Frames in TTR

Eimear Maguire
B.A. (Mod.) Computer Science, Linguistics and Modern Irish
Final Year Project April 2014
Supervisor: Dr. Tim Fernando

School of Computer Science and Statistics
O’Reilly Institute, Trinity College, Dublin 2, Ireland
Declaration

I hereby declare that this project is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university.

______________________________________  _______________________
Eimear Maguire                          Date
Permission to lend

I agree that the Library and other agents of the College may lend or copy this report upon request.

Eiméar Maguire

Date
Acknowledgements

Thank you to Stephen Shaw for passing on the LaTeX code he wrote to format his own report last year. Thank you to Cleo Condoravdi and Sven Lauer for their correspondence, and for passing along a draft of their in-progress paper on anankastic conditionals.

Thank you to Carl Vogel and Martin Emms for seeing me through from a curious shadowing student to (hopefully!) a soon-to-be graduate.

Thank you to my supervisor Tim Fernando, whose seemingly endless patience, thoughtfulness and generosity\(^1\) over the past several months has left me slightly baffled and extremely grateful, and who – knowingly or not – has been a source of calm throughout the year.

Thank you to my family and my friends. Thank you to everyone who, over the past four years, has made Trinity less a place of study than a home. I’m going to miss this bizarre little corner of the universe.

\(^1\)One particular day I counted, and I think I clocked up about 3 hours and 20 minutes in his office. That’s longer than the original cut of Return of the King.
# Contents

1 Introduction  

2 Frames  
2.1 Frames Semantics ........................................ 4  
    2.1.1 Frames ........................................ 5  
    2.1.2 FrameNet ...................................... 7  
    2.1.3 Relations ...................................... 9  
2.2 Frames in Computer Science .......................... 11  
    2.2.1 A sample frame system .......................... 11  
    2.2.2 Implemented frames ............................ 13  

3 Formal Semantic Background  
3.1 Introduction ........................................... 14  
    3.1.1 Possible worlds ................................ 14  
    3.1.2 Propositions as types .......................... 15  
3.2 Examples of interpretation .......................... 15  
    3.2.1 \([A \and B]\) .................................. 15  
    3.2.2 \([A \implies B]\) ................................. 16  

4 Anankastic Conditionals  
4.1 Overview of Anankastic Conditionals .............. 18  
4.2 Formal Description through Possible Worlds .... 22
CONTENTS

4.2.1 Preferences ............................................ 22
4.2.2 ‘want’ .................................................. 22
4.2.3 Anankastic Conditionals ............................... 23

5 TTR ................................. 27
  5.1 Introduction ............................................. 27
  5.2 Notation .................................................. 28
  5.3 TTR in action .......................................... 32

6 TTR Formalisation ............................. 35

7 Conclusion ........................................... 41
Chapter 1

Introduction

This project centres on what is essentially a matter of perspective: whether to describe meaning from the top down or from the bottom up – top-down as typified by possible-worlds semantics, and bottom-up as typified by propositions-as-types semantics. These two formal approaches are considered in the light of frame semantics. Frame semantics is descended from cognitive semantics, and in a sense its formalisation through TTR, a particular version of type theory with records, joins the two traditions (formal and cognitive). The project is, in this sense, driven by various conflicts in semantics – cognitive vs. formal approaches, internal vs. external notions of truth and validity, top-down vs bottom-up constructions of meaning – and the potential for their reconciliation. In the context of this tension, we will attempt to explore some part of linguistics which requires consideration of action, planning and intention, and show that an account originally written through a possible-worlds semantics can be formally reiterated without needing to wield infinite sets of possible worlds.

Anankastic conditionals have been chosen as the specific linguistic phenomenon to implement in TTR. Anankastic conditionals are sentences of the type

- If you want to go to Harlem, you have to take the A Train.

- If we want to finish by midnight, we’ll have to split the work.
• If they want to get to the airport that early, you’re going to have to give them a lift.

• If you use your tax exemption now, you’re going to have to pay more tax next year.

where the inner antecedent (‘you[...]go to Harlem’) can only happen if the inner consequent (‘you[...]take the A Train’) also happens. Anankastic conditionals are perhaps an easy target for description through TTR, as the validity of an anankastic statement is already dependent on what the speaker believes, and the plans/intentions of the subject of the antecedent. In other words, the possible-worlds description already takes into account the mental states of the speaker and subject.

From this point forward, any references to anankastic conditionals can be taken to mean anankastic conditionals under the analysis given by Condoravdi and Lauer. In general, specific examples are also taken directly from their papers. At time of writing, theirs is the most up to date account, and as a specific model needs to be chosen, is the best option. For a more detailed description of its evolution from prior work, and justification of where this account succeeds where others have failed, see Condoravdi and Lauer (2012), Condoravdi and Lauer (2013), and primarily the final version of Condoravdi and Lauer (2014).

Chapter 2 discusses frames in both the context of frame semantics, with a particular focus on the FrameNet project, and the use of frames in AI.

Chapter 3 turns to the formal semantics tradition, briefly describing possible-worlds semantics and propositions-as-types semantics and their varying representations of a few simple examples. This also begins to set out the idea of records and record types as will be revisited in the chapter on TTR itself.

Anankastic conditionals are described in Chapter 4 according to Condoravdi and Lauer’s account.

Chapter 5 introduces the TTR system and notation.

In Chapter 6, the proposed TTR reformalisation of anankastic conditionals is
presented and explained.

Finally, Chapter 7 concludes the project with closing remarks and some notes on potential future work.
Chapter 2

Frames

2.1 Frames Semantics

The description of frame semantics in this section is based on Fillmore and Baker (2010), with additional example frames etc. taken from the FrameNet database¹. Frame semantics is a branch of semantic analysis in which meaning is understood with reference to background knowledge. This background knowledge is characterised as a scene or a frame. Cognitive frames can be defined as the packages of knowledge, beliefs, and patterns of practice that allow humans to understand their experiences (Fillmore and Baker, 2010). Cognitive frames can come from an individual’s cultural background, their speech community, and experiences of simply being human. These frames are not tied to language, but language can be used to evoke them. Frame semantics is the study of how these frames are evoked by linguistic forms, and how those frames are used in understanding the piece text or speech in which those forms appear. There is a careful difference in terminology between invocation and evocation: cognitive frames are *invoked* by an interpreter to make sense of some situation or experience, whereas linguistic frames are *evoked* by linguistic signs for the interpretation of some passage. Evocation of linguistic frames is a cognitive experience,

¹https://framenet.icsi.berkeley.edu
rather than a cognitive process. As for the linguistic forms themselves, the meaning of all content words is assumed to depend on the context provided by some frame.

Sentences that appear different on the surface can be semantically very similar through evocation of the same frame. An example typically given is the *commercial transaction* frame. In a *commercial transaction*, a *seller* gives some *goods* to a *buyer* in exchange for *payment*:

- Mary bought a cake from John.
- John sold a cake to Mary.
- Mary spent €5 on a cake.
- The cake for the party cost Mary €5.
- John got €5 for the cake.

The examples above all use different verbs, but the basic situation imagined is the same one viewed from different perspectives.

### 2.1.1 Frames

Frames are characterised by *frame elements*, which are entities or properties may/must be present in instances of that frame e.g. the buyer, seller etc. above. Frame elements may be core elements, which must be present, or peripheral elements, which do not necessarily appear but which do have a place within the frame if they do. For example, the frame for *intentional deception* has as core elements a *deceiver*, a *victim*, and a *topic* about which they are deceived. The non-core elements listed are the *means* by which they were deceived (e.g. a persuasive argument), a *place* in which the deception occurs and the *time* at which it occurs. Example sentences which evoke this frame are

- I can’t believe they fooled me with that old trick.
• Politicians do nothing but deceive us about money.

• You deceived me about the location of the diamonds.

• Sometimes it may be the case that the press itself stands as a monolith practising deception upon the people.

• Chris misled me with her powers of persuasion.

• It’s hard to fool someone in their own home.

• Some, looking for decent employment, said they were deceived soon after getting off the train.

Their shared connection is less obvious on the surface than the previous cake examples for *commercial transaction*, which all described the same situation in simple language, but the principle remains the same. Whether an element is best classed as core or peripheral is sometimes unclear, as in some frames certain central concepts do not need to be given explicitly. Fillmore and Baker (2010) select the example of time, place and manner: these are usually adjuncts, but are required by some lexical contexts, e.g. the verb *reside* requires a locative. Non-appearing core frame elements (Fillmore and Baker, 2010, p. 328) do so for three main reasons. The first is that the grammatical structure of the sentence evoking the frame may allow (or require) some argument to be absent, e.g. the optionality of the agent in passive sentences. The second and third are for lexical reasons. In the first lexical case, verbs which can be used both transitively and intransitively are considered to be always transitive, with an argument implied whether it is made explicit or not, e.g. *I ate* still requires that the agent ate *something*, but what it was is unimportant and may be omitted. In the second, the argument is already known to all parties, and does not even need to be stated e.g. *I won* presumes that the agent won some particular contest etc. known to all parties.

Lexical units evoke particular frames. Depending on the words chosen to describe a situation, different reactions can be produced in the reader or listener even if the
facts are kept the same. This is so obvious as to be an almost tautological statement, but frame semantics provide a neat semantic justification for this phenomenon. The lexical units listed as associated with intentional deception are deceive, deception, fool, mislead and trick. A single frame can contain multiple shades of meaning depending on the lexical unit used to evoke them. In this sense linguistic units can be thought of as evoking related subtypes of the parent frame. For example, the lexical units for the self movement frame are

\begin{equation}
\text{(2.1) advance.v, amble.v, back.v, barge.v, bop.v, bound.v, burrow.v, bustle.v, canter.v, caper.v, clamber.v, climb.v, clomp.v, [...] make a beeline.v, make.v, march.n, march.v, meander.v, mosey.v, mosey.v, nance.v, pace.v, pad.v, parade.v, plod.v, pounce.v, prance.v, press.v, proceed.v, promenade.v, prowl.v, ramble.n, repair.v, rip.v, roam.v, romp.v, rove.v, run.v, rush.v, sail.v, sashay.v, saunter.v, scamper.v, scoot.v, scramble.n, scramble.v, scurry.v, scuttle.v, shoulder.v, shrink.v, shuffle.n, shuffle.v, [...] slop.v, slosh.v, slouch.v, sneak.v, spring.v, sprint.n, sprint.v, stagger.v, stalk.v, steal.v, step.n, step.v, stomp.v, storm.v, straggle.v, stride.v, stroll.n, stroll.v, strut.v, stumble.v, swagger.v, swim.n, swim.v, swing.v, tack.v, take to the air.v, taxi.v, tiptoe.v, toddler.v, totter.v, traipse.v, tramp.v, tread.v, trek.v, trip.v, troop.v, trot.v, trudge.v, trundle.v, vault.v, venture.v, waddle.v, wade.v, walk.n, walk.v, waltz.v, wander.v, way.n, whisk.v, wriggle.v}
\end{equation}

All of these words evoke the idea of self-motion, but describing an individual as burrowing vs. meandering vs. swimming produces quite different meanings within that.

### 2.1.2 FrameNet

FrameNet is a database of 1,164 frames in English, with over 10,000 word senses supported by over 170,000 manually annotated sentences\textsuperscript{2}. While the original FrameNet

\textsuperscript{2}as of February 2014
itself is in English, FrameNets have been created in its image for other languages, including Chinese, Swedish and Spanish. FrameNet defines the frames themselves, describes how the lexical units combine syntactically and semantically, and how frame elements relate to the lexical unit that evokes the frame in the first place. The annotations of FrameNet represent the frame element realisations for each sentence, and are sets of triples: a frame element name, a grammatical function and a phrase type. The grammatical function and the phrase type are not shown on the web representation, but are included in the downloadable dataset.

The following are two example annotated sentences from the Hound of the Baskervilles:

(2.2) • Perhaps the thought of that lonely walk across the ill-omened moor was weighing heavily upon his mind.

• Perhaps the THOUGHT\textsubscript{Cogitation} of that lonely WALK\textsubscript{Self\_motion} across the ill-omened MOOR\textsubscript{Biological\_area} was weighing heavily upon his mind.

Selecting a capitalised word (a word marked as evoking a frame), the sentence is reproduced with colour-coding showing how other elements of the sentence relate to the frame evoked by that word. The sentence under each of the three frames evoked within it is given below.

The overlapping nature of frames is even more clearly illustrated by the second example:

(2.3) • The steps passed along the path on the other side of the wall under which I crouched.

• The STEPS\textsubscript{Self\_motion} PASSED\textsubscript{Path\_shape} ALONG\textsubscript{Locative\_relation} the PATH\textsubscript{Roadways} ON\textsubscript{Locative\_relation} the OTHER\textsubscript{Increment} SIDE\textsubscript{Part\_orientational} of the WALL\textsubscript{Architectural\_part} UNDER\textsubscript{Locative\_relation} which I CROUCHED\textsubscript{Posture}. 
Frame elements on FrameNet are named to avoid both (a) a small number of abstract names that have to fit all frames (b) a large number of names so specific that they differ depending on the lexical unit evoking the frame. As a middle ground, frame elements in FrameNet are named relative to each frame, so that all sentences within a frame can be annotated in the same way, but each frame is independently readable without needing to understand how a very general name acts in this specific instance.

### 2.1.3 Relations

Frames are not necessarily stand-alone entities, and are linked by hierarchies and networks. Frame-to-frame relations all have frame-element-to-frame-element relations, and eight types of relation are defined between frames (Fillmore and Baker, 2010, p. 330):

- **Generalisation relations:**
  
  - **Inheritance:** Similar to the idea from object oriented programming, where the child frame is a subtype of the parent frame. All of the parent frame’s frame elements are bound to frame elements of the child frame (though not necessarily given the same name), and the child frame may have further frame elements defined. An instance of a frame is also an
instance of the frames it inherits from.

- **Perspective on:** An abstract event can be viewed from different perspective depending on the lexical item, e.g. *scrutinizing for* and *seeking* are both perspectives on a *searching scenario*.

- **Using:** The parent frame provides necessary background knowledge for the child frame, and some but not all of the parent frame’s core frame elements are bound to frame elements in the child frame e.g. *storing* uses *placing*.

- **Event structure relations:**

  - **Subframe:** One frame is a sub-event of a larger more complex event frame, e.g. *activity abandoned state, activity done state, activity finish, activity ongoing, activity pause, activity paused state, activity prepare, activity ready state, activity resume, activity start* and *activity stop* are all subframes of *activity*.

  - **Precedes:** Self-explanatory, a temporal ordering between frames, e.g. pre-x precedes x precedes post-x.

- **Systematic relations:**

  - **Causative of:** The parent frame is the causative of the child frame e.g. *killing* is the causative of *death*.

  - **Inchoative of:** The parent frame causes the beginning of the (stative) child frame e.g. *death* is inchotive of *dead or alive*.

To take the frame *awareness* as an example, *awareness* is a perspective on *awareness situation*, inherits from *mental activity* and is inherited from in turn by *expectation and grasp*, uses *information* and is used in turn by *certainty, examination, fame, impression, reliance on expectation, religious belief, remembering information, representing* and *secrecy status*. The complex network of frames surrounding *awareness*
generated by FrameNet is shown here, though more as an example of the size of a network surrounding a single frame than as a readable example.

2.2 Frames in Computer Science

Frames in artificial intelligence are traced back to Minsky (1975). They have the same slot-filler structure as above, and equivalences are easy to draw. The equivalent to the frame/instance-of-a-frame distinction in frame semantics is a distinction between generic and individual frames. Frames organise knowledge by the kind of object it describes. In this sense, frames in AI are object-oriented. Minsky originally suggested frames as a method of coping with new situations by recognising them as similar to situations already known/learned, but the applications are greater than simple recognition and frame-based languages are now are used in knowledge representation. Where reasoning is involved, Hayes (1981, p. 456) notes that common-sense reasoning often involves reflexive reasoning, where the thinker must reflect on themselves, not just the world being considered. This is worth keeping in mind for the later reconstruction of anankastic conditionals.

2.2.1 A sample frame system

The following sample frame system is taken from Brachman and Levesque (2004, p. 136–141). The chapter defines a simple frame system for example purposes, but other accounts are possible. For explanation’s sake, we will focus on their specific example for now.
In this system, individual frames represent individual objects, and generic frames represent classes or categories of objects. An individual frame will have an extra slot, :INSTANCE-OF, which is filled by the name of the generic frame of which it is an instance. Generic frames can also be specialisations of other more general generic frames, indicated by use of an :IS-A slot. Generic frames can be used to specify default values, and procedures that can be used to change values. For example, an instance of a Lecture frame created with a :Date may have the :DayOfWeek computed automatically by an IF-ADDED procedure defined in the generic Lecture frame to fire on the addition of a date. Similarly an IF-NEEDED procedure can be defined to fire when some slot is left empty, and calculate a value for it by using other fillers in the frame (Brachman and Levesque, 2004, p. 139). In this way, the procedures act as constraints between the slots of a frame. This also means that frames systems are defeasible, because defaults and previous information can be overridden by rules firing upon the inclusion of new data, even when the new data is for a different slot. The reasoning loop of a frame system is (Brachman and Levesque, 2004, p. 140)

- An object or situation is declared to exist, instantiating some generic frame

- Slot fillers which are not provided explicitly in the new frame instance are inherited where possible

- For slots with fillers, any inherited IF-ADDED procedure is run

Each IF-ADDED step may fill new slots, thereby triggering further IF-ADDED procedures and possibly instantiating new frames. If a slot filler is required (by a procedure, the user, the system) then

- If the slot has a filler, that value is returned

- If the slot does not have a filler, any inherited IF-NEEDED procedures are run to calculate the filler

Note that running the IF-NEEDED procedures in the second case may cause calls to IF-ADDED procedures, fill other slots and instantiate new frames.
2.2.2 Implemented frames

As mentioned, this is only an example of how a frame system can work. Some examples of frames in practice are given below:

- **KRL**: A single name can be used for a slot by different frames to mean different relations (Hayes, 1981, pp.453). Objects are described by comparison to others, frequently a prototype representing a stereotypical member of a class Bobrow and Winograd (1977).

- **KL-ONE**: First appearing in 1977 Brachman and Schmolze, an early frame language that heavily influenced other frame languages which came after it. Has primitive structure types (such as *Concept* and *Role*) and structure-forming operations (such as *specialization* and *restriction*). Frames are called *Concepts*, and all frames must be traced back via levels of inheritance to a single superframe.

- **KEE**: A development tool rather than a language, it provides a graphical user interface for the frames. KEE supports the idea of multiple worlds, and to that end a slot can have multiple values rather than a single filler. Slots have facets, which may be values or inference rules for that slot.
Chapter 3

Formal Semantic Background

3.1 Introduction

With all of that in mind, we now contrast two traditions in formal semantics: one which operates from the top down, and one which operates from the bottom up.

3.1.1 Possible worlds

In classical possible-worlds based semantics, meaning is created by constraining in one way or another an infinite set of possible worlds. A proposition is true in a world if matches part of that world e.g. *Mary has a dog* is true if Mary owns a dog at the present time in the real world. According to possible worlds semantics, meaning describes both actual and hypothetical objects, and so we consider not only the real world but also all possible worlds. For example *green* describes the set of all things that are actually green and all things that are hypothetically green. A possible world is a hypothetical world which differs from reality in any way (Kearns, 2011). This may be as immaterial as whether some individual is wearing odd or matching socks, or as major as a completely alternate universe. If in a particular world some item is no longer in the set of *green* things, the meaning of *green* itself does not change, but the set of objects it specifies *in that world* does. In this way, the meaning of a predicate
is independent from the set of items it denotes. Worlds themselves are atomic, in the sense of being maximal: different propositions will be true or not at different timepoints, but when discussing a world, the whole world is held in consideration (including its entire timeline) even if currently focussing on a particular timepoint and particular individuals. As well as possible worlds themselves being maximal, the set of all possible worlds is by necessity infinite. From all possible worlds to certain possible worlds, meaning is specified from the top down.

### 3.1.2 Propositions as types

Propositions-as-types semantics is proof-theoretic rather than model-theoretic and offers a more syntactic and structured representation of meaning. Rather than considering a whole world and checking the truth-value of a proposition at some point, propositions are seen as individual objects that can be considered directly. In this sense meaning is - in contrast to possible worlds - constructed from the bottom up. The Curry-Howard correspondence (a.k.a. the propositions-as-types principle) identifies propositions with types, and proofs with programs (Harper, 2013), serving as a link between computation and logic. According to the propositions-as-types principle, if an object is of some type, then it is a proof of the proposition equivalent to that type.

### 3.2 Examples of interpretation

#### 3.2.1 $[A \text{ and } B]$

In a possible-worlds interpretation, a world will satisfy $[A \text{ and } B]$ if it satisfies both $A$ and $B$ individually. In other words, it must be in the set of worlds that satisfies $A$ and also the set of worlds that satisfies $B$ - an element of the intersection of those two sets.

- $[A \text{ and } B] = [A] \cap [B]$
• \( w \models [A \text{ and } B] \) iff \( w \models [A] \) and \( w \models [B] \),

In a propositions-as-types interpretation, we require an object of type \([A \text{ and } B]\), so instead of selecting the intersection of two sets, their cross product is used. A proposition will satisfy \([A \text{ and } B]\) if it has an element satisfying \(A\) and an element satisfying \(B\). In object/type terms, we require an object that itself contains an object of type \(A\) and an object of type \(B\).

• \([A \text{ and } B] = [A] \times [B]\)

• \( p \in [A \text{ and } B] \) iff \( (\exists a \in [A]), (\exists b \in [B]), p = \langle a, b \rangle \)

It should be self-evident that a tuple of two propositions is much smaller than a world in which two propositions are satisfied.

An expansion from that tuple to a record is straightforward. The example record type for \([A \text{ and } B]\) has two fields: \(\ell_1\), which has to be filled by an object of type \(A\), and \(\ell_2\), which has to be filled by an object of type \(B\).

\([A] \times [B] \leadsto \begin{cases} \ell_1 : A \\ \ell_2 : B \end{cases}\)

The record equivalent to \(\langle a, b \rangle\) is a record where \(\ell_1\) is filled by \(a\) and \(\ell_2\) is filled by \(b\).

\(p = \langle a, b \rangle \leadsto \begin{cases} \ell_1 = a \\ \ell_2 = b \end{cases}\)

With a nod back to frames, the record type is equivalent to the idea of a frame, and the record itself an example of that frame in practice. In the construction of record types, we handle the individual propositions, rather than picking out sets of whole worlds containing them, building a type with fields only for what is needed.

### 3.2.2 \([A \text{ implies } B]\)

To start to tackle something a little more linguistic, we turn now to implication and conditionals. \(A\) can be thought of as some antecedent and \(B\) as some consequent
The material conditional of boolean logic can be written off straight away: its simple truth conditions do not facilitate detailing in any way the types of relations between the antecedent and consequent.

In possible-worlds semantics the approach to modals that has gained the most traction is a Kratzerian account (Kratzer, 1981), which is explained more fully in the chapter on anankastic conditionals. Essentially, however, the set of worlds is constrained twice: first by hard and then by soft constraints. The ‘hard constraints’ are the modal base, which reduces the set of worlds to those in which certain propositions must be true. The ‘soft constraints’ are the ordering source, which further reduce the number of worlds under discussion by selecting the best of that set according to some ranking.

In propositions-as-types, the ‘type’ for an implication is taken as a function space.

- $[A \text{ implies } B] = [A \rightarrow B] = [A] \rightarrow [B]$

- $f \in [A \rightarrow B]$ iff $f$ is a function with domain $[A]$ and range $[B]$

A particular function is a proof of ‘$A$ implies $B$’ if it is a function from an object of type $A$ to an object of type $B$. In the end, however, the propositions-as-types function space doesn’t lend itself particularly well to the variety of conditionals and types of implication found in natural language.

At this point we turn to the use of TTR in an attempt to combine the best of both approaches: the Kratzerian specificity, and the propositions-as-types minimalism. The Condoravdi and Lauer account of anankastic conditionals, the specific type of conditional that will be used, is rooted in beliefs and preferences (though ultimately all within a possible-worlds context). It therefore also provides opportunity to engage with attitudinal states and belief structures. Before that can be attempted, we will need a working understanding of TTR and of anankastic conditionals.
Chapter 4

Anankastic Conditionals

4.1 Overview of Anankastic Conditionals

Anankastic conditionals are conditionals of the form

\[
\text{If you want to } [\text{you}] \text{ go to Harlem, you have to } [\text{you}] \text{ take the A train}
\]

which describe a necessary-precondition relation \(^1\) (usually a necessary-means-of relation) between the internal antecedent and internal consequent. According to the example above, taking the A train is necessary to go to Harlem in an optimal way (e.g. quickly, cheaply). The consequent is construed teleologically i.e. it is about necessary for achieving a goal (rather than, for example, successfully following the law).

An anankastic conditional cannot be identified by surface form alone:

(4.1) If you want to buy a third electric pepper-grinder, you have to re-evaluate your priorities.

\(^1\)Anankastic conditionals are normally described as giving a necessary-means-of relation rather than a necessary-precondition relation. However examples taken as anankastic by Condoravdi and Lauer (e.g. ‘If you want to invite everyone to dinner, your table has to seat at least 20 people.’) include relations which are not ‘means’-based, and so anankastic conditionals are considered, at best, a necessary-precondition relation, with means-of as a particular kind of precondition.
(4.2) If he wants to go there on a Saturday night, he should get his head checked.

In neither (4.1) nor (4.2) is the action in the internal consequent a prerequisite for achieving the desire in the internal antecedent. Furthermore, in these two examples the consequent is related to the entire antecedent (the desire to do something) rather than the internal antecedent (what is desired).

Anankastic conditionals come in strong (*must*) and weak (*should*) varieties: a *should* anankastic conditional conveys that the consequent is the necessary for best achieving the goal of the antecedent. A *must* anankastic conditional conveys that it is the only way, or that it is so vastly preferable to any alternative that the addressee should treat it as the only way. The following example is from Huitink (2005):

(4.3) [Context: You want to go to Harlem, and both the A train and the C train go there. Ruud van Nistelrooij, a football player you are infatuated with, often travels on the A Train]

If you want to go to Harlem, you have to take the A Train.

If you want to go to Harlem, you should take the A Train.

The first example seems false, as there is another way to get to Harlem, but the second true. The weaker anankastic with *should* seems to convey that the inner consequent is best for the agent achieving their goals, while still leaving room for an alternate (inferior) way to achieve them.

They can also perform a variety of functions

(4.4) If you want to get involved, then you should let us know. (Request)

(4.5) If we want to finish by midnight, we should split the work. (Suggestion)

(4.6) If you want to be at the airport on time, I should give you a lift. (Offer)

(4.7) If she wants to get a ticket, she has to start queuing tonight. (Advice)

and have other forms.
(4.8) If you want to go to Harlem, take the A train.

(4.9) If we want to finish by midnight, let’s split the work.

and can be construed specifically or generically. For the sake of simplicity the Con-
doravdi and Lauer account focuses specifically on anankastic conditionals which are strong, specific, and in the form of a modalised declarative.

According to Levinson, citing Davis, want has at least two interpretations:

- want = having a desire
- want = having a desire and intending to act on it

The first interpretation reads as a statement that (as a matter of psychological fact) the individual has a particular desire. The second reads as the individual planning to act on a desire which they have. On the reading in anankastic (and near-anankastic) conditionals, want has the second interpretation. Desires or preferences which are ‘action-relevant’ are effective preferences, and are distinguished from their general preferences and desires. Not all of an agent’s desires or preferences will guide their actions, and desires can contradict each other; once an agent plans to realise certain desires (i.e. if they are effective preferences), then those desires must be consistent.

Incompatible goals to the one in an anankastic statement therefore cannot be taken into account; at best, if the speaker knows that the agent does not effectively prefer the goal in the internal antecedent, they must use a counterfactual version:

(4.10) If you wanted to go to Harlem, you would have to take the A train.

While conflicting goals are ignored, those which are not in conflict with the inner antecedent goal can be taken into account:

(4.11) [Context: There are two trains to Vladivostok, one Russian and one Chinese.

The Chinese train is far more comfortable]

If you want to go to Vladivostok, you have to take the Chinese train.
CHAPTER 4. ANANKASTIC CONDITIONALS

The second seems true, despite being otherwise parallel to the first part of (4.3), if the discomfort of the Russian train is so extreme as to make it an unviable option for the addressee given their priorities and preferences. If the Russian train were much cheaper to compensate, and the addressee ranked cost above comfort, the second example would be judged as false. The different interpretations according to ranking of preferences explain the variability in judgements for the second statement.

In Condoravdi and Lauer’s view of anankastic conditionals, they are not particularly unusual or unique, and their internal machinery is just one particular configuration of elements that work similarly elsewhere. This is based on the identification of compositionality issues shared by anankastic conditionals and other similar conditionals. In this sense, anankastic conditionals can be thought of as part of a larger subset of conditionals which share the same basic semantics.

‘Near’-anankastics are distinguished from similarly formed non-anankastics by want targeting the agent’s effective preference structure, and anankastic conditionals further distinguished from near-anankastics by the teleological construal of the modal, and the expression of a precondition relation (which can be frequently interpreted as means-of). Two examples of near-anankastics are given below:

(4.12) If you want to go to Disneyworld, you should spend at least 5 days there.

(4.13) If you want to use the exemption now, you will have to pay more taxes next year.

The first is teleological, but is a consequence rather than a precondition of fulfilling the goal of going to Disneyworld (assuming that ‘going to Disneyworld’ is taken to mean having an enjoyable time and not just being physically present) - and whether you consider ‘go’ to indicate spending time there or arriving there, the internal consequent clearly cannot take place before either. The second example involves deontic rather than teleological consequences: paying more taxes will fulfil the law, but outside of lawfulness isn’t strictly necessary for using the exemption. The issues these
and other examples share with full anankastic conditionals are detailed in (Condoravdi and Lauer, 2014).

4.2 Formal Description through Possible Worlds

4.2.1 Preferences

Condoravdi and Lauer (2011) model ranked preferences using preference structures, and assume that at any given world \( w \), an agent \( A \) has a set of such structures, \( \mathcal{P}_w(A) \), representing the sources of their preferences.

\begin{equation}
\text{(4.14)} \quad \text{A preference structure relative to an information state } W \text{ is a pair } \langle P, \leq \rangle, \text{ where } P \subseteq \wp(W) \text{ and } \leq \text{ is a partial order on } P.
\end{equation}

\( P_w(A,p) \) expresses that \( p \) is a maximal element of agent \( A \)'s preference structure \( P \) in world \( w \).

An agent is also assumed to have a distinguished preference structure for those preferences which influence their actions i.e. a distinct effective preference structure. This preference structure \( \langle P_w(A), \leq_{P_w(A)} \rangle \) for effective preferences is designated as \( EP \), and must be consistent:

\begin{equation}
\text{(4.15)} \quad \text{A preference structure } \langle P, \leq \rangle \text{ is consistent iff for any } X \subseteq P \text{ if } \cap X = \emptyset, \text{ there are } p, q \in X \text{ such that } p < q.
\end{equation}

The effective preference structure is derived from the agent’s overall set of preference structures, which may contain inconsistencies. The derivation of effective preferences thus must include conflict-resolution via strictly ranking incompatible preferences.

4.2.2 ‘want’

Condoravdi and Lauer (2014) proposes that \textbf{want} is interpreted relative to a preference structure. The value for this preference structure is unspecified, and is instead
resolved by context. Where the contextual parameter $P$ is resoled to be one of the members of $\mathbb{P}_w(a)$

\[(4.16) \text{want}_P(a, \Phi) \text{ is true in } w \text{ iff } P_w(a, \lceil \Phi \rceil)\]

Note that if the preference structure $P$ is not consistent, then $\text{want}_P(a, \Phi)$ and $\text{want}_P(a, \Psi)$ can both be true even if $\Phi$ and $\Psi$ are known to be incompatible. If the targeted preference structure must be consistent however, then if $\Phi$ and $\Psi$ are known to be incompatible, $\text{want}_P(a, \Phi)$ and $\text{want}_P(a, \Psi)$ are also incompatible.

### 4.2.3 Anankastic Conditionals

The interpretation of anankastic conditionals is based on a Kratzerian analysis of modals (Kratzer, 1981). In this analysis modals are interpreted relative to two contextually set parameters, which are functions from worlds to sets of propositions:

- Modal base: specifies a set of background facts held constant in the interpretation of the modal
  
  (will be referenced using $f$)

- Ordering source: orders the set of worlds determined by the modal base according to some ideal
  
  (will be referenced using $g$)

The modal base and ordering source may be thought of as hard and soft constraints, respectively.

$Opt(w, F, g)$ selects optimal worlds based on the current world of evaluation($w$), the modal base ($F$) and an ordering source($g$).

\[
Opt(w, F, g) = \{u \in F | \exists v : u <_{g(w)} v\} \tag{4.17}
\]

\[
v \leq_{g(w)} u \iff \{p \in g(w) | v \in p\} \subseteq \{p \in g(w) | u \in p\} \tag{4.18}
\]
CHAPTER 4. ANANKASTIC CONDITIONALS

$Opt$ picks all worlds $u$ satisfying the facts of the modal base $F$, where there is no world $v$ which is ranked more highly than $u$ by the ordering source as applied to the current world of evaluation $w$. More briefly, these are the best possible worlds given the (a) the facts and (b) what is to be prioritised.

When interpreting a modal,

$$[	ext{Must}_{f,g}(\Phi)] = \{ w | \forall v \in Opt(w, \cap f(w), g) : v \in \lfloor (\Phi) \rfloor \}$$

Starting from a modal base and ordering source, a modal gives us worlds of evaluation, which are all worlds that are elements of the set selected in $Opt(w, \cap f(w), g)$ that are found in an interpretation of $\Phi$. $f(w)$ gives the set of propositions that fully characterise $w$.

A conditional is a generalisation from modals, and has the structure $\text{Modal}[\text{antecedent}][\text{consequent}]$. The antecedent restricts the modal base of the modal, but does not affect its ordering source.

$$\text{If } \Psi, \text{must}(\Phi) \rightsquigarrow \text{Must}[\Psi][\Phi] \quad (4.19)$$
$$\text{If } \Psi, \Phi \rightsquigarrow \text{Nec}[\Psi][\Phi] \quad (4.20)$$
$$[\text{Must}_{f,g}[\Psi][\Phi]] = \{ w | \forall v \in Opt(w, \cap f^+(w), g) : v \in \lfloor (\Phi) \rfloor \}, \quad (4.21)$$

where for any $w, f^+(w) = f(w) \cup \{ \lfloor \Psi \rfloor \} \quad (4.22)$

The interpretation of an "if $x$, (then) $y$" style statement includes the influence of an antecedent, and the worlds $w$ selected satisfy a modal base which includes any further propositions found in the interpretation in the antecedent $\Psi$.

The $\text{Modal}[\text{antecedent}][\text{consequent}]$ representation is inadequate for anankastic conditionals, however:

(4.23) $\text{Must}[\text{you want to go to harlem}][\text{you take the A train}]$

The above orders the worlds where the addressee wants to go to Harlem by how best they satisfy the addressee’s goals (whatever they may be), rather than ordering all worlds by how well they satisfy the addressee’s goals including the assumed goal.
CHAPTER 4. ANANKASTIC CONDITIONALS

of going to Harlem. The use of a double modal structure for conditionals with root
modals was suggested by Frank (1997), and this structure is applied to anankastic
conditionals.

(4.24) \( \text{Nec} [\text{you want to go to Harlem}] [\text{Must}] [\text{you take the A train}] \)

(4.25) \( \text{Nec}_{f_{1, g_{1}}} [\text{want}_{p} (Ad, Harlem)] [\text{Must}_{f_{2, g_{2}}} [\text{ATrain}]] \)

(4.26) \( \text{Nec}_{f_{\text{belS}}, g_{\text{norm}}} [\text{want}_{EP} (Ad, Harlem)] [\text{Must}_{f_{\text{hist}}, g_{\text{epA}}} [\text{ATrain}]] \)

\begin{align*}
    f_{\text{belS}}(w) &= \text{Speaker’s true beliefs at } w, t \\
    g_{\text{norm}}(w) &= \text{what is normal/typical at } w \\
    f_{\text{hist}}^t : \text{historical alternatives of a world at time } t \\
    g_{\text{epA}}(w) &= \text{Addressee’s effective preferences}
\end{align*}

Algorithically, 4.26 describes the following:

Among the worlds in which you have the goal of going to Harlem and
nothing unexpected happens, those in which you realise your goals in an
optimal way are worlds in which you take the A train.

In the formalisation, there is a double-modal structure: one explicit (from the sen-
tence) and one implied, the ‘necessary’. Each of these has a modal base and an
ordering source. Starting from the set of all possible worlds \( w \), this is reduced to the
worlds that are in accordance with the speaker’s true beliefs \( (f_{\text{belS}}(w)) \), and then fur-
ther reduced to the set which is most normal. The ordering source is by ‘normality’ -
there are no strikes, nothing is on the tracks, street traffic does not mysteriously
disappear to make a journey by car faster and easier, etc.. It is then specified that
it is necessary in the worlds under consideration that the addressee wants to go to
Harlem, i.e. that going to Harlem is the maximal element of their effective preference
structure. In terms of \( \text{Opt} \) from earlier, this step is

\begin{align*}
    \forall w' \in \text{Opt}(w, \bigcap f_{\text{belS}}, g_{\text{norm}}) : w' \in [[\text{consequent}]] \\
    f_{\text{belS}}^+(w) &= f_{\text{belS}}(w) \cup [[\text{Addressee effectively prefers to go to Harlem}]]
\end{align*}
With this smaller set of worlds based on the inner antecedent, we move on to the second modal, MUST. The modal base here is the set of worlds in that set at some time \( t \). The time \( t \) is generally the time at which the antecedent is true (the time of wanting rather than achieving the wanted). Those worlds are then ordered by how they best fulfil the addressee’s effective preferences - the maximal element of which has already been specified to be going to Harlem.

\[
\forall w'' \in \text{Opt}(w', \bigcap f_{Hist}^t, g_{epA}) : w'' \in [\text{consequent}]
\]

The modal then states that given all this, the set of worlds we finally end up with must also be worlds in which the addressee takes the A train, \( w'' \in [\text{Addressee takes the A train}] \).
Chapter 5

TTR

5.1 Introduction

TTR, a particular kind of type theory with records, partly descended from Martin-Löf’s intuitionistic type theory (Martin-Löf, 1984), which follows the propositions-as-types principle

- If $p$ is of type $hug(a,b)$, $p$ is a proof that $a$ hugs $b$
- $p$ is a proof that $a$ hugs $b$ if it is a situation in which $a$ hugs $b$

According to the propositions as types principle, a proposition (e.g. $hug(a,b)$) is a type of proof. TTR expresses those propositions along with any required context (e.g. a requirement that $a$ and $b$ are both individuals) in a record type. Rather than unstructured sets and functions, records are structured semantic objects similar in style to the attribute-value matrices already familiar in linguistics. Records and record types are complex semantic objects, whose inner parameters can be accessed and manipulated via labels. TTR also makes use of $\lambda$-calculus for functions.

Cooper and Ginzburg claim in (Cooper and Ginzburg, 2013, p. 3) that TTR can build on previous insights in semantics (such as Montague Semantics), leave room for future development, and enable/allow/support ‘a uniform theory of grammar,
semantics, and interaction’ which can handle problems which appear in possible-worlds semantics and in typed-feature-structure syntax-semantics interfaces.

On a philosophical level, (Cooper, 2012, p. 275) presents TTR as part of a theory of cognition, and an important part at that. For example, in human cognition we are able to consider multiple alternative typings for some observation (whether an object or a situation), and can treat the types themselves as cognitive objects. We also appear to use smaller cognitive objects to construct larger ones whose components can then also be accessed and modified (Cooper, 2012, p. 277). TTR allows us to handle all of these: alternative typings via records which contain the requisite fields of multiple types, types as cognitive objects through the stratification of types, and composition and manipulation of components in its very notation. Taking a view that linguistic processing is unlikely to be totally distinct from all other cognitive processing, TTR is suggested as a way to develop a single type theory not only for linguistic cognitive processing, but all human cognitive processing (Cooper, 2012, p. 277).

5.2 Notation

The notation descriptions here are based on the description of TTR in Cooper (2012) and Cooper and Ginzburg (2013). The description is relatively basic, and there is far more to be said about the specifics of TTR, but the following explanation should be sufficient for understanding the examples in the next section, and the description of anankastic conditionals in the next chapter.

In TTR, a record type is a set of fields, each of which is a label-type pair. The ‘type’ of a field may be dependent on other fields, and so be written as a path leading to the object it depends on. A record, a label-value set, belongs to a record type if its labels include the labels specified by that record type, and have as their value an object of the type specified under that label in the record type. This does not

TTR contains both basic and complex types: basic types can be thought of as
basic ontological categories, and complex types as situations. An example of a basic type might be $\textit{Ind}$, representing an individual, or $\textit{Loc}$, representing a location. The shorthand for ‘$a$ is of type $T$’ is $a : T$.

The simplest complex type is a predicate with arguments. Borrowing the ‘hugging’ example from Cooper and Ginzburg (2013), if a boy called Bill hugs a dog called Dinah, then the situation can be constructed using a two-place predicate ‘hug’ with arity $\langle \textit{Ind}, \textit{Ind} \rangle$: $\textit{hug}(b,d)$. Predicates may have more than one arity: for example, if trying to describe a situation where members of a celebrating sports team hug each other, it may be more sensible to introduce a version of ‘hug’ with an arity of $\langle \{\textit{Ind}\} \rangle$ rather than listing all two-person combinations. The predicate ‘hug’ would then be said to be polymorphic. A type which is constructed by a predicate is called a $\textit{ptype}$.

Using these as building blocks we can now start to build other complex types that describe more general situations (instead of those limited to a particular set of individuals). To construct a type for all situations where some boy hugs some dog, we create a record type:

$$
\begin{align*}
\text{x} & : \textit{Ind} \\
\text{c}_{\text{boy}} & : \text{boy(x)} \\
\text{y} & : \textit{Ind} \\
\text{c}_{\text{dog}} & : \text{dog(y)} \\
\text{e} & : \textit{hug(x,y)}
\end{align*}
$$

This describes situations in which one individual is a boy, another is a dog, and the boy hugs the dog. The $c$ with subscript will be used to denote a constraint/condition. However, its type as written here is actually shorthand: the arguments in $\text{boy(x)}$, $\text{dog(y)}$ and $\text{hug(x,y)}$ are technically labels, not $\textit{Ind}$ as required. Instead of $\textit{ptypes}$ themselves, the ‘real’ notation contains pairs: a function from individuals to $\textit{ptypes}$, and paths to where in the record the argument(s) for that ptype can be found. Expanded to the full notation, the record type for boys hugging dogs is
To talk though the function for \( \text{boy}(x) \) as an example, the function is from some \( v \) of type \( \text{Ind} \) to the ptype \( \text{boy}(v) \), and the value to be used for that argument can be found under the \( x \) label. For the two-place \( \text{hug}(x, y) \), the function in the pair is a function from an individual to a second function, which is itself a function from a second individual to the ptype \( \text{hug}(\text{Ind}, \text{Ind}) \) which takes the individual from the outer and inner function as its arguments. The path to the first variable leads to the \( x \) label, and that for the second leads to the \( y \) label. Path names to the values under labels can be specified as \( r.x \), where \( r \) is some record and \( x \) a label within that record.

The above record type is for all situations in which a boy hugs a dog, but we can do better than representing \( \text{a boy hugs a dog} \) and represent \( \text{Bill hugs Dinah} \). We presuppose that there is an individual named Bill and an individual named Dinah, and specify this context in a record type.

\[
\begin{align*}
x & : \text{Ind} \\
c_{\text{boy}} & : \langle \lambda v : \text{Ind}(\text{boy}(x)), \langle x \rangle \rangle \\
y & : \text{Ind} \\
c_{\text{dog}} & : \langle \lambda v : \text{Ind}(\text{dog}(y)), \langle y \rangle \rangle \\
e & : \langle \lambda v_1 : \text{Ind}(\lambda v_2 : \text{Ind}(\text{hug}(v_1, v_2))) \rangle(x, y) \rangle
\end{align*}
\]

\( \text{Bill hugs Dinah} \) is then expressed with a function that takes this record of this type as context, and returns a record type similar to the previous record type, but with the dog and boy in question traced back to Bill and Dinah.
Cooper and Ginzburg (2013) defines several operations which will be useful in our later description of anankastic conditionals. The first is an asymmetric merge operator \( \boxtimes \), which extends merge \( \times \). The effects of merging two record types is shown by example:

\[
\begin{align*}
&\begin{bmatrix}
  f & : T_1 \\
  g & : T_2
\end{bmatrix} \boxtimes \begin{bmatrix}
  h & : T_1 \\
  k & : T_2
\end{bmatrix} =
\begin{bmatrix}
  l & : T_1 \\
  m & : T_2
\end{bmatrix}
\end{align*}
\]

We will use the version of asymmetric merge used in Larsson (2013), which is identical to merge except in cases where some label is shared by the record type on the right and the record type on the left. In this case, the argument on the right will be given precedence. The following example taken from the same paper illustrates its effects:

\[
\begin{bmatrix}
  t_{right} = 0.090 \\
  t = t_{right} \\
  bg = ... \\
  f = ...
\end{bmatrix} \boxtimes \begin{bmatrix}
  t = 1.4*0.090 \\
  bg = ... \\
  f = ...
\end{bmatrix} = \begin{bmatrix}
  t_{right} = 0.126 \\
  t = t_{right} \\
  bg = ... \\
  f = ...
\end{bmatrix}
\]

The partial order set from Cooper and Ginzburg (2013) will also be useful to describe preference structures, and is specified as Poset(T),

\[
\begin{bmatrix}
  \text{set} & : T \\
  \text{rel} : \left\{ \begin{bmatrix}
    \text{left} & : T \\
    \text{right} & : T
  \end{bmatrix} \right\} \\
  \text{c}_{po} & : \text{po}(\text{rel}, \text{set})
\end{bmatrix}
\]
where \( a : \text{po}(R, S) \) iff \( a = \langle R, S \rangle \) and \( R \) is a partial order on \( S \), and \( R \) is reflexive or irreflexive, antisymmetric and transitive.

Finally, the following definition of \( \oplus \) will also be used.

\[
(5.4) \quad r.set = P.set \cup \{ a \}, \\
r.rel = P.rel \cup \{ \begin{bmatrix} \text{left} = a \ \text{right} = x \end{bmatrix} \mid x \in p.set \} \\
\text{and } r.c_{\text{po}} = \langle r.rel, r.set \rangle.
\]

### 5.3 TTR in action

TTR has various applications for modelling, some examples of which are given below.

Noting Fillmore’s own remarks that frame semantics in its original form is “‘pre-formal’ rather than ‘non-formalist’ ” (Fillmore), Fin directly connects formal and frame semantics via TTR, and provides TTR ‘versions’ of frames described in FrameNet. The slot-filler of frames is easily identified with the label-value structure of TTR. In TTR, the ‘frame’ of frame semantics is equivalent to a record type, and a particular instance is equivalent to an individual record. The core frame elements of ambient temperature as described in FrameNet are

\[
(5.5) \quad \text{Attribute:} \quad \text{The temperature feature of the weather} \\
\text{Degree:} \quad \text{A modifier expressing the deviation of the Temperature from the norm} \\
\text{Place:} \quad \text{The Place where it is a certain Temperature} \\
\text{Temperature:} \quad \text{A quantity or other characterization of the Temperature of the environment.} \\
\text{Time:} \quad \text{The Time during which an ambient environment has a particular Temperature.}
\]
Using the notation that has been introduced, the following record type (Fin, p. 3) expresses a slightly simplified version of the ambient temperature frame from FrameNet:

\[
\begin{bmatrix}
x & : \text{Ind} \\
e\text{-time} & : \text{Time} \\
e\text{-location} & : \text{Loc} \\
ctemp\_at\_in & : \text{temp\_at\_in}(e\text{-time}, e\text{-location}, x)
\end{bmatrix}
\]

The record type above states that at a given time in a given location (the 'e' indicates the event), the temperature will be that given by the value of \( x \).

Verbs are treated as a function from frames to frame types, i.e. a function from records to record types (Cooper, 2012). The verb \textit{run} in the past tense is described as the function (Cooper, 2012)

\begin{equation}
(5.6) \lambda r:\left[ x : \text{Ind} \right] \left( \begin{bmatrix}
e\text{-time} & : \text{TimeInt} \\
ctns & : e\text{-time}.\text{end}<\iota.\text{start} \\
cren & : \text{run}(r.x,e\text{-time})
\end{bmatrix} \right)
\end{equation}

Where \( \iota \) is the speech time and \textit{TimeInt} is the type for a time interval. The above is a function which takes in a record with a field \( x \) of type individual, and returns a record type for records in which that individual runs at some point in the past.

Attitudinal objects are (perhaps surprisingly) easy to model in TTR. In (Cooper, 2005) attitudes have record types as their object. The record type given for \textit{A girl believes a man owns a donkey} is
The belief \( p \) is true of some record \( r \) if \( r \) has all of the fields given in the record type. (Cooper, 2005, p. 8)

Cooper (2005) also suggests a way in which TTR can model the fulfilment of intentions.

If the first is a record type for *A girl seeks a unicorn*, then if \( r \) is a record of this type, then a record \( r_t \) is a successful outcome for \( r.x \)'s search if \( r_t \) is of the second type. The search is successful if some \( r_t \) of this type exists (Cooper, 2005, p. 9).

Other more obviously practical applications include modelling action and perception (Dobnik et al., 2013), and learning perceptual classifications (Larsson, 2013).
Chapter 6

TTR Formalisation

Following on from this formalisation for anankastic conditionals, I propose the following formalisation through TTR for the same account. In the spirit of frame semantics, the TTR formalisation is centred on the use of speaker beliefs and attitudes (rather than worlds). For ease of explanation, the Harlem example will be used as a practical example throughout.

A note on language: throughout this chapter use has been made of terms like ‘alter’, ‘update’, ‘insert’, etc. This is slightly misleading: everything below is a description of a record type rather than a record, and nothing new is ‘generated’ per se, but rather specified. Various procedures may be carried out on individual records, but the record types themselves are as is. In the course of explanation though it may be easier, for example, to think of nec as the result of updating spk_att, instead of as a distinct entity whose fields are constrained to have particular relations to the values found in spk_att.

(6.1) AC:
The type for an anankastic conditional has an attitudinal structure for the speaker - which contains their beliefs and preferences - and two propositions: the inner antecedent and inner consequent (each proposition is an agent-goal pair). The second attitudinal structure, \( \text{nec} \), is equivalent to the earlier requirement of the outer modal that the addressee really does want to go to Harlem. \( \text{wantEP} \) is another record type which contains an \( \text{AttStruct} \) labelled \( \text{revised} \). In the 'revised' attitudinal structure the agent in \( \text{ant} \) has the goal of \( \text{ant} \) as the maximal element of their effective preference structure, and is otherwise the same as the speaker’s original attitudinal structure.

Finally, there is the \( \text{must} \) constraint, which has two arguments. While \( \text{want} \) is further defined below, \( \text{must} \) and \( g \) are treated as black boxes. \( g \), which takes as its arguments an attitudinal structure and an individual, states that that individual optimally achieves their goals - according to whatever that attitudinal structure says those goals are. The \( \text{AttStruct} \) used here is \( \text{nec} \), in which the agent’s maximal effective preference has already been specified as the goal of \( \text{ant} \). As previously mentioned, anankastic conditionals do not specify the type of relation between the inner antecedent and inner consequent beyond a general necessary-precondition relation. Given the statement that that individual optimally achieves their goals (according to \( \text{nec} \)), \( \text{must} \) asserts that this achievement is preceded by the consequent happening.

A value for ‘optimally’ is not, I believe, possible to define here, hence the introduction of black-box assertions. Optimality will be some combination of what the speaker believes about the relevant agent’s preferences, and what the speaker’s own preferences are. For a statement to be valid all that matters is for the speaker to
believe it to be so \(^1\), not for it to be ‘true’ in some external objective sense.

An example is the Chinese train example from the previous chapter, repeated here:

- If you want to go to Vladivostok, you have to take the [Chinese/Russian] train.

The validity of that statement depends upon the relative importance of comfort and cost to the addressee. However, a particular individual may know but disregard the addressee’s preferences, because they believe that despite the addressee’s desire to save money, the Russian train is simply unsuitable. In this case the speaker could still stand by the Chinese train as being a necessity. What this example is intended to illustrate is that the notion of what is ‘optimal’ for a given agent is highly individualised, and different speakers may dismiss or allow alternative solutions based purely on their own beliefs and preferences. To this end, I feel that attempting to write some ‘one-size-fits-all’ function is unsuitable.

(6.2) AttStruct:

\[
\begin{bmatrix}
\text{self} & : & \text{Ind} \\
\text{beliefs} & : & \text{BeliefStruct} \\
\text{prefs} & : & \text{PrefSet}
\end{bmatrix}
\]

The attitudinal structure contains some reference to the self, a set of beliefs, and a set of preferences, all of which are further defined.

(6.3) Prop:

\[
\begin{bmatrix}
\text{agent} & : & \text{Ind} \\
\text{goal} & : & \text{Goal}
\end{bmatrix}
\]

\(^1\)This is not contradictory to the existence of counterfactual anankastic conditionals. Counterfactual anankastics acknowledge a conflict between what the speaker believes the agent’s effective preferences are and what the anankastic requires them to be. If the speaker knows the addressee does not want to go to Harlem then ‘If you want to go to Harlem, you have to take the A train’ is unacceptable, but ‘If you wanted to go to Harlem you would have to take the A train’ is fine.
The record type being used to describe propositions here is a simple agent-goal pair.

(6.4) BeliefStruct:

\[
\begin{bmatrix}
w & : \{\text{Belief}\} \\
\text{others} & : \{\text{Ind, PrefSet}\} \\
c_{\text{ind}} & : \not\exists x, y \in \text{others} | x[0] = y[0], x \neq y
\end{bmatrix}
\]

The belief structure contains a set of beliefs - which are taken to be an atomic type - and, distinct from that, a set of tuples consisting of sets of preferences indexed by the individual believed to hold them. The clear distinction between beliefs and preferences is for admittedly practical purposes, as the preferences of an individual will need to be accessed and altered in the specification of a new maximal effective preference. Note that these individuals are not assigned a full BeliefStruct: the only relevant part of their mental state is the set of their preferences. There is also a constraint to ensure that an individual can only be assigned one set of preferences.

(6.5) PrefSet:

\[
\begin{bmatrix}
\text{ep} & : \text{PrefStruct} \\
\text{genPrefs} & : \{\text{PrefStruct}\} \\
c_{\text{consistent}} & : \text{consistent(ep)}
\end{bmatrix}
\]

A PrefSet contains, as the name suggests, a preference structures, but it also holds one particular preference structure distinguished from the others. This distinguished preference structure is for effective preferences and is therefore constrained by the need to be consistent.

PrefStruct is Poset(T) as described in the chapter on TTR, where T = Goal.

wantEP contains the machinery for the specification of nec in AC itself. Instead of homing in on the set of worlds where a particular goal is a maximal element of an agent’s EP, we insert a new maximal element in the effective preference structure the speaker already theorises they have. Where att : AttStruct and p : Prop,
(6.6) \( \text{wantEP}(\text{att}, p) = \)

\[
\begin{array}{l}
\text{x} = p.\text{agent} : \text{Ind} \\
\text{revised} : \text{AttStruct} \\
\text{c}_{\text{self}} : \text{revised}.\text{self} = \text{att}.\text{self} \\
\text{c}_{\text{P}(a,p)} : ((x = \text{att}.\text{self}) \\
\quad \land \ (\text{revised}.\text{beliefs} = \text{att}.\text{beliefs}) \\
\quad \land \ (\text{revised}.\text{prefs} = \text{newEP}(p.\text{goal}, \text{att}.\text{prefs}))) \\
\quad \lor \ ((\text{revised}.\text{beliefs} = \text{att}.\text{beliefs} \land \\
\quad \quad \text{att}.\text{beliefs}.\text{others} \land \\
\quad \quad \{ x, \text{newEP}(p.\text{goal}, \text{att}.\text{beliefs}.\text{others}[x])\})) \\
\quad \land \ (\text{revised}.\text{prefs} = \text{att}.\text{prefs})
\end{array}
\]

\text{wantEP} is shorthand for a record type specified according to its two arguments, the first of which is an attitudinal structure and the second of which is a proposition. The first field must contain an individual, \( x \), which is the same as the agent of that proposition. The second contains an \text{AttStruct} called \text{revised}. Everything else in the record type is a constraint on that attitudinal structure. The first constraint is that the \text{self} of \text{revised} must be the same as the \text{self} of \text{att}, the given \text{AttStruct}. After this, the differences between the original and this ‘revision’ or ‘update’ appear. There are two cases:

- **Case 1**: the \text{agent} of \( p \) is the speaker themselves

  In this case, their \text{beliefs} about the world are the same, but their own \text{prefs} are updated so that the goal of \( p \) is now the maximal element of their effective preference structure.

- **Case 2**: the \text{agent} of \( p \) is not the speaker

  In this case, their preferences remain unchanged, but their \text{BeliefStruct} is updated to take into account the new goal. The \text{beliefs} of \text{revised} are the same as
those in att, except that the PrefSet indexed under the individual referenced by x now has \( p.goal \) as a maximal element.

Where \( g : \text{Goal} \) and \( p : \text{PrefSet} \),

\[
(6.7) \quad \text{newEP}(g, p) = p \left[ \begin{array}{l}
\text{ep} = \maxEP(g, p.ep) \end{array} : \text{PrefStruct} \right]
\]

\( \text{newEP} \) specifies a record type which is identical to its \( p \) argument (and so is also of type \( \text{PrefSet} \)) apart from the value of \( \text{ep} \), which now has \( g \) as a maximal element.

Finally, where \( g : \text{Goal} \) and \( \text{ep} : \text{PrefStruct} \), and using \( \oplus \) according to the description in the chapter on TTR,

\[
(6.8) \quad \maxEP(g, \text{ep}) = g \oplus \text{ep}
\]

with the additional constraint that \( \forall x \in \text{ep.rel} \), if \( \text{conflict}(g, x) \) then

\[
\begin{bmatrix}
\text{left} = x \\
\text{right} = g
\end{bmatrix} \notin \maxEP(g, \text{ep}).\text{rel}
\]
Chapter 7

Conclusion

Frames across linguistic semantics and computer science were discussed, and the FrameNet project based in Berkeley in particular was examined as a database of linguistic frames which includes in its data hierarchical structures and various relations between frames. Possible-worlds semantics and propositions-as-types semantics were mentioned and compared through and their varying representations of a few simple examples. The idea of moving from sets of propositions to structured records was also introduced. Anankastic conditionals were then described according to Condoravdi and Lauer’s account, which was developed through possible-worlds semantics. After this TTR was introduced both as context and as a short primer on the particular pieces of syntax that would be used to express the rewritten anankastic conditionals. Finally, the proposed TTR reformalisation of anankastic conditionals was presented and explained.

The most obvious direction for future work is to expand the TTR representation to encompass conditionals more generally rather than a particular subtype, and to explore integrating a more specific notion of events (e.g. ‘going to Harlem’) into the formalisation.

This project aimed to balance two conflicting philosophies of meaning, primarily through presenting a bottom-up reformalisation of a theory that had been originally
developed from a top-down possible-worlds perspective. In addition, the project aimed to engage directly with intentions and attitudes. The linguistic phenomenon chosen to encompass all of these elements was anankastic conditionals as described in Condoravdi and Lauer (2014), Condoravdi and Lauer (2012) and Condoravdi and Lauer (2013). This account was reformulated in TTR, using a more minimal conceptual representation than with possible-worlds. Minimal does not necessarily mean brief, however: 'Minimal' is intended to convey the use of only relevant concepts, rather than solutions that can be summarised in a line or two, as with a possible-worlds representation – a record type, or series of record types, may be relatively large when written down, as every object and constraint required must be listed, while representing a truly negligible fraction of a world. This is an inevitable price to be paid for a more syntactic and compositional representation, and one in which there are no ‘passengers’. It is easy to forget how every set of possible worlds is infinite (whatever the constraints), and that each of those worlds encompasses an entire universe from beginning to end. By taking a bottom-up rather than a top-down approach, the reformation re-expresses the essence of the account of anankastic conditionals, while staying firmly away from these infinite alternative realities.
Bibliography


Brachman, R. J. and Schmolze, J. G.


