Introduction to TLC

Carl Pollard
Ohio State University

Linguistics 681 (Algebraic Linguistics
Tuesday, January 20, 2009

These slides are available at:

`http://www.ling.ohio-state.edu/∼plummer/ling681`

(1) **Typed Lambda Calculi (TLCs)**

- Developed by Church and Curry starting in early 1930's
- Can be viewed model-theoretically (Henkin-Montague perspective) or proof-theoretically (Curry-Howard perspective)
- A TLC is specified by giving its *types*, its *terms*, and an *equivalence relation* on the terms.
- There are different kinds of TLCs, depending on what kind of logic its type system is based on.
- Here we focus on *positive* TLC, based on PIPL.
- Positive TLC underlies higher-order logic (HOL), widely used for formalizing theories of NL meaning

(2) **Types of Positive TLC**

a. There are some *basic* types. For concreteness, here we use a set of types motivated by NL semantics (this choice is not relevant for the time being):

Prop (for *propositions*, meanings of sentences)

Ind (for *individuals*, meanings of names)

Bool (for *booleans* or *truth values*, extensions of propositions)

Ent (for *entities*, extensions of individuals)

b. T is a type

c. if A and B are types, so is $A \wedge B$

d. if A and B are types, so is $A \rightarrow B$

(3) **Terms of Positive TLC (1/2)**
Note: we write '$\vdash a : A$' to mean term a is of type A.

a. There are some *nonlogical constants*. In the typical application to NL semantics, these are interpreted as word meanings, e.g.:

$\vdash \mathsf{Fido'} : \mathrm{Ind}$

$\vdash \mathsf{bark'} : \mathrm{Ind} \rightarrow \mathrm{Prop}$

$\vdash \mathsf{bite'} : (\mathrm{Ind} \wedge \mathrm{Ind}) \rightarrow \mathrm{Prop}$

$\vdash \mathsf{give'} : (\mathrm{Ind} \wedge \mathrm{Ind} \wedge \mathrm{Ind}) \rightarrow \mathrm{Prop}$

$\vdash \mathsf{believe'} : (\mathrm{Ind} \wedge \mathrm{Prop}) \rightarrow \mathrm{Prop}$

$\vdash \mathsf{bother'} : (\mathrm{Prop} \wedge \mathrm{Ind}) \rightarrow \mathrm{Prop}$

(4) **Terms of Positive TLC (2/2)**

b. There is a *logical constant* $\vdash * : \mathrm{T}$. In application to NL semantics, this is interpreted as semantic vacuity.

c. For each type A there are variables $\vdash x_i^A : A$ $(i \in \omega)$.

d. If $\vdash a : A$ and $\vdash b : B$, then $\vdash (a, b) : A \wedge B$.

e. If $\vdash a : A \wedge B$, then $\vdash \pi(a) : A$.

f. If $\vdash a : A \wedge B$, then $\vdash \pi'(a) : B$.

g. If $\vdash f : A \rightarrow B$ and $\vdash a : A$, then $\vdash \mathsf{app}(f, a) : B$.
Note: $\mathsf{app}(f, a)$ is usually abbreviated to $f(a)$.

h. If $\vdash x : A$ is a variable and $\vdash b : B$, then $\vdash \lambda_x b : A \rightarrow B$.

(5) **Positive TLC Term Equivalences (1/3)**

Here, t, a, b, p, and f are metavariables ranging over terms.

a. Equivalences for the term constructors:

i. $t \equiv *$;

ii. $\pi(a, b) \equiv a$;

iii. $\pi'(a, b) \equiv b$; and

iv. $(\pi(p), \pi'(p)) \equiv p$

(6) **Positive TLC Term Equivalences (2/3)**

b. Equivalences for λ ('lambda conversion')

(α) $\lambda_x b \equiv \lambda_y [y/x]b$;

(β) $[\lambda_x b](a) \equiv [a/x]b$; and

(η) $\lambda_x [f(x)] \equiv f$, provided x is not free in f.

Note: the notation '$[a/x]b$' means the term resulting from substitution in b of all free occurrences of $x : A$ by $a : A$. This presupposes no free variable occurences in a become bound as a result of the substitution.

(7) **Positive TLC Term Equivalences (3/3)**

c. Equivalences of Equational Reasoning

(ρ) $a \equiv a$

(σ) If $a \equiv a'$, then $a' \equiv a$.

(τ) If $a \equiv a'$ and $a' \equiv a$", then $a \equiv a$".

(ξ) If $b \equiv b'$, then $\lambda_x b \equiv \lambda_x b'$.

(μ) If $f \equiv f'$ and $a \equiv a'$, then $f(a) \equiv f'(a')$.

(8) **The Curry-Howard Perspective, First Pass (1/2)**

Positive TLC can be viewed as a proof theory for PIPL:

- Types correspond to formulas.
- Type constructors correspond to logical connectives.
- Constants correspond to axioms (without hypotheses).
- Variables correspond to hypotheses.
- Term constructors correspond to inference rules:
  a. $(-,-)$ corresponds to $\wedge$-introduction
  b. π and π' correspond to $\wedge$- elimination
  c. app correponds to $\rightarrow$-elimination (modus ponens)

(9) **The Curry-Howard Perspective, First Pass (2/2)**

- λ-binding correponds to $\rightarrow$-introduction (hypothetical proof)
- Terms correspond to proofs.
- Free variables in terms correspond undischarged hypotheses of proofs.
- Terms containing nonlogical constants correspond to proofs from nonlogical axioms.
- Combinators (closed terms without nonlogical constants) correspond to theorems proved with no hypotheses.

(10) **The Henkin-Montague Perspective**

A **(set-theoretic) interpretation** I of a positive TLC assigns to to each type A a set $I(A)$ and to each constant $\vdash a : A$ a member $I(a)$ of $I(A)$, subject to the following constraints:

a. $I(\mathrm{T})$ is a singleton;

b. $I(A \wedge B) = I(A) \times I(B)$;

c. $I(A \rightarrow B) \subseteq I(A) \Rightarrow I(B)$.
Note: the set inclusion can be proper, as long as there are enough functions to interpret all functional terms.

(11) **Variable Assignments**

A **variable assignment** relative to an interpretation is a function that maps each variable to a member of the set that interprets the variable's type.

(12) **Extending an Interpretation (1/2)**

Given a variable assignment α relative to an interpretation I, there is a unique extension of I, denoted by I_α, that assigns interpretations to all terms, such that:

a. For each variable x, $I_\alpha(x) = \alpha(x)$.

b. For each constant a, $I_\alpha(a) = I(a)$.

c. If $\vdash a : A$ and $\vdash b : B$, then $I_\alpha((a, b))$ is $\langle I_\alpha(a), I_\alpha(b) \rangle$.

(13) **Extending an Interpretation (2/2)**

d. If $\vdash p : A \wedge B$, then $I_\alpha(\pi(p))$ is the first component (= projection onto $I(A)$) of $I_\alpha(p)$; and $I_\alpha(\pi'(p))$ is the second component (= projection onto $I(B)$) of $I_\alpha(p)$.

e. If $\vdash f : A \rightarrow B$ and $\vdash a : A$, then $I_\alpha(f(a)) = (I_\alpha(f))(I_\alpha(a))$.

f. If $\vdash b : B$, then $I_\alpha(\lambda_{x \in A} b)$ is the function from $I(A)$ to $I(B)$ that maps each $s \in I(A)$ to $I_\beta(b)$, where β is the variable assignment that coincides with α except that $\beta(x) = s$.

(14) **Observations about Interpretations**

- Two terms $\vdash a : A$ and $\vdash b : B$ of positive TLC are term-equivalent iff $A = B$ and, for any intepretation I and any assignment α relative to I, $I_\alpha(a) = I_\alpha(b)$.
- Another way of stating the preceding is to say that term equivalence (viewed as an equational proof system) is sound and complete for the class of interpretations described in (10-12).
- For any term a, $I_\alpha(a)$ depends only on the restriction of α to the free variables of a.
- In particular, if a is a closed, then $I_\alpha(a)$ is independent of α so we can simply write $I(a)$.
- Thus, an interpretation for the basic types and constants extends uniquely to all types and all closed terms.