Optimality Theory

Optimality Theory, although a general system, arose out of work in phonology.

Phonology was traditionally done in a rule-based fashion, with intermediate representations, much like the syntax we are familiar with. There is a starting point (underlying form), there are a set of rules which apply (perhaps in a particular order, perhaps over and over again) until the surface form is reached.

But it was observed that there are phenomena which, although easy enough to explain in “plain English,” came out strangely or only in a roundabout way from the rules.

Let’s do a little bit of phonology. We’ll look at the phonology of Mohawk, which can illustrate this reasonably well.

1. Penultimate stress (normally)
   a. rágwas /hra-kw-as/ ‘He picks it’
   b. wakashétu /wak-ashet-u/ ‘I have counted it’
   c. wakharatatuhátye /wak-haratat-u-hatye/ ‘I go along lifting up’

2. Contexts in which epenthetic e is inserted:
   a. Between a consonant and a single sonorant (Cen, Cer, Cew)
   b. Between a consonant and a word-final glottal stop.
   c. After a consonant when followed by a consonant cluster (except hC and sC).

3. In some cases, stress skips epenthetic vowels (e below)
   a. tégiks /te-k-rik-s/ ‘I put them together’
   b. wákara /w-akra-s/ ‘it smells’
   c. tákriqṣ /t-ə-rik-ʔ/ ‘I’ll put together side by side’
   d. waʔtkafágnékʔ /waʔ-t-ə-atat-nak-ʔ/ ‘I scratched myself’

The way this has traditionally been handled is with a set of rules that happen in an order.

4. Assign stress (to the penultimate syllable)
   Insert epenthetic vowels

But not all epenthetic vowels are skipped for stress:

5. Unskipped epenthetic vowels (e below)
   a. wakñyék /wak-nyak-s/ ‘I get married’
   b. tekahsérha /te-k-ahsutr-haʔ/ ‘I splice it’
   c. sásáhték /sa-s-ahkt/ ‘go back!’

The natural change to make is to suppose that there are some epenthetic vowels inserted before stress assignment.

6. Insert some epenthetic vowels
   Assign stress
   Insert some more epenthetic vowels

Here’s Michelson’s (1989) proposal about Mohawk stress and epenthesis:

7. Joiner Insertion (insert a onto melody tier in C[...]C)
   e-Epenthesis (insert e onto melody tier in C_R, C_C, C_R, C_C, C_C, C_C)
   V-Insertion I (link unlinked vowel to V slot in C_CC)
   V-Insertion II (link unlinked a [joiner] to V slot)
   Vowel Lengthening (lengthen stressed vowel in _CV)
   Prothesis (insert i in #_CV#)
   V-Insertion III (link unlinked e to V slot)

What this says is this: Insert an (invisible) epenthetic e where called for.

If any of them are before a consonant cluster, make it visible.

Assign stress.

Any invisible e that’s left over, make it visible.

But there’s another way to look at this:

Epenthetic e does not count for stress assignment unless it is needed for syllabification.

A proper syllable in Mohawk is a CVC syllable. Go back and look at where e counts.
We can say that Mohawk has a constraint against counting epenthetic e for stress, but that it is overridden by a constraint requiring syllables to have the shape CVC (CVCs counts as CVC). Isn’t that much easier?

(8) 
\[
\begin{array}{|c|c|}
\hline
\text{epenthetic e} \rightarrow \text{stress} & \text{CVC} \\
\hline
\text{a. ték<e>riks} & * \text{STRESS-E} \\
\hline
\text{b. te.ké.riks} & *! \\
\hline
\end{array}
\]

Isn’t that much easier?

(9) 
\[
\begin{array}{|c|c|}
\hline
\text{CVC} * \text{STRESS-E} \\
\hline
\text{a. wák<e>nyaks} & *! \\
\hline
\text{b. wa.kén.yaks} & \oplus \otimes \\
\hline
\end{array}
\]

The structure of the theory

(10) Constraints are universal (all languages have them—this is “UG”). Grammars differ only in the ranking of the constraints across languages. The child’s task during acquisition is to acquire the rankings.

\[
\begin{array}{|c|c|}
\hline
\text{C}_1 & \text{C}_2 & \ldots \\
\hline
\text{candidate}_1 & * & * & \ldots \\
\hline
\text{candidate}_2 & * & \checkmark \\
\hline
\text{candidate}_3 & \checkmark & \checkmark \\
\hline
\text{candidate}_4 & \checkmark & \checkmark \\
\hline
\text{candidate}_5 & \checkmark & \checkmark \\
\hline
\text{candidate}_6 & \checkmark & \checkmark \\
\hline
\text{candidate}_7 & \checkmark & \checkmark \\
\hline
\text{…} \\
\hline
\end{array}
\]

Optimality Theory has pretty much “taken over” phonology—but it hasn’t gotten very far in syntax yet, although there are a number of people working on a number of different OT-based approaches...

Legendre (2000). An introduction to optimality theory in syntax

GB theory: Inviolable principles in syntax (Case Filter, θ-criterion, ECP, …)
Parameters account for the differences between languages (e.g., bounding nodes for Subjacency).

A potential problem: Parameters are set language-by-language.
Yet patterns and generalizations sometimes only apply to part of a language.

A problem with an optimality character: Expletive subjects in German

(11) a. Es wurde schön gedanzt.  
    it was beautifully danced  
    'The dancing was beautiful.'
    b. Schön wurde getanzt.  

The question: Why is (11c) ungrammatical? Seems: es can satisfy the EPP if nothing else does.

A simple case:

(12) a. It rained.  
    b. Piove.  

We could say that in Italian the EPP is satisfied, but by pro—but there is also a sense in which this renders the EPP as a universal (inviolable) constraint pretty vacuous.

Plus, there is something better about (12b)—it has no meaningless elements in it.

Suppose that we have two constraints:

\[
\begin{array}{|c|c|}
\hline
\text{SUBJECT} & \text{FULL-INT} \\
\hline
\text{The highest A-specifier must be filled with a subject.} & \text{Lexical items must contribute to the interpretation.} \\
\hline
\end{array}
\]

For sentences that mean “it rained” you can’t have it both ways. There is no argument, so if you are going to have the subject position filled (satisfying SUBJECT) you have to fill it with a meaningless element (violating FULL-INT).

In English, it is more important to satisfy SUBJECT, even if that means violating FULL-INT.

(13) \text{SUBJECT >> FULL-INT} \\
(14) \text{English: SUBJECT >> FULL-INT} \\

In Italian, it is more important to satisfy FULL-INT, even if that means violating SUBJECT.
The idea is that both SUBJECT and FULL-INT are universal constraints on language, but languages just differ as to how important each is relative to the other. Interaction with “information status”

(16) a. Was geschah?
   ‘What happened?’

b. Es wurde schön getanzt. (answers general question about the event)
   it was beautifully danced.

c. # Schö n wurde getanzt.

d. * Schö n wurde es getanzt.

(17) a. Wie wurde getanzt?
   ‘How was the dancing?’

b. Es wurde SCHÖN getanzt. (conveys new information)

   c. SCHÖN wurde getanzt. (conveys new & unexpected information)

   d. * SCHÖN wurde es getanzt.

Where things go in the clause depends (in some languages) on its information status. German likes its new things to be at the left edge of the VP, and it likes its noteworthy things to be first in the clause.

So, we need two constraints to do this:

ALIGN-NEW→VP: An element carrying the feature [new] must appear at the left edge of VP.

ALIGN-NOTEWORTHY→CLAUSE: An element carrying the feature [noteworthy] must appear at the left edge of the clause.

The last constraint we need is:

MINPROJ: No maximal projections. (You get a star for every XP in the structure—so trees with less structures are preferred by this constraint).

Now, we can predict the German data from this ranking:

(18) ALIGN-NOTEWORTHY→CLAUSE >> ALIGN-NEW→VP >> MINPROJ >> SUBJ >> FULL-INT

The driving idea behind Optimality Theory (including OT in syntax) is that the constraints involved in grammar are simple and general. This means they are almost always not surface-true, but are only obeyed when it doesn’t violate a more important constraint.
One concern:
Wait—we’re accounting for 3–6 sentences with five constraints. Hmm. We have to be
careful that every new unexpected data point doesn’t lead to the postulation of a specific
higher-ranked constraint to cover that case. This is not a general complaint about OT, but
it is a warning to be responsible when constructing an analysis (like is required in any
framework).

Another, more specific concern:
Constraints are simple and general. Like ALIGN-NOTEWORTHY ➔ CLAUSE?
Is this constraint available in all languages?
Are there languages that have ALIGN-NOTEWORTHY ➔ VP ranked higher?
How many ALIGN-X ➔ Y type constraints are there?

Again, we have no a priori problem with constraints like this, but we need to keep an eye
on the big picture, and what the whole grammar is going to look like in the end.

Even now, OT is fairly new and under development, and OT syntax is even less fully
developed. Initially, we need to keep an open mind and see what we need to explain the
facts of language, without constraining ourselves too much by a priori notions of what
makes a good or a bad constraint. But once we have a foothold, we have to start thinking
about properties of “proper constraints.”

What kind of arguments are there for OT analyses?

Re-ranking typologies:
• The prediction you make when you posit a constraint in OT is that you will see languages
where it is ranked both above and below other constraints in the system. If you find them,
that supports the analysis. But you have to look. If only one permutation of the applicable
constraints is attested, your analysis is suspect.

“Emergence of the unmarked”:
The major difference between parameters (in a Principles & Parameters type system) and
low-ranked constraints is that, while a parameter is off, a low-ranked constraint is just
ranked low. That is, under the right circumstances, you may find that the lower-ranked
constraint still exerts influence (whereas this is unexpected if a parametric option is turned
off).

The role of economy.

OT is based on the idea of optimization, similar in concept to “economy.” The winning
candidate is the one which minimally violates the highest ranked constraints.

Optionality and ineffability.

In general, an OT system does not yield optionality. Some candidate will be the most
optimal, and it will be the (only) grammatical output. So, when it looks like there is
optionality, it is probably not really optionality—the usual approach is to see if they differ
in the input (in the information structure).

On the flip side of this, some candidate will be the most optimal—and it should be
grammatical. Yet, there are supposed to be sentences that you can’t generate. For example:

(22)  a. * Why did you wonder whether John left $t_1$?
    b. * What did the fact that John bought $t_1$ was brought to my attention.

There’s no good way to ask these questions in English. In a P&P approach, we attribute
this to an invalid movement, violating Subjacency (which is fatal).

But what if you start with this input and feed it to an OT system? One candidate will win,
it should be grammatical. What would that be?

In Prince & Smolensky (1993), one approach was considered: The null parse, which is a
candidate among the others. It is a total failure of faithfulness, but in cases where there
appears to be no grammatical output, what this means is that the null parse was the
winning candidate. However, this feels something like a hack.

In Legendre et al. (1998), another approach was considered: Rather than taking the
mapping between the underlying representation and the LF (interpretation) for given, we
assume that the underlying representation provides a target interpretation (why should take
scope in the matrix clause), but the LF interpretation might be one which does not
faithfully reproduce that. That is, the winning candidate for the input attributed to (22a) is
actually (23).

(23) Did you wonder [why John left $t_1$]?

Other possible approaches to optionality.

If there really appears to be optionality that can’t be explained in terms of properties of the
input, there are a couple of ways people have tried to analyze this.

One possibility is that the two candidates don’t differ in any way that is evaluated by the
constraints, but this is an unlikely possibility.
Another possibility is that the candidates have been evaluated by *different grammars*—for example, they might differ in register.

Another is that two constraints are *tied*—that is, violating constraint A is just as bad as violating constraint B, so either a *A or a *B candidate can be optimal.

The last option is somewhat similar to a tie, but is more powerful/灵活—*a partial ordering* that defines a set of rankings, any one of them may be used in a given evaluation. Our own Arto Anttila has studied this kind of interaction in phonology (analyzing variation even in the productions of a single speaker, historical changes, and dialects).

**Grimshaw (1997), Projections, heads, and optimality.**

Grimshaw’s main concern is *do-support* in English, which definitely has a “do it only if you need to” kind of flavor that lends itself to OT

Input: lexical heads, argument structure, tense, semantically meaningful auxiliaries (and information structure?)

Gen makes X'-compatible candidates

Inversion is possible and obligatory in questions, forbidden in non-questions:

(24) a. They will read some books.
   b. * Will they read some books.

(25) a. * Which books they will read?
   b. Which books will they read?

Grimshaw gets this as follows:

**OpSPEC:** Syntactic operators must be in specifier position
   (where they take scope) [cf. "[+wh]-CP"]

**ObHEAD:** A projection must have a head (filled at least with a trace)

**STAY:** Traces are not allowed (don’t move).

In a declarative, we can get away with an IP, no movement involved (abstracting away from the VP-internal subject hypothesis for now):

```
(26) IP
   they
   I
   I'
   VP
   will
   V
   read
   DP
   some books
```

With a *wh*-phrase, which is an *operator*, things change. A wh-operator must take scope over the proposition, so it must appear in a specifier position. There aren’t any specifiers open in (26), so we need to create a new projection (CP) into whose specifier the operator can move.

```
(27) CP
   [which books]i
   C'
   IP
   they
   I'
   I
   will
   V
   t,
   read
   VP
   t,
   i
```

So, to satisfy both *ObHEAD* and *OpSPEC*, we have to move I to C.

```
(28) CP
   [which books]i
   C'
   C
   IP
   they
   I'
   I
   will
   t,j
   C
   VP
   t,l
   read
   t,i
```
You can’t just invert freely, though, because of \textit{stay}. That is violated in (28), but the fact that this candidate wins tells us that

\begin{equation}
\text{ObHead, OpSpec} \gg \text{stay}
\end{equation}

\text{OpSpec} \gg \text{stay} \\
wh\text{-phrases have to move to CP (>>not moving).}
\text{ObHead} \gg \text{stay} \\
if you have a CP, you invert (>>not moving).

So: Trees are only as big as they have to be. \textbf{Matrix declaratives are IPs.}

\textit{do-support:}

Like expletives \textit{(it, there)}, do is an empty element—it doesn’t contribute to the interpretation, and so if you have it, that will be a violation of \textit{full-int}.

\begin{equation}
\begin{array}{ll}
\text{a.} & \text{What she said?} \\
\text{b.} & \text{What did she say?} \\
\text{c.} & \text{What she did say?}
\end{array}
\end{equation}

But if there is no other auxiliary, we seem to have to use \textit{do}. If there is no other auxiliary, the head of IP will be empty, violating \textit{ObHead}. If there is a CP (required for the operator), the head of CP will be empty, *\textit{ObHead}. Although \textit{full-int} dictates against inserting \textit{do}, it seems that \textit{ObHead} is more important.

Why not invert the verb? (Instead of inserting \textit{do})

\textbf{No-lex-mvt: No moving lexical verbs (non-auxiliaries).}

\begin{equation}
\begin{array}{ll}
\text{a.} & \text{What will they read?} \\
\text{b.} & \text{* What do they will read?} \\
\text{c.} & \text{* What will they do read?}
\end{array}
\end{equation}

Inverting \textit{will} satisfies \textit{ObHead} for IP and CP, and inserting \textit{do} doesn’t change that, but it \textit{does} add an extra violation of \textit{full-int}, which we can do without.

\textbf{Subject wh-questions don’t invert.}

\begin{equation}
\begin{array}{ll}
\text{a.} & \text{Who read LGB?} \\
\text{b.} & \text{* Who did read LGB?}
\end{array}
\end{equation}

Why? \textit{Who} can stay in SpecIP and satisfy \textit{OpSpec}. There is no need to build a CP for it, it’s already in a scope-capable Spec position. Building a CP just entails further violations of \textit{stay} (and perhaps \textit{full-int} too).

In embedded clauses (of all kinds), there is no inversion in English. Grimshaw tackles this by positing an even-higher-ranked constraint (called \textit{Proj-Prin} for historical reasons):

\begin{equation}
\text{Proj-Prin: No adjunction to subordinate clauses, and no movement to the head of a subordinate clause.}
\end{equation}

That forbids movement to C in an embedded clause. Ranking \textit{Proj-Prin} \gg \textit{ObHead} means that in embedded clauses, we just have to live with an empty-headed C.

Grimshaw goes on to deal with various other aspects of inversion and non-inversion in English (including negative preposing), but we’ve got a good enough idea now.


We seem to have made some significant progress in the domain of phonology by considering things in terms of optimizing with general, violable, ranked constraints.

It would be interesting if we found that syntax too could be more insightfully described this way—it would suggest that the human language faculty is at its root a kind of optimizer.

Pesetsky starts by reviewing some major concepts of syntax—\textit{the idea being that if OT is to be a theory of syntax (or if anything is going to be a theory of syntax), these are things it will need to deal with. It is also there to serve as an introduction to syntax for those phonologists who are reading the book. We will skip over this stuff here.}

“\textit{Clash & Crash}” and \textit{ineffability}. There is reason to think syntax \textit{doesn’t} act as an OT system predicts. In particular, it looks very much like we find particular inputs which have \textit{no} corresponding output.
Pesetsky also sets up a more complex example that sets up a fight between the Doubly-Filled Comp Filter, the Case Filter, and the Obligatory-Wh-movement constraints:

(35) a. Mary wonders [Bill to buy which book at the store].
    satisfies Case filter, satisfies DFC, violates Wh-movement.

b. Mary wonders [which book for Bill to buy at the store].
    satisfies Case filter, violates DFC, satisfies Wh-movement.

c. Mary wonders [which book Bill to buy at the store].
    violates Case filter, satisfies DFC, satisfies Wh-movement.

So, no constraint “gives way” to the others—they all appear inviolable. Yet:

(36) Mary wonders [which book PRO to buy at the store].

This one is fine because we can make the Case Filter irrelevant, other two are satisfied.

What this means is that if we want to do syntax in OT, we have to do some work, since things look so much like interactions of inviolable constraints. There are things we can say, but we need to be careful that we don’t have to complicate the system beyond the complexity of the facts themselves in order to avoid a “Clash & Crash” model.

However, there may still be a place for OT in syntax even if it isn’t in the building of trees. Suppose that trees are built just as in the pre-OT days, like GB theory, for example, or Minimalist program analyses.

There may still be something OT-like in the pronunciation of these structures. In particular, consider movement.

There is some reason to think that when you move something, you create two positions for the same thing. The GB/MPLT answer to this situation is to say that you pronounce the top position. But that isn’t required.

Why we think that there’s something at the bottom of the chain...

(37) a. He criticized the picture of John. (*BT(C) if he is John)
    (coreference ok)

b. His former teacher criticized the picture of John.

(38) a. Which picture of John did he criticize? (*BT(C) if he is John)
    (coreference ok)

b. Which picture of John did his former teacher criticize? (coreference ok)

(39) a. Which picture of John did he criticize which picture of John?

b. Which picture of John did his former teacher criticize which picture of John?

(40) SILENT TRACE: Don’t pronounce the traces of a moved constituent.

Relative clauses and RECOVERABILITY.

(41) a. * the person [whom that Mary invited — to the party]

b. the person [whom that Mary invited — to the party]

c. the person [whom that Mary invited — to the party]

d. the person [whom that Mary invited — to the party]

(42) a. * the person [to whom that Mary spoke — at the party]

b. the person [to whom that Mary spoke — at the party]

c. * the person [to whom that Mary spoke — at the party]

d. * the person [to whom that Mary spoke — at the party]

(43) a. Mary thinks that Peter is hungry.

b. Mary thinks that Peter is hungry.

The idea is that to whom has some content, whereas whom doesn’t have any (irrecoverable) content.

(44) RECOVERABILITY: The semantic content of unpronounced elements must be recoverable from local context.

French: Complementizer must be pronounced

(45) a. Je crois [CP que Pierre a faim].
    I believe that P is hungry

b. * Je crois [CP que Pierre a faim].

(46) a. * l’homme [CP qui que je connais].

b. * l’homme [CP qui que je connais].

c. l’homme [CP qui que je connais].

d. * l’homme [CP qui que je connais].
    ‘the man I know.’

Proposal: In French we see the operation of LEFTEDGE(CP).

(47) LEFTEDGE(CP): The first pronounced word in CP must be the complementizer.

…except where RECOVERABILITY is at risk; RECOVERABILITY >> LEFTEDGE(CP)
(48) a. * l’homme [cp avec qui que j’ai dansé]  
   b. l’homme [cp avec qui que j’ai dansé]  
   c. * l’homme [cp avec qui que j’ai dansé]  
   d. * l’homme [cp avec qui que j’ai dansé]  

   ‘the man with whom I danced’

RECOVERABILITY favors (48a–b) over (48c–d), but what favors (48b) over (48a)?

(49) TELEGRAPH: Do not pronounce function words (e.g., complementizers)

The ranking RECOVERABILITY >> LEFT_EDGE(CP) >> TELEGRAPH gets us French.

(50) French complex relative clauses

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(51) French declarative clauses

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. Je crois [cp que Pierre a faim].</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. Je crois [cp que Pierre a faim].</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(52) French simple relative clauses

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. l’homme [cp qui que je connais]</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. l’homme [cp qui que je connais]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c. * l’homme [cp qui que je connais]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d. * l’homme [cp qui que je connais]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(53) French embedded questions (irrecoverable wh-words)

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

“Deletion is obligatory, up to recoverability.”—sounds like OT.

The way OT describes different languages is by re-ranking constraints. We have three constraints here, in principle six different strict rankings.

But: English seems to be analyzable with these constraints but with LE(CP) and Tel tied.

(54) English declarative complements

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. I believe [cp that Peter is hungry].</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. * I believe [cp that Peter is hungry].</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(55) English simple relative clauses

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(56) English complex relative clauses

<table>
<thead>
<tr>
<th></th>
<th>RECOVER</th>
<th>LE(CP)</th>
<th>TELEGRAPH</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Look, we got the cluster of differences between French and English by just de-ranking LE(CP) and Tel.

Pronunciation of traces—Are there cases where SILENT TRACE is violated?

It is actually well-known that (in conversational styles of English), you can “resume” an illegal move with a pronoun. Why?

(57) a. * This is the guy who I thought that [[ Mary and — ]] were going to the movies.  
   b. * This is the guy who I thought that [[ Mary and him ]] were going to the movies.

(58) a. * This is the guy who I wondered [[ whether — was going to the movies ]].  
   b. * This is the guy who I wondered [[ whether he was going to the movies ]].

(59) a. * There is one worker who the company fired the employee [[ that treated — badly ]].  
   b. * There is one worker who the company fired the employee [[ that treated him badly ]].

And you can’t do it if it isn’t required—
(60)  a. * This is the guy who I like him.
    b. * This is the guy that he likes coffee.

Why a pronoun? Maybe a silent trace would violate whatever is responsible for island conditions, but a pronoun is closer to silent than the whole wh-phrase—we’ve minimally violated silent trace.

Short discussion of some kid French data that makes it look like, in their relative clauses, they have LE(CP) ranked over silent trace (the reverse of the adult ranking)—they always preserved the complementizer que and put it first, but where recoverability would be put at issue, they would pronounce the wh-phrase in the base position.

All that glitters may not be OT…

Pesetsky puts in a last pessimistic word about the view that all is OT—that is, that, while we seem to find OT-like interactions in the pronunciation, there is still a Clash & Crash aspect to the system.

Point one: We saw that resumptive pronouns can save island violations in relative clauses. So, perhaps what’s wrong with extraction out of islands is that they lose to a competitor with a resumptive (hence no longer a Clash & Crash case)—except that adverbs have no resumptives, and they are subject to island constraints too.

Point two: Even where the resumptives save the structure, the result is marginal.

“The system—as it is set up, and as it functions correctly in many domains—picks winners and losers. There are no silver or bronze medals in OT. Consequently, when we find solver and bronze medalists among possible pronunciations for certain structures, we can be certain that there are non-OT aspects to the interaction of linguistic constraints.”