Case Syncretism in Arabic: A Nanosyntactic Approach

Asst. Lect.
Abbas Talib Alfelugi
The Islamic University, Najaf
abbas.talib@iunajaf.edu.iq

المطابقة في الحالات الاعرابية في اللغة العربية : مقاربة تركيبية جزيئية

المدرس المساعد
عباس طالب الفلوجي
الجامعة الإسلامية ـ النجف الأشراف

الملخص:-

تشير المطابقة بين الحالات العرابية إلى التواجد المشترك لحالتين اعرابيتين مدمجتين بلاحقة واحدة. تحاول الدراسة الحالية إلقاء الضوء على هذه الظاهرة من خلال وصف البيانات العربية التي لم تحظ باهتمام كبير في الأدبيات. تصرف الأسماء العربية لثلاث حالات: حالة الرفع، والنصب، و الجر. تُظهر هذه الحالات نمطين من المطابقة عرضي وغير عرضي. يُفترض أن السبب الأول ناتج عن عمليات صوتية بينما يكون الأخير عام.

النهج المعتمد (nanosyntax) يفترض أن المطابقة هو عام و منظم حيث يحكمها تسلسل هرمي محدد وشرط التقارب. طور Caha (2009) نموذجًا تحليليًا للحالة إلى سمات جزيئية ومكونات اصغر من اللاحقة. علاوة على ذلك، هناك اختلاف جوهري بين الحالة التركيبية والتحليلية. يشير الأول إلى الحالة الصرفية بينما يشير الأخير إلى استخدام حرف الجر للتعبير عن الحالات غير النحوية.

بينت الدراسة أن المطابقة في اللغة العربية نوعان: عرضي وغير عرضي. تحدث التوفيق العرضي في اللغة العربية بسبب الجوانب الصوتية حيث لا تظهر العلامة الاعربية على الأسماء المعتلة. بينما تحدث المطابقة الغير عرضية في صيغة جمع المثنى، والجمع المذكر، والجمع المؤنث. الأسماء المفردة لا تظهر أي نوع من المطابقة فيها.

الكلمات المفتاحية: المطابقة، اللغة العربية، التركيب الجزيئي.

Abstract:-

Case syncretism refers to the co-occurrence of two cases spelled out by one morpheme. The present study attempts to shed lights on this phenomenon by describing Arabic data which received little attention in the literature. Arabic nouns inflect for three cases: nominative, accusative and genitive. These cases show two patterns of syncretism: accidental and non-accidental. The former is assumed to be caused by phonological processes while the latter is universal.

The adopted approach (nanosyntax) assumes that syncretism is universal and systematic. It is governed by specific case sequence and adjacency condition. Caha (2009) developed a decompositional model which analyses case into atomic features and submorphemic Components. Furthermore, a fundamental difference is made between synthetic and analytic case. The former refers to morphological case while the latter refers the use of preposition to express un grammaticalised cases.

The study shows that Arabic exhibits two types of syncretism: accidental and non-accidental. Accidental syncretism occurs in Arabic due to phonological aspects. The defective nouns are not marked for case because they end with a vowel. Thus, they show of non-adjacent cases. Non-accidental syncretism occurs in dual, masculine plural, and feminine plural. Singular nouns do not show any type of syncretism.

Key Words: Case syncretism, Arabic, nanosyntax**.**

1. Introduction

Case syncretism refers to the combination of two or more distinct cases in one morphosyntactic category, i.e. single morpheme that realizes two or more cases. Case syncretism is a language specific phenomenon in which different types of noun classes are marked for case. Some of these noun classes may have syncretic cases or distinct cases. Consider the following declension:

	maxit (fighter, pl.)	maxit, (fighter, sg.)	alpha
nom	maxit-es	maxit-i-s	´alpha
acc	maxit-es	maxit-i-Ø	´alpha
gen	maxit-on	maxit-i-Ø	´alpha

Table (1) Syncretism in Modern Greek adopted from (Caha, 2009, pp. 7)

The above table illustrates that plural nouns in modern Greek show syncretism between nominative and accusative case while the genitive case is marked differently. In contrast, singular nouns show syncretism between accusative and genitive while nominative is marked differently. The last column represented by the word “alpha” shows total syncretism.

Case syncretism can be explained on the basis of distributional grounds (Baerman, 2009, pp. 220). For example, in Classical Armenia, the accusative case is either syncretic with nominative singular nouns or with locative plural nouns.

Syncretism in some examples might be superfluous, resulting from accidental homophony. For example, in Latvian language, -s spelled two cases the nominative and the genitive. The two cases are marked independently of each other. As a result, it is considered an instance of syncretism (Baerman, 2009, pp. 220).

Researchers such as (Baerman, 2009; and Caha, 2009) attempts to explain case syncretism systematically. Syncretism is not an accidental phenomenon and follows certain principles.

Syncretism is a universal phenomenon. Arabic has three case system in which syncretism is demonstrated between accusative and genitive in dual, masculine plural, feminine plural. While some defective nouns have special patterns. Arabic case syncretism has

received little if any attention in the literature. Thus, the presents study aims at investigating case syncretism in Arabic, describing syncretic declensions.

The theoretical framework is adopted from (Caha, 2009) in which case syncretism represents a systematic phenomenon that targets adjacent cases in the case sequence. According to him, there are two types of syncretism accidental and non-accidental. These will be explained in detailed 3.1 among other concepts.

Arabic nouns normally inflect for three cases: NOM, ACC and GEN. Case markers in Arabic are short vowel suffixes: -u for NOM, -a for ACC and -i for GEN but there are substantial exceptions to these markers (Ryding, 2005, pp. 183-204). The following declension paradigm exhibits Arabic case system.

This paper will try to look at case syncretism, attempting to define their atomic features. Along with that, it will try to establish the relation between one case and another. For this purpose, the containment hypothesis is adopted Caha (2009). The decompositional model suggested by Caha, in particular, helps further characterising and distinguishing patterns of syncretism. Consequently, the aim of this paper is to investigate the extent to which the adopted framework is applicable to Arabic case syncretism.

This paper is organised as follows. Section 2 presents a brief introduction to nanosyntax. Section 3 in devoted to case representation. This includes syncretism,decomposition and containment. Section 4 presents declension paradigms and patterns of syncretism are further investigated with reference to Superset Principle and Elsewhere Condition.

2. Nanosyntax

Nanosyntax is a novel approach to the architecture of grammar. It is a late insertion theory Starke (2009, 2011), Caha (2009), Pantcheva (2009, 2011), Fabragas (2009), Taraldsen (2012). It is developed from the cartographic framework of (Cinque (1999), Rizzi (1997). The cartographic research tries “to provide a detailed structural map of language”.

The basic assumption of late Insertion theory is that lexical items are inserted post-syntactically, after Merge (Chomsky, 1995) has

created syntactic structure with morphosyntactic features. “It is only after some steps of derivation that a constituent is large enough to correspond to a morpheme created” (Starke, 2011, pp. 4). This means that lexicon comes strictly after syntax and lexemes correspond to entire phrasal constituent.

Hence, Starke (2011) argued that morphemes cannot feed syntax because they are too big to build syntactic trees. The submorphemic features can be observed through syncretism. Fabragas (2009) defines syncretism as “a mismatch between syntactic structure and lexicon”.

Caha (2009, pp.5) defines it as “when two distinct cases have the same form”. This means that there is one “lexical item which can correspond to more than one syntactic representation”. Therefore, it consists of more than one feature. He further adds that syncretism is a “surface conflation of two different underlying morphosyntactic structures” (Caha, 2009, pp.6).

3.Case Representation

3.1 Syncretism

Syncretism as a phenomenon happens when two distinct cases have one form. Starke provides a restricted theory of syncretism:

1) A lexically stored tree matches a syntactic node if the lexically stored tree contains the node (Starke 2009).

In order to understand the internal organisation of syncretic morphemes, one must look at its beginning. The earliest work in this field started by looking at the English suffix –ed. This suffix has two readings, active and passive.

For example:

a- He folded the sheet. (Active)

b-The sheet was folded. (Passive)

Similarly, consider syncretism in Arabic in the following paradigm:

3) Syncretism in Arabic

Engineers (S.masc.pl)

NOM	muhandis-uuna
ACC	muhandis-iina
GEN	muhandis-iina

From the first sight, we can observe that ACC is syncretic with GEN. This type of syncretism is called non-accidental. It has been demonstrated that non-accidental syncretism is universal (Caha, 2009, pp. 292).

In order to restrict syncretism, Caha combines this view with universal case contiguity (UKC henceforth). This proposal will put syncretism in a systematic fashion.

4) UKC adopted from Caha (2009)

a- Non-accidental syncretism targets contiguous regions in sequence invariant across languages.

b- The case sequence: NOM- ACC- GEN- DAT- INS- COM

5) Syncretism in classical Arabic (adapted from Johnston 1996)

	Theif (sg)	Mecca (Dip)	Queen (S.fem. pl)	Judge (Def N)
NOM	sāriq-u-n	makkat-u	malik-āt-u	qādin
ACC	sāriq-a-n	makkat-a	malik-āt-i	qādiyan
GEN	sāriq-i-n	makkat-a	malik-āt-i	qādin

The three cases can be distinct in singular nouns (Thief). *Mecca* and Q*ueen* represent non-accidental syncretism. It seems that the defective noun *Judge* has violated UKC because it shows syncretism of non-adjacent cases. This type of syncretism is, in fact, caused by phonological processes. This means that it is not the reflex of grammar rather it is the reflex of phonology. Traditional Arabic grammatical theory developed a concept that all nouns are marked for case, but in some of them the case marker is *virtual* or *implied* (muqadar) rather than overt (zaahir) (Ryding 2005, pp. 187).

6) A modified UKC

1- Non-accidental syncretism targets contiguous regions in a sequence invariant across

languages.

2- Accidental syncretism targets non-contiguous regions

3- The case sequence: Nom- ACC- GEN- DAT- INS- COM

3.2 Decomposition

The gist of this section is that case can be decomposed into a set of separate features. To do this, we need to capture natural classes definable by syncretism. This will help us to understand and represent non-accidental and accidental syncretism.

3.2.1 Cross Classification

Jakobson model of cross classification is adopted here since it is simple and convenient. It is

composed of (x,y). (x,y) are further composed of (+x, -x) and (+y, -y). The schematic illustration below indicates the facts.

7) Cross classification (Jakobson, 1962 cited after Caha, 2009)

	+y	-y
+x	NOM	ACC
-x	GEN	DAT

The natural classes captured by such a decomposition are given in (8).

8) a- {+x}: {NOM, ACC}

b- {-x}: {GEN, DAT}

c- {+y}: {NOM, GEN}

d- {-y}: {ACC, DAT}

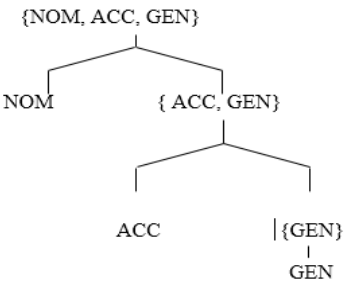
e- {∅}: {NOM, ACC, GEN, DAT}

Firstly, there is no linear ordering between the above mentioned cases. Moreover, the system allows any of the vertical and horizontal cases to syncretised. Furthermore, the system does not work in languages with three or five cases.

3.2.2 Sub-classification

An alternative system proposed by Johnston (1996). Johnston maintains that case can be decomposed into features. Then, the set of cases is sub-classified by features. Sub-classification starts from n (n refers to number) categories of cases. It sub-classified them into individual cases. The cases branch off from the tree one by one.

9) Sub-classification decomposition (adopted from Caha (2009))



Advantages:

a- Each set of cases is sub-classified into component parts rather than two or three cases.

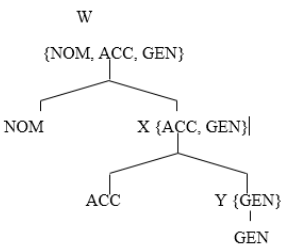
b- one case is considered at a time rather than more than one.

c- The cases branch off at the non-terminal nodes in a universal order.

3.2.3 Cumulative Sub-classification

In sub-classification there are individual cases at terminal nodes and a set of cases at the non-terminal nodes. The proposal is that each set of cases can be characterised by a unique feature.

10) Cumulative sub-classification (adapted from Caha, 2009)



11) Cumulative classification

a- NOM = W

b- ACC = W, X

c- GEN = W, X, Y

What happens is the following:

a- The set of cases {NOM, ACC, GEN} are characterised by the feature W.

b- The set of cases on the non-terminal nodes are partition (X, Y, Z).

c- Once an individual case is taken from the set, it does not belong to them anymore.

d- The system captures natural classes of syncretism.

However, this system is weak since it cannot capture syncretism of non-contiguous cases. For instance, syncretism of Nom- Gen in Arabic. But If this system is supplied with the Elsewhere Condition, then a syncretism between NOM and GEN to the exclusion of ACC (as for the decomposition in (11) is possible, since Phon B (13b) is a proper subset of Phon A (13a). We will divide Cases into two groups, Phon A and Phon B. Phon A represents contiguous syncretism and Phon B represents the non-contiguous one.

12) Elsewhere Condition

In case two rules, R1 and R2, can apply in an environment E, R1 takes precedence over R2 if it applies in a proper subset of environments compared to R2 (Caha, 2009).

13) a- Phon A {NOM, ACC, GEN}

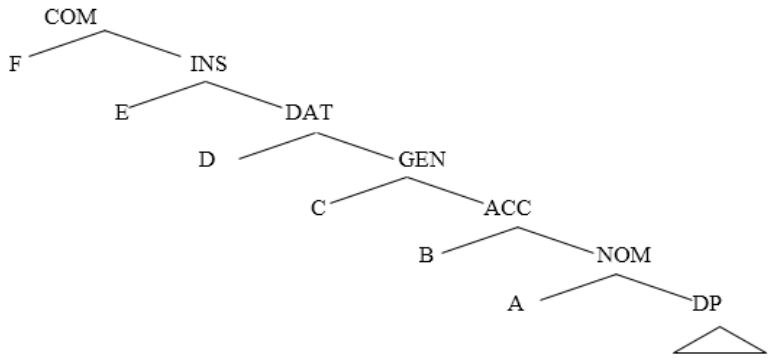
b- Phon B {GEN}

3.3 Containment

The preceding section exhibits the fact that case can be decomposed into features. The proposal of this section is that atomic features combine in the same way phrases and sentences combine, the operation of Merge (Chomsky 1995). The schematic illustration in (14) clarifies the operation.

The tree encodes that NOM is built by the operation of Merge between DP and the feature A. Similarly, ACC is built on the top of Nom by merging the feature B with A. GEN is built on the top of ACC by adding the feature C, and so on. The consequence of this operation is that by omitting the feature C, ACC will appear and by omitting the feature B, NOM will appear. This means that GEN contains ACC and ACC contains NOM by transitivity.

14) Containment (adopted from Caha (2009))



15) Universal case containment

a- In the case sequence, the marking of cases on the left can morphologically contain cases on the right, but not the other way round.

b- The case sequence: NOM- ACC- GEN- DAT- INS- COM (Caha, 2009)

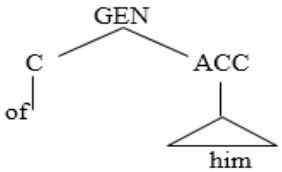
3.3.1 Analytic Case

After proposing case containment, another proposal must take place. That is to say how case interacts with NP movement. Wherever NP movement stops in the hierarchy (the case sequence), all the cases which are lower than NP movement will be case suffix and all cases which are above NP movement will have the structure P + NP + K (Blake, 2001). To see how does that work consider (16).

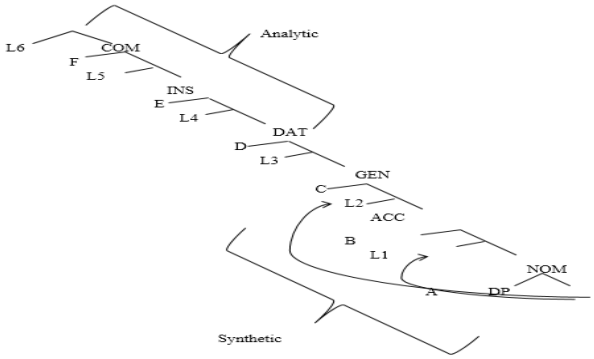
Consider some examples:

a- English has two cases, NOM and ACC. The structure of GEN is given in (17).

17) this is a picture <u>of him</u>



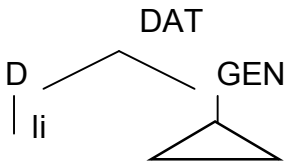
16) Analytic vs synthetic case



b- Arabic has three cases, NOM, ACC and GEN. The structure of of DAT is given in (18).

18) ʾaʿTay-tu l-kitaab-a li-muHammad-in

'I gave the book to the girl'

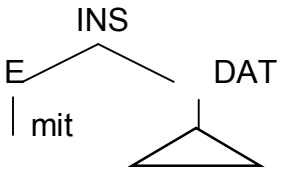


Muhammad-in Caha (2010)

c- German has four cases, NOM, ACC, GEN and DAT. The structure of INS is given in (19).

19) peter hat die suppe <u>mit einem Löffel</u> gegessen

'Peter has eaten the sope with a spoon'

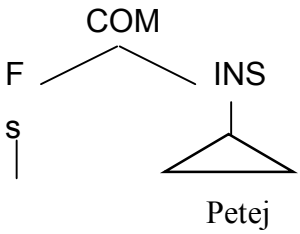


einem Löffel Caha (2010)

d- Russian has six cases, NOM, ACC, GEN, PREP, DAT and Ins. The structure of Com is given in (20)

20) Ex: Anna s Petej napisali pis'mo

'Anna and peter wrote a letter.'



McNally (1993)

The table below summarises the facts:

21) The structure of the Analytic case (adopted from Caha, 2009)

Language	Case	Expression
English	GEN	of+ ACC
Arabic	DAT	li+ GEN
German	INS	mit+ DAT
Russian	COM	s+ INS

The components of this section are:

a- Individual cases are built from atomic features by Merge.

b- The features are ordered in a universal functional sequence.

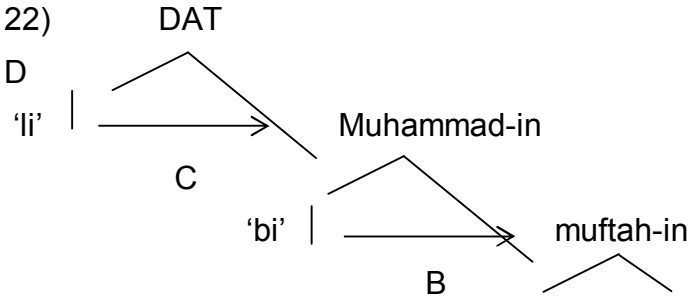
c- Wherever NP movement stops, all cases which are lower than NP movement will be case suffix and all cases which are above it will have the structure P + NP + K.

e- Case affix turns to be case suffix only as a result of NP cyclic movement, since NP cyclic movement targets positions between case features.

3.4 Arabic preposition

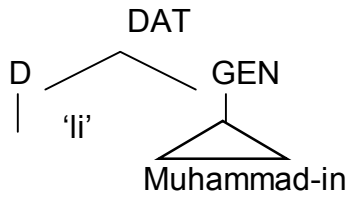
The preposition 'li' in Arabic is attached to GEN to form DAT. Similarly, the preposition 'bi' is attached to ACC to form GEN. The consequence of this operation is that DAT contains GEN, and GEN contains ACC by transitivity.

22)



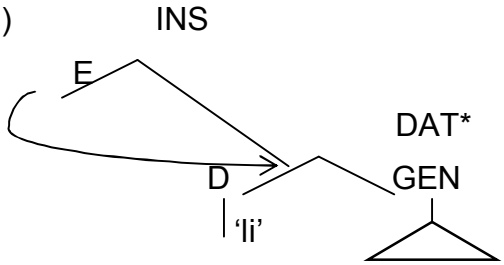
After clarifying the basic assumption of this section, NP movement have the structure P + NP + K. This happens because there is no lexical entry which contains the root node of the cases which are above NP movement. Similarly, in Arabic there is no root node which contains DAT or INS. When we add the feature D to form a DAT, the preposition 'li' is first inserted under the terminal D, consider (23).

23)



Another preposition is instrumental 'bi'. The instrumental preposition 'bi' spells out the instrumental case. However, it is not attached to DAT. Instead, it is attached to GEN. For instance, the DAT PP in the previous example was *li Muhammad-in*, if we added the instrumental 'bi', the structure will be (*bi*)* *li muftah-in*.

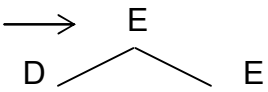
24)



Muhammad-in Caha (2010)

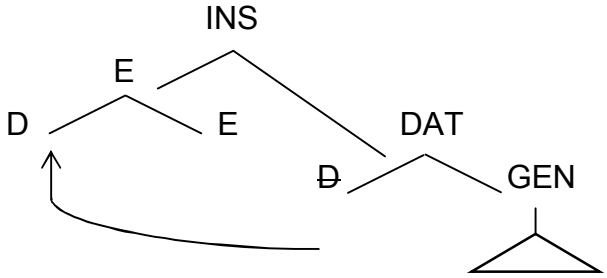
In this way, prepositions will pile up on GEN. Besides, 'bi' cannot be inserted under E alone. It must spell out a constituent with both E and D.

25) 'bi' ⟶



The new structure of INS is given in (26)

26)



Muhammad-in Caha (2010)

In this way, prepositions will not pile up on GEN. Either the feature D will be attached to Gen or the feature E. The conclusion is that the lack of NP movement (the absence of lexical entry) for a given non-terminal node causes the switch of case marking from suffixal to prepositional marking.

4.Declension Paradigms

Declension paradigm is a useful device to show case markers, number and class. This section will further investigate case syncretism, case and number via declension paradigms.

4.1 Case Syncretism

A table is used here to show the predication of case syncretism in Arabic. The prediction of case syncretism in Arabic is given in (27) (Adapted from Caha (2009)).

27)

NOM	ACC	GEN

The paradigm predicates the following:

a- NOM ⟶ ACC

b- ACC ⟶ GEN

c- NOM ⟶ GEN

d- NOM ⟶ ACC ⟶ GEN

The attested types of syncretism in Arabic are summarised in (28)

28) A declension paradigm of case syncretism in Arabic.

Type	NP	NOM	ACC	GEN
Sg	House	bayt-un	bayt-an	bayt-in
S.masc.pl	Engineers	muhandis-uuna	muhandis-iina	muhandis-iina
Def N	Judge	qaaD-in	qaaDiy-an	qaaD-in
Inv N	Complaint	shakwaa	shakwaa	shakwaa

It is observed from (28) that syncretism of NOM – ACC is unattested. Another important fact is that case syncretism in Arabic obeys the Law of Adjacency except the syncretism of the Def N qaaD-in.

4.2 Case and Number

The main idea suggested here is that case, class and number are inseparable entities stored inside a single morpheme. This is clearly manifested in S. Masc.pl and dual.

29) Case and number in Arabic

Type	Phrase	Case marker	Class	NumP

S.masc.pl	-uuna ‘ون’	و NOM	Triptote	ون
	-iina ‘ين’	ي ACC+GEN	Triptote	ين
Daul	-aan ‘ان’	ا NOM	Triptote	ان
	-iina ‘ين’	ي ACC+GEN	Triptote	ين
S.Fem.pl	-at ‘ات’	Damma NOM	Triptote	ات
		Kasra ACC+GEN	Triptote	ات

4.3 Analysing Case Markers

According to the nanosyntactic view, “morphemes are phrases” (Caha, 2009, pp. 218). Under this view, case markers are analysed via hierarchical structure. To achieve this, phrasal movement is proposed by Cinque (2005).

30) Cinque (2005)

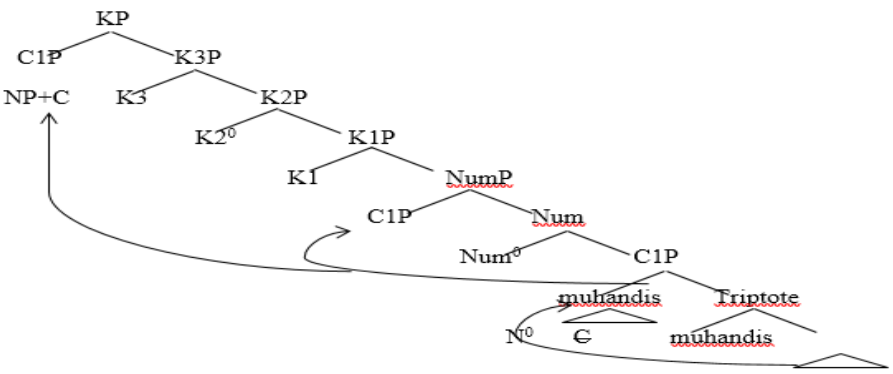
a- Move the constituent which contains the NP only.

b- Movement is leftward.

According to (30), the constituent which contains the NP must be moved. However, it is difficult to move the whole constituent leftward, since in some cases there is a packaging morpheme attached to the NP which contains case and number. And if it is moved, there will be no variation in KP because the packaging morpheme contains its own case.

The problem can be solved by proposing that NP merges with C (case), forming a new constituent that is C1P. This constituent will move across KP (case phrase), which is, in return, will lead to variation in K. To see how does that work, consider (31).

31) Engineers muhandis-uuna/ muhandis-iian

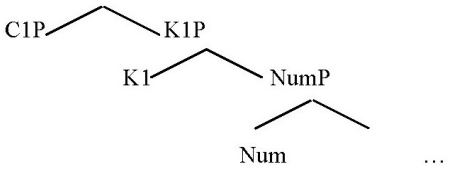


Form the diagram above, it is noticed how the features in the syntactic tree are attached to the noun in a systematic sequence. Firstly, NP is inserted at the bottom of the tree. Secondly, NP + C formed a constituent under the terminal C1P. Thirdly, C1P moved higher than Num. This movement turned Num into a suffix. Finally, C1P moved higher than K. This also turned K into a suffix.

4.4 The order of Cases

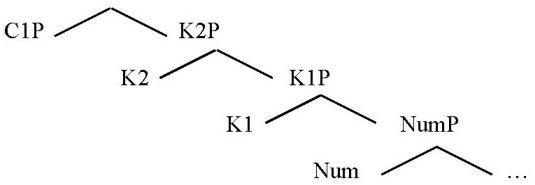
In the preceding section, it is shown that the element which controls the order of morphemes is KP. In unmarked cases, insertion targets the whole constituent. For instance, the insertion of the suffix –iina in S.masc.pl targets two constituents, ACC and GEN. On the other hand, the suffix –uuna targets NOM only.

32)



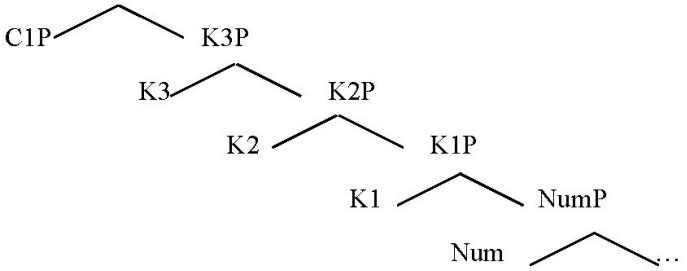
By the addition of another case, the number of case layers' increases.

33)



Again, the number of case layers increases as we go down the tree. An important fact to be notice is that as the number of case layers' increases, the constituent has to be bigger and bigger to fuse case and number.

34)



To conclude, the system works according to two facts:

a- The number of features increases as we go down the tree. This is followed by the addition

of another case layer to the hierarchy.

b- The Superset Principle.

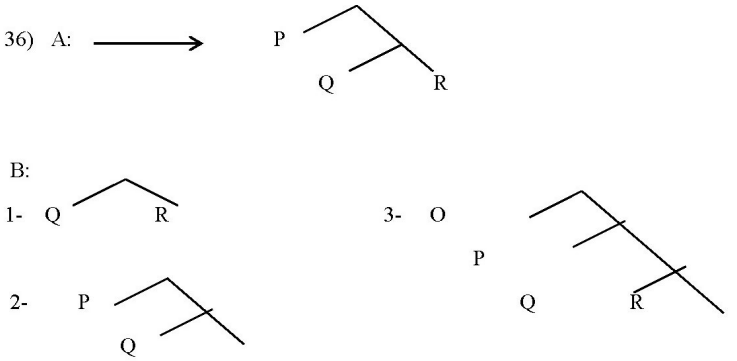
4.5 The Superset Principle

Superset Principle is one of the basic mechanism which accommodates the nanosyntactic view. It is proposed as analogy to subset principle Hall & Marantz (1993). The Superset Principle is the interface spell-out condition which allows a vocabulary item to spell out a certain chunk of the syntactic tree if the lexical entry of that item contains all or a superset of the nodes/features present in the syntax. This means that the spell-out procedure can ignore lexical features, but cannot ignore syntactic features, i.e., all syntactic features must be spelled-out. The Superset principle enables vocabulary items to target a non-terminal node. Thus several syntactic heads can be targeted and spelled out by a single morpheme.

35) Superset Principle

A phonological exponent is inserted into a node if its lexical entry has a (sub-)constituent that is identical to the node (ignoring traces) (Starke, 2005).

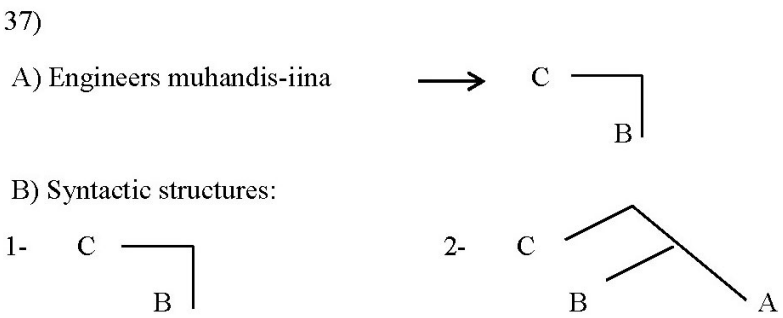
Consider that (A) is a lexical entry and (B) syntactice structures.



Assume that (A) is a lexical entry, a spell out rule paring syntactic structure. According to superset principle (35), it is allowed to spell out any structure which identical to the lexical entry, i.e. (B-1) or (B-2). In other words, syntactic structures (B-1) and (B-2) can be both spelled out using the same lexical entry. However, (B-3) cannot be spelled out by the same lexical entry (A), since the lexical entry (A) does not match (B-3).

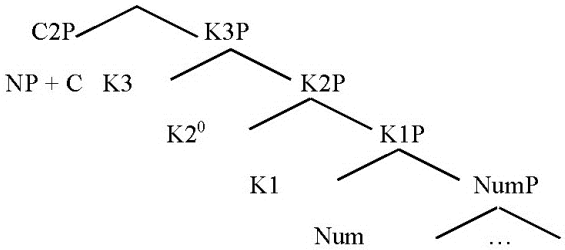
Let us combine this view with data from Arabic. The biggest lexical item in Arabic can spell out two features, B and C. This means that the biggest lexical item in Arabic can spell out the syntactic structure in (37).

It is obvious that the lexical (A) corresponds to (B-1), Since (B-2) has a subset which is not identical to the lexical entry. Thus, it cannot be spelled out by the same lexical entry. The problem is that there is no lexical entry which can spell out (B-2). This can be solved by using separate insertion. Separate insertion is similar to Exhaustive lexicalisation Principle



38) Exhaustive Lexicalization Principle: Every syntactic feature must be lexicalized by a lexical item, even if this item is phonologically null (Fabregas., 2007).

39) Engineers muhandis-uuna/muhandis-iina



It is clear the superset principle is at work in Arabic. The phonological exponent –iina can spell out to features, A and B. However, it is not big enough to spell out the feature A. This complication leads to separate insertion.

4.6 Elsewhere condition

Since Superset Principle is not restrictive enough, another principle is incorporated to restrict the superset principle. This is because it is not the only relevant principle for matching and if it is so, then K1P can also a good candidate for insertion in K2P and K3P, since K1P is lexically stored in K2P and K3P. Consequently, it is needed to restrict the insertion to K2P and K3P only. This is achieved by the Elsewhere Condition. The Elsewhere Condition ensures that at each cyclic node “the most specific wins” (Starke, 2009). This principle is also called “minimise junk” (Starke, 2009).

40) Elsewhere Condition

In case two rules, R1 and R2, can apply in an environment E, R1 takes precedence over R2 if

it applies in a proper subset of environments compared to R2 (Caha, 2009).

The three case suffixes in Arabic have lexically stored trees. The way these case suffixes are stored are in lexicon is given in (41).

41) Root N* muthaqqaf

NOM muthaqqaf-uuna

ACC muthaqqaf-iina

GEN muthaqqaf-iina

This is an immediate consequence that nanosyntax is a late insertion modal and that lexicon is post-syntactic. That is to say, only after syntactic Merge takes place in syntax there can be lexical insertion (De Clercq, 2013).

42) Lexical items (LI henceforth)

a- < / muthaqqaf/, [N*]>

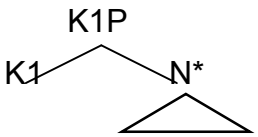
b- < / muthaqqaf-uuna/, [K1]] NOM>

c- < / muthaqqaf-iina/, [K3[K2]]] GEN>

The lexicon in Arabic has an entry for the root N*, (42a). It also has an entry for two case suffixes, -uuna (42b) and -iina (42c). Due to the fact that ACC and GEN are syncretic there is only one LI for them, (42c).

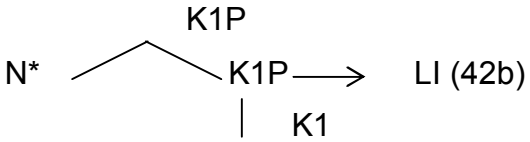
It follows then that we need to spell out the structures given in (42) in order to restricts Superset principle, which in return will restrict insertion to K2P and K3P (Caha, 2009, pp. 25).

43) The spell out of (42a).



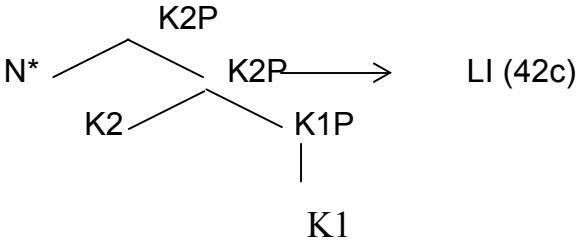
When the first feature of case spine, K1, is merged, generating the syntactic tree in (44). Then the syntactic structure is checked against lexicon. Here, at this stage of derivation, we can spell out K1P via spell out driven movement. Spell out driven movement is significant in two crucial points. Firstly, when N* merges with K1 to create K1P, it immediately evacuates its position and moves to specifier position slightly above the newly merged head. Secondly, after each successful merge the previous merge is overridden.

44)



After the movement is applied, the new structure is checked against lexicon. Due to the bottom-up cyclist of spell out driven movement, the structure in (42c) is spelled out by starting with K2P. Although the structure in (42c) consists of two lexically stored trees and these trees are syncretic. Thus, insertion will target K2P only. And K1P is overridden.

45)



Again the new structure is checked against lexicon. N* merges with K2 and moves to specifier position. Besides, K1P is overridden.

Now N* will merge for the last times due to the fact that Arabic has only three cases and this proves that case interacts with NP movement.

46)

In the same way, the new structure is checked against lexicon. N* merges with K3 and moves to specifier position. Insertion targets K3P only. Besides, K2P is overridden.

Having the Superset Principle and Elsewhere Condition in place, LI (42a) is the winning competitor. This means that -uuna is the only case marker that can be applied in a different environment. Consequently, the case suffix -iina is inserted in K2P yielding to ACC firstly and GEN secondly. As a result, GEN is derived from ACC and not the other way round.

5. Conclusion

This paper examined case in Arabic. Firstly, Arabic marks two types of syncretism. The first one is non-accidental and it is governed

by the *Law of Adjacency*. The second one is accidental and it is caused by phonological processes. This means it is not the reflex of grammar. Secondly, the adopted framework proposed by Caha (2009) predicates and captures case syncretism in Arabic in a systematic order. Thirdly, the analytic case in Arabic has an identical structure to those of English, German and Russian.

The second half of the paper was devoted to declension paradigms. Through them, case marker and number are explored. It has been demonstrated that when case grew bigger, it has to fuse case and number in one morpheme. Next, Superset Principle is at work in Arabic. The conclusion of superset principle is that ACC\GEN are in superset relationship.

References:-

- Blake, B. J. (2001). Case. Cambridge University Press.
- Caha, P. (2008). The case hierarchy as functional sequence. In Scales (pp. 247–267). University of Leipzig.
- Caha, P. (2009). The Nanosyntax of Case [Ph.D. Thesis].
- Caha, P. (2010). The parameters of case marking and spell out driven movement. Linguistic Variation Yearbook 2010, 10, 32–77. https://doi.org/10.1075/livy.10.02cah
- Chomsky, N. (1995). Bare phrase structure. In In Government and Binding Theory and the Minimalist Program: Principles and Parameters in Syntactic Theory (pp. 383–439). Blackwell.
- Cinque, G. (2005). Deriving Greenberg's Universal 20 and Its Exceptions. Linguistic Inquiry, 36(3), 315–332. https://doi.org/10.1162/0024389054396917
- De Clercq, K. (2013). A Unified Syntax of Negation [Ph.D, Thesis].
- Fábregas, A. (2007). The Exhaustive Lexicalisation Principle. Nordlyd, 34(2). https://doi.org/10.7557/12.110
- Fábregas, A. (2009). An argument for phrasal spell-out: Indefinites and interrogatives in Spanish. Nordlyd, 36(1), pp. https://doi.org/10.7557/12.218

- Guglielmo Cinque. (1999). Adverbs and functional heads: a cross-linguistic perspective. Oxford University Press.
- Halle, M., & Marantz, A. (1993). Distributed morphology and the pieces of inflection. In The view from building 20 (pp. 111–176). MIT Press.
- Johnston, J. (1996). Systematic homonymy and the structure of morphological categories [Phd, Thesis].
- Mcnally, L. (1993). Comitative coordination: A case study in group formation. Natural Language & Linguistic Theory, 11(2), 347–379. https://doi.org/10.1007/bf00992917
- Pantcheva, M. (2011). Decomposing path: The nanosyntax of directional expressions [Phd, Thesis].
- Rizzi, L. (1997). The fine structure of the left periphery. In Elements of grammar. Handbook in generative syntax (pp. 281–337). Dordrecht.
- Ryding, K. C. (2005). A reference grammar of modern standard Arabic. Cambridge University Press.
- Starke, M. (2009). Nanosyntax: A short primer to a new approach to language. In Nordlyd 36: Special issue on Nanosyntax (pp. 1–6). University of Tromsø.
- Starke, M. (2011). Towards elegant parameters: language variation reduces to the size of lexically stored trees. University of Tromsø.