# Chain Formation II

Robin Clark
Département de linguistique - LATL
Université de Genève

Décembre 1991

In this document, I will consider a generalization of the $\overline{A}$-chain algorithm I developed earlier (Clark, 1991); the core algorithm will be extended to $A$-chains. Superficially, $A$-chains would seem to differ drastically from $\overline{A}$-chains insofar as the former are subject to an apparently tighter set of conditions, and appear more local, than the latter. Consider, for example, the following contrasts:

(1)a. John$_i$ seems [$_{TP}$ $t_i$ to be sick]

b. *John$_i$ seems [$_{CP}$ ($t_i$) [$_{TP}$ $t_i$ is sick]]

(2)a. The universe$_i$ was thought [$_{TP}$ $t_i$ to be geocentric]

b. *The universe$_i$ was thought [$_{CP}$ ($t_i$) [$_{TP}$ $t_i$ is geocentric]]

(3) *John$_i$ seems that it is likely [$t_i$ to be late]

The above examples, raising in (1), passive in (2) and super-raising in (3), show that simple $A$-movement is subject to the Tensed S Condition (the *TSC*). $\overline{A}$-movement applies rather more freely:

(4)a. How sick did John$_i$ seem [$_{TP}$ $t_i$ to be $t$]

b. How sick did it seem [$_{CP}$ that [$_{TP}$ John was $t$]]

(5)a. What structure was the universe$_i$ thought [$_{TP}$ $t_i$ to have $t$]

b. What structure was it thought [$_{CP}$ that [$_{TP}$ the universe had $t$]]

(6) Who$_i$ does it seem that it is likely that John admires $t_i$

The examples in (4), (5) and (6) show that $\overline{A}$-movement is not subject to the *TSC*. Notice, however, that both $A$- and $\overline{A}$-movement share many properties. For example, neither $A$- nor $\overline{A}$-movement may raise an element out of an adjunct:

(7)a. *Who$_i$ did it seem that John left to irritate $t_i$

b. *Mary$_i$ seemed that John left [$t_i$ to be irritated]

I will argue here that the basic algorithm for positing gaps and forming chains can be generalized across $A$- and $\overline{A}$-movement. The differences between the two can be captured either by sending a special flag to the core chain-formation algorithm or by applying different output conditions to $A$- and $\overline{A}$-chains.

Recall that an algorithm for constructing well-formed chains must take into account the following requirements:

1. Conditions for initiating an $A/\overline{A}$-chain, including, by definition, the search for its tail (gap-positing).
2. The information contained in the chain.
3. Conditions for terminating the chain by positing a tail.
4. Conditions for licensing intermediate links in the chain.
5. Locality and other structural conditions on chain links.
6. Data structures for representing and performing calculations on chains.

As with $\overline{A}$-chains, I will assume that $A$-chains can be represented using a simple list and stack structure, as proposed in Clark (1991). Note, however, that the stack of $\overline{A}$-chains and the stack of $A$-chains must be kept distinct. Although crossing effects have been noted between distinct $\overline{A}$-chains, no such effects have been noted between $\overline{A}$-chains and $A$-chains:

(8)a. *Which sonata do you wonder which violin to play on?

b. ?Which violin do you wonder which sonata to play on?

c. To which teacher did John seem to be a talented student?

In addition to a weak subjacency violation, example (8a) involves crossing between two *wh*-phrases, *which sonata* and *which violin.* Thus, (8b), where the scope of the two *wh*-phrases is reversed, is slightly more acceptable than (8a). Example (8c) shows that there is no crossing effect between $A$-chains and $\overline{A}$-chains; although the $A$-chain headed by *John* crosses the $\overline{A}$-chain headed by *to which teacher*, (8c) is perfectly acceptable.

A related question is whether or not two (or more) $A$-chains can show crossing effects, thus providing additional support for the notion that the memory structure for $A$-chains is a stack. Note, however, that $A$-movement relates a position that has a $\theta$-role but no Case to a position that has Case but no $\theta$-role. We might argue that due to independent properties of the Case and $\theta$ components, the stack for $A$-chains will contain at most one chain with length greater that 1 at any given moment. If this is correct, then there will be no crossing effects for $A$-movement at all and the stack required to store incomplete $A$-chains is trivial, although for reasons that have little to do with processing.[1]

The basic definition of *chain* given in Clark (1991) can be generalized across $A$- and $\overline{A}$-positions as follows:

[1]One should consider, further, the question of clitic chains and whether it is possible to get crossing effects between clitics or between clitic chains and $A$-chains.

(9) ($\text{pos}_1$, ..., $\text{pos}_n$) is an $X$-chain, $X \in \{A, \overline{A}\}$, in phrase marker $P$ if:

(i) $\text{pos}_1$ is an $X$-position in $P$.
(ii) $\text{pos}_n$ is an $A$-position in $P$.
(iii) If $i \neq n$, the $\text{pos}_i$ is an $X$-position in $P$.
(iv) If $i < j$, then $\text{pos}_i$ c-commands $\text{pos}_j$ in $P$.
(v) $\text{pos}_{i+1}$ is subjacent to $\text{pos}_i$ in $P$.

The above condition on chains requires that the tail of all chains be an $A$-position. Furthermore, all intermediate links in the chain must be $A$-positions if the head is an $A$-position or $\overline{A}$-positions if the head is an $\overline{A}$-position. Thus, example (10a) consists of two chains, an $A$-chain and an $\overline{A}$-chain; the $A$-chain is shown in (10c) and the $\overline{A}$-chain is shown in (10b):

(10)a. $[_{CP_0}$ who$_i$ did $[_{IP_0}$ John say $[_{CP_1}$ $t_i$ $[_{IP_1}$ $t_i'$ seems $[_{IP_2}$ $t_i''$ to have been $[_{VP}$ kidnapped $t_i''']]]]]]$

b. $\overline{A}$-chain: (who$_i$, $t_i$, $t_i'$)

c. $A$-chain: ($t_i'$, $t_i''$, $t_i'''$)

The chain in (10c) is subject to slightly different conditions than the chain in (10b). Nevertheless, the same basic algorithm forms both chains, its output being subject to different conditions depending on whether the head of the chain is in an $A$-position or in an $\overline{A}$-position.

Let us turn to the problem of terminating $A$-chains. The basic cases are shown in (11) and can all be shown using passive structures:[2]

(11)a. John$_i$ was $[_{VP}$ seen $t_i]$

b. John$_i$ was $[_{VP}$ believed $[_{TP}$ $t_i$ to be an idiot$]]$

c. John$_i$ was $[_{VP}$ considered $[_{FP}$ $t_i$ an idiot$]]$

That is, $A$-chains terminate either in a thematic complement position or in the position of the thematic subject of a predication. Graphically, the tail of the chain can occur in either of the following configurations:

(12)a. ... $[_{\overline{X}}$ $X^0$ $t]$ ...

b. ... $[_{\overline{X}}$ $X^0$ $[_{YP}$ $t$ $\overline{Y}]]$ ...

These cases are slightly more restricted than the cases associated with $\overline{A}$-chains, reproduced here as (13).[3] Notice that condition (a) of (13) is overly general (at first glance) since it permits $\overline{A}$-chains to terminate in the Spec of CP; the general condition of full interpretation (Chomsky, 1986) will filter out such cases since the operator will fail to be interpretable. Thus, (a) can be left in its most general form:

[2] Raising cases are isomorphic to (11b)

[3] We should note that $\overline{A}$-chains can terminate in the Spec position of a [-tense] TP under certain conditions; infinitival relatives like *the man to fix the sink* provide a good example of this. This runs against the long-held and much cherished dogma that syntactic variables must always bear Case. See Clark's (1990) treatment of control for an analysis which collapses PRO with syntactic variables. I will leave these cases aside for the moment.

(13) Terminate $\overline{A}$-chain formation by positing a gap in:
  a. the Spec position of a YP (where the phrase is [+tense] if YP = TP); or
  b. sister position of an $X^0$ (if $X^0$ can take the head of the chain as a complement); or
  c. an adjoined position.

The ending conditions for $A$-chains are as follows:

(14) Terminate $A$-chain formation by positing a gap in:
  a. the Spec position of a YP (where the phrase is [-tense] if YP = TP); or
  b. sister position of an $X^0$ (if $X^0$ can take the head of the chain as a complement).

Notice that (13) and (14) differ only on a few points. $\overline{A}$-chains may terminate in an adjoined position while $A$-chains cannot; $\overline{A}$-chains may terminate in the Spec position of a tensed clause while $A$-chains terminate in the Spec position of an untensed element. Conceptually, one could collapse the termination conditions in (13) and (14) into a single algorithm which would take two arguments: a displaced element and its chain-type where the latter is either $A$ or $\overline{A}$:

(15) Terminate $X$-chain formation, where $X \in \{A, \overline{A}\}$, by positing a gap in:
  a. the Spec position of a YP (where if $Y$ = [+tense] then $X = \overline{A}$); or
  b. sister position of an $X^0$ (if $X^0$ can take the head of the chain as a complement); or
  c. an adjoined position if $X = \overline{A}$.

A further general output condition on $A$-chains is stated in (16):

(16) The tail of an $A$-chain must be in a $\theta$-position.

As noted above, $\overline{A}$-chains can terminate in $\overline{\theta}$-positions (positions which head an $A$-chain of length greater than 1).

The core part of the chain formation algorithm must specify how intermediate links in the chain are formed. Recall that, for $\overline{A}$-chains, intermediate links were formed by checking the Spec of each maximal projection, excluding TP and VP, between the head of the chain and its tail. If the Spec was empty, then chain formation could continue; otherwise, the chain either terminated by positing a gap or the chain was ill-formed. In addition, the newly projected category had to be a sister to a head ($X^0$); this prevented violations of the adjunct island condition and, in essence, recapitulated conditions on $\theta$-government coded in the definition of barrierhood. We can now generalize, and make more natural, these conditions:

(17)a. $\overline{A}$ Case: If the Spec of a newly projected category is a potential $\overline{A}$-Spec, then that position must be empty.

  b. $A$ Case: If the Spec of a newly projected category is a potential $A$-Spec (that is the category is DP, PP, AP, VP or a [-tense] TP), then that position must be empty.

In fact, these conditions replicate Rizzi's (1990) relativized minimality. The intuition is an $X$-chain is well-formed just in case each $X$-position which intervenes between the head and the tail contains an intermediate trace (an additional link in the chain).

The chain formation algorithm should, then, take place in the following steps: (i) upon encountering a potential argument, the algorithm is invoked; (ii) the character of the position occupied by the potential argument in (i) determines whether the algorithm is invoked using the flag to create an $A$-chain or the flag to create an $\overline{A}$-chain; (iii) the CHECK-SPEC component is invoked for each new category projected subject to the sisterhood convention noted above; if the new category has a non-empty $X$-Spec and the algorithm is building an $X$-chain, then a violation is reported; (iv) TERMINATE-CHAIN is invoked; if successful it ends chain formation and returns the chain; otherwise. step (iii) is repeated.

Let us consider a concrete example of $A$-chain formation:

(18) John is considered an idiot.

The above involves a passive which takes a small clause complement. Thus, $A$-chain formation should be initiated by the presence of a DP in the Spec of TP[4] and should terminate in the Spec of the functional category which serves as the complement to *consider*. I assume that *John* will be encountered first, triggering the projection of a DP. The auxiliary verb *was* triggers the formation of a TP, allowing attachment of *John* as the Spec TP and initiating $A$-chain formation since the Spec TP is an $A$-position. Next the verb *considered* is encountered. This allows for projection of a VP. The VP itself is passed to the CHECK-SPEC component of the chain formation algorithm. Since an $A$-chain is being formed, the Spec of the VP (an $A$-position) is checked; being empty the CHECK-SPEC component is satisfied and a trace is posited in the Spec of VP.[5]

Since *considered*, when selected by a form of *be*, is a passive participle it does not assign a $\theta$-role to its Spec. Thus, TERMINATE-CHAIN fails on the Spec-VP, since this position is a $\overline{\theta}$-position. The parser continues, then, without terminating the $A$-chain. The verb *consider*, in its relevant reading, subcategorizes for a small clause DP.[6] This category is confirmed by the presence of *an* in the input string. Again, its Spec is checked and a trace is dropped. Since the Spec of a functional category must be a $\overline{\theta}$-position, TERMINATE-CHAIN again fails. Finally, the parser encounters the NP *idiot*; CHECK-SPEC is satisfied since the Spec position does not contain any phonologically overt material. Furthermore, TERMINATE-CHAIN is satisfied since the Spec of the NP is a $\theta$-position. Hence, $A$-chain formation terminates with a well-formed chain connecting the Spec of TP with the Spec of a predicate nominal.

Consider next a case of super-raising as in (19):

(19) *John$_i$ seems that [it is like [$t_i$ to be late]]

The crucial point comes when the embedded TP is encountered. Notice that its Spec is not null, but occupied by the expletive *it*. CHECK-SPEC will therefore fail which effectively blocks the chain formation algorithm.

Next, consider a violation of the ECP (and condition A of the binding theory):

---

[4] Given the VP internal subject hypothesis, this is always true.

[5] As we will see below, positing traces will engender a certain amount of parallelism since it is often unclear whether a particular trace is the tail of the chain or merely an intermediate link in a longer chain.

[6] My recent work on small clauses indicates that the structure is more complex than this implies. The added complexity changes none of the essentials of the argument, however. I will therefore leave aside any additional structure.

(20) *John$_i$ seems [$_{CP}$ ($t_i$) [$_{TP}$ $t_i$ is sick]]

Since the embedded TP is [+tense], *A*-chain formation again fails. This occurs independently of whether or not there is an embedded CP present. Thus, the present algorithm effectively mimics the *Barriers* condition on antecedent government.

Consider, next, another case of super-raising, this time out of an adjunct:

(21) *John$_i$ seems that Mary left [$t_i$ to annoy Bill]

Recall that chain formation only continues along lines of $\theta$-government; that is, a chain continues into a new phrase only if that phrase is attached as a sister to an $X^0$. Since the purposive *to annoy Bill* cannot be attached as a sister to an $X^0$ but is, rather, adjoined to an $\overline{X}$, the chain formation algorithm cannot consider its internal structure. The chain formation algorithm will therefore never terminate on (21).

In this technical note, I have considered a simple algorithm for *A*-chain formation. This algorithm is sound in the sense that it reproduces the majors requirements of antecedent government and concordance with condition A of the binding theory. Furthermore, it parallels (and can be collapsed with) $\overline{A}$-chain formation. Thus, it provides a general and simple method of chain formation.