Intro to Syntax, PART FIVE

Omer Preminger, MIT

EGG 2009 / COST-A33, Poznań

Disjoint Reference Effects
What we’ve seen already ....................................................... 3
Starting simple ................................................................. 5
Precedence, or not? .............................................................. 7
c-Command ........................................................................ 12
Non-Coreference Rule .......................................................... 13
Consequences ........................................................................ 19

Binding Theory
A new perspective: Constraints on the distribution of DPs .................. 23
Binding principles ................................................................. 27

References ........................................................................... 35
Disjoint Reference Effects

What we’ve seen already

• In the very first class, we mentioned the following data:
  
  (1) a. Peter$_i$ forgets to lock the door every time he$_i$ leaves the house.
  b. * He$_i$ forgets to lock the door every time Peter$_i$ leaves the house.

  The notation blah$_i$ indicates reference — imagine every individual in the world is assigned a unique index: (1b) is only ungrammatical when he refers to the same individual as Peter; it is perfectly grammatical if the referent of he is different.

  • The phenomenon in (1b) is known as a disjoint reference effect:
    o The sentence is grammatical, but only if the DPs in question refer to different individuals, not if they corefer

    – This formulation assumes that there are only two options: either the reference of two DPs is disjoint, or they corefer
    – Once plural DPs are considered, a third possibility emerges: partial overlap
    – We will temporarily ignore plural DPs — but their behavior is actually the reason for choosing the term disjoint (rather than distinct or different) reference

What we’ve seen already

• We asked if the disjoint reference effect in (1b) — repeated here — could be about precedence (i.e., “what comes first”)

  (1) a. Peter$_i$ forgets to lock the door every time he$_i$ leaves the house.
  b. * He$_i$ forgets to lock the door every time Peter$_i$ leaves the house.

  ◦ and we answered by presenting (2a–b):

  (2) a. Every time Peter$_i$ leaves the house he$_i$ forgets to lock the door.
  b. Every time he$_i$ leaves the house Peter$_i$ forgets to lock the door.

  ◦ he and Peter can corefer both in (2a) and in (2b)

  ➢ THE GOAL: develop a theory that predicts when two expressions can/cannot corefer

Starting simple

• While we will eventually be able to handle data like (1–2) — it’s prudent to start with cases that are a little simpler:

  (3) a. * Peter$_i$ likes him$_i$.
  b. * He$_i$ likes Peter$_i$.

  ◦ but notice:

  (4) a. Peter$_i$ likes himself$_i$.
  b. * Himself$_i$/Heself(?)$_i$ likes Peter$_i$.

  ➢ there seems to be an additional factor going on when two phrases corefer that are arguments of the same predicate (the -self morphology)
Starting simple

• Dealing with too many variables at once is not a good idea
  ⇒ we want to neutralize the -self issue, for the time being
• One way of achieving this is by putting the two coreferring DPs in separate clauses:
  (5)  a. i. Peter \(_i\) thinks [that Lois likes him\(_i\)].  
      ii. Peter \(_i\) thinks [that he\(_i\) likes Lois].  
  b. i. * He\(_i\) thinks [that Lois likes Peter\(_i\)].  
      ii. * He\(_i\) thinks [that Peter\(_i\) likes Lois].
• Another possibility is using “complex” DPs as complements of the verb (instead of “simple” DPs, like him(self)):
  (6)  a. John\(_i\) likes [his\(_i\) sister].  
      b. * He\(_i\) likes [John\(_i\) sister].

Precedence, or not?

• Putting aside that issue of -self morphology, it seems that precedence would still handle this data just fine  
  ◦ e.g., a constraint that would require a DP to come before any coreferential pronouns  
  ➢ But remember, we’ve already seen data that doesn’t obey a precedence-based generalization  
  ⇒ What to do…?

Precedence, or not?

• Logical possibilities:
  I. Some cases obey a precedence-based generalization, and some don’t  
     ⇒ we need to uncover two things:  
     (i) the principle that governs whether precedence will be obeyed  
     (ii) the principle that governs coreference in the “other” cases  
  II. There is an entirely different constraint, which in some cases looks like precedence, but is really about something else entirely  
• Generative syntax has generally pursued approaches of type (II)  
  ◦ for some examples of approaches of type (I), see Jackendoff (1990), Janke and Neeleman (2009), Williams (1997)  
    (NOTE: These authors opt for a type-(I) approach over a type-(II) approach because of data from domains we probably won’t have time to discuss, here.)
**Precedence, or not?**

⇒ Let’s look at some more data!
- We want to tease apart precedence from other potential explanations
- So far, the second of two coreferential DPs has been “buried” within some larger structure
  - an embedded clause (as in (5)), or a “complex” DP (as in (6))
  - While this was necessary to avoid the -self issue, this is also a potential confound
    - since it conflates linear position with other properties — structural properties — that are not necessarily related
⇒ To balance things out, we should try placing the first of two coreferential DPs inside a larger structure, as well

EGG 2009 / COST-A33, Poznań

Intro to Syntax, PART FIVE Omer Preminger, MIT – 9 / 36

---

**Precedence, or not?**

⇒ compare (5a–b), repeated here, with (7a–b):

(5)  
a. i. Peter₁ thinks [that Lois likes him₁].
   ii. Peter₁ thinks [that he₁ likes Lois].

b. i. * He₁ thinks [that Lois likes Peter₁].
   ii. * He₁ thinks [that Peter₁ likes Lois].

(7)  
(6)  
(5)  

⇒ What is the contrast between (5b) and (7b) all about?

• **ATTEMPT #1:** There is something fundamentally different about a pronoun like *his* (compared to a pronoun like *he*), which prevents it from triggering a disjoint reference effect
**Precedence, or not?**

➢ Alas, this is simply not true; compare (8a) with (8b):

(8) a. John’s stories about his travels
   b. * his stories about John’s travels

(5) b. i. * He_i thinks [that Lois likes Peter_i].
    ii. * He_i thinks [that Peter_i likes Lois].

(7) b. i. [His_i mother] thinks [that Lois likes Peter_i].
    ii. [His_i mother] thinks [that Peter_i likes Lois].

• **ATTEMPT #2:** mother has been introduced in between his and Peter
  ➢ if this were enough, then (5b.i) would already be grammatical
  – since there, Lois is in between he and Peter

• Remember, in developing examples like (7b) we were trying to “bury” the first of two coreferential DPs within a larger syntactic structure

---

**c-Command**

⇒ the relevant difference between (5b) and (7b) is whether the pronoun (he/his) is “buried” in additional structure, or not:

(9) *

(10)

DEFINITION

a node α c-commands its sister, and everything dominated by its sister

a node γ dominates a node δ iff there is a monotone downward path (i.e., a path that only goes down, never up) in the tree going from γ to δ
Non-Coreference Rule

- In (9), he c-commands Peter
- In (10), his does not c-command Peter

⇒ We can formulate the following rule:

**NON-COREFERENCE RULE (version 1)**

if α and β are DPs, and α c-commands β, then: α and β cannot corefer

• PROBLEM:

(11) He thinks that he might be late.
(c.f. * He thinks that John might be late.)

Non-Coreference Rule

(12)

```
TP
  DP
    He
        T₀
            T'
                VP
                    CP
                        that
                            {he, John}
                                             might be late
            \phi
            thinks
```

Non-Coreference Rule

Let us define the following term:

**DEFINITION (subject to revision)**

*R-expression:* any DP that is neither a pronoun, nor a pronoun with -self-morphology

- We can now use this term to revise our NON-COREFERENCE RULE:

**NON-COREFERENCE RULE (version 2)**

if α and β are DPs, β is an R-expression, and α c-commands β, then: α and β cannot corefer

⇒ we can account for the pattern in (11), repeated here:

(11) He thinks that he might be late.
(c.f. * He thinks that John might be late.)
Non-Coreference Rule

Let’s look at some predictions made by this NON-COREFERENCE RULE:

I. Other ways of “burying” a DP within another DP should have the same effect as (7b) [= (10)], repeated here:

(7)  
- b. i. [His; mother] thinks [that Lois likes Peter].
  - ii. [His; mother] thinks [that Peter likes Lois].
  - • This prediction is borne out:

(13)  
- a. * He ensured [John would lose the election].
  - b. i. [The rumors [about him]] ensured [John would lose the election].
  - ii. [The rumors [that he; was irresponsible]] ensured [John would lose the election].

Non-Coreference Rule

II. Possessors (e.g., his) should be able to trigger a disjoint reference effect

- a possessor in [Spec,DP] does not c-command anything outside of the (bigger) DP
  - but it does c-command the NP, and everything within the NP:

\[ \begin{array}{c}
  \text{DP} \\
  \Downarrow \\
  \text{DP}_{\text{poss}} \\
  \Downarrow \\
  \text{D}'_{\text{big}} \\
  \Downarrow \\
  \text{D}^0_{\text{big}} \\
  \Downarrow \\
  \text{NP} \\
  \Downarrow \\
  \ldots \\
\end{array} \]

- • We have already seen that this prediction is borne out:

(15)  
- a. \[[\text{DP}_\text{big} \text{John’;s [NP stories about his; travels]}] (8a)\]
  - b. * \[[\text{DP}_\text{big} \text{his; [NP stories about John;’s travels]}] (8b)\]

Non-Coreference Rule

III. If c-command is all that is relevant to the NON-COREFERENCE RULE, it should be able to operate across arbitrarily long distances

(16)  
- a. * He; thinks that John; has won.
  - b. * He; thinks Susan knows that John; has won.
  - c. * He; thinks Mary mentioned that Susan knows that John; has won.
  - • This prediction is also borne out
Consequences

What we have uncovered here has some interesting consequences:

I. There no longer seems to be any formal role for *precedence*
   - There are no cases left that our NON-COREFERENCE RULE gets wrong, and that require a *precedence*-based explanation.
   - Whatever *precedence* effects we thought we were seeing were just a side-effect of *c-command* — coupled with the fact that in English, the *specifier* of an XP normally precedes the *complement*:

   \[
   \text{ENGLISH PHRASE-STRUCTURE}
   \]

\[
\text{XP} \\
\text{specifier} \quad X' \\
\text{X"} \quad \text{complement}
\]

➢ Regardless of whether such a pragmatic constraint exists, we have seen that incorporating it into our model of speakers’ *competence* would be *redundant*:
   - it cannot account for the full range of empirical facts
   - those facts that it *does* capture, are also captured by our NON-COREFERENCE RULE
     - which does not mention *precedence* or linear order


Consequences

- NOTE: it is often observed that unless special context is provided, a *pronoun-first* order (as in (18b)) is **pragmatically** dispreferred, relative to a *pronoun-second* order (as in (18a)):

   (18) a. John’s mother likes him.
   b. ? His mother likes John.

   ➢ Regardless of whether such a pragmatic constraint exists, we have seen that incorporating it into our model of speakers’ *competence* would be *redundant*:
     - it cannot account for the full range of empirical facts
     - those facts that it *does* capture, are also captured by our NON-COREFERENCE RULE
       - which does not mention *precedence* or linear order


Consequences

II. If our formulation of the NON-COREFERENCE RULE is correct, we have essentially constructed a *c-command “detector”*

   ⇒ giving us a unique window into the syntactic structure of utterances
     - When faced with a new construction:
       - we can investigate its syntactic structure by placing pronouns and R-expressions in different positions
         - and testing whether they can corefer
   ➢ As we will see, this is actually only one of a whole family of phenomena that are sensitive to *c-command*
A new perspective: Constraints on the distribution of DPs

- There is another way of looking at our NON-COREFERENCE RULE:

**NON-COREFERENCE RULE** (version 2)

if \( \alpha \) and \( \beta \) are DPs, \( \beta \) is an R-expression, and \( \alpha \) c-commands \( \beta \), then: \( \alpha \) and \( \beta \) cannot corefer

- as a condition on where R-expressions can/cannot appear
  - Given a syntactic structure, with all other DPs in place (and their referential indices fixed):
    - it tells us whether an R-expression, with a particular referential index, can appear in a particular syntactic position

For example:

(19)  

\[
\begin{array}{c}
\text{TP} \\
\text{DP} \quad T_1^0 \\
H_t \\
\phi \\
V_0^0 \\
\text{VP} \\
\text{DP} \\
\text{D'} \\
N^0 \\
\text{NP} \\
\text{PP} \\
\text{DP} \\
\text{D'} \\
S \\
\text{stories} \\
\text{about} \\
\text{childhood} \\
\end{array}
\]

- our NON-COREFERENCE RULE tells us whether each of the following could/couldn’t occupy the specifier of the lower DP:

(20)  

- a. * John
- b. * Mary
- c. Sam

⇒ we therefore need something to regulate the distribution of these expressions, as well
A new perspective: Constraints on the distribution of DPs

- Let us first classify DPs into 3 types:

  A. **anaphors**: must get their reference from some other element in the sentence
     
     *EXAMPLES:* himself, herself, themselves
     
     (22) John₁ likes himselfᵢᵢ.  
     
     **Note:** This is decidedly from the use of the term *anaphor(a)* in other fields (e.g., literary analysis).

  B. **pronouns**: can get their reference from some other element in the sentence, but don’t have to
     
     *EXAMPLES:* he(he/him/his), she(her), they(they/them/their)
     
     (23) Maryᵢ thinks that sheᵢᵢ will win.

  C. **R-expressions**: come with their own reference “built-in”
     
     *EXAMPLES:* John, Mary, the children, the Roman empire

Binding principles

- We already know what regulates the distribution of **R-expressions**:
  
  - what we called the **NON-COREFERENCE RULE**
    
    ➢ We will now rename this **PRINCIPLE C**
    
    (corresponding to “C” on our list of DP-types — namely, R-expressions)

  **PRINCIPLE C**

  an R-expression cannot be *c-commanded* by a coreferential DP

  ⇒ we need a **PRINCIPLE A** and a **PRINCIPLE B**

  - to constrain the distribution of *anaphors* and *pronouns*, respectively
Binding principles

- Let’s start by trying to formulate PRINCIPLE A — which would regulate the distribution of anaphors
  
  - Recall (21a–b), repeated here:

  \[
  \begin{align*}
  \text{(21) a. } & \text{John likes \{"him\_i\}, } \\
  \text{b. John_{i} thinks that Mary likes \{him\_i, } \\
  \text{\} hismelf{\_i}.} \\
  \end{align*}
  \]

  **PRINCIPLE A (attempt #1)**
  
  an anaphor must corefer with another argument of the same predicate

  - While this seems like a necessary condition on the distribution of anaphors, it is not a sufficient condition

  \[
  \begin{align*}
  \text{(24) } & \text{He likes John.} \\
  \text{\{"Himself\,} \\
  \text{\} Hesel(?)\} \\
  \end{align*}
  \]

  - It might be tempting, at this point, to revert to a precedence-based account, if only just for the asymmetry between (24) and (21a)

  Interestingly, there is some empirical support for this particular move, in the specific context of same-clause coreference; see Jackendoff (1990), Janke and Neeleman (2009), Williams (1997).

  - Something along the lines of “an anaphor must corefer with a preceding argument of the same predicate”

  but...

  - RECALL: PRINCIPLE C involves c-command, in a way that cannot be reduced to precedence

  – as we demonstrated in detail, while it was still called the “NON-COREFEERENCE RULE”

  \[
  \begin{align*}
  \text{(25) } & \text{He likes John.} \\
  \text{\{"Himself\,} \\
  \text{\} Hesel(?)\} \\
  \end{align*}
  \]

  - in the interest of uniformity, we will try to have all of these principles refer to c-command

  – rather than some referring to c-command and some to precedence

  (again, given that stating PRINCIPLE C in terms of precedence was shown to be impossible

  **PRINCIPLE A (attempt #2)**
  
  an anaphor must corefer with a c-commanding argument of the same predicate (follows Reinhart and Reuland 1993)
Binding principles

- Now let’s try to formulate **PRINCIPLE B** — which would regulate the distribution of pronouns
  - Consider (21a–b), repeated here, once more:
    (21) a. John_i likes \{“him\_i”, \text{himself}_i\}.
    b. John_i thinks that Mary likes \{”him\_i”, \text{himself}_i\}.
  - As a starting point, we could try the “opposite” of **CONDITION A**:

**PRINCIPLE B (attempt #1)**

- a pronoun must **not** corefer with a c-commanding argument of the same predicate
  (again, follows Reinhart and Reuland 1993)

**PRINCIPLE B (attempt #2)**

- a pronoun’s reference must be **disjoint** from every c-commanding argument of the same predicate
  (follows Lasnik 1981, 1989, Reinhart and Reuland 1993)

Binding principles

- This works perfectly, so long as the DPs in question are **singular**
- Consider (25):

(25) We like \{”me”, \text{myself}\}. [Lasnik 1981, 1989]

  - Clearly, we and me don’t corefer — their references are different
    - but their references overlap
  ➢ This is the reason why, from the very beginning, we chose the term **disjoint reference**, rather than just **different reference**

Binding principles

**Summary:**

**PRINCIPLE A**

- an anaphor must corefer with a c-commanding argument of the same predicate

**PRINCIPLE B**

- a pronoun’s reference must be disjoint from every c-commanding argument of the same predicate

**PRINCIPLE C**

- an R-expression cannot be c-commanded by a coreferential DP
Binding principles

- These principles \((A+B+C)\) are known as the binding principles
  - binding refers to the situation where one DP \(\alpha\) shares an index with, and c-commands, another DP \(\beta\)
- But as we’ve seen, this name is a little bit misleading, since PRINCIPLE B is about more than just c-command + index-sharing
  - that, in fact, was our version #1 of PRINCIPLE B
- It is about disjointness, which is a stronger notion than just “not sharing the same index”

References