Formalizing Affordance:

Logical and Neurological Relations between Grammar and Action

Mark Steedman: *Informatics, U. of Edinburgh*

`http://www.cogsci.ed.ac.uk/~steedman/home.html`

Affordance

- The idea that to perceive an object is to perceive its affordances—that is, the interactions of the perceiver with the world that the object supports or affords—is attractive from the point of view of a number of theoretical positions in cognitive science that emphasize the fundamental role of *actions*.
- However, its influence has been limited by two difficulties.
- One is the idea of "direct" perception of affordances.
- Another stems from the very fact that many such affordances are actions or events. The lack of a formal theory of events in their relation to objects that is applicable to human cognition and/or natural language semantics has been conspicuous.

The Dynamic Event Calculus

- Basic Dynamic Logic:

(1) $\models n \geq 0 \Rightarrow [\alpha](y = F(n))$

(2) $\models n \geq 0 \Rightarrow \langle\alpha\rangle(y = F(n))$

- The particular dynamic logic that we are dealing with here is one that includes the following dynamic axiom (the operator ; is *sequence*, an operation related to functional composition):

  (3) $[\alpha][\beta]P \Rightarrow [\alpha;\beta]P$

- Composition is one of the most primitive *combinators*, or operations combining functions, which Curry and Feys (1958) call **B**, writing the above as $\mathbf{B}\alpha\beta$.

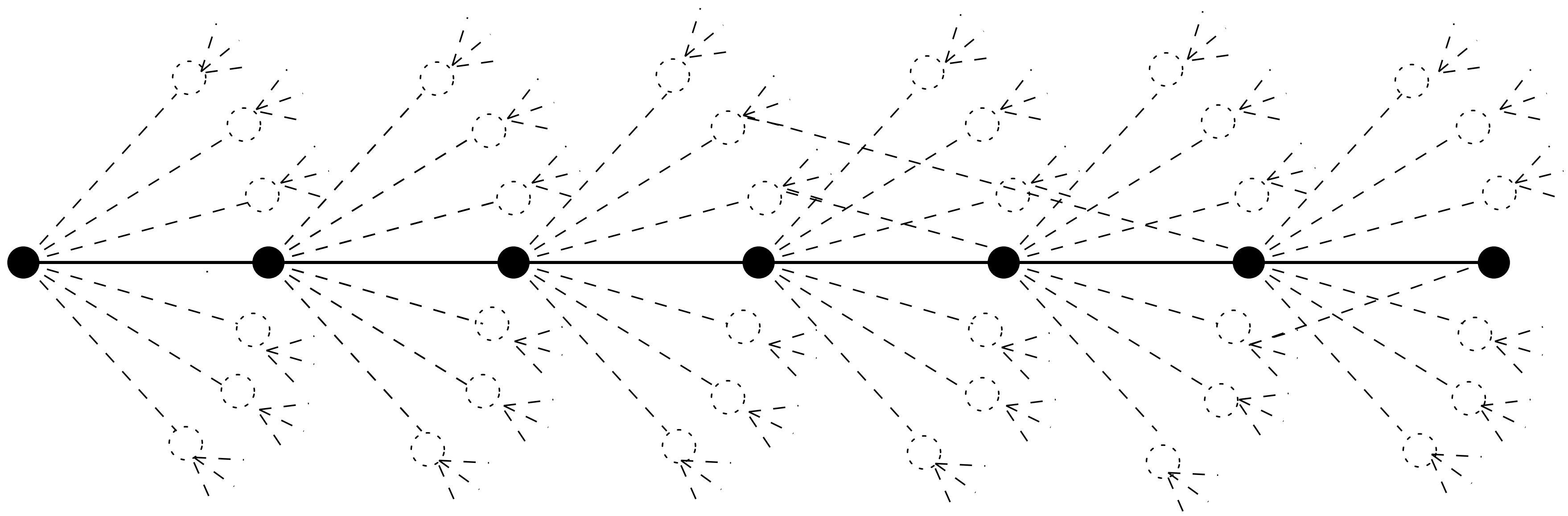


Figure 1: Kripke Model of Causal Accessibility Relation

STRIPS rules as Linear Implication

- We can represent events involving doors in this notation for a world (simplified for purposes of exposition) in which there are two places *out* and *in* separated by a door which may be *open* or *shut*, as follows:

- First we need rules to capture the fact that pushing a door changes it from open to shut and vice versa:

(4) a. $\models shut(x) \multimap [push(x)]open(x)$
 b. $\models open(x) \multimap [push(x)]shut(x)$

(5) a. $\models in \multimap [go\text{-}through(x)]out$
 b. $\models out \multimap [go\text{-}through(x)]in$

- These rules use *linear* logical implication $\multimap$ rather than intuitionistic implication $\Rightarrow$ in those rules that change the value of fluents.

STRIPS updates as Linear Implication (Contd.)

- The effect of such systems can be exemplified as follows: if the initial situation is that you are in and the door is shut:

  (6) $in \wedge door(d) \wedge shut(d)$

  —then the linear rules (4) mean that an attempt to prove the proposition in (7) concerning the state of the door in the situation that results from pushing the door will fail because rule (4a) has removed the fact in question from the database that results from the action $push(d)$:

  (7) $[push(d)]shut(d)$

STRIPS updates as Linear Implication (Contd.)

- On the other hand, attempts to prove the following will all succeed, since they are all facts in the database that results from the action $push(d)$ in the initial situation (6):

(8) a. $[push(d)]open(d)$

b. $[push(d)]door(d)$

c. $[push(d)]in$

- Using linear implication (or the equivalent rewriting logic devices or state update axioms of Thielscher (1999) and Martí-Oliet and Meseguer (1999)) for STRIPS-like rules makes frame axioms unnecessary. Instead, they are theorems of the linear logic representation.

STRIPS Preconditions as Intuitionistic Implication

(9) a. $\models door(x) \wedge open(x)$
$\Rightarrow possible(go\text{-}through(x))$

b. $\models door(x) \Rightarrow possible(push(x))$

- We also need to define the transitive property of the possibility relation, as follows, using the definition 3 of event sequence composition:

(10) $\models possible(\alpha) \wedge [\alpha]possible(\beta) \Rightarrow possible(\alpha;\beta)$

- Steedman (1997) extends this calculus to cover durative and telic events to support natural language semantics of tense and temporality, and investigates further ramifications and qulaifications of the frame problem.

STRIPS Planning in Dynamic Event Calculus

- This fragment gives us a simple planner in which starting from the world (11) in which I am *in*, and the door is *shut* and stating the goal (12) meaning "find a possible series of actions that will get me *out*," can be made to automatically deliver a constructive proof that one such plan is (13):

(11) $in \wedge door(d) \wedge shut(d)$

(12) $possible(\alpha) \wedge [\alpha]out$

(13) $\alpha = push(d); go\text{-}through(d).$

- The situation that results from executing this plan in the start situation (6) is one in which the following conjunction of facts is directly represented by the database:

(14) $out \wedge door(d) \wedge open(d)$

The Monkey and the Bananas

(15) a. $\models [grab(x)]have(x)$

b. $\models at(here + 3 + 3) \Rightarrow possible(grab(bananas))$

- If something is a box you can climb on it:

  (16) $box(b) \Rightarrow possible(climb\text{-}on(b))$

- —and if you are at a place and you climb on a box you are at a place that is higher by 3ft:

  (17) $\models at(p) \multimap [climb\text{-}on(b)]at(p+3)$

The Monkey and the Bananas, (Contd.)

- if two things have nothing on top of them and are not the same thing you can put one on the other:

(18) $\models clear(x) \wedge clear(y) \wedge (x \neq y) \Rightarrow possible(puton(x,y))$

- —and if x is on something and you put it on something else then that something becomes clear and x is on that something else:

(19) a. $\models on(x,z) \multimap [puton(x,y)]clear(z)$
b. $\models clear(y) \multimap [puton(x,y)]on(x,y)$

- It is worth noting that the use of a hybrid logic in which fluents like *clear* can be antecedents to both nonlinear qualification rules like (18) and linear ramification rules like (19) means that we avoid reintroducing the frame problem by having to explicitly state that $clear(x)$ in the consequent of (19b).

The Monkey and the Bananas, (Contd.)

- If the initial state of the world is as follows:

(20) $at(here) \wedge box(b1) \wedge box(b2) \wedge clear(b1) \wedge clear(b2)$

- —then the goal (21a) gives rise to (21b) as one possible plan

(21) a. $possible(\alpha) \wedge [\alpha]have(bananas)$

b. $\alpha = [puton(b1, here); climb\text{-}on(b1);$
$puton(b2, b1); climb\text{-}on(b2); grab(bananas)]$

Formalizing Affordance in DEC

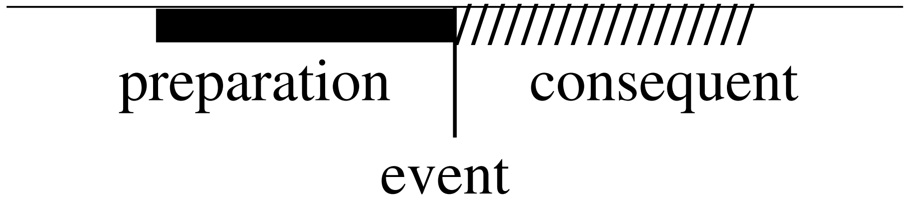


Figure 2: The event nucleus (adapted from Moens and Steedman 1988)

- Some animals can make plans involving tools of the kind in (21) (Köhler 1925).
- Such planning seems to be *reactive* to the presence of the tool and *forward-chaining*, rather than backward-chaining (working from goal to tool).
- This seems a good way for an animal to plan, and implies that actions are accessed via perception of the objects that mediate them.

Formalizing Affordance in DEC (Contd.)

- We can represent such indexing by first defining concepts like *pushing* and *going through* as functions like the following derived from (4) and (5), which it is convenient to call $push'$ and $go\text{-}through'$:

(22) a. $$push' = \lambda x. \left\{ \begin{array}{c} shut(x) \multimap open(x) \\ open(x) \multimap shut(x) \end{array} \right\}$$

b. $$go\text{-}through' = \lambda x. \left\{ \begin{array}{c} in \multimap out \\ out \multimap in \end{array} \right\}$$

- Here the linear implication symbol $\multimap$ is overloaded to signify linear mapping of state to state accompanied by deletion and addition of facts.

Formalizing Affordance in DEC (Contd.)

- The set of such functions $\mathit{Affordances}(\mathit{door})$ constitutes the affordances of doors:

(23) $$\mathit{Affordances}(\mathit{door}) = \left\{ \begin{array}{l} \mathit{push}' \\ \mathit{go\text{-}through}' \end{array} \right\}$$

- The concept of a door can then in turn be defined as a function mapping doors into (second-order) functions from their affordances like pushing and going-through to their results:

(24) $$\mathit{door}' = \lambda x_{door}.\lambda p_{\in \mathit{Affordances}(door)}.px$$

- The operation of turning an object of a given type into a function over those functions that apply to objects of that type is another primitive combinator called **T** or *type raising*, so (24) can be rewritten $\mathit{door}' = \lambda x_{door}.\mathbf{T}x$

Type Raising and Natural Language

- Such a concept of doors is useful for reactive planning, and one can add more affordances to $\mathit{Affordances}(\mathit{door})$ as one's experience increases.
- However, it is a somewhat stultifying representation in human terms, One would like to have the advantages in terms of efficiency of planning that thinking of objects in terms of their affordances allows, while also being able envisage novel uses for doors—for example, using one as a table, or as a raft—when circumstances demand it.
- One of the few sources of information about the natural classifications of objects that permit limited generalization comes from linguistics.

A Language that Lexicalizes a Default Affordance

- Many North American Indian languages, such as the Athabascan group that includes Navaho, are comparatively poorly off for nouns. Many nouns for artefacts are morphological derivatives of verbs.

- For example, “towel” is *bee ’ádít’oodí*, glossed as “one wipes oneself with it”, and “towelrack” is *bee ’ádít’oodí bąąh dah náhidiiltsos*—roughly “one wipes oneself with it is repeatedly hung on it”.

- Such languages appear to lexicalize nouns as a default affordance.

- The next section will argue that *all* languages, whether or not they bear explicit morphological indicators of these kinds, involve both type raising of nominal categories and composition of verbal ones as assumed in the Combinatory Categorial Grammar (CCG) framework.

Combinatory Grammars

- Verbs are associated with a syntactic "category" which identifies them as *functions*, and specifies the type and directionality of their arguments and the type of their result:

(25) likes := $(S\backslash NP)/NP$

- Two general rules of functional application give us context free expressive power:

(26) *Forward Application:* $(>)$

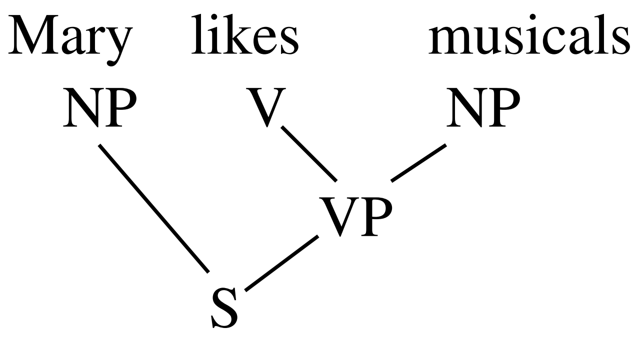
$$X/Y \quad Y \quad \Rightarrow \quad X$$

(27) *Backward Application:* $(<)$

$$Y \quad X\backslash Y \quad \Rightarrow \quad X$$

(28) a.

$$\begin{array}{ccc}
\text{Mary} & \text{likes} & \text{musicals} \\
\overline{NP} & \overline{(S\backslash NP)/NP} & \overline{NP} \\
 & \multicolumn{2}{c}{\overline{\qquad\qquad S\backslash NP \qquad\qquad}{>}} \\
\multicolumn{2}{c}{\overline{\qquad\qquad S \qquad\qquad}{<}} &
\end{array}$$

b. 

Semantics

- Categories and combinatory rules encode the semantic type of their translation:

(29) likes := $(S\backslash NP_{3s})/NP : \mathit{like}'$

(30) *Forward Application:* $(>)$

$$X/Y : f \quad Y : a \quad \Rightarrow \quad X : fa$$

(31) *Backward Application:* $(<)$

$$Y : a \quad X\backslash Y : f \quad \Rightarrow \quad X : fa$$

- They yield derivations like the following:

(32)

$$\begin{array}{ccc}
\text{Mary} & \text{likes} & \text{musicals} \\
\hline
NP_{3sm} : mary' & (S\backslash NP_{3s})/NP : like' & NP : musicals' \\
& \multicolumn{2}{c}{\overline{\hspace{14em}}{>}} \\
& \multicolumn{2}{c}{S\backslash NP_{3s} : like'musicals'} \\
\multicolumn{3}{c}{\overline{\hspace{24em}}{<}} \\
\multicolumn{3}{c}{S : like'musicals'mary}
\end{array}$$

- The derivation yields an S with a compositional interpretation, equivalent under a convention of left associativity to $(like'musicals')mary'$.

Coordination

- Coordination is included via the following rule, allowing constituents of like type to conjoin to yield a single constituent of the same type:

(33) *Coordination:* $(<\&>)$

$$X \quad conj \quad X \quad \Rightarrow \quad X$$

(34)

$$
\begin{array}{ccccc}
\text{I} & \text{loathe} & \text{and} & \text{detest} & \text{opera}\\
\hline
NP & (S\backslash NP)/NP & CONJ & (S\backslash NP)/NP & NP\\
\cline{2-4}
 & \multicolumn{3}{c}{(S\backslash NP)/NP \quad {<}\Phi{>}} & \\
\cline{2-5}
 & \multicolumn{4}{c}{S\backslash NP \quad >} \\
\cline{1-5}
\multicolumn{5}{c}{S \quad <}
\end{array}
$$

Composition

- Functions may nondeterministically *compose*, as well as apply, under the following rule:

(35) *Forward Composition:* ($> B$)

$$X/Y \quad Y/Z \quad \Rightarrow \quad X/Z$$

(36)

$$\begin{array}{cccccc}
\text{I} & \text{requested} & \text{and} & \text{would} & \text{prefer} & \text{musicals} \\
\hline
NP & (S\backslash NP)/NP & CONJ & (S\backslash NP)/VP : will' & VP/NP : prefer' & NP \\
 & & & \multicolumn{2}{c}{\underline{\qquad\qquad\qquad\qquad\qquad}{>}\mathbf{B}} & \\
 & & & \multicolumn{2}{c}{(S\backslash NP)/NP : \lambda x.\lambda y.will'(prefer' x)y} & \\
 & \multicolumn{4}{c}{\underline{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}{<}\Phi{>}} & \\
 & \multicolumn{4}{c}{(S\backslash NP)/NP} & \\
 & & & \multicolumn{3}{c}{\underline{\qquad\qquad\qquad\qquad\qquad\qquad}{>}} \\
 & & & \multicolumn{3}{c}{S\backslash NP} \\
\multicolumn{4}{c}{\underline{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}{<}} & & \\
\multicolumn{4}{c}{S} & & \\
\end{array}$$

Type-Raising

- Combinatory grammars also include type-raising rules:

(37) *Subject Type-raising:* (>T)

$$NP : a \quad \Rightarrow \quad \mathrm{T}/(\mathrm{T}\backslash NP) : \lambda f.f\ a$$

(38)

$$
\begin{array}{cccccc}
\text{I} & \text{dislike} & \text{and} & \text{Mary} & \text{likes} & \text{musicals} \\
\hline
NP & (S\backslash NP)/NP & CONJ & NP & (S\backslash NP)/NP & NP \\
 & & & : mary' & : \lambda x.\lambda y.like'xy & \\
\overline{S/(S\backslash NP)}^{>\mathbf{T}} & & & \overline{S/(S\backslash NP)}^{>\mathbf{T}} & & \overline{S\backslash(S/NP)}^{<\mathrm{T}} \\
 & & & : \lambda f.f\ mary' & & : \lambda f.f\ musicals' \\
\multicolumn{2}{c}{\overline{S/NP}^{>\mathbf{B}}} & & \multicolumn{2}{c}{\overline{S/NP}^{>\mathbf{B}}} & \\
 & & & \multicolumn{2}{c}{: \lambda x.like'x\ mary'} & \\
\multicolumn{5}{c}{\overline{S/NP}^{<\Phi>}} & \\
 & & \multicolumn{4}{c}{\overline{S}^{<}}
\end{array}
$$

Type-Raising and Composition = Permuting and Rebracketing

- We can explain a wide rage of unbounded and fragmenting contructions with CCG:

(39) a. [I dislike]$_{S/NP}$ and [you think Mary likes]$_{S/NP}$ musicals.

b. The musicals which [you think Mary likes]$_{S/NP}$.

(40)

$$
\begin{array}{cccccc}
\text{introduce} & \text{Bill} & \text{to Sue} & \text{and} & \text{Harry} & \text{to George} \\
\hline
(VP/PP)/NP & (VP/PP)\backslash((VP/PP)/NP) \ {<}\mathbf{T} & VP\backslash(VP/PP) \ {<}\mathbf{T} & CONJ & & VP\backslash((VP/PP)/NP) \ {<}\mathbf{B} \\
 & \multicolumn{2}{c}{\overline{\quad VP\backslash((VP/PP)/NP) \quad}\ {<}\mathbf{B}} & & & \\
 & \multicolumn{5}{c}{\overline{\quad VP\backslash((VP/PP)/NP) \quad}\ {<}\Phi{>}} \\
\multicolumn{6}{c}{\overline{\quad VP \quad}\ {<}}
\end{array}
$$

Type-Raising and Composition = Permuting and Rebracketing (Contd.)

- The argument cluster coordination construction (40) is an example of a universal tendency for “deletion under coordination” to respect basic word order: in all languages, if arguments are on the left of the verb then argument clusters coordinate on the left, if arguments are to the right of the verb then argument clusters coordinate to the right of the verb (Ross 1970):

(41)	SVO:	*SO and SVO	SVO and SO
	VSO:	*SO and VSO	VSO and SO
	SOV:	SO and SOV	*SOV and SO

Logical and Neurological Relations Between Grammar and Action

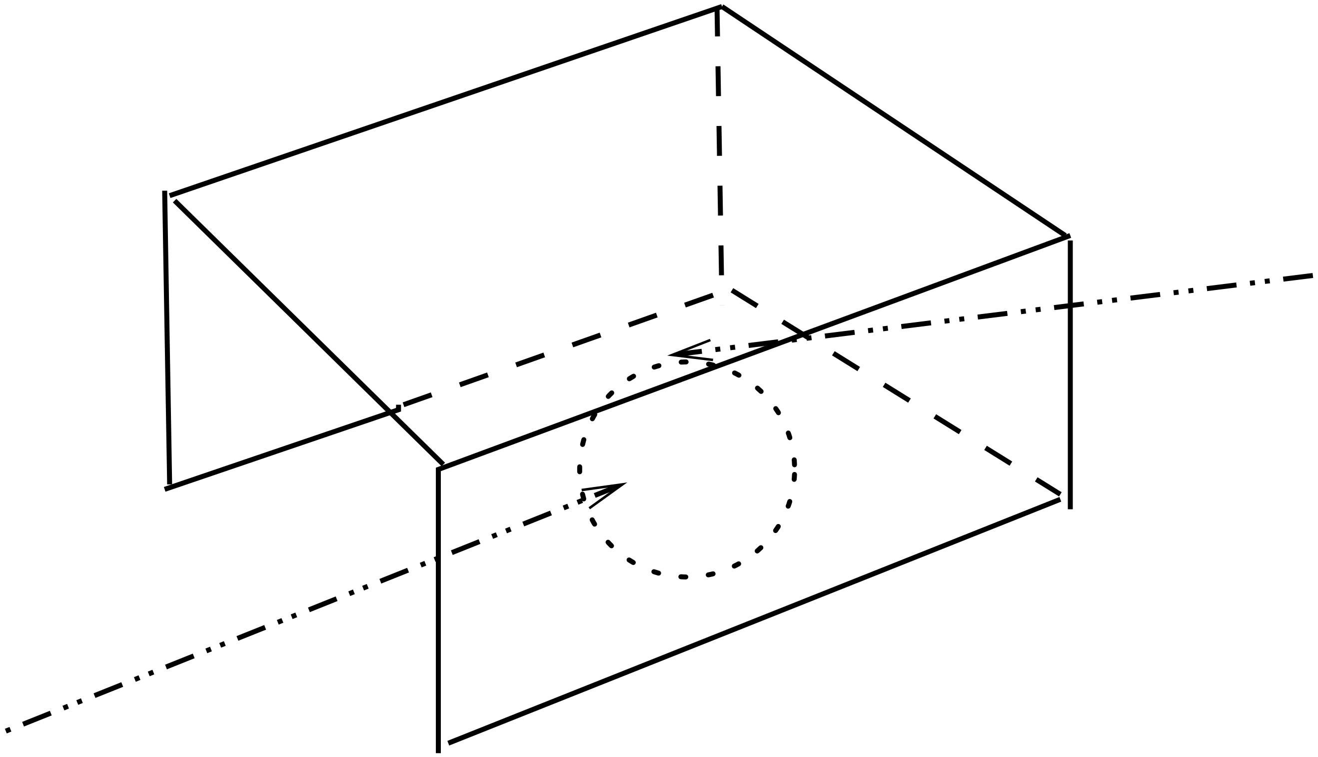
- The ubiquitous appearance of composition and type raising in both affordance-mediated action planning of the most elementary sort on the one hand, and universal grammar on the other, strongly suggests that the language faculty in its syntactic aspect is directly hung onto a more primitive set of prelinguistic operations originally developed for motor planning.
- The left temporal areas that evidence from brain imaging and acquired aphasias suggests are implicated in morphosyntactic processing are immediately adjacent to areas involved in motor planning, suggesting that in evolutionary and developmental terms, the former are built upon the latter.

Logical and Neurological Relations (Contd.)

- The onset of language in infants—and the entire cognitive explosion into the Piagetian operational phase—follows closely on the mastery of motor planning involving the use of tools at the final sixth stage of the Piagetian sensory-motor phase of cognitive development.
- The onset in the child of the ability to compose motor plans such as those needed for composite reaching around an obstacle also closely coincides with the onset of productive language use. It is also argued by Deacon (1988, 1997) and Diamond (1990) to depend the mastery of response inhibition mediated by more frontal areas that are also implicated in language disorders.

Diamond (1990)

- The task:



- The box may be transparent or opaque.
- The child is free to move around.

- The box is open on one side: the child knows this.
- The child knows the object is in the box, either because it can see it through the box, or because it saw it put there, or because it put it there itself by accident.
- Can the child reach the object?

Diamond 1990 (contd., see pp.649-654)

- Until around 6.5 months, children cannot retrieve an object that is entirely inside the box.
- From 6.5 to 8 months they only succeed when they happen to get into a position where their line of sight is through the open side, or when the experimenter turns the box for them. (If the experimenter turns it back again, the child is not helped.) Then they can do a relaxation reach and obtain the object.
- From around 7.5 to 8 months the children increasingly actively change their body position, increasing their chances of the above happening.

Diamond 1990 (contd., see pp.649-654)

- From around 8.5 to 9 months, children begin to separate line of sight from line of reach and achieve seriation of reaching. In this first stage, they deliberately bend over to put themselves in a position where they can see through the open side. They then straighten up so they can no longer see along the line of reach, and do a correct two-dimensional reach.
- From 9.5 to 10.5 they progress to a second stage where they do not need to move to a position where they can look through the opening before doing a 2-D reach.
- At all stages, having a transparent box actually makes things harder, presumably because of the increased difficulty of suppressing a direct relaxation reach. (Thus the problem is not that the children forget that the object is there.)

What Makes Language Uniquely Human?

- The neurological, developmental, and evolutionary conspiracy between serial ordering for motor plans and language suggests that the composition and type raising are prelinguistic primitives that we share with some animals. What more is needed to evolve language?

- One candidate is modal and propositional attitude concepts—that is, functions over propositional entities. (We have so far glossed over the important fact that plans compose actions of type *state* $\rightarrow$ *state*, whereas syntax composes functions of type *proposition* $\rightarrow$ *proposition*.) These induce true recursion in conceptual structures and grammar via the grounded lexicon. There is no evidence that apes entertain such concepts. Perhaps this is *all* they lack.

- Such higher-order concepts are vital to interactive applications and present challenging problems in automated reasoning.

References

References

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

Deacon, T. (1988). Human brain evolution i: Evolution of human language circuits. In Jerison, H. and Jerison, I., editors, *Intelligence and Evolutionary Biology*. Springer-Verlag, Berlin.

Deacon, T. (1997). *The Symbolic Species*. Norton, New York.

Diamond, A. (1990). Developmental time course in human infant and baby monkeys and the neural bases of inhibitory control in reaching. In Diamond, A., editor, *The Development and Neural Bases of Higher Cognitive Functions*, pages 637–676. New York Academy of Sciences, New York.

Köhler, W. (1925). *The Mentality of Apes*. Harcourt Brace and World, New York.

Martí-Oliet, N. and Meseguer, J. (1999). Action and change in rewriting logic. In Pareschi, R. and Fronhöfer, B., editors, *Dynamic Worlds*, pages 1–53. Kluwer, Dordrecht.

Ross, J. R. (1970). Gapping and the order of constituents. In Bierwisch, M. and Heidolph, K., editors, *Progress in Linguistics*, pages 249–259. Mouton, The Hague.

Steedman, M. (1997). Temporality. In van Benthem, J. and ter Meulen, A., editors, *Handbook of Logic and Language*, pages 895–938. North Holland, Amsterdam.

Thielscher, M. (1999). From situation calculus to fluent calculus: State update axioms as a solution to the inferential frame problem. *Artificial Intelligence*, 111:277–299.