What Syntax is For
Pre-Talk
Andy Barss

Tom: Why do we even have syntactic capacity left over?

What would language look like without syntax?

This is a language without syntax (click on the image):

Optimization: Making the Most of the Least

Andy Barss, Linguistics & CogSci
November 1, 2007

Suppose Minimalism is true. Then, major aspects of language follow a Minimization pattern: structures/relations/processes/rule applications
will be shortest/smallest/least/maximally simple
for whatever task is at hand.

(Example tasks: given some proposition p, express p, using a set of lexical items L, that involves the least needed computation).

A question that we need to address while exploring minimization effects:

How does the system doing the minimizing “know” what the minimal solution to the task at hand is? (Competence/ performance distinction on steroids)

Below I’ll:

show some cases where minimizing/following least action doesn’t require a global comparison across alternatives, but “emerges” from a straightforward

show other cases where two analyses differ in their need for explicit global comparison of alternatives.

raise the issue of how interesting LAP becomes if everything works this way.

Minimization: A Non-Linguistic Example (with an easy solution)

You have a full pitcher and an empty goblet.

Task: Determine the minimal pour from the pitcher that will fill the goblet to a particular point.

“Minimal” solution is reached first in an easy way: you can’t pour three ounces in without pouring two ounces in first.

But other cases aren’t so simple. Below I’ll discuss several examples from the linguistics world, some of which are like the pitcher/goblet example, some of which are maybe more mysterious and peculiar.

Quote from Feynman: “Does [the electron] smell the nearby paths, and check them against each other? The answer is, yes, it does, in a way." ”

Chomsky 1986: Proposed a Least Effort Principle for syntactic movement operations.

First, some background.

Raising = displacement of an NP from its semantic argument position to a structurally higher one which gives it Case.

(1) It seems [he is happy]
(2) It seems [he has released the doves]

(1′) He seems [___ to be happy]
(2′) He seems [___ to have released the doves]

(1′′) * It seems [he to be happy]
(2′′) * It seems [he to have released the doves]

Two classic questions:
A: Why does this displacement happen? (in 1′,2′)
B: Why is it obligatory? (1′, 2′ vs. 1′′, 2′′)

(1) It seems [he is happy]
(2) It seems [he has released the doves]

(1′) He seems [___ to be happy]
(2′) He seems [___ to have released the doves]

(1′′) * It seems [he to be happy]
(2′′) * It seems [he to have released the doves]

Answers to A & B:

All NPs need Case.
Finite verbs (*seems, is, has*) can provide it to their subjects.
Infinitive verbs can't. hence, (1′′), (2′′) are *, due to violation of this "Case Requirement".

Classic GB picture: free generation of structures, subsequent filtration by required principles. **For any sentence S, S is grammatical iff it does not violate any grammatical principle/filter.**

(1) It seems [he is happy]
(1′) He seems [___ to be happy]
(1′′) * It seems [he to be happy]

New puzzle: Why is (3) bad?

(3) * He seems [___ is happy]

New type of answer: (3) is bad because (1) is fine.

(1) It seems [he is happy]

(3) * He seems [___ is happy]

(1) and (3) **compete**. Each is viewed as a different way of composing a sentence from the fundamental parts *he, seem, is, happy*.

(1) does the job without movement. (3) involves movement.

Because (1) is grammatical, the movement in (3) is unnecessary.

That is ... (3) violates Least Effort: Move only if necessary.

This seemed a RADICAL IDEA at the time.

Almost immediate objection: this changes what "grammatical" means, giving it a global flavor.

How does the system "compute" the class of relevant competitors and their derivations, and then count the number of movement in each, to converge on the optimal (= grammatical) solution?

Chomsky 2000: Construct the sentence from the bottom up, one Merge following another.

(3) is out now because it cannot be constructed to begin with. It does not directly compete with (1).

Derivation goes like this:
$\oplus$ = Merges (combines into a new constituent with)

a) is $_{[NOM]}$ $\oplus$ happy

b) he $_{[NOM]}$ $\oplus$ [is $_{[NOM]}$ happy]

c) he $_{[\cancel{NOM}\checkmark]}$ $\oplus$ [is $_{[\cancel{NOM}\checkmark]}$ happy]

d) seems $_{[NOM]}$ $\oplus$ [he $_{[\cancel{NOM}\checkmark]}$ [is $_{[\cancel{NOM}\checkmark]}$ happy]]

e) it $_{[NOM]}$ $\oplus$ seems $_{[NOM]}$ [he $_{[\cancel{NOM}\checkmark]}$ [is $_{[\cancel{NOM}\checkmark]}$ happy]]

f) it $_{[\cancel{NOM}\checkmark]}$ $\oplus$ seems $_{[\cancel{NOM}\checkmark]}$ [he $_{[\cancel{NOM}\checkmark]}$ [is $_{[\cancel{NOM}\checkmark]}$ happy]]

No competition, no choice at point (d). Case on *he* is deactivated, cannot be "seen" by higher verb *seems*.

So, on this new view, it's much like the goblet problem I began with.

Sum so far:
two cases of least action/minimization
first has an apparent “ procedural” acount (goblet)
second has two accounts:
competition within a class of competitors, all of which need to be computed and compared -->
global quality, much criticism

local, procedural account

Case 2: Attachment Preferences in Sentence Comprehension.

3-way Ambiguity:

"I saw the spy with the binoculars"
a) I saw the spy, and I used the binoculars to do so
b) I saw that spy who had the binoculars (not the other spy, who didn't)
c) I saw the spy in possession of the binocs (akin to "I saw her with a cigarette")

(a) vs. (b) subject to many experiments.

Classic result: Subjects (in English) prefer the (a) reading to the (b) reading.

Frazier: This is evidence of a Minimal Attachment Strategy in parsing:

Given a new constituent X, attach X into the existing tree in whatever way will minimize added structure.

(a) reading: $[_{VP}$ saw $[_{NP}$ the spy$]$ $]$ --> $[_{VP}$ saw $[_{NP}$ the spy$]$ PP $]$

(b) reading: $[_{VP}$ saw $[_{NP}$ the spy$]$ $]$ –> $[_{VP}$ saw $[_{NP}$ $[_{NP}$ the spy$]$ PP $]$ $]$

(b) is more complex than (a), so (a) is chosen.

Conceptual Puzzler:

If the system is powerful enough to build BOTH structures and compute their differences to determine which is simpler ...

why is it burdened with the MAP to begin with? Doesn't this argument shoot itself in the foot?

Analogy: You're tired, need to get to the summit of the mountain in an energy efficient way. So, you hike all the possible paths, then decide after the fact that one is best; so you pick that one and go back up to the summit.

An alternative account

The (a) reading is consistent with there being just one spy.
The (b) reading requires that there be at least two spies.

(b) is more presuppositionally complex than (a).

Start with no presupps.

Add as needed.

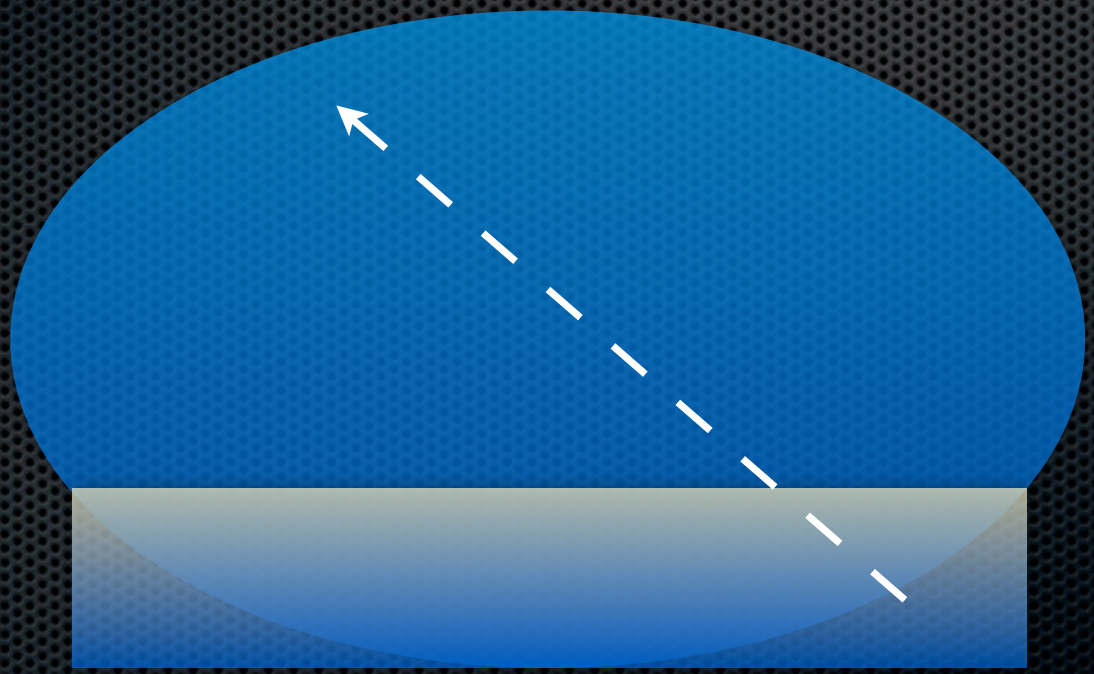
You get to one before you get to two.

So, you prefer reading (a), because it's got fewer presuppositions.

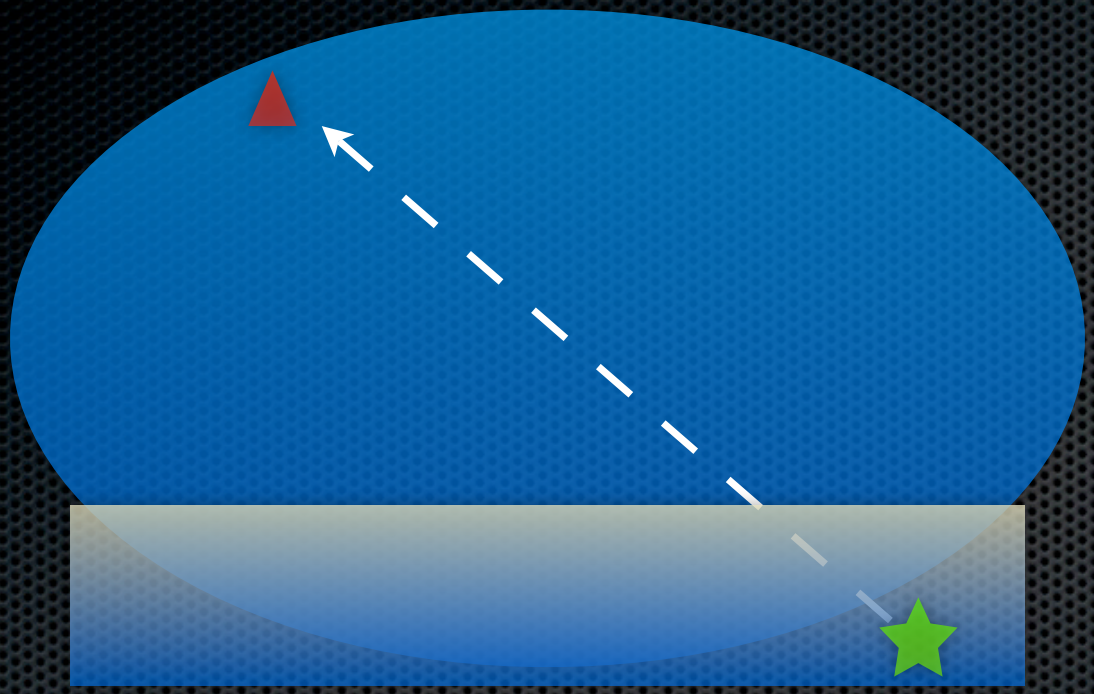
You do NOT need to compare (a) to (b) – structurally or presuppositionally. **You just get to (a) FIRST**.

This makes the optimization again akin to the goblet problem.

Recall the Raising example, and the debate over Global vs. local minimization.
A parallel case: dog navigation (fetching sticks).

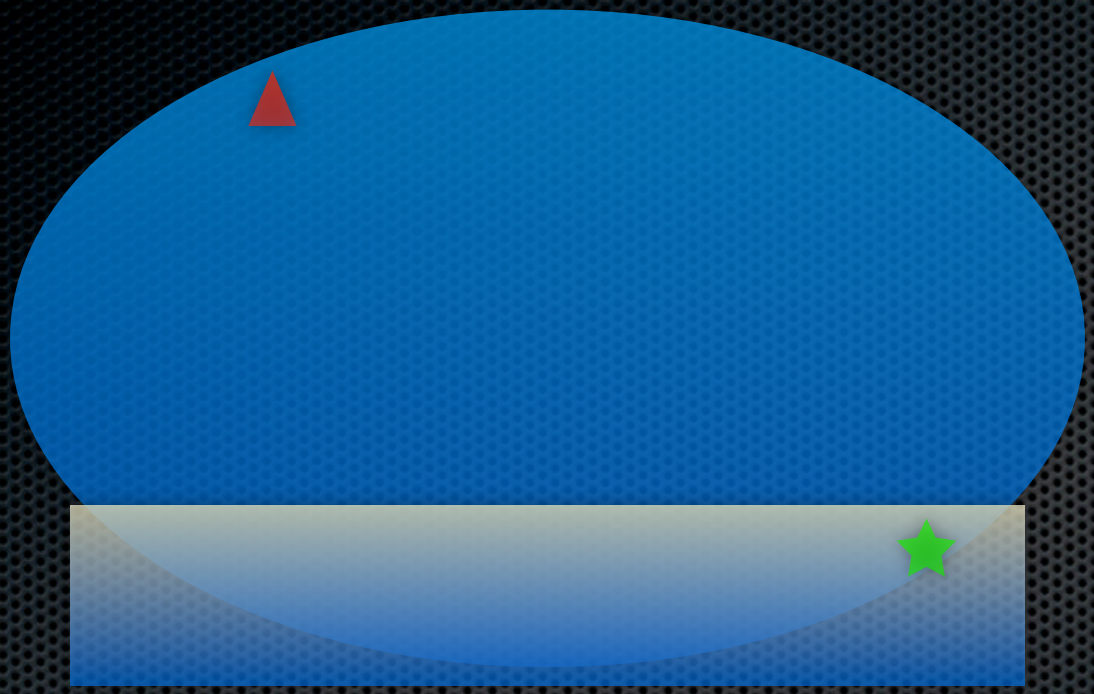


Meet Elvis. He's a welsh corgi.
He also can do calculus.

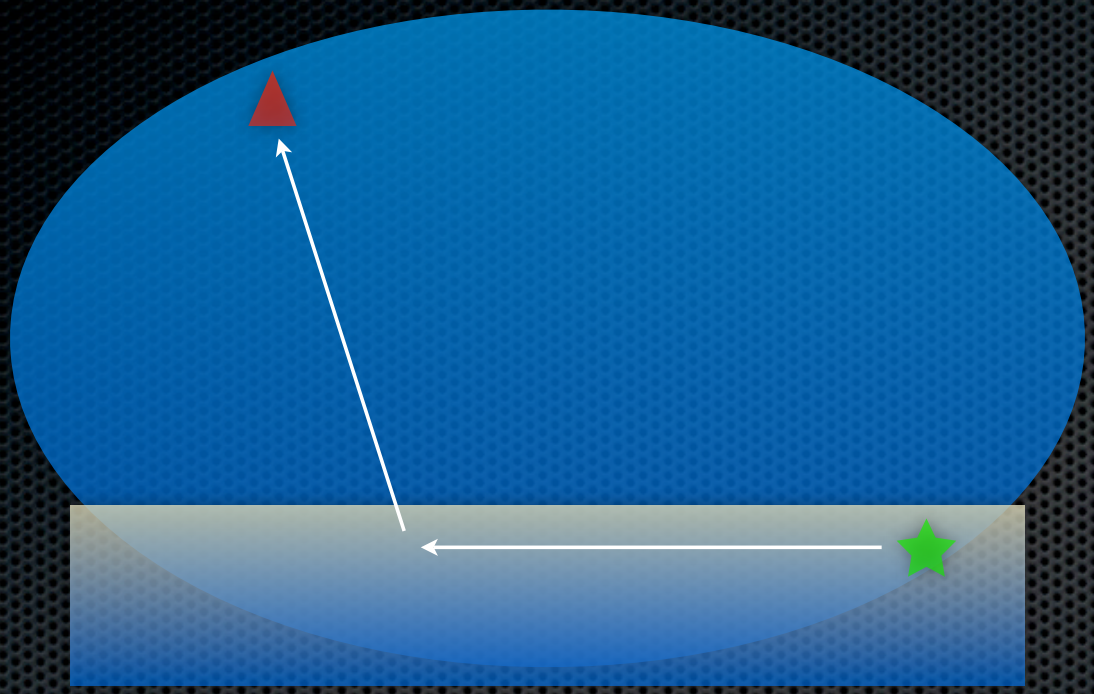


Elvis will chase the stick. What pathway does he follow?

NOT the shortest distance (i.e. the stick's trajectory).



What pathway does he follow? Crucially:
along the beach is faster, but oblique to the stick.
Into the water allows a direct approach to stick, but is slower.
What he does ...
Partway straight along the beach.
Then into the water and straight out to the stick.



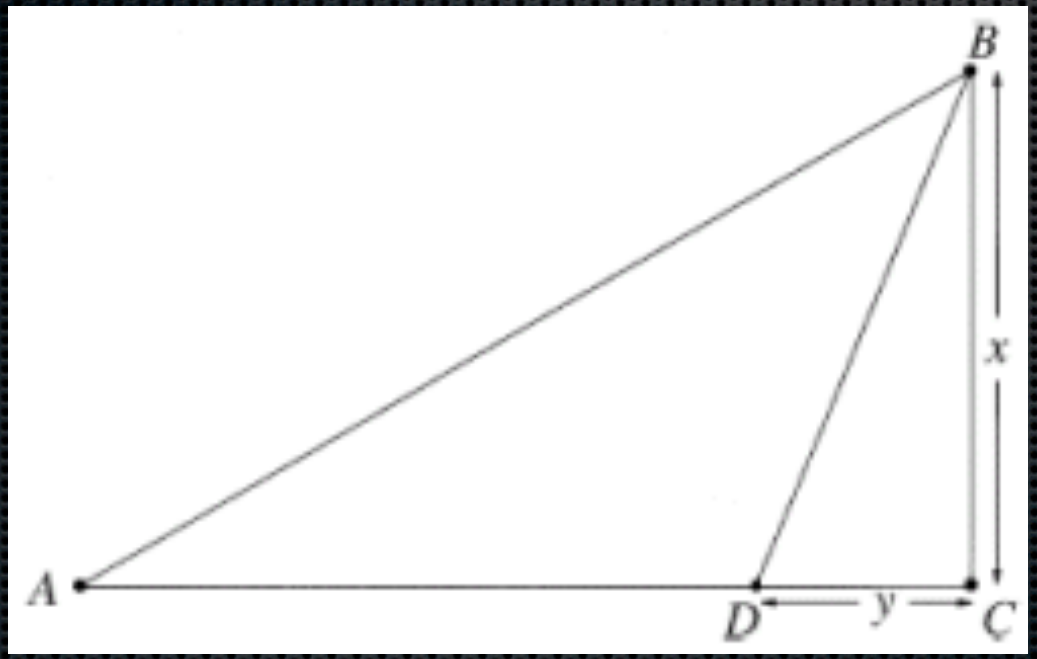
Partway straight along the beach.
Then into the water and straight out to the stick.

There's an infinite number of such routes.
Elvis picks the most efficient one: Least Time/Least Action.
HOW DOES HE DO THAT?

Original account: Global computation (Pennings 2003).
Later: Series of local computations, each one assessing spped of approach to ball. Jump into water when

$$SA_{beach} = SA_{water}$$

(Gallego, J., and P. Perruchet. 2006)



Still Later: No, it really is global (Pennings 200x). Backtracking when trajectory begins in water.

Case 4: Language Acquisition and parameter setting/rule choice.

The fundamental problem you're familiar with:
languages vary, so do grammars
children must figure out their adult grammar from input
input is positive only

Puzzle (origins in Gold 1967, brought into linguistics by Dell 1981, widely adverttised by Manzini and Wexler 1987, Wexler 1994, and many others): how to converge on the right grammar/rule system when input is compatible with MORE THAN ONE GRAMMAR CHOICE in the hypothesis space?

Severe version of this problem:

Grammars G1 and G2 both generate form A
Grammar G1 does not generate form B, but G2 does

Child hears form A, never hears “not B”.

So, G2 generates a superset of the forms G1 generates.

How does the learner converge on G1, and avoid G2?

Dell; Berwick; Manzini/Wexler: Learner follows a Subset Principle:

Given a choice in that situation, always choose the grammar that generates the smaller language.

So ... learner doesn't choose the "larger" grammar, because she is required to follow the strategy of not doing that.

Whew!

Even if we sidestep the apparent circularity in this ‘explanation’,

how does the child decide what the bigger grammar ***is***?

I can tell you, but I'm out of time! Sorry!