Recursion in planning and language

Cem Bozşahin

bozsahin@metu.edu.tr

Cognitive Science Department, The Informatics Institute,
Middle East Technical University (METU/ODTÜ), Ankara

Cognitive Science Seminars, Yeditepe University

February 21, 2014

*Supported by the GRAMPLUS Project at Edinburgh Univ., EU FP7 Ideas Grant 249520

- Language syntax

  and human planning abilities seem disparate from external view.
- Their internal mechanisms are same

  at some level of abstraction.
- That abstraction, automaton, seems to be just right to explain planning in humans and chimpanzees.

  With some unique restrictions.
- Is recursion the key distinction? probably not

- Natural recursion in syntax, or recursion by linguistic value, is not syntactic in nature but semantic.
- Syntax-specific recursion is not recursion by name as the term is understood in AI and theoretical computer science.
- Recursion by name is probably not natural.
- Natural recursion, or recursion by value, is not species-specific.
- Human recursion is not syntax-specific, although the values it operates on are most likely domain-specific, including those for syntax.
- Syntax seems to require no more (and no less) than the resource management mechanisms of an embedded push-down automaton (EPDA).
- We can conceive EPDA as a common automata-theoretic substrate for syntax, collaborative planning, i-intentions, and we-intentions (Searle, 1990).
- What evidence do we have that language (or syntax) was there first in the exploits of a potentially common computational substrate, i.e., of a class of automata?

- Human mind
- Animal mind
- Artificial mind

- Artificial models of human or animal mind
- Computer is the only tool in human history that can change itself

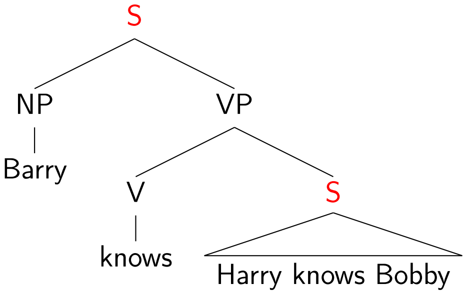
What is at stake in studying recursion and mind?

Hauser et al. (2002); Fitch et al. (2005):

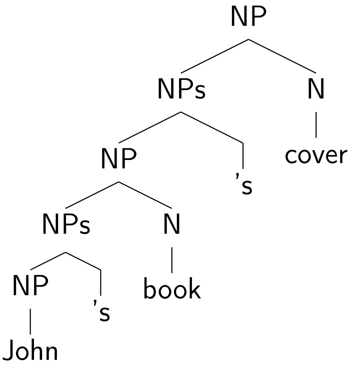
Syntactic recursion is the most unique human capacity.

A one-time cogsci conference was dedicated to recursion in humans:

Speas and Roeper (2009)

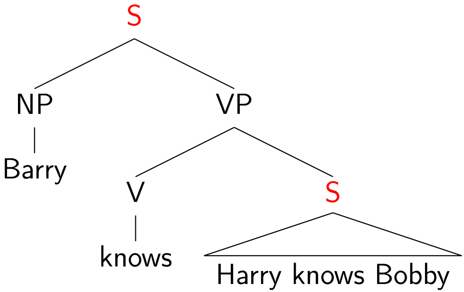


Barry knows Harry knows Jerry knows Bobby.

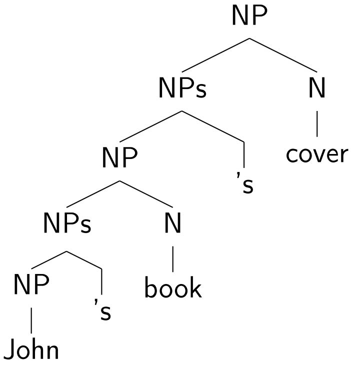


NP
NPs
N
NP
's
cover
NPs
N
NP
's
book
John

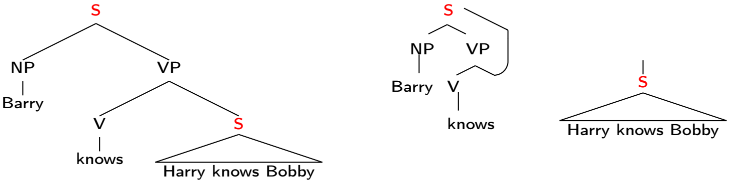
- All these examples are recursion by value
- Another instance of a predicate is taken as a value.



Barry knows Harry knows Jerry knows Bobby.


NP
NPs
N
cover
NP
's
NPs
N
book
NP
's
John

- All these examples are recursion by value
- Another instance of a predicate is taken as a value.
- recursion by name is different than recursion by value.



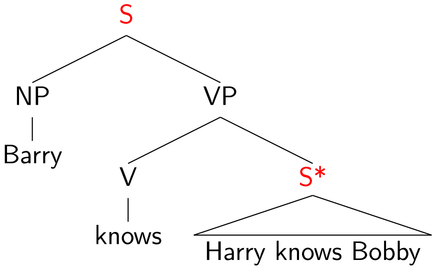
One predication of know on the right,

rather than two, as in recursion by value.

Barry knows Barry knows Barry knows Harry knows Bobby.

Not the native speakers' understanding of knowers and knowees.

Some theories take care of the empirical problem with a single tree:



Two distinct S nodes. (LTAG; Joshi and Schabes, 1992)

Two distinct combination: substitution and adjunction.

The problem is not trees, but not making the distinction.

Some definitions with or without recursion by name.

- Tree: (i) a node called *root* is a tree, denoted as T(root). (ii) The subtrees of a tree T, T(T_1, T_2, $\cdots$ T_m), are partitioned into T_1, T_2, $\cdots$ T_m, where each T_i is a tree.
- Tree: Any tree is a collection of *nested sets*. A collection of non-empty sets is nested if, given any pair X, Y of the sets, either $X \subseteq Y$ or $X \supseteq Y$ or X and Y are disjoint.
  Knuth (1968: 314)

- $\mathbf{Y} \overset{\text{def}}{=} \lambda h.(\lambda x.h(x\ x))(\lambda x.h(x\ x))$ Curry and Feys (1958)
- $\mathbf{U} \overset{\text{def}}{=} (\lambda x\lambda y.y(xxy))(\lambda x\lambda y.y(xxy))$ Turing (1937)

- $\mathbf{Y}h = h\ (\mathbf{Y}h) = h\ (h\ (\mathbf{Y}h)) = \cdots$
- $\mathbf{U}h = h\ (\mathbf{U}h) = h\ (h\ (\mathbf{U}h)) = \cdots$

- $\mathit{fib}(n) = \mathit{fib}(n-1) + \mathit{fib}(n-2) \qquad \mathit{fib}(0) = 0, \mathit{fib}(1) = 1$
- Let
  $\mathit{fib} = \lambda n.\text{if } (n == 0)\ 0 \text{ else if } (n == 1)\ 1 \text{ else } \mathit{fib}(n-1) + \mathit{fib}(n-2)$
- Let
  $h = \lambda f.\lambda n.\text{if } (n == 0)\ 0 \text{ else if } (n == 1)\ 1 \text{ else } f(n-1) + f(n-2)$
  Then $\mathit{fib} = h\ \mathit{fib}$ because $\mathit{fib}\ n = h\ \mathit{fib}\ n,\ \forall n \geq 0$
  And $\mathit{fib} = \mathbf{Y} h$ because $\mathit{fib}\ n = \mathbf{Y} h\, n, \forall n$, and $\mathbf{Y} x = x(\mathbf{Y} x), \forall x$

- Why unnatural? It is not finitely typeable

  read: not finitely representable.

- All human knowledge we can think of is finitely representable.

- Human words are finitely representable even if they take same argument types. (by value)

  See Bozsahin (2012) for impossible words.
- Human plans are finitely representable too.

Interim summary:

- Syntactic recursion in language is not recursion by name.
- Syntactic recursion in programming is recursion by name.

- Recursion by value is limited in humans.
- Recursion by value is attested in non-humans.
- Recursion by value is attested in domains other than language.

- Reactive planning (finite-state)
- Instrumental planning (push-down store of plans, PDA)
- Collaborative planning (embedded push-down store, EPDA)

Lochbaum (1998); Bratman (1992); Petrick and Bacchus (2002); Steedman and Petrick (2007); Grosz and Kraus (1993); Grosz et al. (1999)

- I-intentions (PDA)
- We-intentions (Searle, 1990) (embedded PDA)

- Can multi-agent planning be sum of plans of single agents?

- Searle's (1990) scurrying from rain in the park and corps de ballet in the rain.
- The embedded push-downs of counter-attack in football as common goal
  - A: I chase the ball
  - B: I chase the ball
  - C: I keep close to chasers to join forces
  - D: I keep a watch behind my back
  - E: I keep a watch behind D's back
- No one singly "executes" counter-attack!
- It makes a difference to have a simple stack or stack of stacks (cooperation)

- Finite-state plans: whatever can be afforded by finite history and non-embedded behavior.
- We cannot capture a case where separate actions of an agent match step by step,
- or plans that wait for other plans.

  ex: picking up flowers on the return path of laying them on the floor, ensuring same amount has been picked.

- Instrumental plans: limited context dependency
  In animals and humans
- Julian Jaynes (1976): hapless chimpanzee in captivity.
- Plans that contain other plans and other agents.
- Tomasello et al. (2003): Chimpanzees might have a mind but cannot embed minds within minds (no recursion?)
- Formalizing the dependencies in terms of grammars and automata.

- Plans within plans? Jaynes (1976: 219)

$$\begin{array}{lcl} \text{S} & \rightarrow & \text{FillWater LureKeeper Spit} \\ \text{LureKeeper} & \rightarrow & \text{Coax} \mid \text{Hail} \\ \text{Coax} & \rightarrow & \text{Stalk}\ \ \text{Coax} \mid \text{AskBanana} \end{array}$$

- Spit depends on keeper,
- Coax might fail (spit no more part of this plan)

- External view: finite-state dependencies
- Internal view (planner): contextualized dependencies
- Truly context-free recursion in animals? Probably not (Van Heijningen et al., 2009)
- equiv. to having same dependencies in external and internal view.

- Hauser et al. (2002); Fitch et al. (2005) do not deny non-syntactic recursion in humans

  ((((the hole) in the tree) in the glade) by the stream)
- A closer look at non-syntactic recursion shows striking similarity to language.
- 'Syntax is most unique human capacity' argument weakens.
- A certain kind of recursion seems uniquely human.

(Scurry-in-rain grammar)

$$\begin{array}{lcl} S_i & \rightarrow & \alpha_i \mid A_i \\ A_1 & \rightarrow & \text{run, and do } S_1\text{'s work} \\ \vdots & & \\ A_n & \rightarrow & \text{run, and do } S_n\text{'s work} \end{array}$$

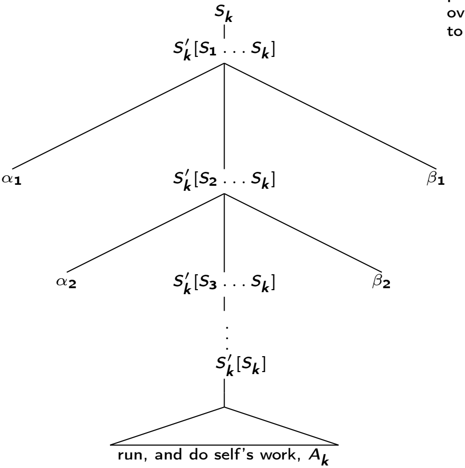
where α_i is a plan with a base case A_i

(Dance-in-rain grammar)

$$
\begin{array}{lcll}
S_i & \rightarrow & S'_i[\pi(S_1, .., S_n)] & \pi(x)\text{: a permutation of } x \\
S'_i[S_j \ldots] & \rightarrow & \alpha_j \, S'_i[\ldots] \, \beta_j & \\
\vdots & & & \\
S'_i[S_i] & \rightarrow & \text{run} & \\
 & & \text{and do } S_i\text{'s work } A_i &
\end{array}
$$

right-hand sides: plans and intentions.
actions and knowledge states are members of RHSs.

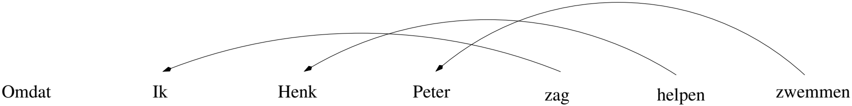
Participant k's grammar for collaborative dancing in the rain.



personal grammar: '$S'_i[S_i]$' rules.
overall collaboration: top rule common to everyone.

α, β contextualize knowledge states and actions (not actions of k)

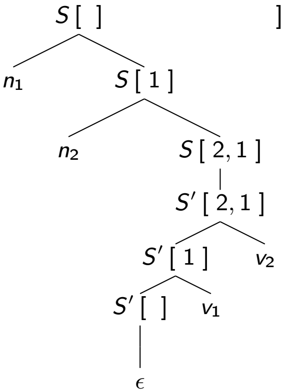
- Such grammars are linear-indexed. (LIG)
- They can only handle human NL at the upper limit.
- And correspond to a stack of stacks as automata.
- Embedded Push-down Automata Vijay-Shanker (1987); Joshi (1990)



‘because I saw Henk help Peter swim ...’

Dutch and Swiss German style dependencies formalized:

$$
\begin{array}{rcl}
S_{[\ldots]} & \rightarrow & n_i \; S_{[i\ldots]} \\
S_{[\ldots]} & \rightarrow & S'_{[\ldots]} \\
S'_{[i\ldots]} & \rightarrow & S'_{[\ldots]} \; v_i \\
S'_{[\,]} & \rightarrow & \epsilon
\end{array}
$$


$S[\ \ ]$
$]$
n_1
$S[\,1\,]$
n_2
$S[\,2,1\,]$
$S'[\,2,1\,]$
$S'[\,1\,]$
v_2
$S'[\ \ ]$
v_1
ϵ

- This automata has predictive power:
- $\{a^n b^n c^n \mid n \geq 0\}$ is strictly not context-free but LIG
- no such grammars for $\{www \mid w \in \{a,b,c\}^*\}$,
  or for $\{w \mid w \in \{a,b,c\}^*$ and $|w|_a = |w|_b = |w|_c\}$,
  $\{a^n b^n c^n d^n e^n \mid n \geq 0\}$ (not a linear metric)

- Instrumental plans and i-intentions: PDA
- Collaborative plans and we-intentions: EPDA
- Syntactic capacity in the limit: EPDA

Temporal integration is not found exclusively in language; the coordination of leg movements in insects, the song of birds, the control of trotting and pacing in a gaited horse, the rat running the maze, the architect designing a house, and the carpenter sawing a board present a problem of sequences of action which cannot be explained in terms of successions of external stimuli. Lashley (1951: 113)

- Some internal mechanism seems to be at work

  for planning and language.
- Linear-indexed grammar and associated automata (EPDA) has been the most explicit proposal what that mechanism might be.
- Not reasoning by analogy: we want to understand limits of natural computation by the mind by explicit proposals for its mechanism.
- Understanding what added explanation can be brought in by a class of automata.
- Such mechanisms seem only depend on recursion by value (natural recursion?)

- Back to Jaynes: consciousness emerged much later than language.
- It taps onto same parts of brain

  and for the same resource as language and planning: combinatory competence.

Conclusion

- Humans appear to be uniquely capable of recursion by value, of the kind that can be afforded by a stack of stacks.
- Various predictions about syntax and other cognitive processes follow from an automata-theoretic way of thinking about them.
- The question remains:

  What evidence do we have that language (or syntax) was there first in the exploits of a potentially common computational substrate, i.e., of a class of automata?

thanks

- Thank you!
- PT-AI reviewers
- Julian Bradfield
- Aravind Joshi
- Simon Kirby
- Umut Özge
- Geoffrey K. Pullum
- Mark Steedman
- Language Evolution and Computation Research Unit (LEC) at Edinburgh Univ.
- GRAMPLUS Project at Edinburgh Univ.

Bozsahin, C. (2012). *Combinatory Linguistics*. De Gruyter Mouton, Berlin/Boston.

Bratman, M. E. (1992). Shared cooperative activity. *The Philosophical Review*, 101(2):327–341.

Curry, H. B. and Feys, R. (1958). *Combinatory Logic*. North-Holland, Amsterdam.

Fitch, T., Hauser, M., and Chomsky, N. (2005). The evolution of the language faculty: Clarifications and implications. *Cognition*, 97:179–210.

Grosz, B. and Kraus, S. (1993). Collaborative plans for group activities. In *IJCAI*, volume 93, pages 367–373.

Grosz, B. J., Hunsberger, L., and Kraus, S. (1999). Planning and acting together. *AI magazine*, 20(4):23.

Hauser, M., Chomsky, N., and Fitch, W. T. (2002). The faculty of language: What is it, who has it, and how did it evolve? *Science*, 298:1569–1579.

Jaynes, J. (1976). *The Origin of Consciousness in the Breakdown of the Bicameral Mind*. Houghton Mifflin Harcourt, New York.

Joshi, A. (1990). Processing crossed and nested dependencies: An automaton perspective on the psycholinguistic results. *Language and Cognitive Processes*, 5:1–27.

Joshi, A. and Schabes, Y. (1992). Tree-adjoining grammars and lexicalized grammars. In Nivat, M. and Podelski, A., editors, *Definability and Recognizability of Sets of Trees*. Elsevier, Princeton, NJ.

Knuth, D. E. (1968). *Fundamental Algorithms, The Art of Computer Programming Vol. 1*. Addison-Wesley, Reading, MA.

Lashley, K. (1951). The problem of serial order in behavior. In Jeffress, L., editor, *Cerebral Mechanisms in Behavior*, pages 112–136. Wiley, New York. reprinted in Saporta (1961).

Lochbaum, K. E. (1998). A collaborative planning model of intentional structure. *Computational Linguistics*, 24(4):525–572.

Petrick, R. P. and Bacchus, F. (2002). A knowledge-based approach to planning with incomplete information and sensing. In *AIPS*, pages 212–222.

Saporta, S., editor (1961). *Psycholinguistics: A Book of Readings*. Holt Rinehart Winston, New York.

Searle, J. R. (1990). Collective intentions and actions. In Philip R. Cohen, Jerry L. Morgan, M. E. P., editor, *Intentions in communication*. MIT Press.

Speas, M. and Roeper, T., editors (2009). *Proceedings of the Conference on Recursion: Structural Complexity in Language and Cognition*. Univ. of Mass, Amherst. forthcoming.

Steedman, M. and Petrick, R. P. (2007). Planning dialog actions. In *Proceedings of the 8th SIGDIAL Workshop on Discourse and Dialogue (SIGdial 2007)*, pages 265–272.

Tomasello, M., Call, J., and Hare, B. (2003). Chimpanzees understand psychological states—the question is which ones and to what extent. *Trends in Cognitive Sciences*, 7(4):153–156.

Turing, A. M. (1937). Computability and λ-definability. *J. of Symbolic Logic*, 2(4):153–163.

Van Heijningen, C. A., De Visser, J., Zuidema, W., and Ten Cate, C. (2009). Simple rules can explain discrimination of putative recursive syntactic structures by a songbird species. *Proceedings of the National Academy of Sciences*, 106(48):20538–20543.

Vijay-Shanker, K. (1987). *A Study of Tree Adjoining Grammars*. PhD thesis, University of Pennsylvania.