Reasoning in and about a Changing World: Representing Time

Alan Bundy
School of Informatics

(slides courtesy of Bonnie Webber)

Reasoning about a changing world: Event Calculus

Both Situation Calculus and STRIPS allow reasoning about a limited type of changing world:

- The environment is static and discrete; the representation will not work for dynamic or continuous environments.
- There is only one agent; the representation will not work for multiple agents.

We need an enriched form of representation that can support more complex forms of reasoning: **Event calculus** $\sim$ continuous situation calculus occurring over time and space.

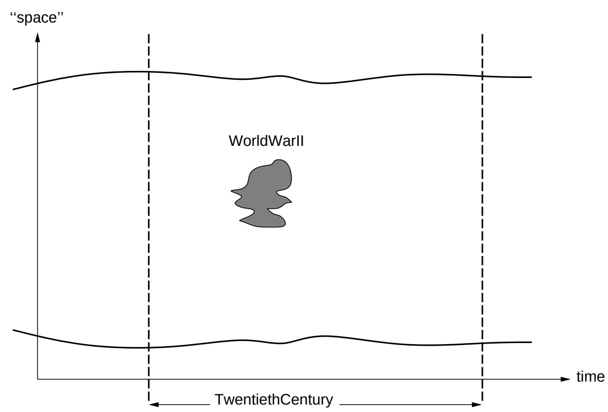


Figure 1: The "TwentiethCentury" as an interval

Informally, an **event** is a chunk of space-time.

An event can have parts called **subevents**.

A temporal **interval** can be considered a special kind of event that includes as subevents, all events that occur within its temporal boundaries.

Example: Flood in Edinburgh

Events are like any other entity, with some things that are true of them and some things that aren't.

To say that there was a flood in week 17, year 2000 in Edinburgh, we could write:

$$\exists\ f.\ f \in \text{Flood} \wedge \text{SubEvent}(f, week17\_AD2000) \wedge \text{PartOf}(\text{Location}(f), \text{Edinburgh})$$

As short-hand, we use the predicate E(c,i):

$$\forall c,i\ .\ E(c,i) \Leftrightarrow \exists\ e.\ e \in c \wedge \text{SubEvent}(e,i)$$

E.g. E(Flood,week17_AD2000)

Location

Location maps an entity to the smallest piece of space that contains it:

$\forall$x,l. Location(x)=l $\Leftrightarrow$
At(x,l) $\wedge$ $\forall$ l_2. At(x,l_2) $\Rightarrow$ SubEvent(l,l_2)

Is this and the fact that $PartOf(Edinburgh, Scotland)$ enough to conclude

$\exists$ f. f $\in$ Flood $\wedge$ SubEvent(f,week17_AD2000)
$\wedge$ PartOf(Location(f), Scotland)

Representing Actions as Events

We can represent actions as events of certain class.

For example, “Fred went to Tesco today” can be represented as a member of the *Journey* class.

$\exists$ j. $j \in$ *Journey* $\wedge$ Target(j,Tesco) $\wedge$
Traveller(j, Fred) $\wedge$ Subevent(j,Today)

Alternatively, we can use compound event types, e.g. Go(x, o, d).

$\forall$ e, x, o, d . $e \in$ Go(x, o, d) $\Leftrightarrow$ $e \in$ Journey $\wedge$
Traveller(e, x) $\wedge$ Origin(e, o) $\wedge$ Target(e, d)

e.g.

$\exists$ j, o. $j \in$ Go(Fred,o,Tesco) $\wedge$ SubEvent(j,Today)

Go(x, o, d) just a class of discrete (time-bounded) “go-ing” events.

Using E(c,i) notation, this can be further abbreviated to

E(Go(Fred,o,Tesco),Today)

Processes

“Unbounded events” begin some time, and end some time, but neither beginning nor ending is intrinsic to the event.

e.g. Fred was hill-walking on Monday can be represented as E(Hill-walk(Fred),Monday).

Alternatively, we use T(c,i) to indicate that an event of type C occurs over *exactly* the interval i.

e.g. T(Hill-walk(Fred), [900,1700]).
Fred was hillwalking for the whole interval between 9am and 5pm

Subevent Property

All processes have the *sub-interval property*: If a process takes place over an interval i, it also takes place over every sub-interval of i.

The subinterval property ensures:

T(Hill-walk(Fred), [900,1700]) $\Rightarrow$
T(Hill-walk(Fred), [1030,1200])

The sub-interval property doesn’t hold of non-process events. E.g.

E(Buy(Fred,Banana3), Yesterday) $\not\Rightarrow$
E(Buy(Fred,Banana3), YesterdayMorning)

Mid-Lecture Exercise

Use the T and E to represent:

> *While we made our escape, the dying Wumpus's cries echoed around the caves.*

- Let Legit be the interval during which "we made our escape".
- Let Dying be the interval during which "the Wumpus died".
- Let Escape(Us) be the class of "escape by us" actions.
- Let Cries(Wumpus) be the class of "the dieing Wumpus's cries echoed around the caves" actions.

Solution to Exercise

T(Cries(Wumpus),Dying) $\wedge$ E(Escape(Us),Dying)

Alternatively,

T(Cries(Wumpus),Dying) $\wedge$ T(Escape(Us),Legit) $\wedge$ Subevent(Legit,Dieing)

Combining Propositions

Naive Combination:

T(At(Agent,Sq)$\wedge$At(Gold,Sq),I)

illegal – 1st argument of T must be term, not sentence.

New Function:

T(And(At(Agent,Sq),At(Gold,Sq)),I)

And takes 2 event categories and returns combined event.

Definition:

$\forall p, q, e.$ T(And(p,q),e)$\Leftrightarrow$T(p,e)$\wedge$T(q,e)

Two Types of Time Interval

Part of reasoning about events involves

- when they happen(ed) with respect to one another
- what follows from that

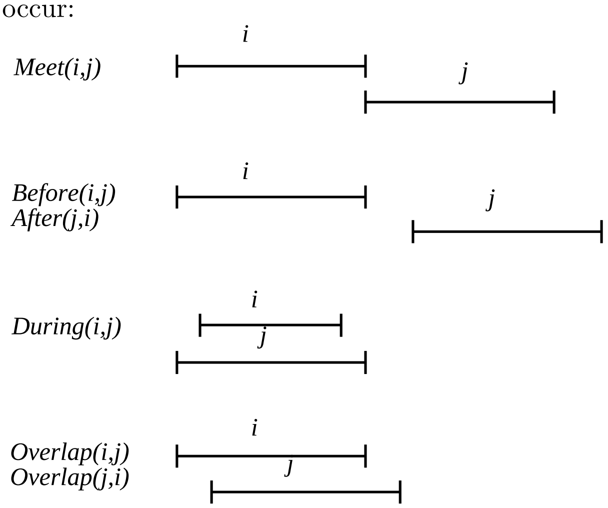
To do this, we allow two types of intervals:

- *extended* intervals (intervals) and
- *point* intervals (moments)

Time(Mom) = the clock time that moment Mom occurs.
Start(Int) = earliest moment in interval Int.
End(Int) = latest moment in interval Int.
duration(Int) =
Time(End(Int))-Time(Start(Int)).

Temporal Relations between Intervals

Can define several relations between events based on relations between the intervals in which they occur:



Defining Temporal Relations

Can define these temporal relations in terms of Time, Start and End.

- Meet(i,j) $\Leftrightarrow$ Time(End(i))=Time(Start(j))
- Before(i,j) $\Leftrightarrow$ Time(End(i))<Time(Start(j))
- During(i,j) $\Leftrightarrow$ Time(Start(j))$\leq$Time(Start(i)) $\wedge$ Time(End(i))$\leq$Time(End(j))
- Overlap(i,j) $\Leftrightarrow$ $\exists k$. During(k,i))$\wedge$During(k,j))

Describing actions

When you climb to the top of the mountain, you're at the top at the end of the climb:

$\forall$ a,m,i_0 $\exists$ i_1 . T(climb(a,m), i_0) $\Rightarrow$
T(at(a,top(m)), i_1) $\wedge$ meet(i_0,i_1)

If you always eat a picnic lunch at the summit:

$\forall$ a,m,i_0 $\exists$ i_1, i_2 . T(climb(a,m), i_0) $\Rightarrow$
T(at(a,top(m)), i_1) $\wedge$ meet(i_0,i_1)
$\wedge$ T(picnic(a), i_2) $\wedge$ during(i_2,i_1)

Q: What is the relationship between i_2 and i_0?

Answer: meet(i_0,i_2) $\vee$ after(i_2,i_0)

Q: One event causing another is a particularly significant relation between events. Only constraint is that the consequence (caused event) cannot start before its cause. What interval relations are possible?

Objects and fluents

Knowing that the prime minister of Britain was a conservative until 1996 and is now Labour, does **not** mean that somebody changed parties.

Knowing that the prime minister of Britain was a woman in the 1980s and is now a man, does **not** mean that somebody changed gender.

The cheer "The king is dead. Long live the king." is **not** a contradiction.

We capture this by treating certain entities as **fluents**, capable of changing their identity and/or properties over time.

Can use the same T notation as before:

T(Labour(PM(Britain)), [1997,2004])
T(Conservative(PM(Britain)), [1980,1997])

But only by declaring that PM is a **fluent** will this not imply that the same person was PM between 1980 and 1997.

Conclusion

- Need to represent continuous time and shared time.
- Event calculus represents space/time intervals and points.
- The E and T macros.
- Process events and the sub-interval property.
- Relations between intervals: Meet, Before, During, Overlap.
- Actions can be described by conditional rules, but using intervals for time rather than situations.
- Fluents: objects can vary in shape and identity over time.