A TREATMENT OF WELSH INITIAL MUTATION

Ingo Mittendorf and Louisa Sadler
University of Essex

Proceedings of the LFG06 Conference

Universität Konstanz

Miriam Butt and Tracy Holloway King (Editors)

2006
CSLI Publications
http://www-csli.stanford.edu/

Abstract

A peculiarity of Welsh and the other Celtic languages is their system of Initial Mutations. These are regular alternations of word-initial phonemes triggered by a variety of lexical and syntactic triggering contexts. This feature of the Celtic languages poses a number of challenges to grammatical description, not least because it requires direct reference to adjacency relations in the linear string. We describe here an approach which covers the full range of mutation processes and their distribution in Welsh using the XLE grammar development environment and the associated finite state and tokenisation tools (Crouch et al., 2006).

1 Introduction

A peculiarity of Welsh and the other Celtic languages is their system of Initial Mutations. These are regular alternations of word-initial phonemes triggered by a variety of lexical and syntactic triggering contexts. This feature of the Celtic languages poses a number of challenges to grammatical description, not least because it requires direct reference to adjacency relations in the linear string. We describe here an approach which covers the full range of mutation processes and their distribution in Welsh using the XLE grammar development environment and the associated finite state and tokenisations tools (Crouch et al., 2006).

The rest of this paper is structured as follows. Section 2 provides some basic background on the system of initial mutations and a brief introduction to the types of conditioning environments. In section 3 we present a word-and-paradigm based view of initial mutation as a morphosyntactic phenomenon, a view which underlies our morphological approach. Following this, section 4 shows how a multiword transducer is defined to account for the distribution of initial mutations determined by specific lexical items. Section 5 then turns to cases of syntactically conditioned mutation, and outlines the c-structure approach to this phenomenon.

2 Initial mutations

A peculiarity of Welsh and the other Celtic languages is their system of Initial Mutations. These are regular alternations of word-initial phonemes such that, under the appropriate circumstances, a word like Welsh *tad* ‘father’ appears as *dad*, *thad* or *nhad*. (1) shows the possible range of alternations in initial consonant phonemes. These alternations can be arranged into different sets, for which the traditional terms are Radical (the citation form), Soft Mutation, Nasal Mutation and Aspirate Mutation.

(1) Welsh Initial Consonant Mutations

Radical	p	t	c /k/	b	d	g	m	ll /ɬ/	rh /r̥ʰ/
Soft Mut	b	d	g	f /v/	dd /ð/	Ø	f /v/	l	r
Nas Mut	mh /m̥ʰ/	nh /n̥ʰ/	ngh /ŋ̊ʰ/	m	n	ng /ŋ/			
Asp Mut	ph /f/	th /θ/	ch /χ/						

As can be seen, there is a wider range of alternations if the initial phoneme is a voiceless consonant (/p/, /t/, /k/) than if it is a voiced or other consonant. Consonants not listed (/n/, /s/, /f/, etc.) show no alternations. Basically two different types of environment can be distinguished in which these mutation forms appear. First, initial mutation can be *triggered* by a range of lexical items including proclitic pronouns, prepositions, determiners and numerals. Each trigger is followed by a specific, lexically determined mutation. The *target* of these mutation triggers, that is, the word that shows the requisite initial mutation, is the word directly following the trigger: a lexical mutation trigger and its mutation target are always adjacent. This means that the target is to some degree unpredictable. For example, in (2) the clitic pronoun *fy* 'my' in (2) triggers NM; in (2a) the target of this mutation is the noun *diddordebau* 'interests'; in (2b), the pre-nominal adjective *prif* 'main'; and in (2c), the numeral *tri* 'three'.[1] The pre-nominal adjective and the numeral in turn trigger their own mutations, SM and AM respectively.

(2) a. *fy niddordebau*
(fy) (NM.diddordebau)
my interests

b. *fy mhrif ddiddordebau*
(fy) (NM.prif) (SM.diddordebau)
my main interests

c. *fy nhri phrif ddiddordeb*
(fy) (NM.tri) (AM.prif) (SM.diddordeb)
my three main interest(s)

There is no connection between the category of the trigger and the triggered mutation: Different prepositions trigger different mutations; different clitic pronouns also trigger different mutations; and so on. (3a) shows the 1SG clitic *fy* 'my' triggering NM; (3b) the 3SG MASC clitic *ei* 'his' triggering SM; and (3c) the 3SG F clitic *ei* 'her' triggering AM.[2] As the

[1] Cardinal numerals are followed by the singular form of nouns in Welsh. This has no bearing on the issue here.

[2] The analysis of *ei thad* in (3c) has been slightly simplified at this point. For a more accurate analysis see (11 b) .

last two examples with the homophonous triggers *ei* ‘his’ and *ei* ‘her’ also illustrate, there is no connection between the phonological makeup of the trigger and the triggered mutation. Initial mutation is not a sandhi-phenomenon.

(3)	a.	*fy*	*nhad*	b.	*ei*	*dad*	c.	*ei*	*thad*
		(fy)	(NM.tad)		(ei)	(SM.tad)		(ei)	(AM.tad)
		my	father		his	father		her	father

Second, initial mutations can be syntactically conditioned, that is, triggered by a syntactic environment. For example, attributive APs, which by default appear in post-nominal position, are subject to Soft Mutation if the head noun is FEM SG; otherwise (with MASC SG nouns or PL nouns of either gender, MASC or FEM), the AP appears in the radical form; cf. (4).

In such syntactic environments it is the first word in the relevant domain which is subject to mutation. In attributive APs this will usually be the adjective, but if the adjective is preceded by an adverb, it will be the adverb; cf. (6). (The adverb in turn triggers its own mutation.) A comparison between the examples in (6) incidentally shows that it would be wrong to view soft-mutated *bwysig* as an (attributive) FEM SG form of the adjective *pwysig*.

(4)	*ci*	*mawr*	*cath*	*fawr*
	(ci)	(RAD.mawr)	(cath)	(SM.mawr)
	dog.M.SG	big	cat.F.SG	big

(5)	*cath*	*ddu*	*fawr*
	(cath)	(SM.du)	(SM.mawr)
	cat.F.SG	black	big

(6)	*agwedd*	*bwysig*	*agwedd*	*dra*	*phwysig*
	(agwedd)	(SM.pwysig)	(agwedd)	(SM.tra)	(AM.pwysig)
	aspect.F.SG	important	aspect.F.SG	very	important
	‘(an) important aspect’		‘(a) very important aspect’		

Attributive AP mutation illustrates why syntactically conditioned mutation should be distinguished from lexically conditioned mutation. Attributive AP mutation is not subject to lexical idiosyncracy. Moreover, as (4c) illustrates, when the FEM SG noun is followed by two APs, each of these is subject to SM independently, and furthermore the trigger (noun) and target (second AP) are not adjacent, which is uncharacteristic of lexical mutation triggers and targets. Note also that lexical triggers are always followed by a target, whereas, of course, attributive APs are optional so that a FEM SG noun is only a trigger when a post-nominal AP is present.

3 Regular Mutation Paradigms

There are a number of different (and sometimes partial) approaches to Celtic initial mutations in the theoretical literature. In some analyses, initial mutations are viewed essentially as phonological processes triggered by syntactic environments, in a framework in which a direct interface between syntax and phonology is assumed (Ball and Müller, 1992). Our approach takes the alternative view that initial mutation is close to inflection in nature and is essentially a morphosyntactic phenomenon. Our approach has much in common with the view of initial mutation in the Goidelic languages proposed in Green (2003) and Stewart (1992): for detailed discussion of this position and criticisms of the phonological view, see those references.

(1) gave an overview over the possible mutation *forms*. These forms, however, cannot simply be equated with what could be call mutation *functions* or *mutation states*. Mutation forms are the morphological exponents of mutation functions with their different values. We assume that each word has a mutation paradigm with different cells filled with the possible mutation forms. There is not necessarily a one-to-one relationship between forms and functions: the paradigmatic nature of mutation forms establishes the different values of the mutation functions.

We illustrate this with a close look at AM in (1). Special AM forms exist for words with an initial voiceless consonant. There are no special forms beginning with other phonemes. This does not of course mean that words with a non-voiceless-stop initial are barred from those syntactic environments where the AM form of voiceless-stop initials is called for. Rather, what happens in such cases is that the radical form "stands in" for the non-existent discrete AM form (the radical is thus the morphological default).

Whatever applies to Aspirate Mutation also applies to Nasal Mutation: here, words with initial /m/, /ɬ/ <ll> and /r̥ʰ/ <rh> have no discrete forms; again the radical stands in. And with words which start with a "non-mutatable" phoneme such as /s/, the radical appears in all mutation environments. A first version of a mutation paradigm could therefore look as in (7).

(7)

	Vl stops			**Vd stops**			**m**	**ll / rh**		**Other C**
Rad	p-	t-	c-	b-	d-	g-	m-	ll-	rh-	s- *etc.*
AM	ph-	th-	ch-	b-	d-	g-	m-	ll-	rh-	s-
SM	b-	d-	g-	f-	dd-	Ø-	f-	l-	r-	s-
NM	mh-	nh-	ngh-	m-	n-	ng-	m-	ll-	rh-	s-

We now turn to the question of the number of mutation functions or states, which we have so far taken to be four (as in (7)), and show that the picture is actually slightly more complicated. There are mutation environments which straightforwardly require the set of Soft Mutation forms, or the Aspirate Mutation set. But in other mutation environments a mixture of such forms appears. First, there are environments which select only a subset of the SM forms. In these environments initial voiceless and voiced stops and /m/ undergo

SM, but if the initial phoneme is <ll> /ɬ/ or <rh> /r̥^{h}/, the radical form is required. This “Restricted Soft Mutation” (SMR) applies, for example, to FEM SG nouns following the definite article; cf. (8).

(8) *y* *gath* / *faner* / *ferch* / *llinell* / *rhwyd*
(y) (SMR.cath) / (SMR.baner) / (SMR.merch) / (SMR.llinell) / (SMR.rhwyd)
the cat / flag / girl / line / net

Second, a further group of triggers (negation particles mostly) is followed by AM forms if the initial phoneme is a voiceless stop, but by SM forms, if available, otherwise; this mutation is usually called Mixed Mutation (MM); cf (9).

(9) *ni* *chanodd* / *ddaeth* / *fudodd* / *lwyddodd* / *redodd*
(ni) (MM.canodd) / (MM.daeth) / (MM.mudodd) / (MM.llwyddodd) / (MM.rhedodd)
not sang / came / moved / succeeded / ran

These cases motivate distinguishing two further mutation states or functions (10).

(10)

	Vl stops			**Vd stops**			**m**	**ll / rh**		**Other C**
Rad	p-	t-	c-	b-	d-	g-	m-	ll-	rh-	s- *etc.*
AM	ph-	th-	ch-	b-	d-	g-	m-	ll-	rh-	s-
MM	ph-	th-	ch-	f-	dd-	Ø-	f-	l-	r-	s-
SM	b-	d-	g-	f-	dd-	Ø-	f-	l-	r-	s-
SMR	b-	d-	g-	f-	dd-	Ø-	f-	ll-	rh-	s-
NM	mh-	nh-	ngh-	m-	n-	ng-	m-	ll-	rh-	s-

A further complication is introduced when we consider the form of words with an initial vowel, which sometimes occur with an initial /h/. This prevocalic aspiration appears with some (but not all) mutation triggers which require the radical or AM on consonants. If these vocalic alternations are taken to be part of the mutation system, two additional mutation functions must be assumed, RAD-H and AM-H, which differ from plain RAD and AM only where words with a vocalic initial are concerned. The examples in (11) contrast plain AM with AM-H, and those in (12) plain RAD with RAD-H.

(11) a. *tri* *chi*
(tri) (AM.ci)
three dog(s)

tri *afal*
(tri) (AM.afal)
three apple(s)

b. *ei* *chi*
(ei) (AM-H.ci)
her dog

ei *hafal*
(ei) (AM-H.afal)
her apple

(12) a. *eich ci* — *eich afal*
(eich) (RAD.ci) — (eich) (RAD.afal)
your.PL dog — your.PL apple

b. *eu ci* — *eu hafal*
(eu) (RAD-H.ci) — (eu) (RAD-H.afal)
their dog — their apple

The inclusion of prevocalic aspiration in the system of Welsh initial phoneme alternations is, incidentally, the reason why we use the term ‘Initial Mutation’ for this phenomenon, and not ‘Initial Consonant Mutation’ that can often be found instead.[3]

The paradigmatic distribution of mutation forms, including pre-vocalic aspiration, leads us to assume the following system of mutation functions and regular mutation forms as their morphological exponents:[4]

(13)

	Vl stops			**Vd stops**			**m**	**ll / rh**		**Other C**	**V**
Rad	p-	t-	c-	b-	d-	g-	m-	ll-	rh-	C-	V-
Rad-H	p-	t-	c-	b-	d-	g-	m-	ll-	rh-	C-	hV-
AM-H	ph-	th-	ch-	b-	d-	g-	m-	ll-	rh-	C-	hV-
AM	ph-	th-	ch-	b-	d-	g-	m-	ll-	rh-	C-	V-
MM	ph-	th-	ch-	f-	dd-	Ø-	f-	l-	r-	C-	d_V-
SM	b-	d-	g-	f-	dd-	Ø-	f-	l-	r-	C-	V-
SMR	b-	d-	g-	f-	dd-	Ø-	f-	ll-	rh-	C-	V-
NM	mh-	nh-	ngh-	m-	n-	ng-	m-	ll-	rh-	C-	V-

A final complication, which is simple to deal with, is that there are some lexical exceptions to the regular patterns shown in (13). These irregularities mostly concern the fact that some words do not have a discrete SM form and use the radical form instead. Among these is a group of recent English loanwords with initial *g-* such as *gêm* ‘game’. In the regular case, words whose initial phoneme is /g/ have SM forms where this phoneme is missing (*gardd* ∼ *ardd* ‘garden’). Words like *gêm* are exceptional in that radical and SM forms are identical (*gêm*). Another word with a rather unusual mutation paradigm is the interrogative *ble* ‘where’. The table in (14) shows a comparison between the regular and the irregular paradigms of *gardd* and *gêm*, and *bardd* ‘bard’ and *ble*, respectively. (Because of the nature of its triggers – negation particles – MM only applies to verbs and has been omitted here; only some mutations apply to *ble*.) Such irregularities (and others) are no problem for our word-and-paradigm based approach.

[3] A different approach to prevocalic aspiration would be to view prefixation of /h/ as a kind of liaison phenomenon: /h/ in fact would “belong” to the preceding word; if the following word starts with a consonant, then /h/ is lost; if, on the other hand, it starts with a vowel, it moves to this word. We do not discuss this alternative further here.

[4] There is a further minor complication concerning vowel initial words undergoing the mixed mutation MM: for reasons of space we suppress discussion of the details of this matter here.

(14)

	gardd	*gêm*	*bardd*	*ble*
Rad	gardd	gêm	bardd	ble
Rad-H	gardd	gêm	(bardd)	*n/a*
AM-H	gardd	gêm	(bardd)	*n/a*
AM	gardd	gêm	bardd	ble
SM	ardd	gêm [!]	fardd	ble [!]
SMR	ardd	gêm [!]	(fardd)	*n/a*
NM	ngardd	ngêm	mardd	mhle [!]

4 Lexical Mutations: The Multiword Transducer

The basic challenge in providing a treatment of lexically conditioned mutations is that the triggering relation is adjacent in the linear string, rather than any more abstract syntactic relation. Within XLE (Crouch et al., 2006) access to the linear relation between strings is possible using a user-defined `MULTIWORD` transducer within the `MORPHOLOGY` component. For those not familiar with XLE, we first give a brief overview of XLE's architecture, before describing our approach using the multiword transducer.

4.1 XLE

An XLE grammar contains a number of different sections, including:

- a RULES section that contains phrase structure rules that are functionally annotated;
- a LEXICON section that lists lexical entries with their c-structure categories and associated constraints;
- and a MORPHOLOGY section that specifies the transducers (finite state or other) used for morphological analysis (in the wider sense of the word).

The main components of interest for the treatment of initial mutations are the MORPHOLOGY section, the LEXICON section in which the tags used in the morphological analysis are mapped to syntactic terminals, and the sublexical rules, from the RULES section, which interface the morphological analysis with the syntactic terminals.

Within the MORPHOLOGY section, a string passes through several sequenced components (here described from the persective of analysis but fully reversible):

- The TOKENIZE section specifies the transducer whose main task it is to break up a parse string into individual words (or, properly speaking, tokens). (This is also the place to deal with sandhi phenomena.)
- The next section lists the transducers that are used for the morphological analysis proper and that map surface strings on to lexical strings and vice versa, i.e. they

pair tokens with morphological analyses which consist of a stem (usually the citation form is chosen) and a number of tags that encode any morpho-syntactically relevant information about a lexical item. In morphological analysis tokens are analysed individually one after the other, and thus at this stage adjacent tokens are not accessible to each other.

– The third section makes provision for the processing of multiword units. At this stage, adjacent tokens are once again accessible, and morphological analyses of individual words are concatenated. Multiword expressions may be built from the morphology and the lexicon via a built-in transducer and, if desired, marked with a special tag. In addition a user-defined multiword transducer can be specified to manipulate the concatenated string.

The morphologically analysed parse string is then passed to the grammar proper where it receives its c- and f-structure analysis. The grammar LEXICON lists all the stems[5] and tags used in the morphological analysis. The tag entries take the form of ordinary lexical entries, that is, in an XLE lexicon they consist of the stem (= the tag), a category label, a morphcode signalling whether to use the output of the morphological analyser (*XLE,* always required for tags) or not (*), and associated constraints if any. Some simple (and simplified) examples, stem and tags for *cath* 'cat' specifying that this is a FEM SG noun, are given in (16),[6] based on the morphological analysis in (15).

(15)	**Surface**	**Lexical**
	cath	`cath +Noun +F +Sg`

```
(16)  cath   N      XLE   (^ PRED)='cath'.
      +Noun  NSFX   XLE.
      +F     DGEND  XLE   (^ GEND)=fem.
      +Sg    DNBR   XLE   (^ NUM)=sg.
```

Finally, (sublexical) c-structure rules (Kaplan et al., 2004) describe the possible constituents (stem and tags sequence) of a category, N in the case of *cath.* (All sublexical constituents are appended with _BASE in these rules.) The sublexical rule for a noun is given in (17), again somewhat simplified.[7]

(17)	N −− >	N_BASE	*cath*
		NSFX_BASE	+*Noun*
		DGEND_BASE	+*F*
		DNBR_BASE.	+*Sg*

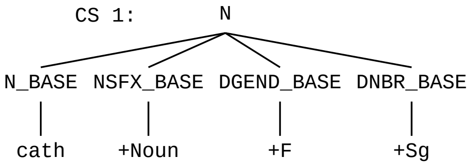
[5] It is actually not necessary to list every single lexical item/stem in the XLE lexicon. XLE allows the use of blanket entries, which is especially useful for open word classes such as nouns, adjectives, etc (Kaplan et al., 2004).

[6] ^ equals ↑ in XLE notation, and ! equals ↓. It is ParGram policy to use lower case for atomic values (Butt et al., 2002).

[7] This rule also shows an XLE simplification of the usual LFG functional annotations in that a c-structure category without annotations is understood to be annotated with ↑=↓. (Some restrictions apply.)

Given as input the morphological analysis shown in (15), the tag entries in (16) and the (sublexical) rule in (17), XLE produces the c-structure in (18) and the f-structure in (19) for the string *cath*.

(18)



(19)

```
"cath"

   [PRED 'cath'       ]
  1[GEND fem, NUM sg  ]
```

4.2 The Multiword Transducer

Before we can outline how our multiword transducer works and fits into the XLE architecture described in the previous section, and how we use it to deal with lexical mutations, we need to look at the (now "real" and unsimplified) morphological analyses of a lexical mutation trigger (the personal pronoun clitic 3SG MASC *ei* 'his') and of a mutation target (the FEM SG noun *cath* 'cat'). Both are shown in (20). We give two different mutation forms for the noun so that we can examine both a grammatical and an ungrammatical construction below.

(20)

Surface	**Lexical**
ei	`+Rad+ ei +Pron +Pers +Proclit +3Sg +M +SM+`
cath	`+Rad+ cath +Noun +F +Sg`
gath	`+SM+ cath +Noun +F +Sg`

The very last tag in the morphological analysis of the mutation trigger *ei* is a tag that encodes the initial mutation that this trigger governs, that is, the mutation state that the target must be in. For *ei* this would be Soft Mutation (`+SM+`, boxed in the example).

The very first tag in the morphological analysis of each and every word is a tag that encodes the mutation state of this word. One possible analysis for the mutation form *cath* would be Radical (`+Rad+`, boxed); one possible analysis for the mutation form *gath* would be Soft Mutation (`+SM+`, boxed). The mutation trigger *ei* also starts with such a tag, but this is immaterial in this context. Please note that we use the same set of tags for "mutation state" and "mutation governed", this difference being reflected solely in terms of position in the lexical string (start/end).

After each word (or rather token) has been morphologically analysed, XLE concatenates all the morphological analyses of the parse string. Given the two mutation forms of *cath* listed in (20), we might arrive at the two concatenated strings shown in (21).

(21) +Rad+ ei +Pron ... | +SM+ +Rad+ | cath +Noun ...
+Rad+ ei +Pron ... | +SM+ +SM+ | cath +Noun ...

As can be seen, the final mutation tag of the trigger *ei*, which constrains the mutation state of the target, and the initial mutation tag of the target encoding its mutation state are now adjacent. And only in the second of these concatenated strings do the two tags match, whereas in the first they differ. This first string (representing **ei cath*) is, in fact, ungrammatical because the target *cath* would show the wrong mutation.

It is at this point that our multiword transducer comes into action, checking that lexical mutation requirements have been satisfied by performing a test that checks whether the two mutation tags do in fact match.

There are several ways in which this test for matching mutation tags can be implemented. The way the test is performed now consists of two separate replacement operations. (The reason for keeping these two operations separate will become clear below in section 4.3):

1. Renaming the first of two matching mutation tags (the one originating from the mutation trigger): `+SM+` is replaced with `+T-SM`, `+Rad+` with `+T-Rad`, etc.[8] (22 b) shows a couple of lines from this section of the transducer.
2. Deleting the second of two matching mutation tags (the tag originating from the mutation target). The reason for this deletion has to do with syntactic mutations, which we will examine below in section 5. A couple of lines from this section of the transducer are shown in (22 a).

(22) a.
```
[..]  <- "+Rad+" || "+T-Rad" _
.o.
[..]  <- "+SM+" || "+T-SM" _
.o.
...
```

b.
```
"+T-Rad" <- "+Rad+" || _ "+Rad+"
.o.
"+T-SM" <- "+SM+" || _ "+SM+"
.o.
...
```

(23) shows the successful transformation of the concatenated grammatical string. If the two mutation tags do not match, no replacement takes place – and the test has failed; see (24).

[8] Instead of different replacement tags, one single blanket tag could be used instead (`+MutOK`, for instance). But because of the second replacement below this would mean that no indication whatsoever of the specific mutation triggered would remain in the Grammar proper. At least as a check some record should survive, if only at the sublexical level.

(23)

+Rad+ ei +Pron +Pers +Proclit +3Sg +M +T-SM cath +Noun +F +Sg

⇑

+Rad+ ei +Pron +Pers +Proclit +3Sg +M +SM+ +SM+ cath +Noun +F +Sg

ei *gath*

(24)

+Rad+ ei +Pron +Pers +Proclit +3Sg +M +SM+ +Rad+ cath +Noun +F +Sg

=

+Rad+ ei +Pron +Pers +Proclit +3Sg +M +SM+ +Rad+ cath +Noun +F +Sg

ei **cath*

The final component of the treatment of Welsh lexical initial mutations involves the sublexical rules for mutation triggers in the grammar.

From the above it should have become clear that – provided the multiword test was successful – the lexical string of a mutation trigger will end in a renamed mutation tag (`+T-Rad`, `+T-SM`, etc.) instead of the mutation tag of its original morphological analysis (`+Rad+`, `+SM+`, etc.). These renamed tags have lexical entries in our grammar as shown in (25). Their (sublexical) category is `TMUT(_BASE)`, and there are no other tags of this category. There are no further constraints associated with them.

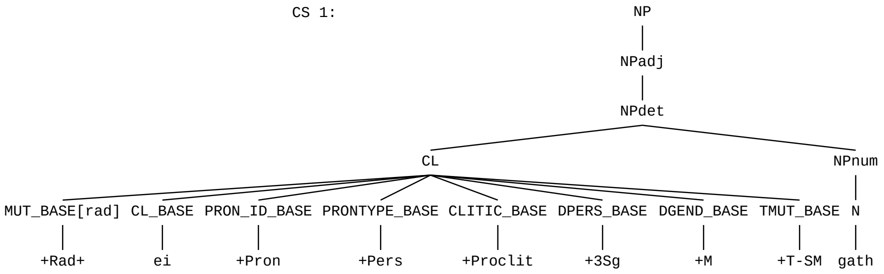
(25)
```
+T-Rad  TMUT  XLE.
+T-SM   TMUT  XLE.
...
```

If we now include this category in the sublexical rules for lexical mutation triggers, we have ensured that these are in fact followed by the correct mutation forms. (26) shows the sublexical rule for pronoun clitics like *ei* ‘his’ and (27) shows the c-structure for *ei gath*, with morphemes shown (split in two parts because of the size of the tree).

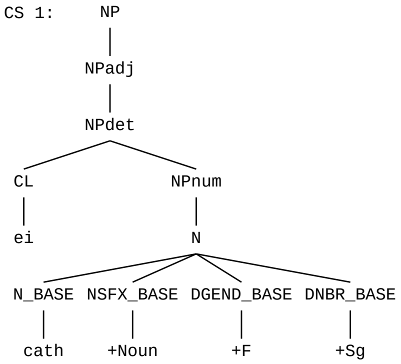
(26) CL -->

```
        (MUT_BASE[rad])                  +Rad+
        CL_BASE                          ei
        PRON_ID_BASE                     +Pron
        PRONTYPE_BASE                    +Pers
        CLITIC_BASE                      +Proclit
        { DPERS_BASE | CPERS_BASE }      +3Sg
        (DGEND_BASE: (^ INDEX)=!)        +M
        TMUT_BASE                        +T-SM
        MWE_BASE*.                       +MWE
```

(27) a.



b.



4.3 Multiword Mutation Triggers

In the example above, the lexical mutation trigger was a single word. There are, however, some mutation triggers which are themselves multiword expressions (MWE) in our grammar. This introduces a slight complication into our own multiword transducer and makes necessary a minor adjustment.

One such multiword mutation trigger is the preposition *ar gyfer* ‘for’, which is followed by the radical of the mutation target. An example is given in (28). The morphological analyses for *ar gyfer* and the noun *cath* are shown in (29). Note the final `+Rad+` tag in the analysis for *ar gyfer* (boxed) that specifies the mutation governed by this preposition.

(28) *ar gyfer cath*
for cat
'for a cat'

(29)

Surface	**Lexical**
ar gyfer	`+Rad+ ar% gyfer +Prep +Nom +Rad+`
cath	`+Rad+ cath +Noun +F +Sg`

In the MORPHOLOGY section we specify that multiword expressions should be built from the Morphology (and the Lexicon) and should receive the tag `+MWE` (30).[9]

(30)
```
BuildMultiwordsFromMorphology:
      Tag = +MWE
```

XLE will then attach this tag to the multiword analysis of *ar gyfer*:

(31)

Surface	**Lexical**
ar gyfer	`+Rad+ ar% gyfer +Prep +Nom +Rad+ +MWE`
cath	`+Rad+ cath +Noun +F +Sg`

This tag will be attached *before* the multiword transducer for lexical mutation checking comes into operation, that is, the architecture is as in (32).

(32)
```
welsh-multiword.fst
        .o.
  BuildMultiwords
        .o.
    MORPHOLOGY
```

The mutations transducer should then work with the output of `BuildMultiwordsFromMorphology` and has to accommodate one (or several) `+MWE` tags across which the check should be performed. (33) shows the modified version of the transducer. Note the addition of `["+MWE"]*` (= any number of `+MWE` tags including none) in the replacement context vis-à-vis the version in (22) without it.

(33)
```
[..]  <- "+Rad+" || "+T-Rad" ["+MWE"]* _
.o.
[..]  <- "+SM+" || "+T-SM" ["+MWE"]* _
```

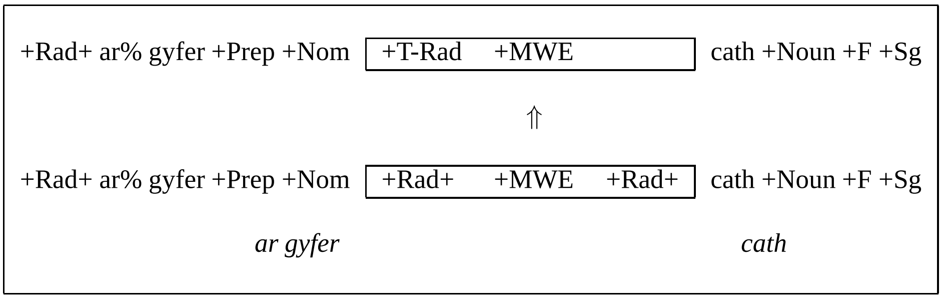
[9] The purpose of this tag is to make it possible in the grammar proper to give MWEs preferential treatment over single word analyses via so-called OT marks (Frank et al., 2001). If the constraints associated with the tag `+MWE` include the appropriate OT mark, non-MWE analyses will be dispreferred.

```
.o.
...
"+T-Rad" <- "+Rad+" || _ ["+MWE"]* "+Rad+"
.o.
"+T-SM" <- "+SM+" || _ ["+MWE"]* "+SM+"
.o.
...
```

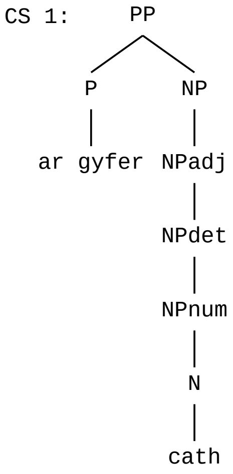
The presence of `["+MWE"]*` in these rules is the reason why there need to be two separate replacement operation sections in the multiword transducer, one for the first of the two matching tags and one for the second. These rules must work around `["+MWE"]*` (= any number of `+MWE` tags). They cannot be executed in one go because they would then have to include `["+MWE"]*` – which would imply replacing *any number of* `"+MWE"` with *any number of* `"+MWE"`, a result that we very definitely do not want.

(34) shows the successful transformation of the concatenated morphological analyses for *ar gyfer cath*, with the corresponding c-structure and sublexical analysis shown in (35) and (36).

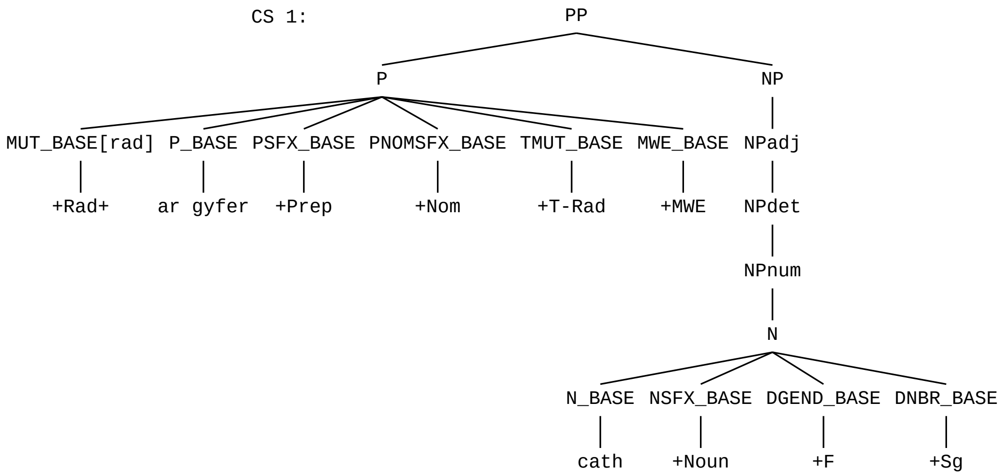
(34)



(35)



(36)



5 Syntactic mutations

Recall that in addition to lexically triggered initial mutations, there are mutations which are triggered by syntactic environments. These two types of initial mutation have in common the fact that the exact mutation target is not predictable. Syntactic mutations apply to the first word in the relevant environment.

5.1 Syntactic Mutations as Categories

A (comparatively simple) example of a syntactic mutation is that governing (post-nominal) attributive APs, given above in (4)-(6) and repeated here as (37)-(39). The first word in a post-nominal AP appears in the Radical if the head noun is PL or MASC SG, but is soft-mutated if the head noun is FEM SG (37). All post-nominal APs are subject to this syntactic mutation (38). The mutation applies to the entire AP (i.e., the first word in the AP), not specifically to the adjective (see (39)).

(37)	*ci*	*mawr*	*cath*	*fawr*
	(ci)	(RAD.mawr)	(cath)	(SM.mawr)
	dog.M.SG	big	cat.F.SG	big

(38)	*cath*	*ddu*	*fawr*
	(cath)	(SM.du)	(SM.mawr)
	cat.F.SG	black	big

(39)	*agwedd*	*bwysig*	*agwedd*	*dra*	*phwysig*
	(agwedd)	(SM.pwysig)	(agwedd)	(SM.tra)	(AM.pwysig)
	aspect.F.SG	important	aspect.F.SG	very	important
	‘(an) important aspect’		‘(a) very important aspect’		

(40) shows possible analyses for the two mutation forms *mawr* and *fawr* of the adjective 'big'. As with all morphological analyses, these start with a tag encoding the word's mutation state (boxed in the example).

(40)	**Surface**	**Lexical**	
	mawr	`+Rad+`	`mawr +Adj`
	fawr	`+SM+`	`mawr +Adj`

The tags, `+Rad+`, `+SM+` etc. are in the lexicon, as shown in (41).[10]

(41)
```
+Rad+  MUT[rad]  XLE.
+SM+   MUT[sm]   XLE.
...
```

The easiest way to understand how we constrain syntactic mutations is through examination of the AP rule (42). The initial element of the right hand side of this rule is a disjunction of the relevant mutation categories. The associated (inside-out) constraints state that if the modified noun is FEM SG, the AP must be soft-mutated (`MUT_BASE[sm]`), that is, start with an initial soft mutation segment, and otherwise that it must begin with the radical (`MUT_BASE[rad]`). The mutation categories are then followed by the remaining constituents of the AP, whatever they are.

(42)

AP –>
{ MUT_BASE[sm]: ((ADJ $\in\uparrow$) GEND) $=_c$ fem
((ADJ $\in\uparrow$) NUM)$=_c$ sg
| MUT_BASE[rad]: { ((ADJ $\in\uparrow$) GEND)$=_c$ masc
((ADJ $\in\uparrow$) NUM)$=_c$ sg
| ((ADJ $\in\uparrow$) NUM)$=_c$ pl } }
... + *remaining constituents of AP*

That is, our treatment of syntactic mutations involves mutations mapping to syntactic categories which appear constituent-initially.

For this to work, it is crucial that the sublexical rules for an adjective, or a pre-adjectival adverb modifying the adjective, or indeed (almost) any other lexical category, do not start with a mutation category. If these rules did start with a (non-optional) mutation category, `MUT_BASE[rad/sm]` in (42) could not appear at the supralexical level in the c-structure

[10] The category names chosen currently have the format of a complex category `MUT[`*`value`*`]` where the value (in square brackets) can be passed to the left hand side of a rule in so-called parameterized rules (Crouch et al., 2006). This was necessary in an earlier approach to mutation in our grammar, but complex categories are no longer necessary in our current approach and could be replaced by simple categories such as `MUTrad`, `MUTsm` etc. We are, however, keeping them for the time being as they may become useful again for possible further improvements to the way we handle syntactic mutations.

because it would be associated with the lexical item it morphologically originates from. The sublexical rule for an adjective only contains the stem and any tags appearing after the stem as shown in (43), while the morphological analysis for adjectives as shown in (40) involves an initial mutation tag.

(43) `A -->` A_BASE *mawr*
ASFX_BASE *+Adj*
+ further (optional) sublexical constituents

Initial mutation tags are thus either consumed by lexical mutation triggers (i.e., deleted by our multiword transducer), in the case of lexically induced mutation, or they are treated supralexically as categories in c-structure rules in the case of syntactically conditioned mutation.

5.2 Syntactic Mutations as Edge Inflections

The treatment of syntactic mutations outlined in this section has the perhaps unexpected feature that it treats the initial mutation tag as mapping to a syntactic terminal in its own right, in apparent violation of lexical integrity. Within the context of our implemented grammar, the reasons for this treatment are largely of a practical nature, for this greatly simplifies the rule set required. Nonetheless, it is basically equivalent to treating syntactically conditioned initial mutation as a type of edge inflection, and an alternative direct encoding of an edge inflection approach is possible, though more complicated and less compact to state and more susceptible to coding error. We illustrate such a comparable approach as edge inflection with the rather simpler case of Basque case marking.

Although Basque is predominantly head-final, adjectival modifiers and demonstratives follow the noun. NPs (or perhaps DPs) in Basque can be inflected for case, number and determinedness (and a few other features). This inflection is marked on the last NP constituent only. Some examples are given in (44); case is always ABS[olutive].

(44)	a.		b.		c.	
		zaldia		*zaldi txikia*		*zaldi txiki hau*
		horse.ABS.SG.DET		horse small.ABS.SG.DET		horse small this.ABS.SG
		'(a/the) horse'		'(a/the) small horse'		'this small horse'

Clearly, whatever phrase structure rules and constraints we assume, we would have to ensure that only the last NP constituent is inflected, and that this inflection is passed up towards the top level of the NP. Respecting lexical integrity strictly, we could pass inflectional information upwards not only by using suitable f-structure annotations on sublexical constituents but also by using XLE's parameterized rules, which contain complex categories on both sides of the rule that hold a value (in square brackets) and whose value could be passed from the right hand side to the left hand side of the rule.

The morphological analysis for *txikia* in (44 b) would be as in (45), capturing the fact (*inter alia*) that the adjective *txikia* is inflected for case (`+Abs`). The lexical entry for the tag `+Abs` is given in in (46) where the tag's category has the case value *abs*. The sublexical rule for adjectives, for instance, as shown in (47), where the case value of the `CASE(_BASE)[_case]` category, whichever this is, would be passed to the left hand side as value of `A[_case]` and all functional information would likewise be passed up via the annotations on the inflectional sublexical categories. This value could again be passed up to the AP as in (48) and from there to the NP as in (49). Similar rules can be written for nouns, NPs and demonstratives. But since non-final NP constituents appear uninflected, we would have to write additional rules for uninflected Ns, NPs, As, APs, etc.

(45) **Surface** **Lexical**
txikia `txiki +Adj +Sg +Art +Abs`

(46) `+Abs CASE[abs] XLE`.

(47) A[_case] −− > A_BASE *txiki*
ASFX_BASE *+Adj*
NUM_BASE: ((ADJUNCT $ ^)=!; *+Sg*
ART_BASE: ((ADJUNCT $ ^)=!; *+Art*
CASE_BASE[_case]: ((ADJUNCT $ ^)=!. *+Abs*

(48) AP[_case] −− > A[_case].

(49) NP[_case] −− > N AP[_case].

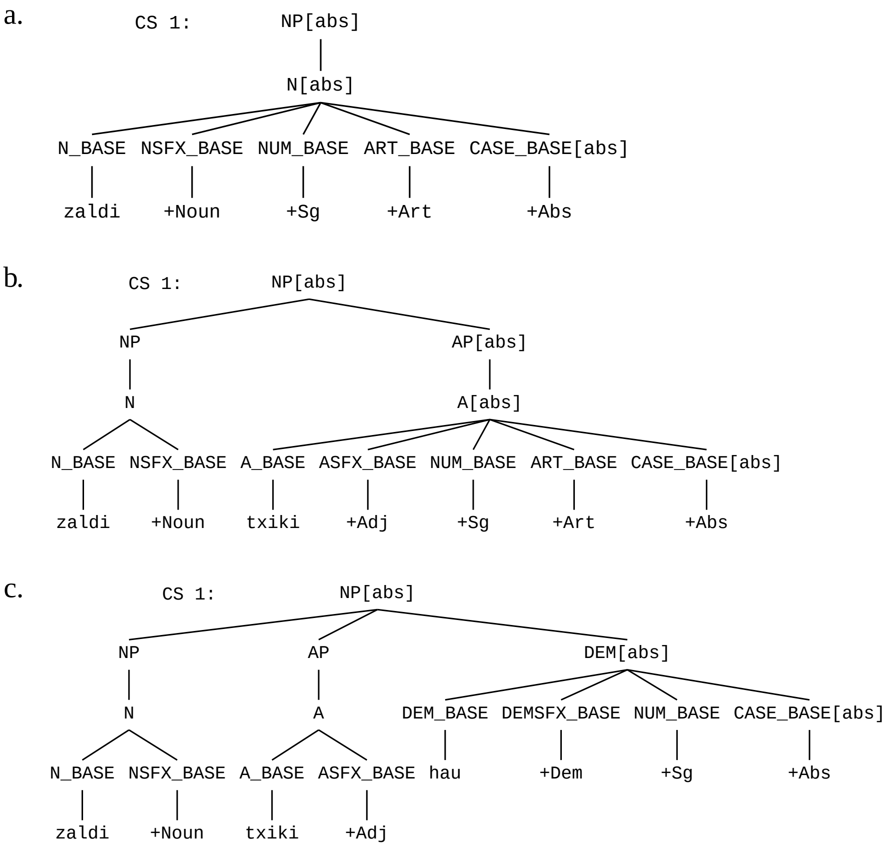
In fact, since the case value is stored in the parameter, we could forego the functional annotation on `CASE_BASE[_case]` in (47) and provide this information wherever `NP[_case]` is instantiated as in (50).

(50) { NP[abs]: (^ CASE)=abs
| NP[erg]: (^ CASE)=erg
| NP[dat]: (^ CASE)=dat *etc.* }

Turning to the NP rule, (51) encodes a flat NP analysis. What is crucial here is the number of disjuncts that take account of the requirement that the last NP constituent, whatever it is, is the locus of case inflection (i.e., a complex category). The distinction between complex (= inflected) and simple (= uninflected) categories in the rule ensures that inflection only appears where it is licensed in the c-structure. (52 a-c) shows the resulting trees for (44 a-c) with all morphemes displayed.

(51) NP[_case] −− > { N[_case]
| N AP* AP[_case]
| N AP* DEM[_case] }.

(52)


a.
CS 1:
NP[abs]
N[abs]
N_BASE NSFX_BASE NUM_BASE ART_BASE CASE_BASE[abs]
zaldi +Noun +Sg +Art +Abs
b.
CS 1:
NP[abs]
NP
AP[abs]
N
A[abs]
N_BASE NSFX_BASE A_BASE ASFX_BASE NUM_BASE ART_BASE CASE_BASE[abs]
zaldi +Noun txiki +Adj +Sg +Art +Abs
c.
CS 1:
NP[abs]
NP
AP
DEM[abs]
N
A
DEM_BASE DEMSFX_BASE NUM_BASE CASE_BASE[abs]
N_BASE NSFX_BASE A_BASE ASFX_BASE hau +Dem +Sg +Abs
zaldi +Noun txiki +Adj

This Basque example shows how an inflectional value originating from the sublexical level can be passed up without necessarily passing up any functional information alongside. It seems to us that a similar approach to syntactic mutation in Welsh is possible, but would be extremely complex: while in Basque the inflection in question appears only on the final word in the NP, in Welsh the nature and distribution of syntactic mutations is much wider, inducing serious complications into the c-structure.

5.3 Default Mutation

One last point remains to be explained: why we deleted the second of two matching mutation tags in our multiword transducer dealing with lexical mutations. This is, in fact, not strictly necessary, but it gives us the considerable practical advantage of being able to specify *one* syntactic mutation as a default. As mentioned above, (almost) all sublexical rules end up without an initial mutation category. If syntactically governed, the mutation category appears in the supralexical c-structure rules; if lexically governed the corresponding tags are deleted even before they can enter the grammar proper.

Syntactic mutations almost always involve either Soft Mutation or the Radical. If we now include the mutation category corresponding to, say, the `+Rad+` tag in the sublexical rules, and make this category optional, we only have to specify those syntactic mutations that do not involve the radical. This means that if we choose not to specify a syntactic mutation in the c-structure rules (and no lexical mutation applies), the mutation category/tag can remain with the lexical item it originates from, and if its only possible value is *rad*, the (overtly mutationally ungoverned) lexical item will default to its radical mutation state, but none other. (53) shows the slightly modified sublexical rule for an adjective vis-à-vis (43) above.

```
(53)  A -->  (MUT_BASE[rad])   +Rad+
             A_BASE            mawr
             ASFX_BASE         +Adj
             + further (optional) sublexical constituents
```

Acknowledgements

The work reported on here was carried in the project Verb Initial Grammars: a Multilingual, Parallel Approach: see `http://users.ox.ac.uk/˜cpgl10015/pargram/index.html`. We are grateful for the financial support of the ESRC (research grant RES-000-23-0505, to Dalrymple and Sadler). We thank participants at LFG06 for useful comments, and especially John Maxwell and Ron Kaplan for much discussion of the multiword transducer.

References

Ball, Martin J and Nicole Müller. 1992. *Mutation in Welsh*. London: Routledge.

Butt, Miriam, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The Parallel Grammar Project. In *Proceedings of Coling 2002, Workshop on Grammar Engineering and Evaluation*, pages 1–7. Online: http:www2.parc.com/isl/members/thking/coling02pg.pdf.

Crouch, Dick, Mary Dalrymple, Ron Kaplan, Tracy King, John Maxwell, and Paula Newman. 2006. XLE documentation. Tech. rep., Palo Alto Research Center, Palo Alto, CA.

Frank, Anette, Tracy Holloway King, Jonas Kuhn, and John Maxwell. 2001. Optimality Theory style constraint ranking in large-scale LFG grammars. In Peter Sells, ed., *Formal and Empirical Issues in Optimality Theory*, pages 367–97. Stanford, CA: CSLI Publications.

Green, Antony Dubach. 2003. The independence of phonology and morphology: the Celtic mutations. *ZAS Papers in Linguistics* 32:47–86. also online http://www.ling.uni-potsdam.de/ green/cv/independ.pdf.

Kaplan, Ron, John Maxwell, Tracy Holloway King, and Richard Crouch. 2004. Integrating Finite-state Technology with Deep LFG Grammars. In *Proceedings of the*

Workshop on Combining Shallow and Deep Processing for NLP (ESSLLI). Online: http://www2.parc.com/isl/groups/nltt/pargram/esslli04fst-xle.pdf.

Stewart, Thomas. 1992. *Mutation as Morphology: Bases, Stems and Shapes in Scottish Gaelic*. Ph.D. thesis, Ohio State University. Online http://www.ohiolink.edu/etd/send-pdf.cgi?osu1086046888.