Lecture 3 – Temporal Logics in AI (I)

1. Introduction to Temporal Reasoning

2. Types of Logics and Techniques

   (a) Method of temporal arguments

   (b) Method of reification

3. Three Formalisms

   (a) BTK – Bacchus et. al.

   (b) STL – Shoham

   (c) Situation Calculus – McCarthy

Tasks Requiring Temporal Reasoning

1. *Prediction* – Given a description of the world over some period of time, and the set of rules governing change, predict the world at some future time.

2. *Explanation* – Given a description of the world over some period of time and the rules governing change, produce a description of the world at some earlier time that accounts for the world being the way it is at the later time.

3. *Planning* – Given a description of some desired state of the world over some period of time, and the rules governing change, produce a sequence of actions that would result in a world fitting that description.

4. *Learning new rules* – Given a description of the world at different times, produce the rules governing change which account for the observed regularities in the world.

Davis' Methodology

According to Davis' methodology, our first task is to define the microworld and analyze it in three respects. The first level of analysis is the ontology of our domain.

1. **The definition of the microworld: its entities, its relations, and the rules that govern them. This definition is called the ontology of the domain.**

From the tasks listed above, there are a number of fundamental entities that could be included in our ontology:

1. A class of temporal entities and relations between them. The tasks mention specific time points and durations.

2. A state or description of the world.

3. Facts about the world true in some states of the world, false in others.

4. Actions whose effects change the state of the world.

As far as "rules that govern them", there are "rules governing change" and "regularities in the world". It appears that we will need some forms of representation for both the static and dynamic aspects of the world; what changes when actions are performed and what remains unchanged.

Davis' Methodology Con't

The next level of the analysis is

1. **The analysis of the type of information that the system must record and the types of inferences it must perform. The result of this analysis is generally a formal language, in which facts of the microworld can be expressed and axioms or rules of inference, which characterize inferences that are reasonable in the microworld.**

– Variety of inference types needed: Abduction, reasoning with partial information.

– Variety of choices as regards formal language. (We will consider several)

– Extremely complex microworld!

Approaches to Temporal Reasoning

Our temporal representations must be capable of addressing a number of dfiferent issues. We list a few of them:

1. Representation of change.

2. Representation of the occurence of events.

3. Representation of constraints on possible states of the world.

4. Representation of causal laws.

In addition there are a number of issues related to the particular temporal structures we use:

1. Representation of quantitative relations among times and durations.

2. Distinguishing between past, present and future.

3. Reasoning about processes and continuous time.

Two Broad Classes of Temporal Reasoning

– **Change-Based Approaches** - characterized by their event-driven nature. The world is in a particular state until an action, event, or execution of a program, for instance, changes the worlds state. The notion of time is generally implicit in the formalism and one is unable to reason about time explicitly on the object level.

- Situation Calculus - McCarthy **
- Event Calculus - Kowalski and Sergot
- Dynamic Logic - Pratt

– **Time-Based Approaches** - characterized by their time-driven nature. Time progresses independent of any explicit event or activity. There is a time structure, facts are true at various points in the structure, and one can reason about time on the object level.

- Interval Logic - Allen
- Point-Based Logic - McDermott
- Modal Temporal Logic - Shoham **
- BTK - Bacchus, Tenenberg and Kooman **
- Discrete Fluent Logic - Sandewall **

FOPC and Temporal Reasoning

– Some properties of FOPC:

"A fact f is true"

- Deals with universal truths.
- Context independent.
- Monotonic.

– Some properties of Temporal Reasoning:

"A fact f is true at a point t"

- Deals with contingent truths.
- Context dependency.
- Type of nonmonotonicity.

– Obviously two alternatives:

1. Do not use FOPC.
2. Extend it in a variety of ways.

Representing Temporal Information

"A fact is true" at a "point in time"

"A formula is true" in a "model"

– Several useful techniques:

1. Method of Temporal Arguments.

2. Method of Reification.

– Outside FOPC

1. Modal Logics

2. Non-standard semantics.

In discussing the two techniques, we will assume a simple time structure consisting of situations.

Definition 1 (Situation) *A* situation*, or* state *is a snapshot of the world at a given point in time. At different points in time, the world can be in different states.*

Situations will be considered as first-class entities. Consequently they can be used as arguments to first-order predicates and terms. Later on, we will use more explicit and complex temporal entities such as timepoint structures and interval structures, but for the time being situations will suffice.

Method of Temporal Arguments

- Fact f is a predicate constant P.
- Introduce temporal variables and constants as terms in the language.
- Temporal terms used as arguments to predicates.
- Essentially an *n*-sorted logic.

– "Fact f is true in situation $S1$"

$$P(S1)$$

– "Location of $block1$ in situation $S1$ is the $table$

$$Location(S1, Block1, Table)$$

– or alternatively using a functional term

$$location(S1, Block1) = Table$$

Pros and Cons:

- No distinction between temporal and atemporal aspects either in the syntax or semantics.

  $P(red(S1))$ and $P(red(house1))$ are legal formulas.

- Syntactic complexity: repeating temporal arguments.

  When using functions: $Dead(S1, pres(S1, USA))$

- Apparent lack of expressibility.

  "Effects of actions come after causes"

- On other hand: developed completely within FOPC framework.

  Access to results and theorem provers.

Method of Reification

- Predominantly an AI approach. Montague, McCarthy.
- Time-varying relations are represented as terms rather than predicates.
- The terms denote *fluents*. A fluent is a function on situations. A fluent whose range is the boolean truth values is called a *boolean* or *propositional* fluent.

–The time-varying relation $Location$ would denote a boolean-valued fluent:

$$Location(b, l) : \langle situation \mapsto \{true, false\}\rangle.$$

Syntactically, we can use the following representation:

$$Holds(location(b, l), s)$$

where

- $Holds$ is a predicate constant,
- $location(b, l)$ denotes a boolean fluent,
- s denotes a situation.

– An alternative representation is

$$value(location(b, l), s) = true$$

where $value$ is a function constant taking a fluent and situation as arguments. Both representations have their uses.

– In addition, time-varying terms such as the color of a block, $color$, denote fluents of arbitrary object domains,

$$color(b) : \langle situation \mapsto \{Red, Green, Blue\}.$$

Syntactically, we can use the following representation:

$$value(color(b), s) = Blue.$$

Pros and Cons:

- Philosophically suspect: Introduce fluents into the ontology.

- Sorted logics increase syntactic complexity of the language.

- Since properties are "thingified" as terms, we have the ability to
  - quantify over, and in,
  - use as arguments to functions that return fluents themselves.

For example:

one can easily express the fact that all changes in values of fluents are caused by actions

$$\begin{aligned}&\forall f, s, s'.s < s' \wedge value(f, s) \neq value(f, s') \rightarrow \\ &\quad \exists a.Causes(a, f, value(f, s')).\end{aligned} \tag{1}$$

Three Temporal Formalisms

1. BTK is an example of the time-based approach and uses the method of temporal arguments. It is the most classical in nature and is formulated completely withing the confines of classical FOPC with sorts.

2. STL is an instance of the time-based approach and uses a technique similar to reification but does not quite go all the way. It is a sort of half-animal with a refied and modal flavor.

3. The situation calculus (SC) is an instance of the change-based approach and uses the method of reification. It was the first temporal representation used in AI (1963) and is still quite popular, although its usage and limitations are now a topic of current debate.

- BTK is covered to introduce the idea of a sorted logic and the syntax used in the introductory paper.

- STL is introduced to show the most popular way to define non-standard semantics and to clean up Allen and McDermott's work.

- SC is introduced for obvious reasons.

BTK

– BTK is a two-sorted FOPC using the method of temporal arguments

Syntax

In the 2-sorted case

- the vocabulary $\Lambda = \langle C, F, R, d \rangle$, contains a mapping d defined on $C \cup F \cup R$;
- For each constant $c \in C$, $d(c)$ is either t, or u, representing the sort (t temporal, u atemporal);
- For each $r \in R$, $d(r)$ is a tuple $\langle k_1, \ldots, k_n, k_{n+1}, \ldots, k_m \rangle$, where $d(k_1) =, \ldots, = d(k_n) = t$ and $d(k_{n+1}) =, \ldots, = d(k_m) = u$, where the k_i's specify the type of argument to r;
- For each $f \in F$, $d(f)$ is a structure $\langle k_1, \ldots, k_n, k_{n+1}, \ldots, k_m \to k \rangle$, where again $d(k_1) =, \ldots, = d(k_n) = t$ and $d(k_{n+1}) =, \ldots, = d(k_m) = u$, and $d(k) = t$ or u.

If $d(k) = t$ then it is called a *temporal function* and if $d(k) = u$ then it is called an *atemporal function*. So for both relations and functions, the first n arguments are temporal and the last m arguments are atemporal.

The list of variables is:

$$\xi_1^t, \xi_2^t, \xi_3^t, \ldots, \xi_1^u, \xi_2^u, \xi_3^u, \ldots$$

In each case the superscript shows the sort of variable.

We can now define terms as follows:

Definition 2 *Let* $\Lambda = \langle C, F, R, d\rangle$ *be a vocabulary of the 2-sorted logic. Then*

- *For* $c \in C$, c *is a term of sort* $d(c)$.
- *The variable* ξ_i^j *is a term of sort* j.
- *If* g *is a function symbol of* Λ *with* $d(g) = \langle t_1, \ldots, t_n, u_{n+1}, \ldots, u_m \to k\rangle$, *and* $\mu_1, \ldots, \mu_n$ *and* $\mu_{n+1}, \ldots \mu_m$ *are terms of* Λ *of sorts* t *and* u, *respectively, then*

  $$g(\mu_1, \ldots, \mu_n, \mu_{n+1}, \ldots, \mu_m)$$

  is a term of Λ *of sort* $d(k)$.

Definition 3 *An* atomic formula *of* Λ *is an expression of* Λ *of the form:*

$$r(\mu_1, \ldots, \mu_n, \mu_{n+1}, \ldots, \mu_m)$$

where r *is a relation symbol of* Λ, $d(r) = \langle t_1, \ldots, t_n, u_{n+1}, \ldots u_m\rangle$, *and* $\mu_1, \ldots, \mu_n, \mu_{n+1}, \ldots, \mu_m$ *are terms of* Λ *of sorts* t *and* u, *respectively.*

The definition of *formula* **is exactly the same as for the 1-sorted case.**

Semantics

An *interpretation* I of the 2-sorted vocabularly Λ consists of:

1. *Domains* D_t and D_u each of which is a non-empty set.

2. For each constant symbol c of Λ, an element $c_I \in D_{d(c)}$.

3. For each function symbol g of Λ with
$$d(g) = \langle t_1, \ldots, t_n, u_{n+1}, \ldots, u_m \to k \rangle$$
a mapping g_I of $D_t^1, \times, \ldots D_t^n \times D_u^{n+1}, \times, \ldots D_u^m$ into D_k.

4. For each relation symbol r of Λ with
$$d(r) = \langle t_1, \ldots, t_n, u_{n+1}, \ldots u_m \rangle$$
a mapping r_I of $D_t^1, \times, \ldots D_t^n \times D_u^{n+1}, \times, \ldots D_u^m$ into $\{1, 0\}$.

A *valuation* v of an interpretation I is the sequence

$$v_j^t, v_j^u;\ j = 1, 2, 3, \ldots$$

where each $v_j^t \in D_t$ and each $v_j^u \in D_u$. We say that v_j^k is the value of the variable ξ_j^k with respect to the valuation v. If v and w are both valuations of I, we write $v \equiv_{k,j} w$ to mean that $v_i^k = w_i^k$ for all $i \neq j$.

Now, given these preliminaries, we can modify the 1-sorted version of $VALUE$ as follows:

2. If v is the variable ξ_j^k then $VALUE(v, I, v) = v_j^k$.

6. If $\gamma = (\exists \xi_j^k)\alpha$ then $VALUE(\gamma, I, v) = 1$ if there is a valuation w of I such that $w \equiv_{k,j} v$ and $VALUE(\alpha, I, w) = 1$; otherwise $VALUE(\gamma, I, v) = 0$.

7. If $\gamma = (\forall \xi_j^k)\alpha$ then $VALUE(\gamma, I, v) = 0$ if there is a valuation w of I such that $w \equiv_{k,j} v$ and $VALUE(\alpha, I, w) = 0$; otherwise $VALUE(\gamma, I, v) = 1$.

BTK Temporal Ontology

Given the 2-sorted logic, BTK, we would like to give the temporal subdomain D_t some structure. For example, the temporal subdomain could be the integers, rationals, or the reals, dependent on whether we wanted to work with discrete, dense, or continuous time. We must be careful though. Problems with completeness.

– Example:

Given our choice of temporal domain, how do we use it with BTK? We simply add the axioms which characterize our temporal structure to the axiomatization of first-order logic. The first-order rules of inference will, in union with our extended axiom set, provide a complete proof theory. Let's look at an example. Below we list the first-order axioms which characterize the linearly-ordered temporal structures:

$$\forall x . x \leq x$$

$$\forall xy . x \leq y \wedge y \leq x \rightarrow y = x$$

$$\forall xyz . x \leq y \wedge y \leq z \rightarrow x \leq z$$

$$\forall xy . x \leq y \vee y \leq x.$$

– One of the advantages of FOPC framework

BTK Blocks World Example

– World: Blocks world where blocks can be moved and painted

We will use the following sorts:

1. timepoints (t),
2. individuals (u).

Our vocabulary, $\Lambda = \langle C, F, R, d \rangle$, contains the following:

1. $C = TC \cup C'$ where

$$TC = \{0, 1, \ldots\} \text{ and } \forall x \in TC.d(x) = t,$$

$$C' = \{Block_1, \ldots Block_n, Table, Red, White, Blue\}$$
$$\text{and } \forall x \in C'.d(x) = u,$$

2. $F = TF \cup F'$ where

$$TF = \{+\} \text{ and } d(+) = \langle t \times t \to t \rangle.$$

$$F' = \{top\} \text{ and } d(top) = \langle t \times u \to u \rangle.$$

3. $R = \{At, Color, Clear, Move, Paint, Location\}$
$\cup\{Block, Trivial, Ab_x, \leq\}$ where

$$d(At) = d(Color) = d(Move) = d(Paint) = \langle t \times u \times u \rangle,$$

$$d(Clear) = d(Trivial) = d(Ab_7) = \langle t \times u \rangle,$$

$$d(Location) = d(Block) = \langle u \rangle,$$

$$d(Ab_n) = \langle t \times u \times u \rangle \text{ for n = 1...4},$$

$$d(\leq) = \langle t \times t \rangle,$$

The list of variables for sorts is:

$$\xi_1^t, \xi_2^t, \xi_3^t, \ldots, \xi_1^u, \xi_2^u, \xi_3^u, \ldots$$

To make the axioms more readable we will use the sort type (possibly subscripted) for variables:

$$t, t_1, t_2, t_3, \ldots, u, u_1, u_2, u_3, \ldots \text{etc.}$$

We will use the temporal structure $\langle \mathcal{N}, 0, S, <, + \rangle$, assuming that the proper axioms are added to any of our scenarios.

BTK Axioms

Initial Situation

All blocks are on the table and are white.

$$\forall u.Block(u)\rightarrow At(0, u, Table). \tag{2}$$

$$\forall u.Block(u)\rightarrow Color(0, u, White). \tag{3}$$

Laws of Motion

Blocks change their location when moved and their color when painted.

$$\begin{aligned}\forall t, u, u'.Move(t, u, u')\wedge\neg Ab_3(t, u, u')\\ \rightarrow At(t+1, u, u').\end{aligned} \tag{4}$$

$$\begin{aligned}\forall t, u, u'.Paint(t, u, u')\wedge\neg Ab_4(t, u, u')\\ \rightarrow Color(t+1, u, u').\end{aligned} \tag{5}$$

Frame Axioms

Blocks change their location and color only if there is a reason to change.

$$\forall t, u, u'. At(t, u, u') \wedge \neg Ab_1(t, u, u') \\ \rightarrow At(t+1, u, u'). \tag{6}$$

$$\forall t, u, u'. Color(t, u, u') \wedge \neg Ab_2(t, u, u') \\ \rightarrow Color(t+1, u, u'). \tag{7}$$

$$\forall t, u, u'. Move(t, u, u') \rightarrow Ab_1(t, u, u'). \tag{8}$$

$$\forall t, u, u'. Paint(t, u, u') \rightarrow Ab_2(t, u, u'). \tag{9}$$

Qualification Axioms

Known conditions which prevent a block from being moved.

$$\forall t, u, u'. \neg(Block(u) \wedge Clear(t, top(t, u))) \vee \neg Clear(t, u') \\ \vee u' = top(t, u) \rightarrow Ab_3(t, u, u'). \tag{10}$$

State Constraints

Since we do not use types for the blocks worlds, we must identify our objects.

$$Block(Block_1) \wedge \ldots \wedge Block(Block_n). \tag{11}$$

$$Location(Table). \tag{12}$$

$$\forall u.Block(u) \rightarrow Location(top(u)). \tag{13}$$

$$Color(Red) \wedge Color(White) \wedge Color(Blue). \tag{14}$$

A location is clear if there are no trivial objects on it.

$$\forall t, u.Clear(t, u) \equiv \neg \exists u'.\neg Trivial(t, u') \wedge At(t, u', u). \tag{15}$$

$$\forall t, u.\neg Ab_7(t, u) \rightarrow Trivial(t, u). \tag{16}$$

Querying the System

We can use the axiomatization for various purposes. For example, to answer the question "is Block1 White after it is moved on top of Block2?", we would first add

$$Move(0, Block1, top(Block2))$$

to our theory and then try to prove

$$Color(1, Block1, White).$$

Here is another query involving explanation. "Given that Block1 is Red at time 5, can this be explained by the performance of a paint action?". We would first add

$$Color(5, Block1, Red))$$

to our theory and then try to prove

$$\exists t.0 \leq t < 5 \wedge Paint(t, Block1, Red)).$$

Traditional Temporal Structures

Definition 4 (Point Structure) *A* point structure *is ant ordered couple* $\langle T, < \rangle$ *of a non-empty set* T *with a binary relation* $<$ *on* T*.*

"Time Flows"

$$\text{TRANS: } \forall x, y, z.(x < y < z \supset x < y)$$

"Absence of stops"

$$\text{IRREF: } \forall x.\neg(x < x)$$

"Directionality of Time"

$$\text{ASYMM: } \forall x, y.(x < y \supset \neg(y < x))$$

"No Branching"

$$\text{LIN: } \forall x, y.(x = y \vee x < y \vee y < x)$$

"No branching toward the left"

$$\text{L-LIN: } \forall x, y, z.((y < x \wedge z < x) \supset (y < z \vee y = z \vee z < y))$$

"Beginning of time"

$$\exists x. \neg \exists y. y < x$$

"End of time"

$$\exists x. \neg \exists y. x < y$$

"Eternity"

$$\text{SUCC: } \forall x \exists y. y < x$$

$$\forall x \exists y. x < y$$

"Infinite Divisibility"

$$\text{DENS: } \forall x \forall y. (x < y \supset \exists z. x < z < y)$$

"Stepwise succession"

$$\text{DISC: } \forall x \forall y. (x < y \supset \exists z. (x < z \wedge \neg \exists u. x < u < z))$$

$$\forall x \forall y. (x < y \supset \exists z. (z < y \wedge \neg \exists u. z < u < y))$$

Some Temporal Structures

DENSE LINEAR ORDERS: TRANS, IRREF, LIN, SUCC, DENS

$$\mathcal{Q} = \langle Q, < \rangle \text{ Rationals}$$

DISCRETE LINEAR ORDERS: TRANS, IRREF, LIN, SUCC, DISC

$$\mathcal{Z} = \langle Z, < \rangle \text{ Integers}$$

– The latter is not first-order definable!!.