Formal Semantics in Modern Type Theories
(and Event Semantics in MTT-Framework)

Zhaohui Luo
Royal Holloway
University of London
This talk

I. Formal semantics in Modern Type Theories: overview
   ❖ MTT-semantics is both model-theoretic and proof-theoretic
   ❖ HoTT-logic for MTT-semantics in Martin-Löf’s TT
     ❖ paper in Proc. of LACompLing18

II. Event semantics in MTT-framework
   ❖ (Neo-)Davidsonian event semantics and problems
   ❖ Event semantics in MTT-framework
     ❖ Events in MTT-semantics
     ❖ Event structure with dependent types
I. Overview of MTT-semantics

❖ Natural Language Semantics – study of meaning
  (communicate = convey meaning)

❖ Various kinds of theories of meaning
  ❖ Meaning is reference (“referential theory”)
    ❖ Word meanings are things (abstract/concrete) in the world.
    ❖ c.f., Plato, ...
  ❖ Meaning is concept (“internalist theory”)
    ❖ Word meanings are ideas in the mind.
    ❖ c.f., Aristotle, ..., Chomsky.
  ❖ Meaning is use (“use theory”)
    ❖ Word meanings are understood by their uses.
    ❖ c.f., Wittgenstein, ..., Dummett.
Type-Theoretical Semantics

Montague Semantics
- R. Montague (1930–1971)
- Dominating in linguistic semantics since 1970s
- Set-theoretic, using simple type theory as intermediate
- Types (“single-sorted”): e, t, e→t, ...

MTT-semantics: formal semantics in modern type theories
- Examples of MTTs:
  - Martin-Löf’s TT: predicative; non-standard FOL
  - pCIC (Coq) & UTT (Luo 1994): impredicative; HOL
  - Ranta (1994): formal semantics in Martin-Löf’s type theory
  - Recent development on MTT-semantics
    ➔ full-scale alternative to Montague semantics
Recent development on rich typing in NL semantics

- Asher, Bekki, Cooper, Grudzińska, Retoré, ...
  - S. Chatzikyriakidis and Z. Luo (eds.) Modern Perspectives in Type Theoretical Sem. Springer, 2017. (Collection on rich typing & ...)

- MTT-semantics is one of these developments.

Advantages of MTT-semantics, including

- Both model-theoretic & proof-theoretic – offering a new perspective not available before (explicated later today)
MTT-semantics: basic categories

<table>
<thead>
<tr>
<th>Category</th>
<th>Semantic Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>Prop (the type of all propositions)</td>
</tr>
<tr>
<td>CNs (book, man, ...)</td>
<td>types (each common noun is interpreted as a type)</td>
</tr>
<tr>
<td>IV</td>
<td>A → Prop (A is the “meaningful domain” of a verb)</td>
</tr>
<tr>
<td>Adj</td>
<td>A → Prop (A is the “meaningful domain” of an adjective)</td>
</tr>
<tr>
<td>Adv</td>
<td>ΠA:CN.(A→Prop)→(A→Prop) (polymorphic on CNs)</td>
</tr>
</tbody>
</table>

In MTT-semantics, CNs are types rather than predicates:
- “man” is interpreted as a type Man : Type.
- Man could be a structured type (say, Σ(Human,male))
- A man talked.
- ∃m:Man.talk(m) : Prop, where talk : Human → Prop and Man ≤ Human (subtyping – crucial for MTT-semantics; see later.)
Rich type structure ("many-sorted", but types have structures):
- Existing types in MTTs: Table, $\Sigma x: \text{Man.handsome}(x)$, ...
- Newly introduced types to MTTs: Phy•Info (representing copredication)
- Type-theoretic representations for various linguistic features
  (Adj/Adv modifications, coordination, copredication, coercions, events, ...)

Selectional restrictions: meaningfulness v.s. falsity

(#) Tables talk.
- Montague: $\forall x: e. \text{table}(x) \Rightarrow \text{talk}(x)$ (well-typed, false in the intended model)
- MTT-sem: $\forall x: \text{Table.talk}(x)$ (ill-typed as talk: Human $\rightarrow$ Prop; meaningless)

Note:
- Well-typedness corresponds to meaningfulness (c.f., [Asher11] and others)
- Typing in MTTs is decidable, while truth/falsity of a formula is not.
Modelling Adjective Modification: Case Study
[Chatzikyriakidis & Luo: FG13, JoLLI17]

<table>
<thead>
<tr>
<th>Classical classification</th>
<th>example</th>
<th>Characterisation of Adj(N)</th>
<th>MTT-semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>intersective</td>
<td>handsome man</td>
<td>N &amp; Adj</td>
<td>$\sum x: \text{Man.handsome}(x)$</td>
</tr>
</tbody>
</table>
| subsective               | large mouse        | N (Adj depends on N)       | large : $\Pi A: \text{CN. } A \rightarrow \text{Prop}$  
large(mouse) : Mouse $\rightarrow \text{Prop}$ |
| privative                | fake gun           | $\neg N$                   | $G = G_R + G_F$  
with $G_R \leq_{\text{inl}} G$, $G_F \leq_{\text{inr}} G$ |
| non-committal            | alleged criminal   | nothing implied            | $\exists h: \text{Human. } H_{h,A}(\ldots)$       |

- $H_{h,A}(\ldots)$ expresses, eg, “h alleges ...”, for various non-committal adjectives $A$; it uses the Leibniz equality $=_\text{Prop}$. [Luo 2018] (*)
- cf, work on hyperintensionality (Cresswell, Lappin, Pollard, ...)

LACompLing 2018
Note on Subtyping in MTT-semantics

❖ Simple example
A human talks. Paul is a handsome man.
Does Paul talk?
Semantically, can we type talk(p)?
(talk : Human→Prop & p : Σ(Man,handsome))
Yes, because p : Σ(Man,handsome) ≤ Man ≤ Human.

❖ Subtyping is crucial for MTT-semantics
❖ Coercive subtyping [Luo 1999, Luo, Soloviev & Xue 2012] is adequate for MTTs and we use it in MTT-semantics.
❖ Note: Traditional subsumptive subtyping is inadequate for MTTs (eg, canonicity fails with subsumption.)
MTT-semantics is both model/proof-theoretic

- Model-theoretic semantics (traditional)
  - Meaning as denotation (Tarski, ...)
  - Montague: NL $\rightarrow$ (simple TT) $\rightarrow$ set theory

- Proof-theoretic semantics
  - Meaning as inferential use (proof/consequence)
  - Gentzen, Prawitz, Martin-Löf (meaning theory)

- MTT-semantics
  - Both model-theoretic and proof-theoretic – in what sense?

- What does this imply?
MTT-semantics is model-theoretic
- NL $\rightarrow$ MTT (representational/model-theoretic)
- MTT as meaning-carrying language
  - types representing collections
  - signatures (eg, subtyping [Lungu 2018]) representing situations
  - Cf, set theory in Montague semantics

MTT-semantics is proof-theoretic
- MTTs have proof-theoretic meaning theories
  - Judgements can be understood by means of their inferential roles.
  - Use theory of meaning (Wittgenstein, Dummett, Brandom)
  - Proof-theoretic semantics (Gentzen, Prawitz, Martin-Löf, ...)
- Proof technology: reasoning based on MTT-semantics on computers (eg, [Chatzikyriakidis & Luo (JoLLI14)])
Importance for MTT-semantics

❖ Model-theoretic – powerful semantic tools
  ❖ Much richer typing mechanisms for formal semantics
  ❖ Powerful contextual mechanism to model situations

❖ Proof-theoretic – practical reasoning on computers
  ❖ Existing proof technology: proof assistants (Coq, Agda, Lego, …)
  ❖ Applications to NL reasoning

❖ Leading to both of
  ❖ Wide-range modelling as in model-theoretic semantics
  ❖ Effective inference based on proof-theoretic semantics

Remark: new perspective & new possibility not available before!
Advanced features in MTT-semantics: examples

❖ Copredication
  ❖ Linguistic phenomenon studied by many (Pustejovsky, Asher, Cooper, Retoré, ...)
  ❖ Linguistic feature difficult, if not impossible, to find satisfactory treatment in a CNs-as-predicates framework. (For a mereological one, see [Gotham16].)

❖ Anaphora analysis/resolution via Σ-types
  ❖ [Sundholm 1986, Ranta 1994] in Martin-Löf’s type theory

❖ Linguistic coercions via coercive subtyping [Asher & Luo 2012]

❖ Several recent developments
  ❖ (today) Event semantics in MTT-framework [Luo & Soloviev (WoLLIC17)]
  ❖ Propositional forms of judgemental interpretations [Xue et al (NLCS18)]
  ❖ CNs as setoids [Chatzikyriakidis & Luo (J paper for Oslo meeting 2018)]
  ❖ (today) HoTT-logic for MTT-sem in Martin-Löf’s TT (current proceedings)
MTT-semantics in Martin-Löf’s TT with H-logic

- Martin-Löf’s type theory for formal semantics
  - Sundholm, Ranta & many others (all use PaT logic)
- PaT logic: propositions as types (Curry-Howard)
  - P is true if, and only if, p : P for some p.
  - But Martin-Löf goes one step further: types = propositions!
  - This is where a problem arises [Luo (LACL 2012)].
- Proof irrelevance (*)
  - Example: a handsome man is (m,p) : Σx:Man.handsome(x)
    - Two handsome men are the same iff they are the same man – proof irrelevance (any two proofs of the same proposition are the same.)
    - But in MLTT with PaT logic, this would mean every type collapses! Obviously, that would be absurd.
- So, MLTT with PaT logic is actually inadequate for MTT-sem, which has been mainly developed in UTT so far.
MLTT\textsubscript{h}: Extension of MLTT with H-logic

- **H-logic** (in Homotopy Type Theory; HoTT book)
  - A proposition is a type with at most one object.
  - isProp(A) = \prod_{x,y:A} (x = y).
  - Logical operators (examples):
    - \( P \implies Q = P \rightarrow Q \) and \( \forall x:A.P = \prod_{x:A} P \)
    - \( P \lor Q = |P+Q| \) and \( \exists x:A.P = |\sum_{x:A} P| \)
      where \( |A| \) is propositional truncation, a proper extension.

- MLTT\textsubscript{h} = MLTT + h-logic
  - Proof irrelevance is “built-in” in h-logic (by definition).
  - Claim: MLTT\textsubscript{h} is adequate for MTT-semantics.
  - Details in the short paper of LACompLing18 proceedings.
II. Events in MTT-Semantic Framework

❖ Davidson’s event semantics [1967]
❖ Original motivation: adverbial modifications (*)
  (1) John buttered the toast.
  (2) John buttered the toast with the knife in the kitchen.
  Does (2) imply (1)? (Cumbersome in MG with meaning postulates.)
❖ Events make it natural without meaning postulates.
  In neo-Davidsonian notation with thematic roles (1980s):
  (1’) \( \exists e: \text{Event. butter}(e) \)
  & agent(e)=john & patient(e)=toast
  (2’) \( \exists e: \text{Event. butter}(e) \& \text{with}(e,\text{knife}) \& \text{at}(e,\text{kitchen}) \)
  & agent(e)=john & patient(e)=toast
  Obviously, (2’) \( \Rightarrow \) (1’)
Problems in Event-semantics + Montague

❖ For example, “event quantification problem” (EQP)
❖ Incompatibility between event semantics and MG.

(1) Nobody talked.

Intended neo-Davidsonian event semantics is (2):
(2) $\neg \exists x : e. \text{human}(x) \land \exists v : \text{Event. talk}(v) \land \text{agent}(v,x)$

But the incorrect semantics (3) is also possible – it is well-typed:
(3) $\exists v : \text{Event.} \rightarrow \exists x : e. \text{human}(x) \land \text{talk}(v) \land \text{agent}(v,x)$

which moves the event quantifier “$\exists v : \text{Event}$” in (2) to the left.
Some proposed solutions to EQP

❖ Many different proposals
  ❖ Purpose: to force scope of event quantifier to be lower.
  ❖ Only mention two of them here.

❖ Champollion’s quantificational event sem. [2010, 2015]
  ❖ talk : (Event→t)→t with talk(E) = ∃e: Event. e∈E & talk(e)
  ❖ Trick: taking a set E of events as argument, but talk(e) ...
  ❖ Debatable: intuitive meanings, compositionality & complexity

  ❖ Use Abstract Categorial Grammar (see, eg, [de Groote 01])
  ❖ ACG structure prevents incorrect interpretation.

❖ Our proposal: dependent event types (solution to EQP & ...)
Dependent event types [Luo & Soloviev (WoLLIC17)]

- DETs: refining event structure by (dependent) typing
- Applications include
  - A solution to EQP
  - Selection restrictions in MTT-event semantics
- Refined types of events: Event → Evt(…)
  - Event types dependent on thematic roles agents/patients
  - For a:Agent and p:Patient, consider DETs
    Event, $\text{Evt}_A(a)$, $\text{Evt}_P(p)$, $\text{Evt}_{AP}(a,p)$
- Subtyping between DETs:
  \[
  a : A \quad A \leq B \\
  \Rightarrow \\
  a : B
  \]
DET-solution to EQP

(1) Nobody talked.

Neo-Davidsonian in Montague’s setting (repeated):
(2) \( \neg \exists x : e. \text{human}(x) \land \exists v : \text{Event. talk}(v) \land \text{agent}(v,x) \)
(3) \( \exists v : \text{Event.} \neg \exists x : e. \text{human}(x) \land \text{talk}(v) \land \text{agent}(v,x) \)

The incorrect (3) is well-typed.

Dependent event types in Montague’s setting:
(4) \( \neg \exists x : e. \text{human}(x) \land \exists v : \text{Evt}_A(x). \text{talk}(v) \)
(#) \( \exists v : \text{Evt}_A(x). \neg \exists x : e. \text{human}(x) \land \text{talk}(v) \)

where (#) is ill-typed since the first “x” is outside scope of “\( \exists x : e \)”. 
Selectional restrictions

❖ Recall:
   (#) Tables talk.
   ❖ Montague: $\forall x: e. \text{talk}(x)$ – well-typed but false, as talk : $e \rightarrow t$
   ❖ MTT-sem: $\forall x: \text{Table}. \text{talk}(x)$ – ill-typed as talk : Human$\rightarrow$Prop

❖ What happens with events?
   ❖ Neo-Davidsonian: talk : Event$\rightarrow$t or talk : Event$\rightarrow$Prop
   ❖ Montague: $\forall x: e \exists v: \text{Event}. \text{talk}(v) \& \text{agent}(v) = x$ (well-typed)
   ❖ MTT-sem: $\forall x: \text{Table} \exists v: \text{Evt}_A(x). \text{talk}(v)$
     (Also well-typed (!) because Table $\leq$ Agent)

So?
Three ways to enforce selectional restriction with events:
1. Refined typing for verb phrases (like talk)
2. Refining the typing of thematic roles (like agent)
3. Refining event types (next slide)

Approach 1 & 2: Instead of the neo-Davidsonian typing
talk : Event → t, or agent : Event → e, we consider

- talkₕ : Human → Event → Prop (Davidson’s original proposal) or
- talkₜ : ∏h:Human. Evtₐ(h) → Prop (dependent typing) or
- agentₕ : Event → Human (with codomain being Human)

Tables talk. (Ill-typed – table x is not a human.)

(#) ∀x:Table ∃v:Event. talkₕ(x,v) & agent(v)=x (ill-typed)
(#) ∀x:Table ∃v:Event. talk(v) & agentₕ(v)=x (ill-typed)
(#) ∀x:Table ∃v:Evtₐ(x). talkₜ(x,v) (ill-typed)
Approach 3: refined DETs

Let $T \subseteq_c \text{Agent}$. (example for subtypes of Agent)
- $\text{Evt}_A[T] : T \rightarrow \text{Type}$
- $\text{Evt}_A[T](a) = \text{Evt}_A(c(a)), \text{ for any } a : T.$

Examples

- Men talk. (OK because Man $\subseteq$ Human)
- $\forall x : \text{Man} \; \exists v : \text{Evt}_A[\text{Human}](x). \; \text{talk}(v)$
- Tables talk. ($\text{Evt}_A[\text{Human}](x)$ ill-typed as $x$ is not a human.)
- (#) $\forall x : \text{Table} \; \exists v : \text{Evt}_A[\text{Human}](x). \; \text{talk}(v)$
- John picked up and mastered the book. (b:Book $\subseteq$ Phy•Info)
- $\exists v : \text{Evt}_{AP}[\text{Human, Phy} \bullet \text{Info}](j,b). \; \text{pick-up}(v) \; \& \; \text{master}(v)$
Underlying formal systems

- Systems extended with dependent event types
  - $C_e$ – Church’s simple type theory + DETs (with subsumptive subtyping)
  - UTT[E] – the modern type theory UTT + DETs (with coercive subtyping as specified in E)

- Theorem.
  - $C_e$ (like UTT[E]) has nice meta-theoretic properties including, e.g., normalisation and logical consistency.
  - Proof. Faithfully embedding $C_e$ into UTT[E].

(***
Related (and some future) work on DETs

❖ Original idea
  ❖ Came from my treatment of an example in (Asher & Luo 12)
    ❖ Evt(h) to represent collection of events conducted by h : Human.
  ❖ Further prompted by de Groote’s talk at LENLS14 (on EQP etc.)

❖ Other applications of DETs
  ❖ For example, problem with negation in event semantics
    ❖ Krifka’s solution [1989]: a mereological negation system
    ❖ Champollion’s solution [2015] (as mentioned above)
    ❖ DETs solution: details to be worked out.

❖ DEPs dependent on other parameters
  ❖ Dependency on other thematic roles, say time/location/...:
    Reasonable? Useful?
  ❖ Dependency on other kinds of parameters than thematic roles?
References (1)

References (2)

❖ S. Chatzikyriakidis and Z. Luo. Formal Semantics in Modern Type Theories. ISTE/Wiley Science Publishing Ltd. (to appear)
References (3)

References (4)

References (5)