

# Outlines of Agreement Syntax

László Drienkó

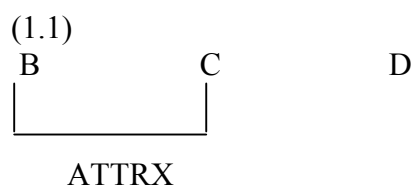
ELTE University, Budapest, Hungary

## Abstract/Introduction

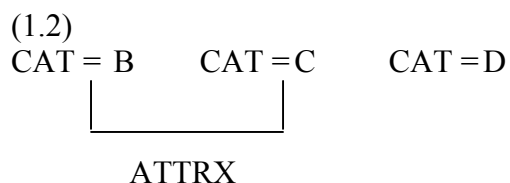
This paper is dedicated to the demonstration of how phenomena in linguistics - particularly, in syntax – can be interpreted in terms of a generalized notion of 'agreement'. The computational basis for our present work is included in the findings reported by Drienkó (2004). In the agreement approach context the linguist's task is to determine relevant features of linguistic elements, the agreement of which features ensures grammaticality. Although Drienkó (2004) provides some solutions for 'problematic' cases, it is also desirable to show the compatibility of the model with more standard issues. After a brief outlining of the agreement approach in the first section, the second part of this paper examines, in the light of agreement properties, such issues as conjugation, inflection, subcategorisation, anaphora, control, raising, and unbounded dependencies.

## 1. Agreement

We assume that a phrase or a sentence is representable as an agreement relation, i.e. as a sequence of elements which satisfies the agreement conditions/constraints for that relation. We shall call a sequence – or more generally, a set - of elements, together with its agreement constraints a PATTERN. Graphically, we represent a pattern as in (1.1)



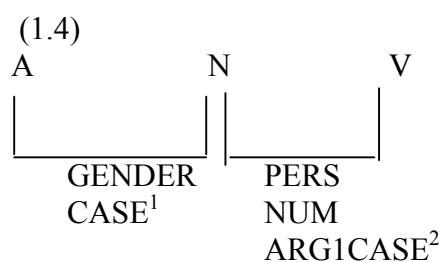
where B, C, and D represent the elements which are configurationally linked together, and the line connecting B and C indicates the explicit agreement constraint that B and C should have the same value for attribute ATTRX. Actually, B, C, and D represent values of input elements with respect to a certain attribute (or attributes). In this work input elements will be represented by their syntactical category (CAT) values. Thus (1.1) is a simplified way of writing (1.2).



For instance, a pattern for (1.3) could look like (1.4).

(1.3)

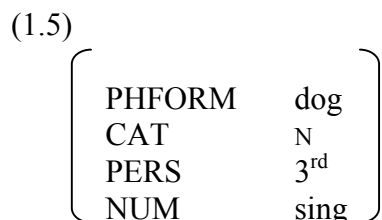
Krasivaya	d'evushka	rabotayet
beautiful-FEM-NOM	girl-FEM-NOM	work-PRES-3 <sup>rd</sup> -sing
The/a beautiful girl works/is_working		



A, N, and V here represent adjectival, nominal, and verbal elements respectively.

We consider the operation of the language faculty to be a mapping from input elements to patterns. Patterns are finite sequences of (symbolic or representational) elements. Each pattern is associated with a finite (possibly empty) set of agreement conditions/constraints.

Input elements are simple attribute-value structures (AVS) consisting of a finite number of attribute-value pairs. (1.5) exemplifies such an AVS.



<sup>1</sup> Since there are two features/attributes that must be checked (GENDER, CASE), arguably, there are two agreement relations between the adjective and the noun. It is also possible to think of a single GENDER-CASE attribute whose values could be fem-nom, fem-acc, ... masc-nom, masc-acc, ... neutr-nom, neutr-acc, ... (corresponding to the respective suffixes. Cf. our discussion of conjunctions of constraints below.)

<sup>2</sup> Agreement of ARG1CASE values ensures that the subject – argument 1 – of the verb be in the nominative case.

Agreement constraints are finite sets of symbolic representations of requirements for agreement relations. Basically, they have three parts (cf. (1.6)).

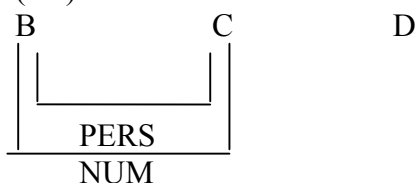
(1.6)

1 3 4 PERS FIRST-TO-FIRST

The first part specifies the elements that take part in the agreement relation. The second part specifies which attribute must have the same value for all the elements taking part in the agreement relation.<sup>3</sup> The third part, the agreement strategy, is relevant for recursive cases – see below –, and it can also include restrictions/conditions as to the application of the stated agreement constraint.<sup>4</sup> Thus an agreement constraint is fulfilled, if the elements enlisted in the first part of the constraint have the attribute in the second part specified, and its value is the same for all the elements (while using the necessary agreement strategy in the recursive case).

Agreement constraints may be (finite-length) conjunctions or disjunctions of other constraints. Requiring that all 'subconstraints' of a constraint be satisfied can be interpreted as a logical AND relationship. Our default interpretation of the relationship between constraints is conjunction, since we require that all constraints of a pattern should be fulfilled. In this respect, (1.7) and (1.8), for instance, are equivalent, so graphically we will use only one line for several (AND) agreement relations between the same elements.

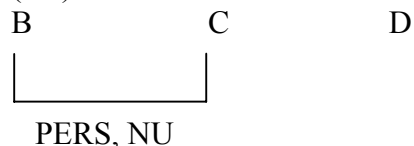
(1.7)



Constr<sub>1</sub> : 1 2 PERS FIRST-TO-FIRST

Constr<sub>2</sub> : 1 2 NUM FIRST-TO-FIRST

(1.8)



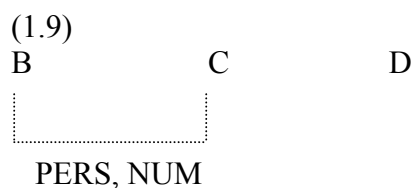
<sup>3</sup> We use the word 'element' both for input elements and for symbolic or representational elements constituting patterns. In the general case it will cause no problem, but it should be kept in mind that the elements of patterns are representations of features of input elements.

<sup>4</sup> The notion of Conditional Agreement is introduced in section 2.1. Cf. (2.24) and (2.31).

Constr:

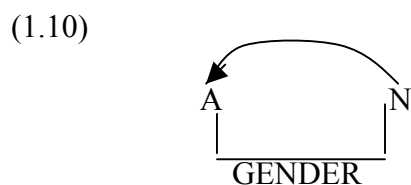
$$\text{Constr}_1 \wedge \text{Constr}_2 = (1 \ 2 \ \text{PERS} \ \text{FIRST-TO-FIRST}) \wedge (1 \ 2 \ \text{NUM} \ \text{FIRST-TO-FIRST})$$

Agreement constraints can be applied disjunctively, as well. This means that, given e.g. constraint *Constr*:  $\text{Constr}_1 \vee \text{Constr}_2$ , either  $\text{Constr}_1$  or  $\text{Constr}_2$  should be satisfied. In the examples below our OR relations will not be exclusive. Disjunctive agreement relations will be indicated with dotted lines. In (1.9) for instance either the person or the number value must be the same for B and C.



Elements can be mapped onto patterns recursively. We use arrows to show recursion in patterns. Thus, e.g. (1.10) is a recursive pattern licensing a potentially infinite sequence of alternating adjectival and nominal elements.

However, in recursive cases it may not be straightforward how to check agreement since several input elements can be mapped on a single element of a pattern. This is where agreement strategies come into play. Given, for instance, pattern (1.10)



and input elements

$i_1$ : PHFORM = *krasivaya*, CAT = A,  
GENDER = fem

$i_2$ : PHFORM = *malen'kiy*, CAT = A,  
GENDER = masc

$i_3$ : PHFORM = *devushka*, CAT = N,  
GENDER = fem

$i_4$ : PHFORM = *malchik*, CAT = N,  
GENDER = masc,

let us assume that input elements are mapped onto (1.10) in the following order:  $i_1$   $i_3$ ,  $i_2$ ,  $i_4$ , - i.e. *krasivaya devushka malen'kiy malchik*. Then it may be desirable that  $i_1$  agree with  $i_3$  and  $i_2$  with  $i_4$ . This type of agreement strategy will be called FIRST-TO-FIRST since agreement checking observes the order in which elements are mapped.

Another possibility – absurd for our current example, but useful in other cases – is checking agreement in semi-reverse order, that is the checking process observes mapping order for one element in the pattern, but input elements mapped on another element of the pattern are taken in reversed order. Such a strategy may be called LAST-TO-FIRST. Imagine that input elements  $i_1$ -  $i_4$  are mapped on (1.11) in the order  $i_1$   $i_2$ ,  $i_4$ ,  $i_3$ , i.e. *krasivaya malen'kiy malchik devushka*. Then strategy LAST-TO-FIRST guarantees that, again, *malen'kiy malchik* and *krasivaya devushka* be checked. We shall indicate the agreement strategy explicitly – in brackets below the line representing the agreement - only when it is not the default FIRST-TO-FIRST.



Note (1.11) can be seen as representing a sort of nested dependency, since in the event of encountering input sequence  $A_1, A_2, \dots, A_n, N_n, \dots, N_2, N_1$  the LAST-TO-FIRST strategy will couple the elements as the indices show.

Of course, there may be many more strategies to check agreement for recursive cases. It is for linguistic research to determine which strategy is optimal for which case.

Agreement constraints can be conditional in that they can depend on specified properties of input elements. In the end of 2.1 we provide two examples of how the notion of conditional agreement may be useful.

## 2.1 Conjugation, Inflection, Subcategorization

Agreement relations readily lend themselves for representing conjugation paradigms. Consider the Spanish sentences in (2.1).

(2.1)

Yo soy húngaro/húngara.

Tú eres húngaro/húngara.

Él es húngaro.

Ella es húngara.

Nosotros somos húngaros.

Nosotras somos húngaras.

Vosotros sois húngaros

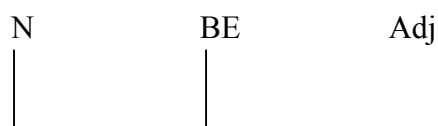
Vosotras sois húngaras

Ellos son húngaros

Ellas son húngaras

The person-number agreement constraint in (2.2) ensures the appropriate form of verb 'ser'('be') in (2.1) as required by the pronoun preceding it

(2.2)



PERS, NUM  
ARG1CASE

Agreement of the *ARG1CASE* values (say 'ARG1CASE=nom' for both the (pro)nouns and the verbs) prescribes that the subject – the first argument - of the finite verb must have the nominative case.<sup>5</sup>

Note that for the conjugation of the verb 'be' in English it is not necessary to represent the six different person-number values corresponding to 1<sup>st</sup> person singular ... 3<sup>rd</sup> person plural.

---

<sup>5</sup> In fact, there can be several types of predicative agreement relations, so that it might be useful to differentiate them. One way to do so would be to use SVOARGCASE instead of just ARGCASE for representing constraints responsible for the appropriate grammatical cases of the arguments of a verb subcategorizing for a subject and an object.

Indeed, consider the NBE (noun-be agreement) attribute value specifications in (2.3) for the lexical items involved.

(2.3)

I: NBE = am	YOU: NBE = are	WE: NBE= are	THEY: NBE= are
AM: NBE = am	ARE: NBE = are		
HE: NBE = is	SHE: NBE = is	IT: NBE = is	
IS: NBE = is			

For instance, the NBE values for 'we', 'you', 'they', and 'are' are the same, so 'we are', 'you are', and 'they are' are all correctly licensed. Thus three possible values of a single attribute can characterise the present conjugation of 'be'. This seems to be a simple way of handling exceptions.<sup>6</sup>

For other verbs of English, two values appears to be enough, corresponding to the 3<sup>rd</sup> person singular case (+3ps) and the rest (-3ps):

(2.4)

I: NV = -3ps	YOU: NV = -3ps	WE : NV = -3ps	THEY: NV = -3ps
HE: NV = +3ps	SHE: NV = +3ps	IT: NV = +3ps	
GO: NV = -3ps	GOES: NV = +3ps		

With six NBE values, we can encode the subject-verb agreement in (2.1) similarly (cf. (2.5)).

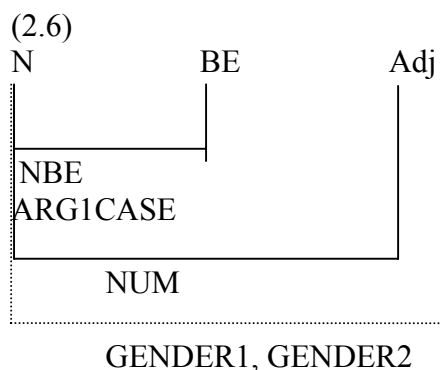
(2.5)

YO: NBE= 1ps	TÚ: NBE=2ps	EL/ELLA: NBE=3ps
NOSOTROS: NBE= 1pp	VOSOTROS: NBE=2pp	
NOSOTRAS: NBE= 1pp	VOSOTRAS: NBE=2pp	ELLOS/ELLAS: NBE = 3pp
SOY: NBE= 1ps	ERES: NBE=2ps	ES: NBE=3ps
SOMOS: NBE= 1pp	SOIS: NBE=2pp	SON: NBE = 3pp

In (2.1) there is also number and gender agreement between the (pro)noun and the adjective. (2.6) can handle it.

---

<sup>6</sup> Of course, any name for attributes or for their values can be invented at our convenience.



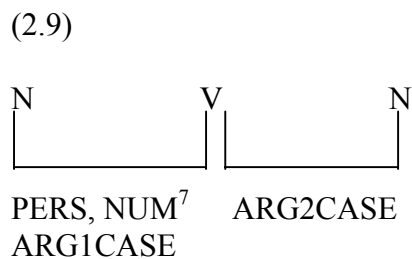
Since the singular first and second person pronouns can take both masculine and feminine form adjectives while the gender of the other pronouns is determined more directly, the dotted line indicates alternatives: we require either the agreement of GENDER1 or the agreement of GENDER2. The desired effect is achieved through the following value specifications (2.7).

(2.7)	
YO, TÚ:	GENDER1 : nonfem GENDER2 : fem
EL, ELLOS, NOSOTROS, VOSOTROS:	GENDER1: nonfem
ELLA, ELLAS, NOSOTRAS, VOSOTRAS:	GENDER1: fem
HUNGARO:	GENDER1 : nonfem GENDER2 : nonfem
HUNGARA:	GENDER1 : fem GENDER2 : fem

A similar analysis for

(2.8) *She loves him*

would require the pattern



with the following attribute values.

<sup>7</sup> For better readability we still write PERS, NUM instead of just the NV attribute introduced earlier.



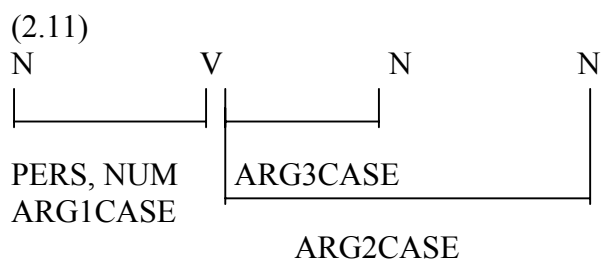
SHE: NUM = sing PERS = 3<sup>rd</sup> ARG1CASE = nom ...  
 LOVES: NUM = sing PERS = 3<sup>rd</sup> ARG1CASE = nom ARG2CASE = acc ...  
 HIM: NUM = sing PERS = 3<sup>rd</sup> ARG2CASE = acc ...

Note that the transitivity of the verb *love* can be expressed through its agreement property that it fits into a pattern requiring an object – a noun after the verb.

Verbs may subcategorize for more than two arguments. In

(2.10) He gives her money

'gives' takes a subject, an indirect object, and a direct object. The corresponding pattern is (2.11)



and the attribute value specification for the lexical items should imply

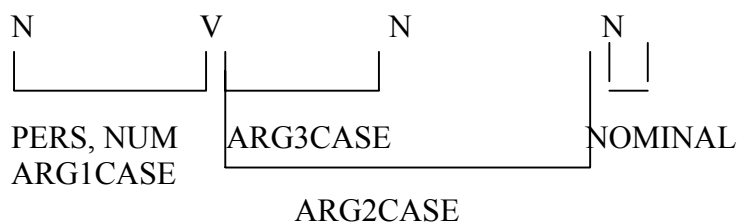
HE: ARG1CASE=nom ...  
 GIVES: ARG1CASE=nom ARG2CASE=acc ARG3CASE= dat ...  
 HER: ARG2CASE=acc ARG3CASE=dat ...  
 MONEY: ARG1CASE=nom ARG2CASE=acc ARG3CASE=dat ...

In order to exclude sentences like

(2.12) \*He gives money her

we have to differentiate between nouns and pronouns, since pronouns must not occupy the final position in (2.11). To do so, let us introduce feature *NOMINAL=yes* for nouns. Furthermore, let us require an agreement of the last N in (2.11) with itself on attribute *NOMINAL*. Thus (2.11) will look like (2.13).

(2.13)



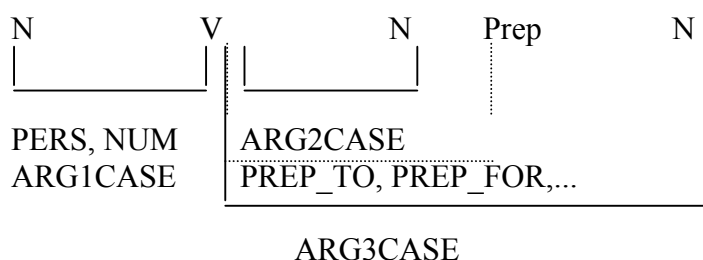
Because NOMINAL is not specified for pronouns, agreement checking will fail with sentences like (2.12).

On the other hand

(2.14) He gives money to her

is licensed by pattern

(2.15)



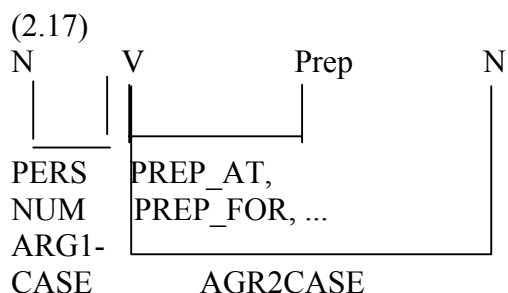
assuming the same attribute value specifications as above. The optional agreement on attributes PREP\_TO, PREP\_FOR, etc. represents a choice between several prepositions that can possibly fit in the pattern – as licensed by the verb. Note, that *AGR3CASE* = *dat* guarantees that the noun after the preposition cannot be in the nominative case.

Similar arguments may apply in the case of verbs with compulsory prepositions or morphemes on the noun (cf. (2.16)).

(2.16) *He looked at her*

For (2.16) a *PREP\_AT* = *yes* property may specify that *look* can take the preposition *at*.

The very same pattern



together with

LOOKED: ..., PREP\_AT = yes, PREP\_FOR = yes, ARG2CASE = acc, ...  
 FOR: ..., PREP\_FOR = yes, ...  
 AT: ..., PREP\_AT = yes, ...  
 HER: ..., ARG2CASE = acc, ...

also licenses, e.g.: *He looked for her.*

Of course, there are other ways to guarantee that the preposition be followed by a non-nominative (pro)noun. For instance, a *PREPARGCASE=nonnom* agreement property could tell the system that the argument of a preposition is a non-nominative (pro)noun. However, in (2.15) and (2.17) it is unnecessary. As (2.15) and (2.17) show, we need not refer directly to the preposition-noun correlation.<sup>8</sup>

The Hungarian sentence

(2.18) *Mari segít Janinak*  
 Mary helps John<sub>dat</sub>

can be analysed with

<sup>8</sup> This may relate to some evolutionary aspect of the appearance of prepositional phrases. In, e.g., (2.17) the final (pro)noun is considered to receive its accusative (non-nominative) case from the verb rather than from the preposition. This may well correspond to a 'primordial' situation when there existed only, say, "verbs" and "nouns". Then e.g. *look her* may have been a correct phrase. Later on the appearance of prepositions could modify the semantic relation between 'look' and 'her', but case assignment might remain the noun's task.

(2.19)



PERS  
 NUM ARG2CASE  
 ARG1-  
 CASE

and

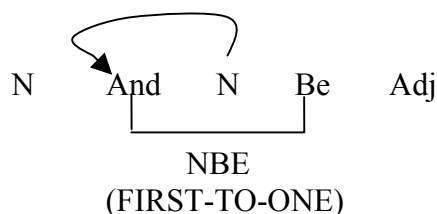
MARI:           ..., ARG1CASE= nom,       ...  
 SEGÍT:       ..., ARG1CASE=nom,       ARG2CASE = dat, ...  
 JANINAK:                   ...,       ARG2CASE = dat, ...

Subject-verb agreement is more complicated in recursive cases. For, e.g.,

(2.20) Jack and Jill (and ... ) are sad.

we can declare that the *NBE = are* feature of 'be' is determined by an agreement with 'and', assuming *NBE = are* for 'and'. This seems logical, since it is co-ordination that causes the plurality of the phrase. (2.21) sketches a corresponding pattern.

(2.21)



Strategy FIRST-TO-ONE ensures that agreement be checked only once, i.e. between the first 'And' (FIRST) and the single 'Be' (ONE). Of course, 'Be' could be agreed with any 'And', not only with the first one. Unfortunately, there can be more complicated cases. Consider the Hungarian sentences in (2.22).

(2.22)

Te és én vagyunk itt  
 Ti és én vagyunk itt  
 Mi és ti vagyunk itt  
 Ő és én vagyunk itt  
 Ő és te vagytok itt  
 Ő és ti vagytok itt

Ők és én vagyunk itt  
 Ők és te vagytok itt  
 Ők és ő vannak itt  
 Ők és te és ti és én és ő vagyunk itt..  
 Ő és te és ők és ti és ő és ő vagytok itt.  
 Ők és ő és ő és ők ő vannak itt.

(én : I, te: you-2<sup>nd</sup> –sing, ő: he/she/it mi: we, ti: you-2<sup>nd</sup>-pl, ők: they  
 vagyunk: we are, vagytok: you-all are, vannak: they are, itt: here, és: and)

As the data reveal, the subject-verb agreement is effected somewhat semantically.

We can argue that the word 'és' ('and') determines the NUM=plural feature. However, the PERS value is less straightforward. But we note the following regularities:

(2.23)

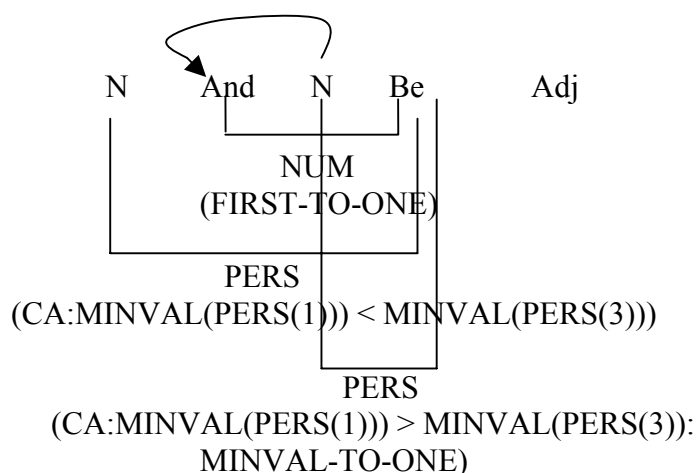
PERS values of co-ordinated (pro)nouns:

PERS value of verb:

1 <sup>st</sup>	2 <sup>nd</sup>	1 <sup>st</sup>
1 <sup>st</sup>	3 <sup>rd</sup>	1 <sup>st</sup>
2 <sup>nd</sup>	3 <sup>rd</sup>	2 <sup>nd</sup>
3 <sup>rd</sup>	3 <sup>rd</sup>	3 <sup>rd</sup>

That is, PERS values observe a certain hierarchy. The occurrence of PERS= 1<sup>st</sup> entails the same value for the verb. The occurrence of PERS=2<sup>nd</sup>, if no PERS=1<sup>st</sup> is present entails PERS=2<sup>nd</sup> for the verb. The verb will have the PERS=3<sup>rd</sup> feature only if no first or second person pronouns are present. Alternatively, we can say that the verb agrees in person with the (pro)noun which has the least PERS value. We can use strategy MINVAL-TO-ONE to handle the situation. This strategy selects that element of several others (mapped recursively on the same representational element of a pattern) which has the minimum value (MINVAL) on an attribute and checks whether it agrees with the value for a single (ONE) element (mapped non-recursively on another representational element of the pattern) (cf. (2.24)).

(2.24)



For (2.24), strategy MINVAL-TO-ONE guarantees the proper agreement between a noun mapped on the 3<sup>rd</sup> element of the pattern and 'be' (mapped on the 4<sup>th</sup>). However, it may well be the case that the PERS value for the first noun (mapped on the 1<sup>st</sup> element) is less than the minimal value for the nouns mapped on the 3<sup>rd</sup> element. Then the first noun will determine agreement. To tackle the situation we introduce the notion **Conditional agreement**. 'Conditional' here means that the agreement relation is required to hold only under specified circumstances, i.e., only if some conditions are fulfilled. Thus, '(CA:MINVAL(PERS(1))) < MINVAL(PERS(3))' in (2.24) represents a conditional agreement relation between the first and the fourth elements of the pattern, which agreement relation must be observed only if the minimal PERS value for the first element (the PERS value for the single noun itself) is less than the minimal value for the nouns mapped on the third element. The other conditional constraint in (2.24) reads similarly, except that it applies in the opposite case – MINVAL(PERS(1))) > MINVAL(PERS(3)) – , plus it specifies strategy MINVAL-TO-ONE as the necessary strategy instead of the default FIRST-TO-FIRST.<sup>9</sup>

There is an interesting feature of Hungarian verb suffixation with respect to (in)definiteness, as pointed out by László Kálmán (personal communication). In the present tense there is an opposition between definite and indefinite conjugation. This is shown in (2.25)-(2.26).

(2.25) Látom                                      a                      fiút  
           See-PRES-SING-1-DEF            def. article       boy-ACC  
           'I see the boy'

<sup>9</sup> There may be more controversial cases besides those referred to in (2.22). A more detailed analysis, however, would go beyond the objectives of this paper.

(2.26) Látok egy lányt  
 See-PRES-SING-1-INDEF indef. article girl-ACC  
 'I see a girl'

The combination of the two cases results in ungrammatical sentences (cf. (2.27)-(2.28)).

(2.27) \*Látok a fiút és egy lányt és ...  
 Intended: 'I see the boy and a girl and ...'

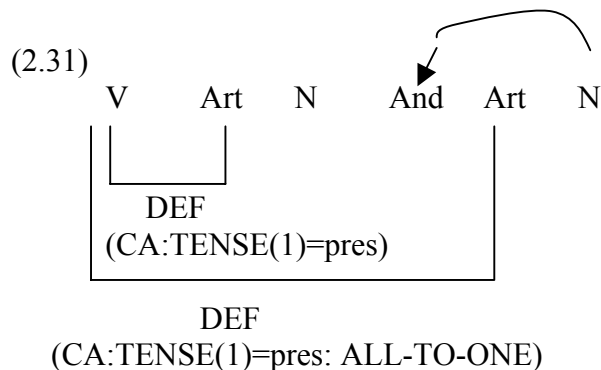
(2.28) \*Látom egy fiút és a lányt és ...  
 Intended: 'I see a boy and the girl and ...'

However, the opposition disappears in the past tense, as can be seen in (2.29)-(2.30).

(2.29) Láttam a/egy fiút  
 See-PAST-SING-1-DEF def. article/indef. article boy-ACC  
 'I saw the/a boy'

(2.30) Láttam a fiút és egy lányt és ...  
 'I saw the boy and a girl and ...'

The situation can be characterised by (2.31).



A: ..., DEF= def, ...

LÁTOM: ... DEF = def, TENSE = pres...

LÁTTAM: ..., TENSE = past, ...

EGY: ..., DEF= indef, ...

LÁTOK: ..., DEF = indef, TENSE = pres ...

The conditional agreement requirements in (2.31) demand that agreement checking with respect to definiteness must be done only if the verb (mapped on the first element) is in the present tense. The ALL-TO-ONE strategy guarantees that every article (ALL) mapped on the 5<sup>th</sup> element of the pattern is agreed with the single verb (ONE), mapped on the 1<sup>st</sup> element.<sup>10</sup>

<sup>10</sup> Some speakers may feel that the verb must share the (in)definiteness property only with the article that immediately follows it. Adopting that view would make the recursive agreement constraint (between the 1<sup>st</sup> and the 5<sup>th</sup> elements) in (2.31) unnecessary .

## 2.2 Anaphora

Anaphora means that certain elements in the sentence are predetermined by their *antecedents*.

In

(2.32) He loves himself

for instance, *he* predetermines – is the antecedent of – *himself*. That is

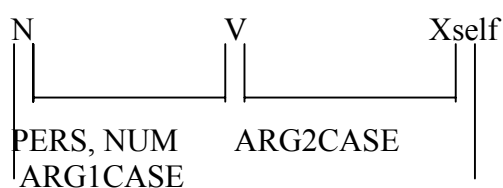
(2.33) \*He loves herself

or

(2.34) \*He loves myself

is incorrect. To analyze such sentences we need a pattern like

(2.35)



PERS, NUM, GENDER

and the attribute values for the lexical items involved should imply

HE:	PERS = 3 <sup>rd</sup>	NUM = sing	GENDER = masc	ARG1CASE=nom ...
LOVES:	PERS = 3 <sup>rd</sup>	NUM = sing	ARG1CASE = nom	ARG2CASE = acc
HIMSELF:	PERS = 3 <sup>rd</sup>	NUM = sing	ARG2CASE = acc	GENDER = masc

In this setting,

\*He loves herself

would qualify as incorrect because of the gender mismatch, while in

\*He loves myself

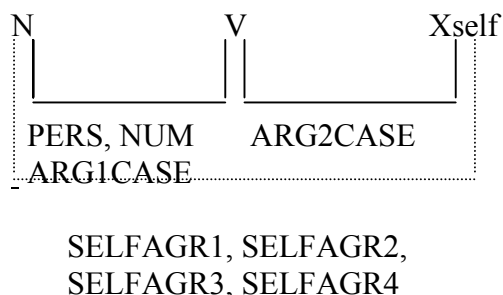


the PERS values would fail to match. However, gender assignment for the reflexive pronouns, except 'himself', 'herself', and 'itself', may be problematic. Alternatively, the PERS, NUM, GENDER agreement constraints in (2.35) can be replaced with SELFAGR ones, together with the following value specifications:

- |   |  |
|---|--|
| I: SELFAGR1=i_myself                                  | MYSELF: SELFAGR1=i_myself  |
| YOU: SELFAGR1=you_yourself<br>SELFAGR2=you_yourselves | YOURSELF: SELFAGR1=you_yourself<br><br>YOURSELVES: SELFAGR2=you_yourselves |
| HE: SELFAGR1=he_himself                               | HIMSELF: SELFAGR1=he_himself   |
| SHE: SELFAGR1=she_herself                             | HERSELF: SELFAGR1=she_herself  |
| IT: SELFAGR1=it_itself                                | ITSELF: SELFAGR1=it_itself   |
| WE: SELFAGR1=we_ourselves                             | OURSELVES: SELFAGR1=we_ourselves   |
| THEY: SELFAGR1=they_themselves                        | THEMSELVES: SELFAGR1=they_themselves                                       |

The disjunctive agreement of either SELFAGR1 or SELFAGR2 in (2.36) now rules out sentences like (2.33) or (2.34) plus it can guarantee the correctness of both 'You love yourself' and 'You love yourselves'.

(2.36)



(As introduced earlier, discontinuous lines indicate disjunctive agreement, i.e. alternative agreement possibilities.)

Furthermore, one might wish not to deem sentences like (2.37) ungrammatical.

(2.37) I love ourselves  
She/he/it loves themselves

The desired effect can be achieved by specifying attribute values of SELFAGR1- SELFAGR4 as follows plus by using pattern (36):

I:  
SELFAGR1=i\_myself  
SELFAGR2=i\_ourselves

MYSELF:  
SELFAGR1=i\_myself

YOU:  
SELFAGR1=you\_yourself  
SELFAGR2=you\_yourselves

YOURSELF:  
SELFAGR1=you\_yourself

YOURSELVES:  
SELFAGR2=you\_yourselves

HE:  
SELFAGR1=he\_himself  
SELFAGR2=he\_themselves

HIMSELF:  
SELFAGR1=he\_himself

SHE:  
SELFAGR1=she\_herself  
SELFAGR3=she\_themselves

HERSELF:  
SELFAGR1=she\_herself

IT:  
SELFAGR1=it\_itself  
SELFAGR4=it\_themselves

ITSELF:  
SELFAGR1=it\_itself

WE:  
SELFAGR1=we\_ourselves

OURSELVES:  
SELFAGR1=we\_ourselves  
SELFAGR2=i\_ourselves

THEY:  
SELFAGR1=they\_themselves

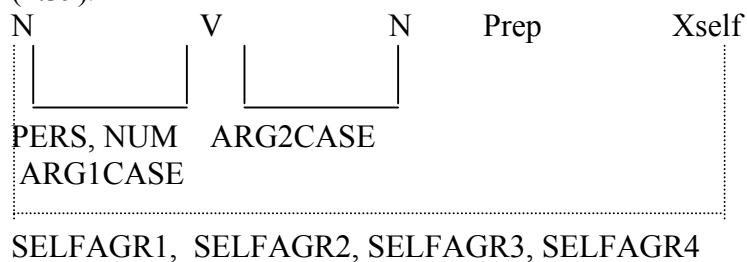
THEMSELVES:  
SELFAGR1=they\_themselves  
SELFAGR2=he\_themselves  
SELFAGR3=she\_themselves  
SELFAGR4=it\_themselves

Similarly, for

(2.38) Joe painted pictures of himself

we may suppose an agreement of the SELFAGR values of *Joe* and *himself*, as shown in (2.39).

(2.39).



Pattern (2.39) also licenses sentences like (2.40).

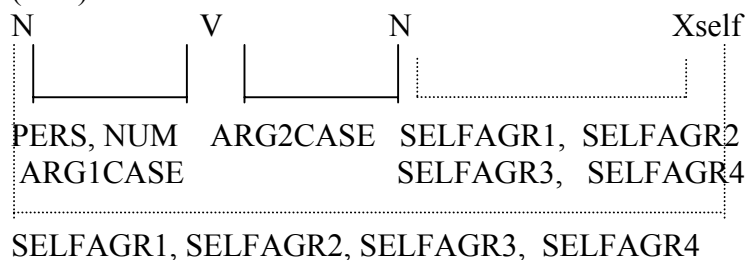
- (2.40) You painted pictures of yourself/yourselves.  
 I painted pictures of myself/ourselves.  
 She painted pictures of themselves

Now consider the examples in (2.41).

- (2.41) I showed you myself (in the mirror)  
 I showed you yourself (in the mirror)  
 I showed you yourselves (in the mirror)  
 You showed me myself (in the mirror)  
 You showed me yourself (in the mirror)  
 You showed me yourselves (in the mirror)  
 He showed me himself/themselves/myself/ourselves (in the mirror)

In the sentences of (2.41) the antecedent of the reflexive pronoun can be either the subject or the indirect object. This is reflected in (2.42).

(2.42)



Of course, the object forms of the pronouns must have the necessary values:

ME:  
 SELFAGR1=i\_myself  
 SELFAGR2=i\_ourselves

HIM:  
 SELFAGR1=he\_himself  
 SELFAGR2=he\_themselves

HER:  
 SELFAGR1=she\_herself  
 SELFAGR3=she\_themselves

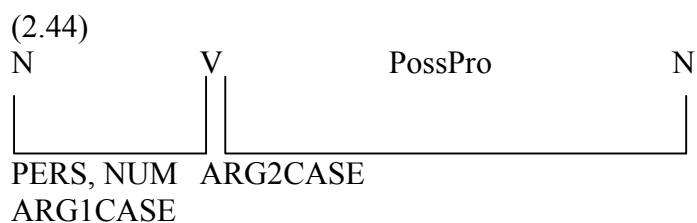
US:  
 SELFAGR1=we\_ourselves

THEM:  
 SELFAGR1=they\_themselves

With possessive pronouns the situation is a bit different. Consider

(2.43) Joe likes his father.

Here we may not require an agreement of the relevant values of *Joe* and *his*. In fact, we may not require any syntactic correlation between the two elements. See (2.44)



(2.47) He promised her to go

Here 'he' is the subject of 'go'.

In the strict sense, the problem of control (and raising) does not seem to be of truly syntactic nature. A pattern

(2.48) N V N to V

can underlie both sentences above. Or we can differentiate between the two cases distinguishing the respective word categories:

(2.49) N V<sub>oc</sub> N to V  
N V<sub>sc</sub> N to V

However, to demonstrate the applicability of the arguments set forth in the previous sections, we provide an analogous analysis.

Our straightforward idea is that we can modify (2.48) so that it incorporates an option for the verb with regards to whether it chooses to control its subject or its object. (2.50) contains this option. (Other agreement properties are omitted.)

(2.50)

N	V	N	to	V
SC	OC			

PROMISED:	PERSUADED:	HIM:	HE:	JOE:
SC=yes	OC=yes	OC=yes	SC=yes	SC=yes, OC=yes

Of course, (2.50) licenses sentences like those in (2.51).

(2.51)  
 He promised/persuaded Joe to go  
 Joe promised/persuaded Joe to go  
 Joe promised/persuaded him to go

Similar arguments may hold for sentences with *seem*, a subject raising verb.

(2.52) He seems to be happy

(2.53) He seems happy

The respective patterns for (2.52) and (2.53) are given in (2.54) and (2.55)

(2.54) N V to BE Adj



SR

(2.55) N V Adj



SR

SEEMS:

SR=yes

HE:

SR=yes

To also allow other verbs, not only 'strictly' subject-raising ones, one just has to keep to the following recipe. Specify *SR1 = yes* for subject-raising verbs and *SR2 = yes* for other verbs that you want to allow. Specify *SR1 = yes* and *SR2 = yes* for (pro)nouns that you wish to allow as subjects. Require disjunctive agreement between the noun and the verb on attributes SR1 or SR2.<sup>11</sup>

Arguably, in (2.54) and (2.55) the configuration of word categories itself determines that only subject raising verbs can fit into the pattern. This can be guaranteed through agreement of the verbs with themselves on attribute SR. That is, in this respect, (2.56) and (2.57) are equivalent with (2.54) and (2.55) respectively.

(2.56) N V to BE Adj



SR

(2.57) N V Adj



SR

<sup>11</sup> Note that the same effect can be achieved by simply extending feature *SR = yes* for all the legal verbs for the pattern – provided the linguist does not insist on theoretical distinction in similar cases.

Here again, the notion of raising verbs can be extended in the same fashion that was described above.

The object of an object-raising verb – like *believe* – becomes the subject of the following predicate:

(2.58) He believes her to be happy

(2.59) He believes her happy

The respective patterns are (2.60) and (2.61).

(2.60)

N	V	N	to	BE	Adj
		ObjR			

(2.61)

N	V	N	Adj
		ObjR	

BELIEVES:	HER:
ObjR=yes	ObjR=yes

The class of verbs fitting (2.60) and (2.61) can be expanded in the same ways as mentioned above in connection with subject raising verbs.

## 2.4 Wh-questions, unbounded dependency

The term unbounded dependency refers to sentences where one element is moved a distance away from its 'original' place. The distance can be made arbitrarily long by embedding sentences recursively. The notion is closely related to wh-questions, which are regarded as a prototypical case. In this section we investigate the analysis of wh-questions.

First of all we point out again that we think in terms of patterns. In accordance with that, we may not view questions as the result of some transformation of the declarative sentence. We





CAN: AUX=yes                      LOVE:  
       AUXN1=3<sup>rd</sup>sing                NONF=yes  
       AUXN2=non3<sup>rd</sup>sing

HE, JOHN: AUXN1=3<sup>rd</sup>sing        I, YOU, WE, THEY: AUXN2=non3<sup>rd</sup>sing

Note that we need not refer to the word category 'auxiliary'. We may think of auxiliaries as a subcategory of verbs. Specifying AUX=yes for a verb means that it is - or it can be used as - an auxiliary.<sup>12</sup> The AUXN1 or AUXN2 requirement is a simplified form of person-number agreement between the auxiliary and the noun. Basically, it is only necessary because of the 'do/does' and the 'have/has' dichotomies. The fulfilment of condition NONF=yes guarantees the nonfinite (unaffixed) form of the main verb.

Besides (2.66), (2.67) also allows sentences like (2.68)

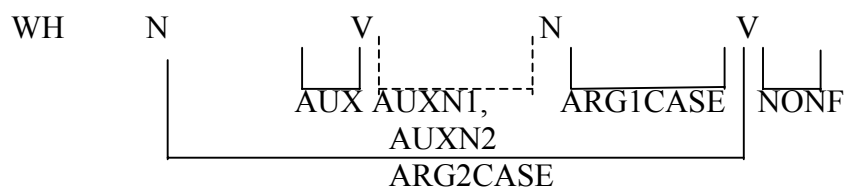
- (2.68)
- Who does John love
- Who can they love
- Who can you do
- Who does he do

The analysis of

(2.69) Which girl    does    he    love

(an instance of 'pied-piping') is quite similar. See (2.70).

(2.70)



Clearly, sentences like (2.71) should be avoided.

<sup>12</sup> As a matter of fact, we do not even need the AUX=yes feature to specify auxiliaries, because the AUXN1/AUXN2 properties can do the job as well, since they are only specified for auxiliary verbs. However, we keep the AUX attribute for better readability of patterns.

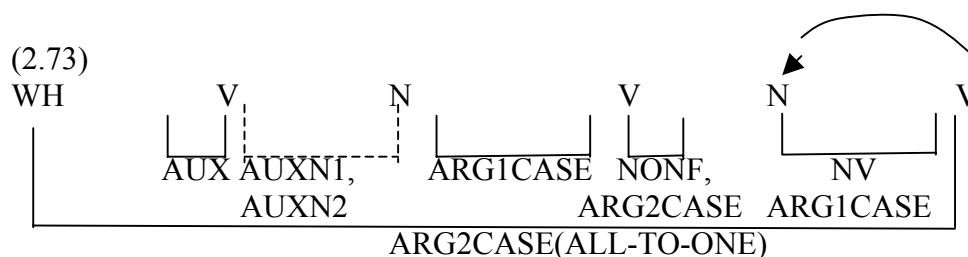
(2.71) \*Where/why/how/when girls does he like

To do so we may break down WH into subcategories by, e.g. introducing feature ATTRWH=yes for *which*, *what* and *whose*, and requiring the agreement of WH with itself on ATTRWH.

Unbounded dependencies can be arbitrarily long since clauses can be embedded recursively. Consider

(2.72) Who does Jim know Peter believes they know Eve trusts

For the analysis we can use (2.73).



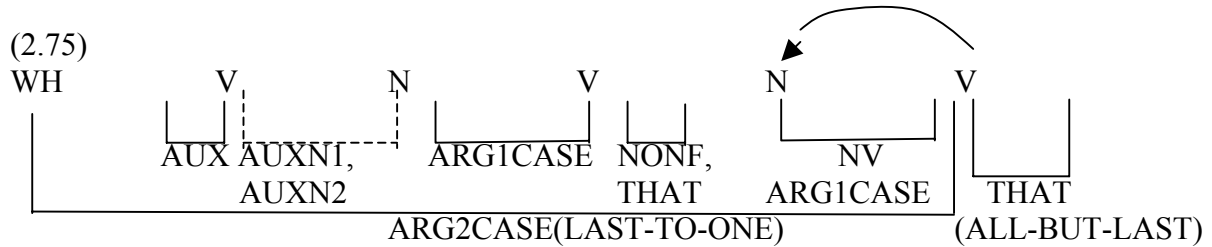
Unfortunately, (2.73) only guarantees that the main verbs of the sentence are transitive verbs (having ARG2CASE specified). Thus, for instance, sentence (2.74) would be legal.

(2.74) Who does Jim hurt Peter climbs they drink Eve picks

Obviously, the problem lies in the difference between verbs like 'know', 'believe', 'suppose', etc. that is, verbs which can take an object clause with or without complementizer 'that' and other 'normal' transitive verbs.

A way of augmenting the situation is to assume feature THAT=yes for verbs which can be followed by a clausal object. Supposing that all 'that-verbs' are transitive, the THAT=yes requirement can take the place of the ARG2CASE correlations in (2.73). However, the last verb should not necessarily be a that-verb. Although, it must be transitive. The desired effect can be achieved through introducing recursive agreement strategy ALL-BUT-LAST for the THAT agreement relation of the 6<sup>th</sup> element of the pattern and a LAST-TO-ONE ARG2CASE agreement between the 1<sup>st</sup> and the 6<sup>th</sup> elements. The ALL-BUT-LAST agreement guarantees that all verbs mapped on the 6<sup>th</sup> element will be checked for "thatness",

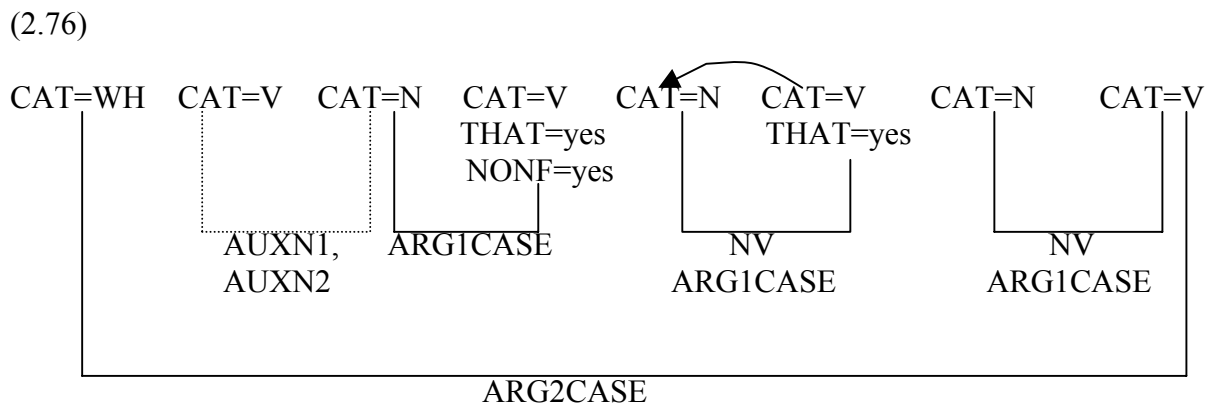
with the exception of the last verb. On the other hand, the last verb will be checked with respect to transitivity due to the LAST-TO-ONE ARG2CASE relation between the 1<sup>st</sup> and the 6<sup>th</sup> elements. Thus (2.75) can rule out sentences like (2.74).



Note the semantics-related component of the above problem. In this work we discuss issues concerned with syntax. At this point, we would like to emphasize the possible applicability of our approach to other fields or to modelling the interactions of various modules of the language.

As (1.2) shows, we refer to elements of patterns by the CAT values of the input AVSs they represent. However, it is also possible to use several attributes for representing input elements, not just the CAT values. Furthermore, such attributes need not be restricted to characterise only a single linguistic module. It is possible to integrate syntactic, semantic, phonological or other features simultaneously into patterns.

Pattern (2.76) – which is equivalent with (2.75) – demonstrates how mapping onto patterns can be determined by several attributes.



(2.75), and (2.76) also demonstrate that the construction of a pattern corresponding to the same collection of linguistic phenomena may not be uniquely determined by the linguistic phenomena alone. In the general case there are several ways to construct a pattern. (2.75) and

(2.76) consist of six, and eight atomic representational elements respectively, and they also differ in terms of agreement relations. The sequence of elements in (2.75) can be directly thought of as states of a deterministic finite-state automaton. The automaton implementation of (2.76) would imply non-determinism, since after processing the state corresponding to the third verb (the 6<sup>th</sup> element) there would be two transitions both labelled 'CAT=N'.

Thus it may be concluded that the definition of a pattern may be influenced by both theory and practice.

### **3. Conclusions**

In this paper we presented an approach toward modelling linguistic phenomena which is based on the observation that there is a wide range of agreement properties that can be established between linguistic elements. In particular, the agreement framework was applied to issues in syntax. The selection of topics was meant to be both representational and capable of demonstrating the possible usefulness of the method. We suspect that the agreement approach can be employed in a wider domain of phenomena in syntax, as well as possibly in other fields. The verification of the suspicion demands further research.

### **About the Author**

Mr Drienkó works within the theoretical linguistics programme at ELTE University, Budapest, Hungary.

Email: dri@axelero.hu

### **References**

Drienkó, L. (2004). Agreement Mapping System Approach to Language. *Journal of Language and Linguistics*. Vol. 3. No. 1. 38-61.