A New Approach to Tough-Constructions

Sandiway Fong and Jason Ginsburg
University of Arizona and Osaka Kyoiku University

1. Introduction

Tough-constructions (TCs) have been the subject of much research, such as Chomsky (1964), Rosenbaum (1967), Ross (1967), Postal (1971), Lasnik & Fiengo (1974), Chomsky (1977, 1981), Hornstein (2001), and Hicks (2009), among others. These constructions are of interest because of the property whereby the subject is the understood object. In (1), the tough-predicate "easy to please" lacks an overt object. This empty object of 'please' is understood to be'John', which is the subject of the sentence.[1]

(1) John is easy (for us) to please. (Chomsky, 1977:103)

Chomsky (1977) proposes that a TC involves wh-movement whereby a wh-phrase/operator is base generated in the embedded clause and undergoes wh-movement to the clause edge. Chomsky later, in Chomsky (1981), proposes that a TC subject obtains a theta-role by binding the embedded clause operator. Under this analysis, the derivation of (2a) proceeds as in (2b), where the embedded complementizer is represented as 'for'.

(2) (a) John is easy (for us) [to please](Chomsky, 1977:102)

(b) John is easy (for us) [$_{\overline{S}}$ [who$_i$ for] PRO to please t$_i$] (Chomsky, 1977:103)

A wh-operator 'who' is base generated in object position of 'please', from where it undergoes wh-movement to the embedded clause edge, and the TC subject 'John' binds 'who', thereby giving them an identical theta-role and reference. In more recent work, this wh-phrase is indicated as being an 'Op' to represent its status as a null wh-operator, as shown in (3).

(3) John$_i$ is tough [$_{CP}$ OP$_i$ [$_{TP}$ PRO to please t$_i$]]. (Hicks, 2009:536)

This type of wh-movement analysis of TCs is supported by examples such as (4) in which there is both TC movement and typical wh-movement.

(4) (a) *What sonatas is this violin easy to play on?

(b) the violin is easy [$_{\overline{S}}$ [which$_i$ for] PRO to play sonatas on t$_i$] (Chomsky, 1977:105)

According to Chomsky (1977), in (4a), wh-movement of 'what sonatas' is blocked by a null wh-operator in the embedded clause, as shown in (4b), in which the wh-operator, represented as 'which', has moved to the TC predicate edge, thereby preventing any further wh-movement, such as movement of 'what sonatas', through the embedded clause.

(5a), like (4a), also involves TC movement and typical wh-movement, but it is well-formed. To account for this example, Chomsky proposes that this construction has the structure in (5b) in which the PP 'on what violin' is adjoined to the VP. Thus, 'what violin' is able to undergo wh-movement to the matrix clause edge without passing through the TC predicate; if it were to pass through the edge of the TC predicate, then it should be ill-formed since the null operator should block tough-movement.

(5) (a) What violin is the sonata easy to play on? (Chomsky, 1977:105)

(b) the sonata is [$_{AP}$ easy [$_{\overline{S}}$ [which$_i$ for] PRO to play t$_i$]] on what violin (Chomsky, 1977:106)

* We would like to thank the audience of WCCFL 31 for their helpful comments.

[1] The 'tough' predicate also lacks an overt subject; this subject is generally considered to be PRO.

There are a variety of problems for this type of TC analysis. Introduction of an empty operator is necessary to account for wh-movement, but it creates an extra level of complexity. Is a null operator, which is not pronounced, really required, and if so, how does it influence the interpretation of a TC? Also, it isn't clear how theta-role assignment works. Can an R-expression give a theta-role to an Op via binding? Lastly, Chomsky's analysis requires the PP "on this violin" to be an underlying complement in (4a), but the PP "on what violin" to be an adjunct in (5b), even though the TC predicate "easy to play on" is pronounced identically in both constructions. A more uniform analysis would simply give them the same structure, either as a complement or as an adjunct, in both constructions.

Hicks (2009) draws on Chomsky (1977, 1981) and Kayne (2002) for inspiration. Hicks adopts Chomsky's wh-movement account combined with a doubling constituent analysis, in the spirit of Kayne, whereby TC movement results from movement of a null operator out of a doubling constituent. The proposed doubling constituent structure is shown in (6), in which the operator 'Op' and 'everyone' share a theta-role.

(6) [$_{DP}$ D [$_{NP}$ [$_{N}$ Op] [$_{DP}$everyone]]]

According to Hicks, the derivation of (7a) proceeds along the lines of (7b).

(7) (a) Everyone is tough for us to please (Hicks, 2009:547)

(b) [$_{TP}$ Everyone is [$_{aP}$ tough [$_{AP}$ [$_{PP}$for us] tough [$_{CP}$ [$_{DP}$Op ~~everyone~~] C$_{Int}$ PRO to [$_{vP}$ ~~[$_{DP}$Op everyone]~~ PRO please ~~[$_{DP}$Op everyone]~~]]]]]

The doubling constituent is base generated as the object of 'please'. The TC complement clause contains an interrogative C that attracts the Op, which is a wh-element. When Op moves, it smuggles (Collins 2005a, 2005b) its wh-constituent complement with it to the embedded clause edge, from where it can be attracted by a probe in the matrix clause. Note that under this analysis, the r-expression TC subject is not base generated in its surface position, but rather undergoes movement, which is both A and A'-movement.[2]

A disadvantage of wh-movement accounts of tough-constructions of the sort proposed by Hicks is that a TC adjective sometimes, but not always, must take an interrogative CP complement. Under a wh-movement analysis, the complement clause of 'tough' in the TC (8a) is an interrogative CP. On the other hand, the complement of 'tough' in the expletive version (8b) is not an interrogative. In the latter case, there is no evidence to our knowledge that there is anything akin to wh-movement in this example, and there is no reason to assume that the non-finite clause would be interrogative.

(8) (a) Linguists are tough to please.

(b) It is tough to please linguists. (Hicks, 2009:535-36)

Our new approach does not face this last problem presented by (8a-b), since the complement of a TC adjective is never an interrogative CP. We demonstrate how this analysis accounts for typical TCs, such as (1), as well as the potentially problematic examples (4), (5), and (8).

2. Proposals

We propose a computational model of syntax that accounts for TCs via a stack mechanism and a doubling constituent. The stack mechanism has the advantage of doing away with probe-goal search of Phase Theory (Chomsky, 2001). The doubling constituent mechanism has the advantage of accounting for how a TC subject and null object obtain theta-roles and corefer.

We propose that natural language makes use of a stack mechanism, which is an ordered list; the last element that is pushed onto the stack is the first element that is popped out of the stack. For economy reasons, we propose there is no search. Only the top of the stack element (TOS) is visible. We stipulate that all operations, e.g. Agree, must operate on the TOS. This severely constrained mechanism

[2] This analysis also requires that the doubling constituent move to the embedded vP clause edge in order to be accessible to the interrogative C.

is sufficient to handle the TC cases. A Syntactic Object (SO) with an uninterpretable/unvalued feature is pushed onto a stack.[3] A probe can only agree with the TOS, if Agree is possible. SOs stacked below the TOS cannot become visible for agreement until Agree applies to the TOS, fully valuing its unvalued features. A fully-licensed SO is popped off the top of the stack. All operations (Merge, Agree) involve the TOS element. Any incomplete SOs, e.g. with unvalued features, are stacked as they are built, and removed as they are valued, in strict stacking order.

We also utilize a doubling constituent. Following Kayne (2002) and Hicks (2009), we propose that tough-movement (TM) involves the following doubling constituent structure.

(9) [$_D$ [$_N$ *pro* [$_N$ [$_D$ r-expr]]]]

There is a null D head that takes a *pro* as its complement, where the term *pro* is used loosely to refer to a null NP.[4] This *pro* takes the r-expression as its complement. We propose that *pro* has a unique property in that it passes its theta-role on to its complement. This notion of theta-role sharing follows Hicks (2009).

Note that these proposals can be implemented formally. They have been validated using a computer model, which automatically generated the derivations presented in this paper.

3. Derivations

In this section we demonstrate how our proposals predict typical TCs.

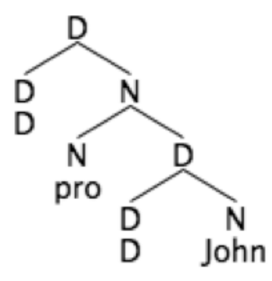
The typical TC (10) (originally presented as (1) above) has the derivation in (11).

(10) John is easy to please. (Chomsky, 1977:103)

Initially, *pro* and the DP 'John' Merge, forming a doubling constituent, and 'John' is pushed onto the stack, as a result of its being unlicensed, since it lacks Case and a theta-role (11a).[5] At the point that the v*P is completed, 'PRO' and 'John' are on the stack, since they both lack Case (11b).[6] Then, at the point that the matrix T is Merged, only 'John' is left in the stack (11c). Since 'John' is the top of the stack (TOS) element, T is able to Agree with it and assign it Case. At this point, an EPP feature on T forces Remerge of the TOS 'John' with T (11d).

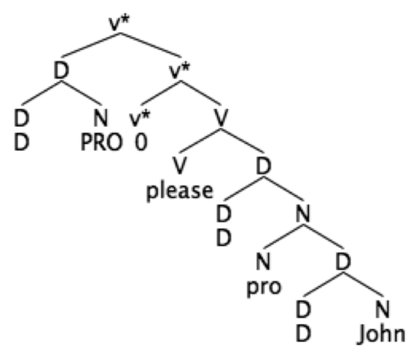
(11)

(a)



Stack: [D [D D [uCase]] [N John]]

(b)



Stack:
[D [D D [uCase]] [N PRO]]
[D [D D [uCase]] [N John]]

---

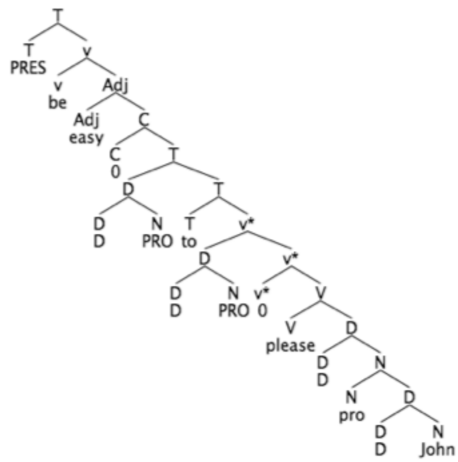
[3] For the sake of simplicity, we refer to unchecked features as being unvalued, leaving aside the issue of whether or not interpretability and feature valuation are separate. See Pesetsky & Torrego (2007) among others for discussion of this issue.

[4] We do not attach strong theoretical importance to the term *pro* here. More work is needed to determine the exact nature of this element.

[5] The feature 'uCase' signifies unvalued Case. We have indicated this unvalued Case feature as being on D, the idea being that the entire DP needs Case.

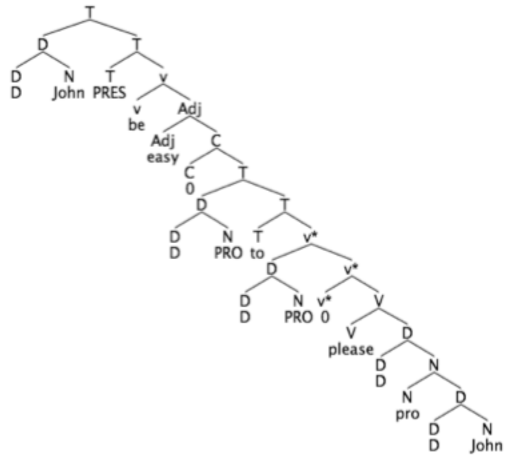
[6] We assume that PRO requires some type of Case, such as null Case Chomsky & Lasnik (1993), and thus have indicated it as having a 'uCase' feature.

(c)



Stack: [D [D D [uCase]] [N John]]

(d)



Stack: [ ]

This analysis predicts the data in (4-5) above, which involve both tough-movement and typical wh-movement, as in (12b) and (13b) below. (12a) and (13a) are typical TCs. However, (12b) and (13b) involve wh-movement together with TM, and these examples differ in their well-formedness. (12b) does not permit wh-movement and TM, whereas (13b) does. Crucially, in (12b), the base position of the wh-phrase ‘what sonata’ is higher than the base position of the TC subject ‘the violin’, as shown in (12c). On the other hand, in (13b), the base position of the TC subject ‘the sonata’ is higher than the base position of the wh-phrase ‘what violin’, as shown in (13c). We propose that this difference in the base positions of the TC subjects and wh-phrases accounts for the well-formedness differences between these two examples (12b) and (13b).

(12) (a) The violin is easy to play **the sonata** on.

(b) ***What sonata** is the violin easy to play on?

(c) [easy to play **what sonata** on **the violin**]

(13) (a) The sonata is easy to play on **this violin**.

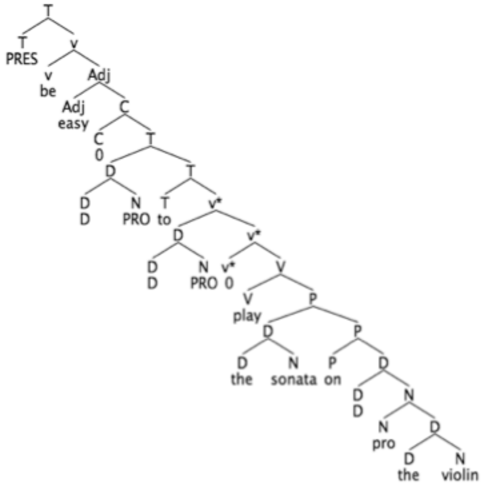
(b) **What violin** is the sonata easy to play on?

(c) [easy to play **the sonata** on **what violin**]

The TC (12a) has the derivation (simplified) shown in (14). The doubling constituent is base generated within an embedded PP, which we assume has full argument structure, along the lines of Pesetsky (1995), with ‘the sonata’ in specifier position and the doubling constituent in complement position. At the point that the matrix T is Merged, ‘the violin’ is in the stack, since it lacks Case (14a). Thus, it Agrees with T, its Case is checked, it is popped off the stack, and an EPP feature on T forces it to remerge with T, resulting in (14b) which converges successfully.

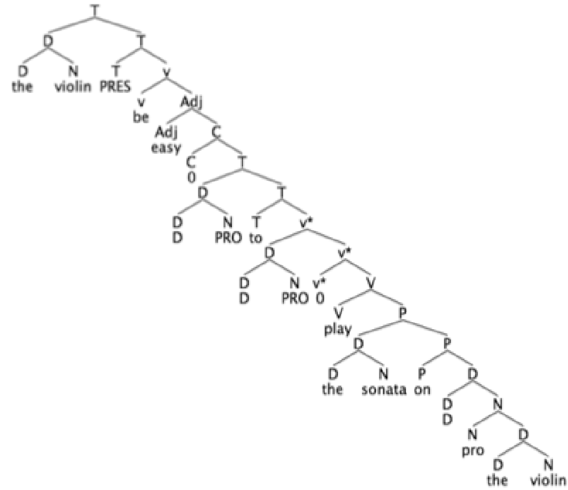
(14)

(a)



Stack: [D [D the [uCase]] [N violin]]

(b)

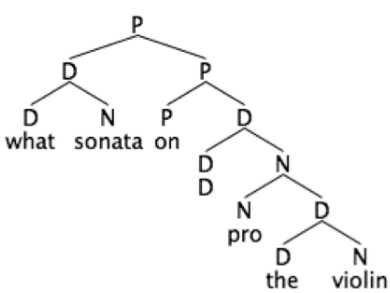


Stack: [ ]

This analysis accounts for the ill-formed (12b) as follows. The doubling constituent is base generated as the complement of the P and the PP has the structure shown in (15a) with the wh-phrase 'what sonata' in its specifier position. The DP 'the violin' is Merged into the derivation first, and thus it goes onto the stack (since it is unlicensed), followed by the wh-phrase, which is Merged later. We assume that the wh-phrase goes onto the stack because it has an unvalued scope feature.[7] Since the wh-phrase is Merged later than 'the violin', the wh-phrase ends up as the TOS. At the point at which the matrix T is Merged (15b), the wh-phrase is still at the top of the stack, since it has an unvalued Scope feature. Since the wh-phrase is the TOS, and T can only agree with the TOS, T is unable to Agree with and assign Case to 'the violin'. The wh-phrase already obtained Case in its base position, and thus is not able to undergo Agree with T to obtain Case. The TOS wh-phase thus blocks T from agreeing with 'the violin,' which is unable to get Case, thereby causing the derivation to crash.

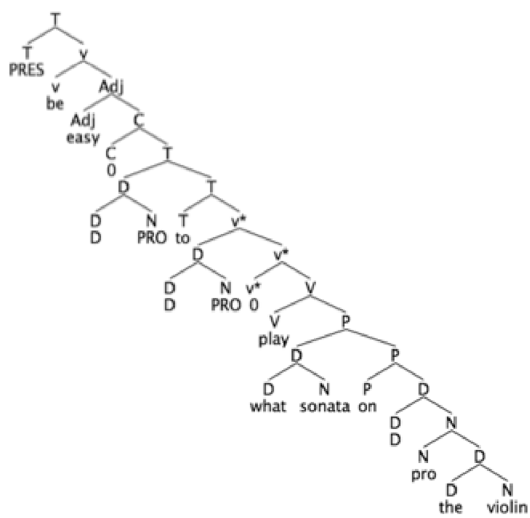
(15)

(a)



Stack: [D [D what [uCase, uScp]] [N sonata]] / [D [D the [uCase]] [N violin]]

(b)



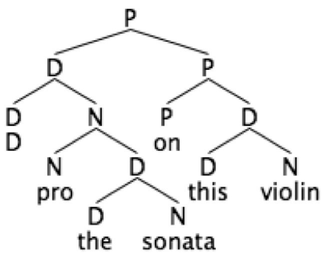
Stack: [D [D what [uScp]] [N sonata]] / [D [D the [uCase]] [N violin]]

[7] It is crucial to this analysis that the wh-phrase has some type of unvalued feature. We indicate this as as being a scopal feature 'Scp', however, the exact nature of this feature is not crucial for this analysis.

The well-formedness of examples (13a-b) is accounted for as follows. Example (13a) is a typical TC. In this case, the doubling constituent is base generated in the specifier of the PP (16a). At the point that the PP is completed, the doubling constituent is the TOS, since its head 'pro' has unchecked Case. The r-expression 'the sonata', which is also unlicensed, is at the bottom of the stack. The doubling constituent is eventually licensed ('pro' obtains Case) and popped off the stack. At the point that the matrix T is Merged, only 'the sonata' is left in the Stack (16b). Thus, T is able to Agree with 'the sonata', which gets assigned Case. An EPP feature on T forces 'the sonata' to remerge and the derivation converges (16c).

(16)

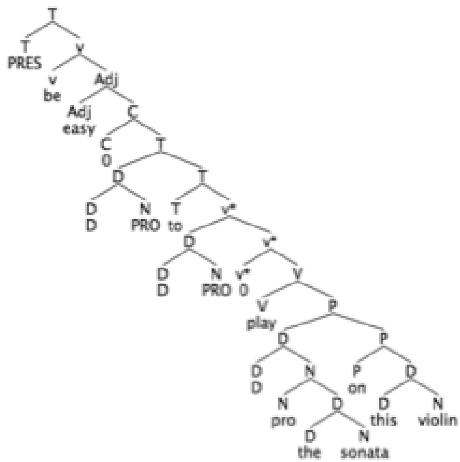
(a)



Stack:

[D [D D [uCase]] [N [N *pro*] [D [D the [uCase]] [N sonata]]]]
[D [D the [uCase]] [N sonata]]

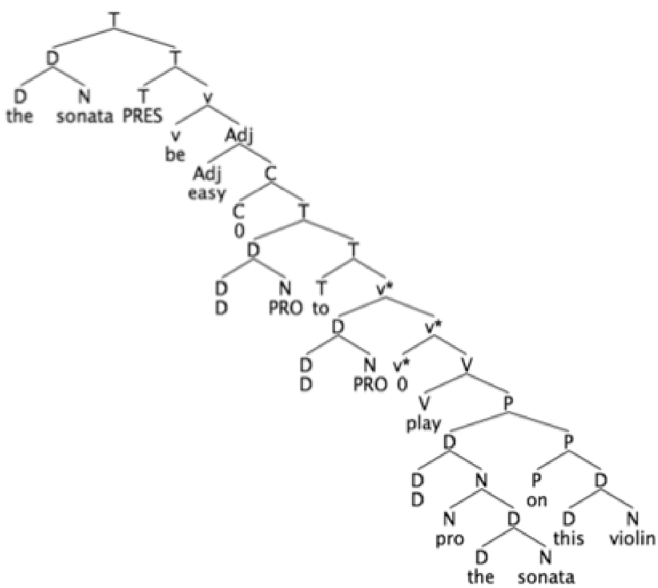
(b)



Stack:

[D [D the [uCase]] [N sonata]]

(c)



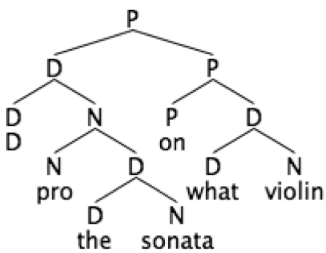
Stack:

[ ]

In (13b), the wh-phrase (the complement of P) is initially Merged before the doubling constituent (the specifier of P). Thus, at the point that the PP is completed (17a), the wh-phrase is at the bottom of the stack. At the point that that the matrix T is Merged (17b), 'the sonata' is at the top of the stack. Therefore, it is able to Agree with T and undergo remerge, resulting in (17c). At this point, only the wh-phrase remains in the stack. Lastly, an interrogative C*[8] Agrees with the wh-phrase; here we assume that a valued Scope feature on the interrogative C* values the unvalued Scope feature on the wh-phrase. The wh-phrase is then popped off the stack and an EPP feature on C forces it to remerge with C (17d).

(17)

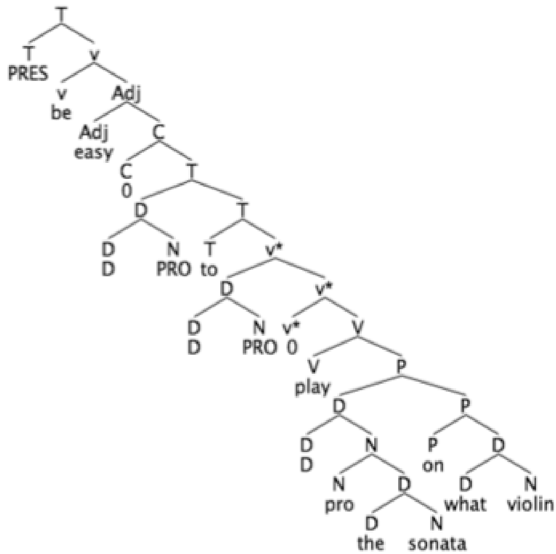
(a)



Stack:

[D [D D [uCase]] [N [N *pro*] [D [D the [uCase]] [N sonata]]]]
[D [D the [uCase]] [N sonata]]
[D [D what [uScp]] [N violin]]

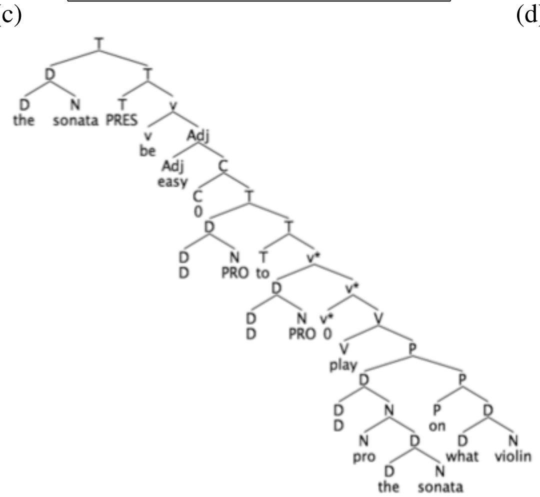
(b)



Stack:

[D [D the [uCase]] [N sonata]]
[D [D what [uScp]] [N violin]]

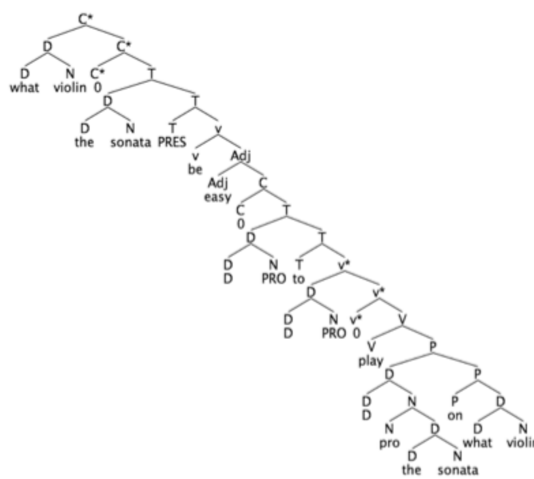
(c)



Stack:

[D [D what [uScp]] [N violin]]

(d)



Stack: [ ]

Crucially, both (12b) and (13b) require TC movement and typical wh-movement, yet only (13b) is well-formed. Our analysis is able to predict this distinction. In (12b), as shown in (15), the wh-phrase 'what sonata', via its TOS position, blocks T from Agreeing with 'the violin', which is unable

[8] The '*' indicates that C is a phase head.

to get Case. In (13b), as shown in (17), ‘the sonata’ is the TOS when the matrix T is Merged, so it is able to Agree with T and obtain Case. Then ‘what violin’ moves to the TOS position, where it can be successfully licensed by C.

An advantage of this analysis is that a TC adjective never requires an interrogative complement, thus doing away with the problem that (8a-b), repeated below, poses for wh-movement proposals of TCs, since these proposals would require the adjectival complement in (18a) to be interrogative, but not that in (18b).

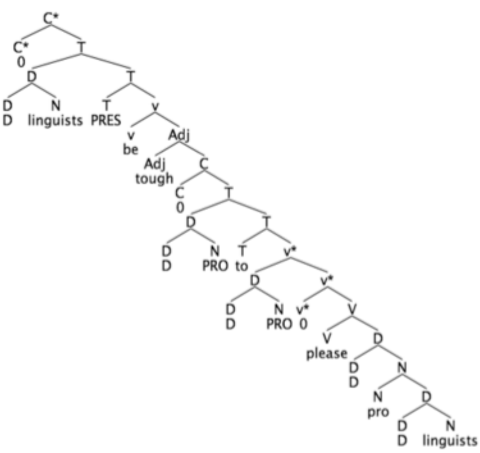
(18) (a) Linguists are tough to please.

(b) It is tough to please linguists. (Hicks, 2009:535-36)

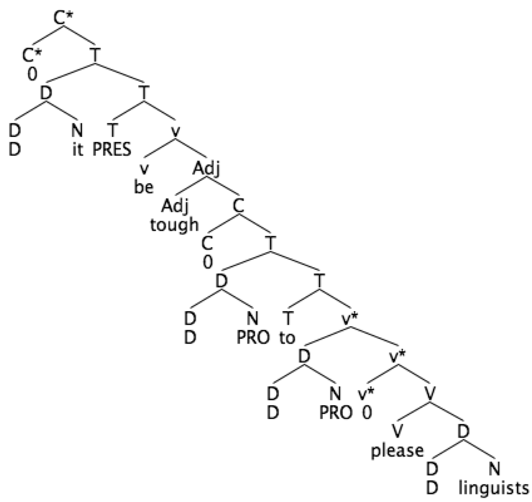
The derivation (simplified) of (18a), shown in (19a), is accounted for as discussed above, with Agree occurring, via the stack, between T and the r-expression ‘linguists’. On the other hand, in (18b), there is no doubling constituent, and the expletive ‘it’ is simply Merged with the matrix T, as shown in (19b).

(19)

(a)



(b)



4. Conclusion

We have presented a new approach to tough-constructions that utilizes a stack mechanism and doubling constituents. The stack mechanism does away with the need for search in probe-goal computation and maximizes efficiency. The doubling constituent proposal accounts for how the r-expression and antecedent receive identical interpretations and theta-roles.

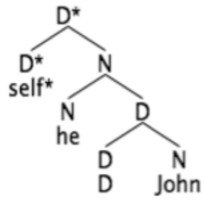
In addition, this analysis can be extended to account for a variety of typical coreference (Binding) data. Hicks (2009) develops a doubling constituent analysis, in the spirit of Kayne (2002). However, Hicks accounts for TC effects, not coreference phenomena. Kayne (2002) focuses on coreference, not TCs. Our analysis not only can account for TCs, but it can also account for standard coreference data.

In Fong & Ginsburg (2012), we propose that there are two kinds of doubling constituents. One type, an anaphor, is a phase with a D phase head ‘self’, as shown in (20a), where ‘D*’ represents a DP phase head.[9] Other types of doubling constituents are not phases, such as (20b). Note that the TC doubling constituent structure also falls into this latter structure. If we assume this phasehood distinction, we can explain a variety of coreference phenomena.

[9] We assume that there is some type of morphological merger that produces ‘himself’ from the D ‘self’ and the NP ‘him’.

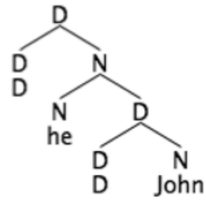
(20)

(a) $John_1$ praises $himself_1$



DP doubling constituent is a phase

(b) $*John_1$ praises him_1



DP doubling constituent is not a phase

Phasehood translates into a delay in accessibility. We propose that an element on a stack only becomes accessible when the smallest containing phase becomes fully licensed. In (20a), the r-expression becomes accessible when the containing D*P is licensed. In (20b), the r-expression becomes accessible when the containing v*P is licensed. This proposal can account for typical coreference (Binding) data. Results of our previous work (Fong & Ginsburg, 2012) translate directly into this stack implementation. Due to lack of space, we leave an in-depth discussion of how this analysis extends to coreference phenomena for future work.

References

Chomsky, Noam (1964). *Current issues in linguistic theory*. Mouton, The Hague.

Chomsky, Noam (1977). On wh movement. Culicover, Peter W., Thomas Wasow & Adrian Akmajian (eds.), *Formal syntax*, Academic Press, New York, 71–132.

Chomsky, Noam (1981). *Lectures on Goverment and Binding*. Foris, Dordrecht.

Chomsky, Noam (2001). Derivation by phase. Kenstowicz, Michael (ed.), *Ken Hale: a life in language*, MIT Press, Cambridge, MA, 1–52.

Chomsky, Noam & Howard Lasnik (1993). The theory of principles and parameters. Jacobs, Joachim, Arnim von Stechow, Wolfgang Sternefeld & Theo Vennemann (eds.), *Syntax: An international handbook of contemporary research*, de Gruyter, Berlin, 506–569.

Collins, Chris (2005a). A smuggling approach to raising in English. *Linguistic Inquiry* 36, 289–298.

Collins, Chris (2005b). A smuggling approach to the passive in English. *Syntax* 8, 81–120.

Fong, Sandiway & Jason Ginsburg (2012). Computation with doubling constituents: Pronouns and antecedents in phase theory. Di Sciullo, Anna Maria (ed.), *Towards a biolinguistic understanding of grammar: Essays on interfaces*, John Benjamin Publishing Company, Amsterdam, 303–338.

Hicks, Glyn (2009). Tough-constructions and their derivation. *Linguistic Inquiry* 40, 535–566.

Hornstein, Norbert (2001). *Move! A minimalist theory of construal*. Blackwell, Malden, MA.

Kayne, Richard (2002). Pronouns and their antecedents. Epstein, Samuel David & T. Daniel Seely (eds.), *Derivation and explanation in the Minimalist Program*, Blackwell Publishing, Malden, MA, 133–166.

Lasnik, Howard & Robert Fiengo (1974). Complement object deletion. *Linguistic Inquiry* 5, 535–571.

Pesetsky, David (1995). *Zero syntax: Experiencers and cascades*. MIT Press, Cambridge, MA.

Pesetsky, David & Esther Torrego (2007). The syntax of valuation and the interpretability of features. Karimi, Simin, Vida Samiian & Wendy Wilkins (eds.), *Phrasal and clausal architecture: Syntactic derivation and interpretation. In honor of Joseph E. Emonds*, John Benjamins, Amsterdam, 262–294.

Postal, Paul (1971). *Cross-over phenomena*. Holt, Rinehart, & Winston, New York.

Rosenbaum, Peter S. (1967). *The grammar of English predicate complement constructions*. MIT Press, Cambridge, MA.

Ross, John (1967). *Constraints on variables in syntax*. Ph.D. thesis, Massachusetts Institute of Technology.

Proceedings of the 31st West Coast Conference on Formal Linguistics

edited by Robert E. Santana-LaBarge

Cascadilla Proceedings Project Somerville, MA 2014

Copyright information

ISBN 978-1-57473-462-1 library binding

Ordering information

Orders for the library binding edition are handled by Cascadilla Press.
To place an order, go to www.lingref.com or contact:

Cascadilla Press, P.O. Box 440355, Somerville, MA 02144, USA
phone: 1-617-776-2370, fax: 1-617-776-2271, sales@cascadilla.com

Web access and citation information

This entire proceedings can also be viewed on the web at www.lingref.com. Each paper has a unique document # which can be added to citations to facilitate access. The document # should not replace the full citation.

This paper can be cited as:

Fong, Sandiway and Jason Ginsburg. 2014. A New Approach to Tough-Constructions. In *Proceedings of the 31st West Coast Conference on Formal Linguistics*, ed. Robert E. Santana-LaBarge, 180-188. Somerville, MA: Cascadilla Proceedings Project. www.lingref.com, document #3020.