node decorating it with a requirement for a e_s type. The difference with subordinating conjunctions will lie in the fact that this requirement will be built in the main tree and not in a LINKed tree as it is the case with subordinating conjunctions. Such an analysis will unify the analysis of *pu* with the one given for subordinating conjunctions. Evidence that this account is on the correct track, is the fact that *pu* as well as subordinating conjunctions cannot be followed by any of the subjunctive markers *na* or *as*. With the trigger of these markers being the non-existence of a situation node, parsing of *pu* will immediately block further parsing of any of these two markers according to fact:

(59) **Lipame* *pu* *na/as* *ton* *dis.*
be-sorry.1SG COMP SUBJ him.CL-ACC saw.2SG
'I'm sorry that you saw him.'

Under this analysis, the actions of the complementizer *pu* will involve the construction of the situation argument node along with its decoration with a requirement for a e_s type:[27]

(60) Sample lexical entry for *pu*

IF	$?Ty(t), \langle\uparrow\rangle\top$
THEN	$\texttt{make}(\langle\downarrow_0\rangle); \texttt{go}(\langle\downarrow_0\rangle); \texttt{put}(?Ty(e_s));$ $...gofirst(?Ty(t))$
ELSE	abort

The above entry correctly captures proclisis with *pu*.

3.6. Negation

The last proclitic environment we are going to look at concerns cases involving the negation particles *en/min*, 'not'. As already said, proclisis obtains in the presence of these elements as exemplified below:

(61) *En* *ton* *iksero* *(*ton).*
NEG him.CL-ACC know.1SG him.CL-ACC
'I do not know him.'

(62) *Min* *ton* *dis* *(*ton).*
NEG him.CL-ACC see.2SG.PNP him.CL-ACC
'Do not see him.'

In Bouzouita (2008a,b), proclisis found in negative environments is accounted for by using a negative feature *[+NEG]*. This feature is assumed to be projected by the negation marker in the type *t* requiring node. However, there is a way to avoid the *[+NEG]* trigger and still be able to capture negation. There are two types of negation in CG, one used in indicative and one in subjunctive or negative imperative environments:

[26]See Chatzikyriakidis, 2010 for an alternative DS proposal based on Roussou's (2007) proposal that factive *pu* is the same as relative *pu*.

[27]The statement $\langle\uparrow\rangle\top$ says that in order for *pu* to be parsed, structure must exist above this node. This captures the fact that *pu* is parsed in an embedded type t node (in other words it introduces a verbal complement) and not on the initial type t requiring node.

(63) *En/*min ton ida.*
NEG him.CL-ACC saw.1SG
'I did not see him.'

(64) *Thelo na min/*en pais.*
want.1SG SUBJ NEG go.PNP
'I want you not to go/ I do not want you to go.'

(65) *Na min pais.*
SUBJ NEG go.PNP
'Do not go.'

Furthermore, the negation marker *min* when combined with the Perfective Non Past (PNP) verbal form can give rise to a negative imperative reading. In that sense, sentence (65) is also possible without the subjunctive marker *na* as witness the example below:

(66) *Min pais.*
NEG go.PNP
'Do not go.'

Thus, it seems that the different types of negation interact with modality/tense/aspect in different ways. Giving an analysis of negation in CG might then involve the contribution of a number of restrictions as regards modality or aspect by the two different negation markers that will capture the different semantics in each case as well as the combinatorial properties of the two. Of course, a study of what these restrictions actually are and how to formally encode them is well beyond the scope of this paper. However, given that modality/tense/aspect information is encoded inside the complex situation argument node, it is quite straightforward to assume that the negation markers will project their restrictions inside this node. Whatever the exact locus of the projection of this information (presumably the lower functor node), negation markers will have to built the higher type e_s node. In that respect, the lexical entry for negation markers will have to project at least a *[+NEG]* feature as well as the type e_s requiring node. The lexical entry is shown below:

(67) Lexical entry for negation

IF	$?Ty(t)$
THEN	$\texttt{make}(\langle\downarrow_0\rangle); \texttt{go}(\langle\downarrow_0\rangle);$
	$\texttt{put}(?Ty(e_s)), ...;$
	$gofirst(?Ty(t)), \texttt{put}([+NEG]).$
ELSE	abort

Given the above entry, we do not have to refer to the *[+NEG]* feature to capture negation, since proclisis with negation will be captured by the same mechanisms used for modality/tense markers as well as subordinating conjunctions, namely the type e_s requiring trigger. In that sense, the entry for the clitic stays the same as in (56).

With a third trigger added, we will have to see its interaction with the already presented triggers (the unfixed node and the enclitic trigger). The first thing we need to look at is the interaction of the type e$_s$ requiring trigger with the other proclitic trigger, namely the unfixed node trigger, i.e. whether constructions where both triggers are present are allowed in CG or not. Such structures are commonly found in CG and can involve a number of different structures, e.g. a fronted object+negation+clitic, a subordinating conjunction+fronted object+clitic, a wh+future particle+clitic and a number of similar constructions. In these cases, both triggers are satisfied, so the interpretation of the disjunction must be inclusive. An exclusive interpretation of disjunction will undergenerate, since it will rule out examples like the ones below:

(68)

An	*en*	*ton*	*iksero...*
if	NEG	him.CL-ACC	know.1SG

‘If I do not know him...’

(69)

Pios	*enna*	*ton*	*piasi*	*tilefono?*
who	FUT	him.CL-ACC	catch	phone

‘Who is going to call him?’

In the above example both an unfixed node and a type e_s requiring daughter node will exist when the clitic comes into parse. Assuming an exclusive interpretation of the disjunction between the unfixed node trigger and the type e_s requiring node trigger, examples like the above would be ruled out contrary to fact. Thus, an inclusive interpretation of disjunction is needed here. On the contrary, the interaction of the type e_s requiring trigger or/and the unfixed node trigger with the enclitic trigger is of the same type as the one between the unfixed node trigger and the enclitic trigger, i.e. the case where both a type e_s trigger and the enclitic trigger are satisfied will vastly overgenerate, since it will predict constructions like the ones shown below to be grammatical contrary to fact:

(70)

**An*	*dite*	*ton,...*
if	saw.2PL	him.CL-ACC

‘If you see him...’

(71)

**Pu*	*ida*	*ton,*	*milisa*	*tu.*
when	saw.1SG	him.CL-ACC	spoke.1SG	him.CL-GEN

‘When I saw him, I spoke to him.’

Thus, interpretation of disjunction in this case should be inclusive. The lexical entry thus stays as in (56).

3.7. Variation environments

3.7.1. The Case of oti

One of the facts that make clitic positioning in CG rather complex, is the existence of environments where variation in clitic positioning is possible. The most striking of these environments is the case of the non-factive complementizer *oti* ‘that’. This complementizer, unlike its factive counterpart discussed above, can trigger both proclisis and enclisis. The relevant facts are repeated below:

(72)

a.

Ipen	*oti*	*ethkiavasen*	*to.*
said.3SG	COMP	read.3SG	it.CL-ACC

b.

Ipen	*oti*	*to*	*ethkiavasen.*
said.3SG	COMP	it.CL-ACC	read.3SG

‘S/He said that s/he read it.’

Proclisis with *oti* is straightforwardly captured assuming an analysis in line with the approach I proposed for complementizer *pu*, i.e. an analysis where *oti* builds the situation argument node and decorates it with a type e_s requirement:

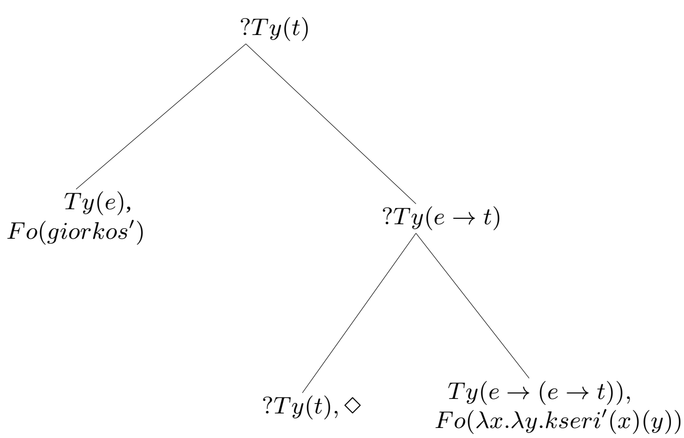
(73) Sample lexical entry for *oti*

IF $?Ty(t), \langle\uparrow^*\rangle\top$
THEN $\mathtt{make}(\langle\downarrow_0\rangle); \mathtt{go}(\langle\downarrow_0\rangle); \mathtt{put}(?Ty(e_s));$
$...gofirst(?Ty(t))$
ELSE abort

However, only half the problem is solved, since *oti* can also involve enclitic positioning. Nothing in the entry we have given above will predict this fact. So, what can explain enclitic positioning with *oti*? Looking at the diachrony of *oti*, specifically its behaviour in Medieval Modern Greek (MMG) and Medieval Cypriot Greek (MCG), might give us a way out of the problem. Clitic positioning with *oti* in MMG is prevalently postverbal, with 30 of the 38 tokens showing postverbal positioning and only 8 of them preverbal (Pappas, 2004). The situation in the Cypriot *chronicles of Makhairas* and *Boustronios* with respect to clitic positioning of *oti* is quite harder to tell. Pappas (2004) does not give a count for *oti* for MCG. A first look at the *chronicle of Makhairas*, reveals a number of cases involving *oti*. However, in their vast majority, these cases involve the use of *oti* as a causal subordinator and not as a non-factive complementizer. However, in both its guises, *oti* can be found with both proclisis and enclisis, with enclisis being prevalent in both cases (5/8 for non-factive *oti* and 36/40 for causal *oti*). I do not know if and how positioning of causal *oti* has influenced positioning with complementizer *oti* or vice versa. What is rather interesting is Pappas' (2004: 36) discussion concerning the classification of *oti* in the category clause initial. Pappas provides two interesting views on the function of *oti*, one given by Mackridge (1985) and one given by Jannaris (1968). Both these writers, argue that *oti* is actually more of a coordinating conjunction rather than a subordinating one. Mackridge states that *oti* shows a pattern of "incomplete subordination" while Jannaris (1968) that in many instances *oti* "corresponds to our modern colon(;)...". The use of *oti* as a coordinating rather than a subordinating conjunction might give us a way out of the variation problem. I will then assume that *oti*, can be also parsed as a coordinating conjunction. In this scenario, *oti* will build a LINK transition in the same sense coordinating conjunctions do in DS. There are a number of issues with respect to such a transition. Coordinating conjunctions induce this LINK transition from a type complete to a type requiring node of the same type. However, in the case of *oti* things are different, since no complete type *t* node will exist by the time *oti* will come into parse. This is because a main verb like *ksero* 'know' will subcategorize for an NP subject and a VP object. Therefore, when the complementizer *oti* will come into parse in a sentence like (74), no complete type *t* node will exist, since the value for the VP object (the internal type *t* requiring node) will have not yet been provided:[28]

(74) *O Giorkos kseri oti kseris ton.*
the George know.3SG COMP know.2SG him.CL-ACC
'George knows you know him.'

(75) Before parsing *oti* in *o Giorkos kseri oti kseris ton*



[28]Again, the situation nodes are omitted of ease of exposition.

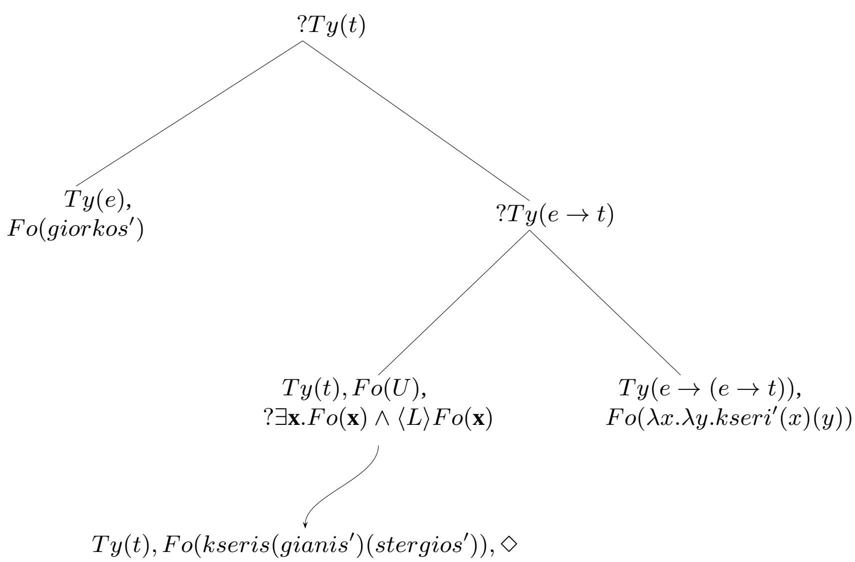
The pointer is at the internal type *t* requiring node. The subject has been parsed but no complete type *t* can be given until the embedded clause is parsed. I assume in that respect that *oti* projects a type *t* value and a formula metavariable ($Fo(U)$) in the embedded type *t* node with the requirement that this formula metavariable gets substituted by a proper formula value that is the same as the one found on the top of the LINKed tree ($?\exists\mathbf{x}.Fo(\mathbf{x}) \wedge \langle L\rangle Fo(\mathbf{x})$. Then, a LINK relation is created and the LINKed tree is created and decorated with a type *t* requirement ($\mathtt{make}(\langle L\rangle);\mathtt{go}(\langle L\rangle);\mathtt{put}(?Ty(t))$). The full lexical entry is shown below:

(76) Final lexical entry for *oti*

IF $?Ty(t), \langle\uparrow^*\rangle\top$
THEN $\mathtt{make}(\langle\downarrow_0\rangle);\mathtt{go}(\langle\downarrow_0\rangle);\mathtt{put}(?Ty(e_s))$
$...gofirst(?Ty(t))|$
$\mathtt{put}(Ty(t), Fo(U), ?\exists\mathbf{x}.Fo(\mathbf{x}) \wedge \langle L\rangle Fo(\mathbf{x}))$
$\mathtt{make}(\langle L\rangle);\mathtt{go}(\langle L\rangle);\mathtt{put}(?Ty(t))$
ELSE abort

Given this lexical entry, the result of parsing (74) using the LINK strategy is shown below:

(77) Parsing *o Giorkos kseri oti kseris ton* 'George knows that you know him.'



The only possible substituent for the Fo(U) metavariable in the embedded type t node given the situation above is the Fo formula value of the top node of the LINKed tree ($Fo(kseris(gianis')(stergios'))$). Given substitution, we get the correct semantic results. Thus, treating *oti* both as a regular complementizer and as a coordinating conjunction will predict variation to be possible.[29]

3.8. The Case of epidi

Subordinate clauses introduced with the causal subordinator epidi (giati and *afu* as well) also exhibit variation in CG. The relevant data are repeated below:

[29]For those speakers that accept variation with *pu*, it can be argued that parsing of *pu* as a coordinating conjunction is also possible, possibly in analogy to non-factive *oti*.

(78)

a.

Epidi	*enevriases*	*me,*	*en*	*kamno*	*tipota.*
because	made-angry.2SG	me.CL-ACC	NEG	do.1SG	nothing

b.

Epidi	*me*	*enevriases,*	*en*	*kamno*	*tipota.*
because	me.CL-ACC	made-angry.2SG	NEG	do.1SG	nothing

'I'm not doing anything because you pissed me off.'

The variation attested with *epidi* is problematic, since one would expect epidi to pattern with the rest of the subordinate conjunctions in terms of clitic positioning and exhibit only proclisis rather than variant positioning. Thus, assuming an analysis similar to the one I have argued for the rest of the subordinating conjunctions, capturing proclisis is the easy case. Under such an analysis, the causal subordinator will project a type e_s requirement that will function as the clitic's proclitic trigger. However, this is again half the analysis since we also need to capture enclisis. How is enclisis going to be captured? In order to answer this question, it would be good to look at the properties of the causal subordinator in question or causal subordinators in general. The first thing worth mentioning is that this variant positioning exhibited by *epidi* has been also attested for the equivalent subordinator in Medieval Spanish (Bouzouita, 2008a). The cause subordinator *ca* 'because' in MedSp shows the same characteristics, exhibiting both proclisis and enclisis (see Bouzouita, 2008a for the actual examples). Bouzouita (2008a: chapter 3), following a generally accepted line of view (Lapeza 1978; Menendez Pidal 1980), notes that sentences introduced with *ca* can function as either subordinate or coordinate sentences. The reason for this structural duality can be traced back to Latin where two different kinds of causal relations existed, one pattering with coordinate and one with subordinate structures. The same structural duality is found in Ancient Greek (AG, Tzartzanos, 1940). It is worth mentioning that the variation attested with *epidi* in CG is also attested in MCG, so variation is not an innovation of the modern CG system. A quick check in the *chronicle of Makhairas* returned four instances of clitics appearing in *epidi* clauses. Half of them exhibited proclisis and the other half enclisis, so variation with *epidi* was already at stake in MCG. Of course, this kind of evidence is not conclusive for the assumption that epidi can be parsed as a coordinating conjunction as well. [30] However, what I am going to propose is that *epidi*, just like *oti* can be parsed as both a subordinating and a coordinating conjunction. Whether this behaviour has its roots in the structural duality of AG subordinates of cause or it is an innovation of the MCG or even the modern CG system is something that for the moment cannot be proven. Going back to the analysis of *epidi* let us start with the proclitic case involving *epidi*. In this case, parsing *epidi* will give us a type e_s requirement, and thus proclisis will obtain. In the second case, *epidi* will be parsed as a coordinating conjunction where no such a requirement will exist. Actually, the analysis of *epidi* as a coordinating conjunction will be much closer to the DS analyses of classic coordinating conjunctions like *and* or *but* than the analysis provided for *oti*. This is because *oti* was assumed to build a LINK relation from a type *t* requiring node to another type *t* requiring node or, according to the alternative analysis provided, the LINK relation was created from a type complete node but with no complete formula value (metavariable). This is unlike the analyses of coordinate conjunctions like *and* or *but* where the LINK relation is built from a type and formula complete node into a type *t* requiring node. In the case of *epidi* however, one can maintain an analysis where the causal conjunction builds a LINK relation from a type and formula complete node to a type *t* requiring one. Let us see how this will be done. Say we want to parse a subordinate of cause structure involving a clitic in enclitic position followed by the main clause:

(79)

En	*su*	*milao,*	*epidi*	*enevriases*	*me.*
NEG	you.CL-GEN	talk	because	made-angry.2SG	me.CL-ACC

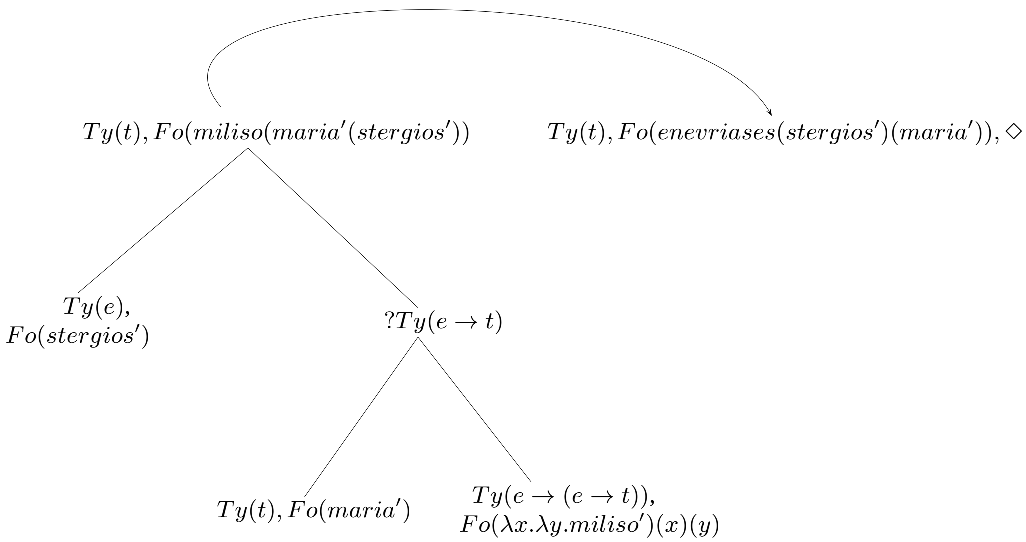
'Because you made me angry, I'm not talking to you.'

First we parse the main clause obtaining a type *t* and a formula value in the initial node. At that point *epidi* comes into parse and creates a LINK relation from the type *t* complete node to a type *t* requiring node and leaves the pointer at the latter node.

[30] Another way to explain the variant positioning exhibited with causal subordinates is to attribute it to variant positioning found with the complementizer *oti*. Given that *oti* could be either a non-factive complementizer or a subordinate of cause conjunction, variation exhibited by non-factive *oti* could spread to causal *oti* and from causal *oti* to subordinates of cause in general. However, no data are available to vindicate such a speculation and as such at the moment it cannot be anything more than a speculation.

The pointer is left at the type *t* requiring node of the LINKed tree. No proclitic trigger exists and thus parsing of the clitic is impossible. On the other hand, the verb can be parsed since its triggering point ($?Ty(t)$) is satisfied, and now parsing of the clitic is possible giving rise to enclitic positioning:

(80) After parsing *epidi*



At that point the LINK evaluation rule for causal subordinates applies. A LINK evaluation rule is basically a rule that defines the exact semantic correlation of the two LINKed trees (e.g. conjunction in coordinated structures). The LINK evaluation rule for causal subordinates will be minimally different to the one assumed for coordinate conjunctions like *and* or *but* in that it links the two sentences via a causal relation:[31]

(81) LINK evaluation rule for *epidi*

$$\frac{\{\{Tn(X), Ty(t), Fo(\alpha), ...\}, \{\langle L^{-1}\rangle Tn(X), Ty(t), Fo(\beta), \Diamond ...\}...\}}{\{\{Tn(X), Ty(t), Fo(\alpha) \leftarrow Fo(\beta), \Diamond, ...\}, \{\langle L^{-1}\rangle Tn(X), Ty(t), Fo(\beta), ...\}...\}}$$

The $\leftarrow$ symbol is taken to encode the causal relation.[32] Formally $Fo(\alpha) \leftarrow Fo(\beta)$ reads as: $Fo(\alpha)$ has a cause in case $Fo(\beta)$ is true. In simpler terms, $Fo(\alpha)$ is the cause of $Fo(\beta)$.[33] The analysis of causal subordinates in line with coordinating conjunctions will give us enclisis than proclisis. Given that subordinates of cause can be also parsed as regular subordinates, variation with *epidi* is predicted to be possible. The same line of reasoning can be applied to the other causal subordinators like giati and *afu*, 'because' and 'since (causal)'. [34]

3.9. Complex Conjunctions/marker with tze

In this section, I will take a look at the behaviour of complex subordinators and the complex negation marker 'en tze' with respect to positioning. All these complex subordinators/markers are consisted of two parts. The first part is a marker or subordinate conjunction, while the second part is the coordinate conjunction *tze*. The first part of the conjunction is comprised

[31] Similar rules are assumed for relative, conditional and temporal clauses. See Gregoromichelaki (2006) and Chatzikyriakidis, 2010 for more details.

[32] Taken from Litschitz (1998).

[33] The formal details of what a causal relation is are not going to be discussed here. The interested reader is directed to Lewis (1986) for a discussion of causal relations and Liftschitz (1998), Shafer (1998) for causal logic models using situation calculus and event trees respectively.

[34] The difference in pragmatic import associated with the use of the different causal subordinators need not bother us here.

of proclitic triggers (e.g. negation marker en or subordinate conjunctions), however the complex conjunction/marker as a whole fails to trigger proclisis and enclisis obtains instead. For example, negation *en* and the conditional conjunction *ean*, as discussed earlier, categorically trigger proclisis. However, the complex particles/markers *en tze* ‘NEG’ and *ean tze* ‘even though’ fail to do so and enclisis is found in these cases instead. The culprit behind this behaviour seems to be the coordinating conjunction *tze*, which in a way cancels triggering of preverbal positioning. Indeed, this is the analysis already pursued in Agouraki (2001), where *en tze* is argued to involve conjunction of two CPs. The negation marker occupies the head of the CP1 phrase, while conjunction *tze* occupies the SpecCP2, the verb occupying $CP2_0$. However, in order for *tze* to cancel out a proclitic trigger, it needs to follow this trigger and not just be within the same domain with this element. For example, coordination between two clauses where the second of these involves the negation marker following *tze* regularly trigger proclisis as the example below shows:

(82) *Tuto en alithja tze en ton adiko pu alakse gnomi.*
this is truth and NEG him.CL-ACC blame COMP changed.3SG opinion
‘This is true and I do not blame him for changing his mind.’

A good way to test whether *tze* really cancels out proclisis in general or this is just an idiosyncratic behaviour of the complex elements involving *tze*, is to try and find cases where *tze* appears inside the clause rather than in its usual position at the beginning of the clause. Fortunately in Modern Greek and CG as well, there are a number of constructions where *tze* is not used as a coordinating conjunction, but rather starts off the sentence. *Tze* is followed by an affirmative sentence, followed by another instance of tze, followed by the negative version of the affirmative sentence. Such examples are shown below::

(83) *Tze ksero ton tze en ton ksero.*
and know.1SG him.CL-ACC and NEG him.CL-ACC know.1SG
‘I know him but not too well (free translation)’

(84) *Tze sevome ton tze ektimo ton.*
and respect.1SG him.CL-ACC and appreciate.1SG him.CL-ACC
‘I both respect and have an appreciation for him’

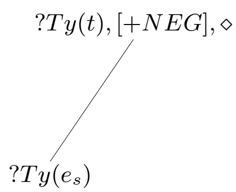
So, how are structures like the above are going to help us decide whether *tze* cancels out proclisis across the board when following proclitic triggers? Well, fortunately structures like these can be embedded within subordinate clauses, in which case they follow the subordinate conjunction. The crucial sentence in these cases is the affirmative one given that the negative one will be proclitic in any case. Indeed, as we see from the example below, *tze* cancels out proclisis that is induced by the conditionals *ean* and *otan* respectively:

(85) *Ean tze theli to tze en to theli, kalitera na min ton*
if and want.3SG it.CL-ACC and NEG it.CL-ACC want.3SG better SUBJ NEG him.CL-ACC
piesume.
push1PL
‘If he does not want it that much, let us not push him.’

(86) *Otan kapion tze sevese ton tze ektimas ton, tote tzinos enna*
when someone and respect.2SG him.CL-ACC and appreciate.CL-ACC him.CL-ACC then he FUT
adapodosi.
payback.3SG
‘When you both respect and appreciate someone, s/he is going to pay you back.’

In effect when *tze* follows any of the proclitic triggers, then it cancels out proclisis. Now, in giving an analysis of *en tze*, we must first look at the elements comprising the complex negation particle *en tze* and see what the existing DS analyses are, if any, with respect to each of these elements separately. The negation marker has already received an analysis. According to this, the negation marker builds the situation argument node and then returns to the type *t* requiring node where it further projects a *[+NEG]* feature on that node. On the other hand, coordinating conjunctions have traditionally been characterized in DS as LINKed structures, the mainstream assumption being that they induce a LINK relation from a type complete node to a node with a requirement of the same type (Cann et al., 2005). The reasoning behind such treatment is that coordinating conjunctions start a new domain, be it sentential, verbal or nominal, an assumption rather independent from the theoretical model used here. However, no actions are induced in that same domain by coordinating conjunctions in contrast to subordinate conjunctions which further introduce structure in that new LINKed domain. In that sense, a clitic following a coordinating conjunction is a sentence initial clitic, since it is the first element inducing structure in the domain. The reason that proclisis cannot obtain with *en tze,* and enclisis is only possible might very well be attributed to this latter fact, i.e. that *tze* actually starts a new domain different than the one *en* is parsed. In that sense, the situation argument node projected by the negation marker is parsed in a different domain, given that *tze* will induce a LINK transition from the tree where negation is parsed to another tree structure. The first thing we need to look at in formalizing this assumption, is the node from which the LINK transition associated with *tze* is going to be created. When *tze* comes into parse in a sentence like (85), the only word already parsed will be the negation operator *en*:

(87) After parsing *en*



The pointer returns to the type *t* requiring node. The problem is that now, no node is type complete. In that sense, the assumption that coordinating conjunctions project a LINK relation from a type complete node to a node requiring the same type cannot be maintained in this case. The other way to look at the problem is to assume that these complex elements are parsed as one element and not separately. Such an assumption can be further backed by the fact that nothing can intervene between the two elements in all these constructions and furthermore, while the two elements can appear on their own, the interpretation that they give rise to in combination is not a compositional accumulation of the content of the entries of each of the separate elements. For example, *an tze* 'even though', cannot be captured assuming an analysis where the two elements are treated as separate entries, since in this case what we get is the actions induced by the conditional *an* plus the actions of the coordinating conjunction *tze*, which by no means capture the semantics of concession. Of course, none of these two diagnostics is enough to guarantee that the two elements involve one single lexical entry, since examples involving two inseparable elements but still are treated as separate entries or cases where a combination of two words does not give to compositional meaning are all over the place in natural language. Examples of the first kind include cases of mood/tense particles and the verb or the clitic and the verb in Greek, whereas examples of the second involve various idiomatic expressions.[35] For the moment, it is impossible to dwell into a discussion of whether or not these elements are to be parsed as a single entry or not. For the moment I will make the assumption that one lexical entry is indeed involved in these cases. Given an analysis for *en tze* as involving one lexical entry, we can assume that *en tze* first marks the type *t* requiring node with the *[+NEG]* feature while it projects a type t value and a formula metavariable in that same node with the requirement that the value for the metavariable's substituent is to be shared with the formula value of the LINKed tree, as in the case of *oti*. Then, a LINK transition to a type *t* requiring node is created. This new LINKed domain will be the domain in which the rest of the sentence is going to be parsed. What I am then assuming is that the sentence is parsed within the context of a negative specification, within the context of negation. In that sense, such an analysis is quite close to the DS analysis of HTLD or relative clause structures. Left dislocated arguments in HTLD constructions or the relativized elements in relative clauses can be seen as setting the context in which the rest of the clause in HTLD structures or the relative clause in relative

[35]An anonymous reviewer is gratefully thanked for reminding me that the arguments used are enough to prove that the cases in question are indeed cases where one complex lexical entry is involved.

clause structures are going to be parsed. In the same sense, *en tze* can be seen as providing the context (a negative context) in which the sentence is to be parsed. The lexical entry for *en tze* is shown below:

(88) Lexical entry for *en tze*

IF $?Ty(t)$
THEN $\texttt{put}([+NEG], Ty(t), Fo(U), ?\exists\mathbf{x}.Fo(\mathbf{x}) \wedge \langle L\rangle Fo(\mathbf{x}));$
$\texttt{make}(\langle L\rangle); \texttt{go}(\langle L\rangle); \texttt{put}(?Ty(t))$
ELSE abort

The result in tree notation is shown below:

(89) After parsing *en tze*

$$?Ty(t), [+NEG], Ty(t), Fo(U), ?\exists\mathbf{x}.Fo(\mathbf{x}) \wedge \langle L\rangle Fo(\mathbf{x}) \qquad ?Ty(t)$$

Notice that no type e_s requiring node is assumed to be projected by *en tze*. This is done for reasons of simplicity only, since it does not make any difference in the analysis presented, given that the type e_s requiring node will be projected in the domain where the LINK starts, i.e. in a tree different to the one where the clitic is going to be parsed as we will see. This type e_s requiring node cannot constitute a proclitic trigger, as the clitic will be parsed in a different domain, namely the domain of the LINKed tree. After parsing *en tze* the pointer is at the type *t* requiring node of the LINKed tree. Let us say we want to parse *en tze ida ton* ‘I did not see him’. *En tze* is parsed first giving us the structure just discussed. Then, the rest of the sentence is parsed in the LINKed type *t* requiring domain. Assuming that all metavariables (subject and object) have been substituted by values from the context, we end up with the following structure (the metavariable in the type t node of the first tree is substituted by the value found in the LINKed type t node):

(90) After metavariable substitution

$$Ty(t), Fo(ida(giani)(stergios')), [+NEG] \qquad Ty(t), Fo(ida(giani)(stergios'))$$

The core of the proposed analysis is that the complex negation particle *en tze* links a type t node with a type t requiring one. The node where the LINK starts (the type t node), carries the *[+NEG]* feature. The pointer ends up at type *t* requiring node after parsing *en tze*. It is at that domain where the rest of the sentence is parsed. The sentence is parsed given a negative context, i.e. the complex negation marker sets a negative context against which the sentence is going to be parsed. This treatment of complex negation is an effective way to get out of the enclisis problem. Proclisis is not possible with *en tze* since the clitic is parsed in a domain where no proclitic triggers exist (the clitic will be parsed on the LINKed tree). In that respect, the only possibility with *en tze* is enclisis. The same reasoning can be argued to apply for the other complex markers/conjunctions like *ean tze* ‘even though’. *Ean tze*, in the same sense as *en tze*, will be parsed as one complex element linking a type *t* requiring node with another node. A LINK evaluation rule will then do the appropriate copying, so that the semantics of concession are captured. There are a number of details that need to be taken care of for the case of *ean tze*, since concessive clauses just like subordinate clauses will already be linked with the consequent clause via a LINK relation. I will not flesh out the exact details of how the *ean tze* conjunction (or other complex conjunctions with *tze*) works. What is crucial, is that there is a way to account for these cases fairly easy within DS.

3.10. *Clitic Positioning with Imperative verbs*

We have provided a mechanism for capturing the full range of clitic positioning restrictions in CG. However, all these positioning restrictions are relevant for non-imperative verbs only, since in imperative environments these restrictions disappear and enclisis is the case across the board. The examples below shown cases of imperative sentences where a fronted constituent is present but proclisis cannot obtain:

(91) *TUTO pe mu/ *TUTO mu pe.*
this tell.2SG.IMP me.CL-DAT
'Tell me this.'

(92) *TORA fer mu to/ TORA mu to fer(e).*
now bring.2SG.IMP me.DAT it.CL-ACC
'Bring it to me now.'

Proclisis with mood/tense markers or the negation marker are in principle not possible since these elements are incompatible with imperatives in general. Given this situation someone might argue that what the account provided does is to account for only a subpart of the clitic positioning behaviour of CG, given that clitic positioning with imperatives is not captured. However, as we will see the lexical entry we have proposed for CG will work for imperatives as well. As the reader might already inferred the difference in the behaviour between indicative and imperative verbs as regards clitic positioning will reside in the lexical entries for the two different types of verbs. As we have just mentioned, imperative verbs are in principle incompatible with tense/mood markers or the negation marker in CG. In this sense, we will need a mechanism irrelevant of the behaviour of clitics that will rule out the cases where a mood/tense or negation marker is followed by an imperative verb. The obvious way to rule these cases lexically is via a restriction in the entry for imperative verbs that will abort in case a requirement for a e_s type exists in the 0 node below the type *t* node, given that all these elements will project this node along with the type e_s requirement (see the discussion on ?). Thus, the lexical entry for the imperative verb in CG, or even in languages with similar behaviour like for example SMG will have to have a trigger that aborts in case such a situation occurs:

(93) Lexical entry for imperative verbs in CG (ACTIONS are omitted)

IF	$?Ty(t)$	
THEN	IF	$\langle\downarrow_0\rangle ?Ty(e_s)\bot$
	THEN	ACTIONS
	ELSE	abort
ELSE	abort	

Of course the above entry does not suffice to exclude cases where an unfixed node has been parsed first followed by the clitic, and then the imperative verb. Furthermore, we cannot exclude these cases on the basis of the unfixed node strategy given that cases where a fronted element parsed on an unfixed node followed by an imperative exist in CG:

(94) *TON GIORKO (i)ksero.*
the.ACC George.ACC know.1SG
'Tell me this.'

(95) *TORA ela.*
now come.2SG.IMP
'Come now.'

Chatzikyriakidis (2009) argues that imperatives in Grecia Salentina Greek (SMG as well) involve a restriction in their entries for imperative verbs that basically aborts in case any fixed node is present in the tree structure:[36]

[36] See Chatzikyriakidis (2009) for the relevant argumentation.

(96) The restriction as given by Chatzikyriakidis (2009)

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

The restriction basically says that all nodes below carry a requirement for a proper treenode address, thus are all unfixed.[37] This restriction includes the restriction we have provisionally provided for excluding tense/mood and negation markers, since it aborts in case any fixed node is present. However, this restriction does more than this, since it provides us with an account of the asymmetry in clitic positioning between imperative and non-imperative verbs. Let us see why. Positing such a restriction, one will not only exclude cases where a tense/mood or negation marker has already been parsed (since a fixed node will exist in this case) but also in general all cases in which a fixed node is present when the verb comes into parse. Parsing a clitic, will always involve the projection of a fixed node, thus cases where an unfixed node plus a clitic have been parsed will be ruled out for imperatives.[38] Thus, assuming that imperatives in CG involve the restriction as this is posited for GSG and SMG in Chatzikyriakidis (2009, 2010), the asymmetry in the behaviour of clitic positioning in imperative and non-imperative verbs is explained.

3.11. Some notes on the Diachrony of the Cypriot Greek positioning System

The modern CG system emerged from its medieval counterpart, Medieval Cypriot Greek (MCG). Clitic positioning in MCG was quite similar to the positioning found today. There is however a major difference between the two systems. MCG in contrast to CG does not exhibit proclisis with fronted elements. The data below taken from Pappas (2004) are illustrative:

(97) Clitic positioning in the Cypriot Chronicles (vols. 1-56, adapted from Pappas, 2004)

Environment	Preverbal	Postverbal
Clause initial	0	208
Reduplicated object	0	15
Function word	101	3
Fronted constituent	1	14
Subject	0	29
Gerund	0	19
Imperative	0	4

As we can see, only one instance out of 15 of a fronted constituent triggers proclisis. The same goes for subjects even though we do not have a separate count for fronted subjects. Thus, the two systems differ in that in MCG the factor fronted element does not play a role in clitic positioning, whereas in CG a fronted element can trigger proclisis.

The account proposed for CG involves the use of three generalized parsing strategies that are used as triggers for parsing the clitic. These strategies do not refer to the actual elements parsed but are general actions of structure building. For example the unfixed node trigger refers to the existence of a structurally underspecified element whereas the enclitic trigger refers to the presence of any functor (verbal) type. This way of dealing with the complex positioning of CG, besides providing

[37] Note that in case where no nodes exist below the current type t requiring one, the statement is trivially true, given that universal quantification does not carry any sort of existential statement.

[38] I assume that dative clitics, even though underspecified, always project a fixed node. This is done in order to capture the fact that these clitics even though underspecified never function as subjects. In this sense, I follow Chatzikyriakidis (2009, 2010) rather than Kempson & Cann (2008.

an economical account of the facts, is further backed by the evidence of MCG and the transition from MCG to CG. As we have seen, in MCG fronted constituents do not trigger proclisis. Proclisis with these elements is captured via the unfixed node trigger, which furthermore captures proclisis with Wh elements. MCG like CG exhibits proclisis with Wh elements. Thus, the lexical entry for MCG would still involve the unfixed node trigger but this would have to be specified to apply to Wh elements only, i.e. the lexical entry would exclusively refer to the parsing strategy of the unfixed node as applied to Wh elements. Given this, the transition from MCG to CG involves the generalization of the unfixed trigger to apply for all elements that can be parsed on unfixed nodes. In effect we have a transition from an item-specific trigger (the unfixed node strategy as applied to Wh elements only) to an item-neutral one (general unfixed node strategy). The transition from MCG to CG is shown below:

(98) The transition from MCG to CG

MCG				CG		
IF		$?Ty(t)$		IF		$?Ty(t)$
THEN	IF	$\langle\downarrow_1^+\rangle Ty(x)\vert$		THEN	IF	$\langle\downarrow_1^+\rangle Ty(x)\vert\vert$
		$\langle\downarrow_*\rangle ?\exists\mathbf{x}.Tn(\mathbf{x}) \wedge Fo(WH)\vert$	$\mapsto$			$\langle\downarrow_*\rangle ?\exists\mathbf{x}.Tn(\mathbf{x})\vert$
		$\langle\downarrow_0\rangle ?Ty(e_s)$				$\langle\downarrow_0\rangle ?Ty(e_s)$
	THEN	ACTIONS			THEN	ACTIONS
ELSE		abort		ELSE		abort
ELSE	abort			ELSE	abort	

As you see in the above entries the unfixed node trigger specialized to apply for Wh elements only in the MCG entry ($\langle\downarrow_*\rangle ?\exists\mathbf{x}.Tn(\mathbf{x}) \wedge Fo(WH)$) is generalized to all elements parsed on an unfixed node in the CG entry by dropping the second part of the conjunction ($\langle\downarrow_*\rangle ?\exists\mathbf{x}.Tn(\mathbf{x})$). In that respect, the transition from MCG to CG involves the emergence of a new proclitic environment, i.e. the fronting constituent environment. Such a transition is easily explainable given the DS perspective as a process where a parsing strategy acting as a parsing trigger for the parse of a specific class of elements only (Wh elements) gets generalized and serves as a generalized parsing trigger for all other elements using this strategy (Wh elements plus fronted constituents). What is rather welcomed in such an analysis is that even though the CG system looks more complicated on a descriptive basis than the MCG one, since it involves more proclitic environments than CG, the entry for CG is less complicated than the respective MCG one. In that respect, the passage from MCG to CG involved reducing the complexity of the entry as well. It is worth noting that the transition from MCG to CG, as shown in Chatzikyriakidis (2010) parallels the transition to other MG dialects via their respective medieval ancestors, with routinization (in the sense of Pickering & Garrod, 2004) and parsing trigger generalization being the leading factors of change in all cases. Reasons of space do not allow me to further expand on the issue in this paper. However, the interested reader is directed to Chatzikyriakidis (2010) for a quite detailed sketch of the transition from Koine Greek to the Medieval Greek dialectal systems and from the latter systems to the modern positioning ones. Furthermore, the interested reader is directed to Bouzouita (2008) for a similar account as regards the transition from Medieval to Modern Spanish.

4. Some general notes on clitic positioning and clitics in general

Abstracting away from the technicalities of the proposed account, one might plausibly ask what this account has to offer with respect to the nature and properties of clitics. I consider that DS accounts of clitics (this account included) have a number of advantages over other theoretical frameworks and can explain some interesting aspects of the syntax of clitics from a wholly new perspective. Firstly, what I consider very important is that the word-affix debate, as already discussed in 3.1, is irrelevant for DS, since assuming that lexical entries are basically just packages of information, what we need in order to assume that a lingusitic element has a separate lexical entry, is that this element provides distinct procedural information as regards the parsing process. Such information is definitely conveyed by clitics whether these are considered words or not. In this sense, no pre-theoretical decision on the issue needs to be taken in order to provide an account of clitics within the DS framework. What is more, the account will be compatible with both options (clitics being words and clitics being affixes) and as such we avoid the problem of an analysis being dependent on whatever pre-theoretical decision we take on the issue. Furthermore, clitics under the account proposed, and pretty much all DS accounts on clitics, are treated as ordinary pronouns in that they are assumed to project a type value and a formula metavariable that needs to be updated from the context. However, what is exceptional as regards clitics is that these create their own structure, and, in so doing, they have become more complex than just being a pronoun occurring in a certain position. Thus, the lexical entry for a pronoun has as its triggering point the node in which type and formula information is going to be provided (*?Ty(e)*). However, in the case of clitics, this triggering point is a type *t* requiring node. Starting from this node, clitics create their own structure in which they project type and formula information (by building a fixed node or an unfixed one). This behaviour of clitics is independently needed given the locality restrictions associated with clitics. Given these restrictions, a type *e* requiring trigger for clitics will predict that locality violations in the case of clitics should be fine contrary to fact (given parsing of the clitic on a node introduced by *ADJUNCTION). This behavior, the fact that clitics induce their own structure, is one of the differences between clitics and pronouns and follows the natural observation as regards positioning of pronouns on the one hand and clitics on the other, the former being freer compared to the latter. Lastly, the whole adjunct vs argument debate as regards the status of clitics and the NPs (or DPs depending on the framework used) that are doubled by clitics in doubling constructions is again not something that constitutes a problem in giving a DS analysis. Given that clitics project a Fo metavariable in one of the object nodes, a proper formula value needs to act as a substituent for this metavariable. This substituent can very well come from the context in which case no doubling occurs, but can however also come from the natural language string itself, in which case doubling occurs by unification of the two nodes, the one that hosts the clitic and the unfixed node that hosts the doubled NP.[39]

Clitic positioning in a lexicalized framework like DS is defined within the lexical entries for clitics in each case, i.e. it is basically treated as a lexical phenomenon. From a synchronic point of view positioning restrictions can be seen as a number of generalized triggering points that if satisfied regulate clitic positioning. From a diachronic point of view, these restrictions can be seen as the result of a number of processes, including routinization in the sense of Picekring & Garrod (2004) as well as parsing/production (speaker/hearer) mismatches due to the availability of different parsing strategies for parsing one and the same element. Indeed, similar claims have been made for languages like Spanish (Bouzouita, 2008) and an example of the transition from MCG to CG has been given in this paper as well. Typologically, at least in the case of Greek, we are dealing with three types of systems as Condoravdi & Kiparsky (2001) have already pointed out. The SMG one, the CG one and the Pontic Greek (PG) one. The three systems can be described as involving the same set of actions but different triggering points (see Chatzikyriakidis, 2010: chapters 3,4,5 for more details). These triggering points may seem arbitrary from a synchronic point of view but make sense once one looks at the forms the respective medieval dialects had. Thus, typologically different systems in Greek are described via different triggering points, the latter being dependent on the form of the medieval dialect in each case. In the same vein, similar clitic systems in different language families e.g. the systems of Modern Spanish and SMG should have developed from pretty much similar ancestoral systems, which is to a large extent true for the cases in question (check Chatzikyriakidis, 2010 for Greek and Bouzouita, 2008 or Spanish). Of course, this last claim needs to be further tested.

5. Conclusions

In this paper, I provided an account of CG positioning by arguing that the apparent complexity of proclitic positioning can be derived assuming two generalized proclitic triggers. Following Bouzouita (2008), the first proclitic trigger was argued to be the existence of an unfixed node, i.e. that the presence of an unfixed node constitutes a trigger for parsing the clitic. This assumption correctly captures variation with fronted elements, given that these can be parsed using more than one parsing strategy. In that sense, the account correctly predicts the claim made in Pappas (2010), namely that focused fronted elements do not necessarily trigger proclisis. Building on assumptions by Gregoromichelaki (2006) and Cann (2011), according to which every sentence contains an obligatory situation argument, I argued that the second proclitic trigger consists in the existence of a type e_s requiring node. I then assumed that subordinating conjunctions, modal/tense-aspect markers and negation project a requirement for such a type. Enclisis on the other hand is captured assuming a generalized trigger which allows the clitic to be parsed in case any verbal type has been parsed first. These three triggers are adequate enough to

[39]This is based on the assumption that clitics allow more structure to be built below the node they are hosted. If this is not allowed, then no doubling is possible. This is presumably the difference between doubling and non-doubling languages, as well as the reason why strong pronouns cannot be doubled (see Cann et al., 2005: chapter 4; Gregoromichelaki, 2010 among others).

capture the complexity of the CG positioning system as regards strict enclitic and proclitic environments. Then, variation in positioning with *oti* and subordinates of cause was captured assuming that these elements can be also parsed as coordinating conjunctions (i.e. LINK structures). Lastly, the account was shown to be grounded in diachronic considerations as well, showing that the CG positioning system derived from the respective MCG one via parsing trigger generalization, according to which an item specific trigger changes to an item neutral one.

Acknowledgements

This paper has benefited from a number of discussions with Ruth Kempson, Ronnie Cann, Stavroula Tsiplakou, Theoni Neokleous. Spyros Armosti and Theoni Neokleous are also thanked for commenting on some of the CG data as well as providing judgments. The AHRC is gratefully thanked for funding part of this work. Lastly, Alex Deligianidis is thanked for providing crucial technical advice on parts of this paper.

References

Agouraki, Y., 1997. On the Enclisis-Proclisis Alternation. In: Drachman, G., Malikouti-Drachman, A., Fykias, J., & Klidi, C., (eds), *Proceedings of the 2nd International Conference on Greek Linguistics*. Graz, W. Neugebauer Verlag, 393-404.

Agouraki, Y., 2001. The position of clitics in Cypriot Greek. In: Ralli, A., Joseph, B., & Janse, M., (eds), *Proceedings of the First International Conference on Greek Linguistics*, 1-18.

Blackburn, P. & Meyer-Viol, W., 1994. Linguistics, Logic and Finite Trees.

Bošković, Ž., 1995. Participle Movement and Second Position Cliticization in Serbo-Croatian. *Lingua 96*, 245-266.

Bošković, Ž., 2001. *On the Nature of the Syntax-Phonology Interface: Cliticization and Related Phenomena*. Elsevier.

Bouzouita, M., 2008a. *The Diachronic Development of Spanish Clitic Placement*. Phd Thesis, King's College, London.

Bouzouita, M., 2008b. At the Syntax-Pragmatics Interface: Clitics in the History of Spanish. In: Cooper, R., Kempson, R. (eds), *Language Evolution and Change*. Cambridge, Cambridge University Press.

Bresnan, B., 2001. *Lexical-Functional Syntax.* Oxford, Blackwell.

Cann, R., 2011. Towards an Account of the English Auxiliary System: Building Interpretations Incrementally. In: Kempson R., Gregoromichelaki, E., & Howes C., (eds), *The Dynamics of Lexical Interfaces*. Stanford, CSLI publications.

Cann, R., & Kempson, R., 2008. Production Pressures, Syntactic Change and the Emergence of Clitic Pronouns. In: Cooper, R. & Kempson, R., (eds), *Language in Flux: Dialogue Coordination, Language Variation, Change and Evolution*, London, College Publications, 221-263.

Cann, R., Kempson, R., & Marten, L., 2005. *The Dynamics of Language*. Oxford, Elsevier.

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

Chatzikyriakidis, S., 2009a. Clitics in Grecia Salentina Greek: A Dynamic Account. *Lingua* 119. Elsevier, 1939-1968.

Chatzikyriakidis, S., 2009b. The Person Case Constraint in Modern Greek: A Unified Dynamic Syntax Account. *Newcastle Working Papers in Linguistics* 15, 40-62.

Chatzikyriakidis, S., 2010. Clitics in 4 Dialects of Modern Greek: A Dynamic Account. PhD thesis, King's College, London.

Chatzikyriakidis, S., 2011. A Dynamic Account of Clitic Climbing: A First Sketch. In: Kempson R., Gregoromichelaki, E. (eds), *The Dynamics of Lexical Interfaces*. Stanford, CSLI publications.

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

Crysmann, B., 1997. Parallel Morpho-Syntactic Constraints in European Portuguese Cliticization. In: Butt, M., & Holloway King, T., (eds.), *proceedings of the LFG97 Conference.* CSLI publications, Stanford.

Gregoromichelaki, E., 2006. *Conditionals in Dynamic Syntax*. Phd thesis, King's College, London.

Gregoromichelaki, E., 2010. 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*. Amsterdam, John Benjamins. Jannaris, A.N., 1968. *Historical Greek Grammar*. Georg Olms Verlagsbuchhandlung: Hildesheim.

Kempson, R., & Kurosawa, A., forthcoming. At the Syntax Pragmatics Interface: Japanese Relative Clause Construal. In: Hoshi., H. (ed.), *Language Mind and Brain*. Kuroshio publishers.

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

Kordoni, V., 2004. Between shifts and alternations: Ditransitive constructions. In: Müller, S., *Proceedings of the HPSG04 Conference*. Stanford, CSLI publications.

Lewis, D., 1986. *Philosophical Papers, volume II*. New York, Oxford University Press.

Lifschitz, V, 1997. On the Logic of Causal Explanation. *Artificial Intelligence*96, 451-465.

Luis, A. & R. Otoguro. 2004. Proclitic Contexts in European Portuguese and their Effect on Clitic Placement. In: *Proceedings of LFG04*, CSLI on-line Publications.

Mackridge, P., 1985. *The Modern Greek Language: a Descriptive Analysis of Standard Modern Greek*. Oxford University Press, Oxford.

Newton, B., 1972. *Cypriot Greek: Its Phonology and Inflections.* The Hague: Mouton.

Pappas, P., 2004. Variation and Morphosyntactic Change in Greek: From Clitics to Affixes. *Palgrave Studies in Language Change and Language History*, Palgrave.

Pappas, P. A., 2010. Object clitic placement in the history of Cypriot Greek. In: Heselwood, B., Upton, C. (Eds.), Proceedings of Methods XIII: Papers from the Thirteenth International Conference on Methods in Dialectology 2008. Frankfurt: Peter Lang.

Paraskeua, M., 2007. *A Dynamic Syntax Account of Object Clitics in Cypriot Greek*. Msc Thesis, King's College, London.

Pickering, M., & Garrod, S., 2004. Towards a Mechanistic Psychology of Dialogue. *Brain and behavioral science* 27.

Revithiadou, A., 2006. Prosodic Filters on Syntax: An Interface Account of Second Position Clitics. *Lingua 116*, 79-111.

Roussou, A., 2000. On the Left Periphery: Modal Particles and Complementizers. *Journal of Greek Linguistics* 1, 65-94.

Roussou, A., 2008. *Selecting complementizers.* Ms., University of Patras.

Shafer, G., 1998. Causal Logic. *Proceedings of the 13th European Conference on Artificial Intelligence* (ECAI 98), John Wiley & Sons, 1998, 711-720.

Tenny, C., & Pustejovsky., J., 2000. A History of Events in Linguistic Theory. In: Tenny, C., & Pustejovsky., J. (eds.), *Events as Grammatical Objects*. Stanford, CSLI publications.

Terzi, A., 1999a. Cypriot Greek clitics and their Positioning Restrictions. In: Alexiadou A., Horrocks, G. & Stavrou, M. (eds.), *Studies in Greek Syntax*. Dordrecht, Kluwer, 227-240.

Terzi, A., 1999b. Clitic combinations, their Hosts and their Ordering. *Natural Language and Linguistic Theory* 17, 85-121

Tzartzanos, A., 1940. *Sintaktiko tis Arxeas Elinikis Glosis.* [Syntax of the Ancient Greek Language]. Kiriakidis, Thessaloniki, 1994.

Tsiplakou, S., 2009. Code-switching and Code-mixing between Related Varieties: Establishing the Blueprint. *The International Journal of the Humanities* 6, Issue 12, 49-66.

Tsiplakou, S., in press. Competing grammars in Cypriot Greek? Evidence from language alternation. *Proceedings of the 18th International Conference on Historical Linguistics*, Montreal, 6-11 August 2007. Amsterdam, John Benjamins.

Tsiplakou, S., Panagiotidis, P. & Grohmann, K.K., 2007. Properties of Cypriot Greek Wh-question formation. In: *Proceedings of the 7th International Conference on Greek Linguistics*, 83-98.

Appendix A. Computational Rules

(A.1) ELIMINATION

$$\frac{\{...\{\langle\downarrow_0\rangle(Fo(a),Ty(X)),\langle\downarrow_1\rangle(Fo(b),Ty(X\rightarrow Y),...,\diamond)\}...\}}{\{...\{Fo(b(a)),Ty(Y),\langle\downarrow_0\rangle(Fo(a),Ty(X)),\langle\downarrow_1\rangle(Fo(b),Ty(X\rightarrow Y),...,\diamond)\}...\}}$$

Condition:$\langle\downarrow_i\rangle$?ϕ does not hold and i $\in\{0,1\}$

(A.2) COMPLETION

$$\frac{\{...\{Tn(n),...\},\{<\uparrow_i>,Tn(n),...,Ty(X)\,..,\diamond\}...\}}{\{...\{Tn(n),...,<\downarrow_i>\,Ty(X),...,\diamond\},\{<\uparrow_i>\,Tn(n),...,Ty(X),...\}...\}}$$

Where i $\in$(0,1,*)

(A.3) ANTICIPATION

$$\frac{\{...\{Tn(n),...,\diamond\},\{\langle\uparrow\rangle Tn(n),?X...\}...\}}{\{...\{Tn(n),...\},\{\langle\uparrow\rangle Tn(n),?X...,\diamond\}...\}}$$

(A.4) THINNING

$$\frac{\{...\{...,X,...,?X,...,\diamond\}...\}}{\{...\{...,X,...,\diamond\}...\}}$$

(A.5) *ADJUNCTION

$$\frac{\{...\{Tn(n),?Ty(t),\diamond\}\}}{\{...\{Tn(n),?Ty(t)\},\{\langle\uparrow^*\rangle Tn(n),?Ty(e),?\exists\mathbf{x}.Tn(\mathbf{x}),\diamond\}\}}$$

(A.6) MERGE

$$\frac{\{...\{...,DU,DU',...\}...\}}{\{...\{...,DU\sqcup DU',...\}...\}}$$

Where $\diamond\in$ *DU'* and *DU* $\cup$ *DU'* is consistent

(A.7) TOPIC STRUCTURE INTRODUCTION

$$\frac{\{\{Tn(0),?Ty(t),\diamond\}\}}{\{\{Tn(0),?Ty(t)\}\},\{<L>Tn(0),?Ty(e),\diamond\}}$$

(A.8) TOPIC STRUCTURE REQUIREMENT

$$\frac{\{\{Tn(0), ?Ty(t)\}, \{< L > Tn(0), Fo(a), Ty(e), \diamond\}}{\{\{Tn(0), ?Ty(t), ? < D > Fo(a), \diamond\}\}, \{< L > Tn(0), Fo(a), Ty(e)\}}$$