PROCESSES AND EVENTS IN GEOGRAPHICAL SPACE

Session III: Formal Treatments

Antony Galton, University of Exeter, UK
COSIT 2009

What does 'formal' mean?

A formal system represents facts about some domain in such a way as to allow those facts to be manipulated using rules that refer only to the *form* of the representations, and not their *meanings*.

Example: *Arithmetic*. What do you need to know in order to compute 1791×1827?

The key to a successful formalisation is to design the formal rules so that they deliver results agreeing with what is expected in the domains the system is to be applied to.

- It is not necessarily the case that every domain can be successfully formalised.
- Computing necessarily involves formalisation (at least implicitly), so what we can do with computers is limited to what can be formalised.

Logical notation: A quick reminder

Formula	Truth condition	Meaning
$A \wedge B$	A and B are both true	"A and B"
$A \vee B$	At least one of A and B is true	"A or B"
$A \rightarrow B$	Either A is false or B is true	"If A then B"
$A \leftrightarrow B$	A and B are either both true or both false	"A if and only if B"
$\neg A$	A is false	"It is not the case that A"
$\exists x A$	There is at least one element x such that A is true	"For some x, A"
$\forall x A$	A is true for every element x	"For all x, A"

Temporal Logic

Classical logic was not designed for the expression of time and change.

There are two main ways of building temporality into logic:

- The **modal** approach: Extend the logical apparatus with operators expressing temporality.
- The **first-order** approach: Incorporate temporality into non-logical vocabulary.

In the modal approach, time is built into the formal framework in which we express propositions.

In the first-order approach, the formal framework is the same as before, and time is part of the subject-matter, i.e., what we express propositions about.

The Modal Approach: Tense Logic

Temporal operators resemble the *tenses* of natural language:

Formula	Interpretation
r	It is cold
Pr	It was cold, it has been cold
Fr	It will be cold
Hr	It has always been cold
Gr	It will always be cold

The Modal Approach: Tense Logic

Temporal operators resemble the *tenses* of natural language:

Formula	Interpretation
r	It is cold
Pr	It was cold, it has been cold
Fr	It will be cold
Hr	It has always been cold
Gr	It will always be cold

Combination of operators:

HFr	It has always been going to be cold
FPr	It will have been cold

The Modal Approach: Tense Logic

Temporal operators resemble the *tenses* of natural language:

Formula	Interpretation
r	It is cold
Pr	It was cold, it has been cold
Fr	It will be cold
Hr	It has always been cold
Gr	It will always be cold

Combination of operators:

HFr	It has always been going to be cold
FPr	It will have been cold

An axiom:

$p \rightarrow GPp$ What is true now will always have been true

An extension of Tense Logic: Hybrid Logic

How can we say more exactly *when* something is true? (I.e., not just past, present, or future.)

Let t stand for the proposition "It is 12th July 2009", and r for "It is raining". Then the formula

$$P(t \wedge r) \vee (t \wedge r) \vee F(t \wedge r)$$

states that it was, is, or will be raining on that day.

This can be abbreviated to

$$\Diamond(t \wedge r)$$

which in Hybrid Logic notation is

$$@_t r$$

Logical notation: Another quick reminder

In **first-order predicate calculus** (FOPC),

- An **atomic formula** has the form $P(a_1, a_2, \ldots, a_n)$.
- Here $a_1, \ldots, a_n$ are **terms** denoting individual entities.
- P is a **predicate** expressing some relation that holds amongst individual entities taken n at a time.
- The formula asserts that this relation holds for the n entities picked out by the a_i.
- The terms a_i are called **arguments** of the predicate.

Other formulae are obtained from atomic formulae by means of the logical connectives $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\neg$, $\exists$, and $\forall$.

The Method of Temporal Arguments

Times are assumed to be individual entities that can be referred to by terms, which in turn can be used as arguments to predicates.

- It rained on 12th July 2009:

$$Rain(day_{12-07-2009})$$

The Method of Temporal Arguments

Times are assumed to be individual entities that can be referred to by terms, which in turn can be used as arguments to predicates.

- It rained on 12th July 2009:

$$Rain(day_{12-07-2009})$$

- Napoleon invaded Russia in 1812:

$$Invade(napoleon, russia, year_{1812})$$

The Method of Temporal Arguments

Times are assumed to be individual entities that can be referred to by terms, which in turn can be used as arguments to predicates.

- It rained on 12th July 2009:

$$Rain(day_{12-07-2009})$$

- Napoleon invaded Russia in 1812:

$$Invade(napoleon, russia, year_{1812})$$

Note: This method does not readily distinguish between processes and events. Nor does it specify exactly how the process or event is related to the given time.

Reification

In a reified system, the event or process is expressed by a term, the fact of its occurrence by a predicate. There are two kinds of reification: *type-reification* and *token-reification*.

Reification

In a reified system, the event or process is expressed by a term, the fact of its occurrence by a predicate. There are two kinds of reification: *type-reification* and *token-reification*.

- Method of temporal arguments:

$$Invade(napoleon, russia, year_{1812})$$

Reification

In a reified system, the event or process is expressed by a term, the fact of its occurrence by a predicate. There are two kinds of reification: *type-reification* and *token-reification*.

- Method of temporal arguments:

$$Invade(napoleon, russia, year_{1812})$$

- Type-reification(the event term denotes an event *type*):

$$Occurs(invade(napoleon, russia), year_{1812})$$

Reification

In a reified system, the event or process is expressed by a term, the fact of its occurrence by a predicate. There are two kinds of reification: *type-reification* and *token-reification*.

- Method of temporal arguments:

$$Invade(napoleon, russia, year_{1812})$$

- Type-reification(the event term denotes an event *type*):

$$Occurs(invade(napoleon, russia), year_{1812})$$

- Token-reification (the event term denotes an event *token*):

$$\exists e(Invade(napoleon, russia, e) \wedge Occurs(e, year_{1812})).$$

Exactly what does *Occurs* mean?

In interpreting $Occurs(E, t)$ there is a potential ambiguity:

- Does it mean that t is the exact interval over which E occurred?
- Or does it just mean that E occurred sometime within the interval t?

It is usual to choose the first of these interpretations. This is secured by means of an axiom such as

$$\forall e \forall i \forall i' (Occurs(e, i) \wedge i' \sqsubset i \rightarrow \neg Occurs(e, i'))$$

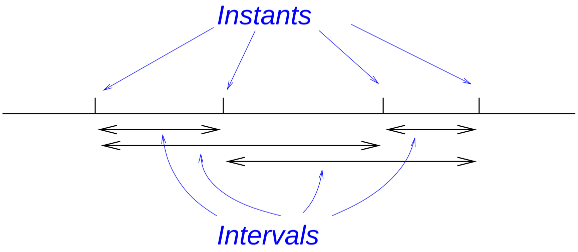
(here $i' \sqsubset i$ means that i' is a proper subinterval of i).

Given this, the second interpretation can be expressed as

$$\exists i' (i' \sqsubseteq i \wedge Occurs(e, i')).$$

Time Itself: Instants and Intervals

- Instants are durationless. They represent the meeting-points of contiguous intervals. E.g., “2.45 p.m. exactly”.
- Intervals have duration. An interval is bounded by instants at the beginning and end. Instants may be
  - “Standard”: 1812, June 1812, 24th June 1812.
  - “Arbitrary”: from 4 p.m. to 5.30 p.m. on 24th June 1812.
  - Defined by events: The reign of Louis XIV.



The Logic of Instants

Primitive relation: $t \prec t'$
Interpretation: Instant t precedes (i.e., is earlier than) instant t'.

A **predecessor** of instant t is any instant t' such that $t' \prec t$.
A **successor** of instant t is any instant t' such that $t \prec t'$.

The formal properties of the ordering of the instants are expressed by means of *axioms* written as first-order formulae.

In any application context, the axioms should be chosen to capture the properties of the temporal ordering that are required for reasoning within that context. In principle, different applications may require different models of time (there is not "one true model" for time — probably).

Axioms for Instants I

1. *Irreflexivity*. No instant precedes itself:

$$\forall t(\neg t \prec t)$$

Axioms for Instants I

1. *Irreflexivity*. No instant precedes itself:

$$\forall t(\neg t \prec t)$$

2. *Transitivity*. A successor of a successor of t is a successor of t:

$$\forall t \forall t' \forall t''(t \prec t' \wedge t' \prec t'' \rightarrow t \prec t'')$$

Axioms for Instants I

1. *Irreflexivity*. No instant precedes itself:

$$\forall t(\neg t \prec t)$$

2. *Transitivity*. A successor of a successor of t is a successor of t:

$$\forall t \forall t' \forall t''(t \prec t' \wedge t' \prec t'' \rightarrow t \prec t'')$$

3. *Linearity*. Of any two distinct instants, one precedes the other:

$$\forall t \forall t'(t \prec t' \vee t = t' \vee t' \prec t)$$

Axioms for Instants I

1. *Irreflexivity*. No instant precedes itself:

$$\forall t(\neg t \prec t)$$

2. *Transitivity*. A successor of a successor of t is a successor of t:

$$\forall t \forall t' \forall t''(t \prec t' \wedge t' \prec t'' \rightarrow t \prec t'')$$

3. *Linearity*. Of any two distinct instants, one precedes the other:

$$\forall t \forall t'(t \prec t' \vee t = t' \vee t' \prec t)$$

4. *Past unboundedness*. Every instant has a predecessor:

$$\forall t \exists t'(t' \prec t)$$

Axioms for Instants I

1. *Irreflexivity*. No instant precedes itself:

$$\forall t(\neg t \prec t)$$

2. *Transitivity*. A successor of a successor of t is a successor of t:

$$\forall t \forall t' \forall t''(t \prec t' \wedge t' \prec t'' \rightarrow t \prec t'')$$

3. *Linearity*. Of any two distinct instants, one precedes the other:

$$\forall t \forall t'(t \prec t' \vee t = t' \vee t' \prec t)$$

4. *Past unboundedness*. Every instant has a predecessor:

$$\forall t \exists t'(t' \prec t)$$

5. *Future unboundedness*. Every instant has a successor:

$$\forall t \exists t'(t \prec t')$$

Axioms for Instants II

6. *Density*. Between any two distinct instants there is a third instant which precedes one and is preceded by the other:

$$\forall t \forall t' (t \prec t' \rightarrow \exists t'' (t \prec t'' \wedge t'' \prec t'))$$

Axioms for Instants II

6. *Density*. Between any two distinct instants there is a third instant which precedes one and is preceded by the other:

$$\forall t \forall t'(t \prec t' \rightarrow \exists t''(t \prec t'' \wedge t'' \prec t'))$$

7. *Continuity*. If a set S of instants is non-empty, has non-empty complement, and contains the predecessors of all its members, then there is an instant all of whose predecessors, and none of whose successors, is in S:

$$\forall S(\exists t S(t) \wedge \exists t \neg S(t) \wedge \forall t \forall t'(S(t) \wedge t' \prec t \rightarrow S(t')) \rightarrow \exists t \forall t'(t' \prec t \rightarrow S(t) \wedge t \prec t' \rightarrow \neg S(t)))$$

N.B. Axiom 7 is *not* a first-order formula (because of the "$\forall S$", which makes it *second-order*).

Models for the axioms

A model for the axioms consists of a set of instants with a precedence relation defined on them, such that all the axioms come out true when understood as referring to this model.

If we use the *rational numbers* ($\mathbb{Q}$) to represent instants, with the usual "less than" relation ($<$) to represent precedence, then axioms 1–6, but not 7, are all satisfied. For this model, axioms 1–6 are *complete*, i.e., every true statement about temporal precedence in the model is a logical consequence of the axioms.

If we use the *real numbers* ($\mathbb{R}$), then again axioms 1–6 give the complete first-order theory. But now axiom 7 (continuity) is also satisfied. The orderings of $\mathbb{Q}$ and $\mathbb{R}$ can only be distinguished by means of second-order logic.

Discrete time

If we drop Axiom 6 (density) and replace it with

6′ If an instant has a predecessor (successor), then it has an immediate predecessor (successor):

$$\forall t \forall t' (t \prec t' \rightarrow \exists t'' (t'' \prec t' \wedge \forall t''' (t''' \prec t' \rightarrow t''' \preceq t'')) \wedge \exists t'' (t \prec t'' \wedge \forall t''' (t \prec t''' \rightarrow t'' \preceq t''')))$$

then we obtain the theory of *discrete* time — satisfied, for example, by the set of integers ($\mathbb{Z}$).

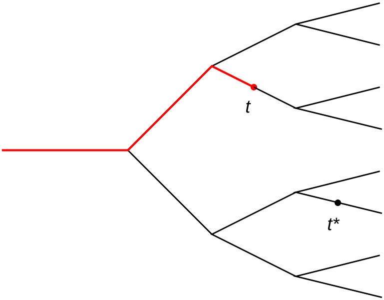
This might be an appropriate model for representing phenomena which only display significant structure on a time-scale of more than a day. Time can then be modelled as a discrete sequence of days.

Branching time

If we drop Axiom 3 (linearity), we can capture the asymmetry between the past and the future, i.e.., each instant has a unique past but many possible futures, as expressed by

3′. *Past-linearity*: Of any two distinct instants which both precede some third instant, one precedes the other:

$$\forall t \forall t' \forall t'' (t \prec t'' \wedge t' \prec t'' \rightarrow t \prec t' \vee t = t' \vee t' \prec t)$$



The logic of intervals

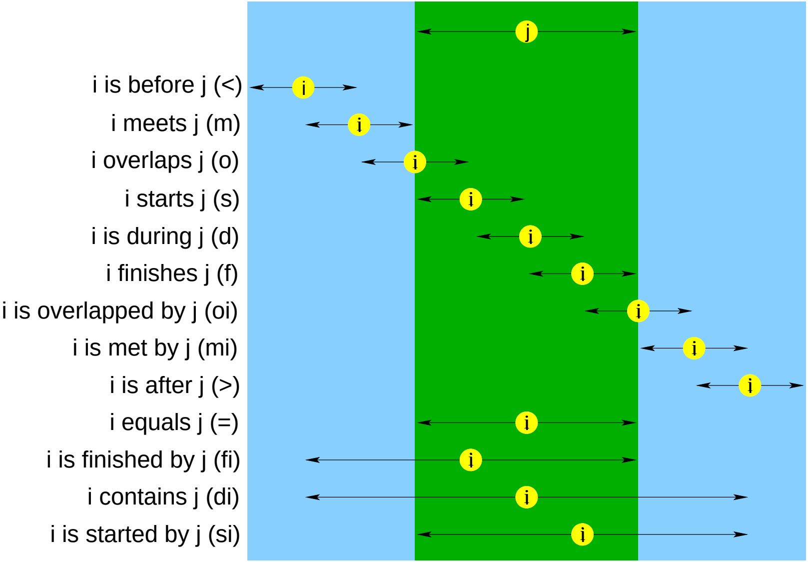
Allen (1984) argued that instants have no empirical reality and that all reasoning about temporal phenomena should be based on a model of time in which intervals are primitive elements, not constructed as aggregates of instants.

He showed that there is a set of 13 basic qualitative relations between intervals, forming a jointly exhaustive and pairwise disjoint (JEPD) set. (Cf. RCC8, the JEPD set of 8 qualitative relations between equidimensional spatial regions.)

These can all be defined in terms of a single primitive relation, *meets*, denoted m .

The formula $a \, \mathrm{m} \, b$ means that interval a ends exactly as interval b begins.

The 13 Qualitative Relations Between Intervals



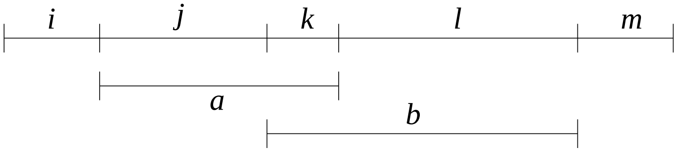
The Interval Calculus

Name	Symbol	Definition
is before	$<$	$a < b \equiv \exists j (a \mathrm{m}\, j \mathrm{m}\, b)$
meets	m	Primitive
overlaps	o	$a \mathrm{o}\, b \equiv \exists i \exists j \exists k \exists l \exists m (i \mathrm{m}\, j \mathrm{m}\, k \mathrm{m}\, l \mathrm{m}\, m \wedge i \mathrm{m}\, a \mathrm{m}\, l \wedge j \mathrm{m}\, b \mathrm{m}\, m)$
starts	s	$a \mathrm{s}\, b \equiv \exists i \exists j \exists k (i \mathrm{m}\, a \mathrm{m}\, j \mathrm{m}\, k \wedge i \mathrm{m}\, b \mathrm{m}\, k)$
finishes	f	$a \mathrm{f}\, b \equiv \exists i \exists j \exists k (i \mathrm{m}\, j \mathrm{m}\, a \mathrm{m}\, k \wedge i \mathrm{m}\, b \mathrm{m}\, k)$
is during	d	$a \mathrm{d}\, b \equiv \exists i \exists j \exists k \exists l (i \mathrm{m}\, j \mathrm{m}\, a \mathrm{m}\, k \mathrm{m}\, l \wedge i \mathrm{m}\, b \mathrm{m}\, l)$
equals	$=$	$a = b \equiv \exists i \exists j (i \mathrm{m}\, a \mathrm{m}\, j \wedge i \mathrm{m}\, b \mathrm{m}\, j)$
is after	$>$	$a > b \equiv b < a$
is met by	mi	$a \mathrm{mi}\, b \equiv b \mathrm{m}\, a$
is overlapped by	oi	$a \mathrm{oi}\, b \equiv b \mathrm{o}\, a$
is started by	si	$a \mathrm{si}\, b \equiv b \mathrm{s}\, a$
is finished by	fi	$a \mathrm{fi}\, b \equiv b \mathrm{f}\, a$
contains	di	$a \mathrm{di}\, b \equiv b \mathrm{d}\, a$

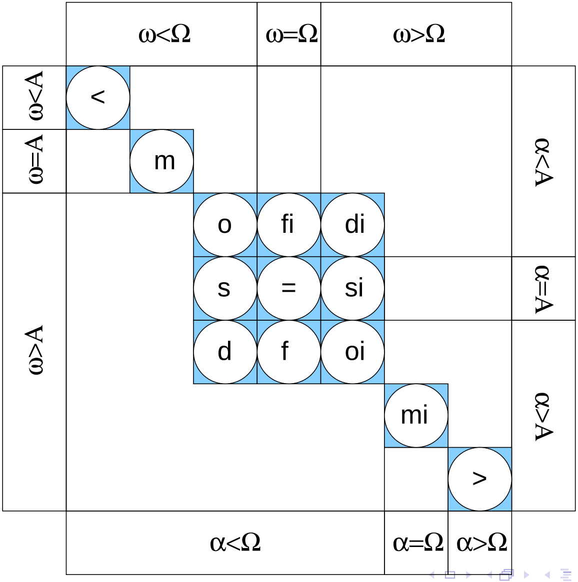
How the definitions work

The following diagram illustrates the definition

$$a \circ b \equiv \exists i \exists j \exists k \exists l \exists m (i \, \mathrm{m} \, j \, \mathrm{m} \, k \, \mathrm{m} \, l \, \mathrm{m} \, m \wedge i \, \mathrm{m} \, a \, \mathrm{m} \, l \wedge j \, \mathrm{m} \, b \, \mathrm{m} \, m)$$



Freksa's Construction: Relations between (α, ω) and (A, Ω)



Conceptual Neighbourhood

The following definition is due to Freksa (1992):

> *Two relations between pairs of events are (conceptual) neighbours, if they can be directly transformed into one another by continuously deforming (i.e., shortening, lengthening, moving) the events (in a topological sense).*

Freksa's conjecture: "If a cognitive system is uncertain as to which relation between two events holds, uncertainty can be expected particularly between neighbouring concepts."

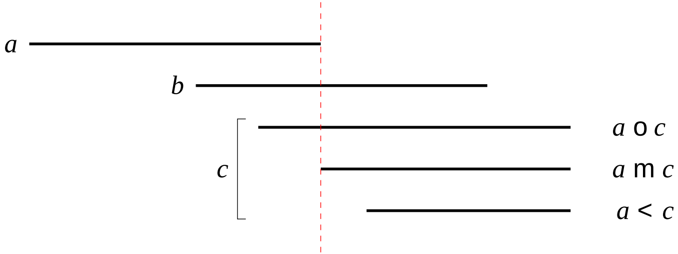
These ideas can be applied to spatial relations as well as temporal ones (cf., RCC, the double-cross calculus, etc.).

Compositional reasoning with intervals

A typical **composition rule** for the Interval Calculus is

- If *a* overlaps *b* and *b* overlaps *c* then *a* either is before, meets, or overlaps *c*.

$$\forall a \forall b \forall c (a \,\mathrm{o}\, b \wedge b \,\mathrm{o}\, c \rightarrow a < c \vee a \,\mathrm{m}\, c \vee a \,\mathrm{o}\, c),$$



Interval Algebra

The Interval Algebra $\mathcal{A}$ includes, in addition to the 13 base relations of the Interval Calculus, all possible *disjunctions* of these. The notation used is, e.g., $\{<, \mathrm{m}, \mathrm{o}, \mathrm{fi}, \mathrm{di}\}$, defined by

$$a\{<, \mathrm{m}, \mathrm{o}, \mathrm{fi}, \mathrm{di}\}b \equiv a < b \vee a \,\mathrm{m}\, b \vee a \,\mathrm{o}\, b \vee a \,\mathrm{fi}\, b \vee a \,\mathrm{di}\, b$$

There are $2^{13} = 8192$ relations in $\mathcal{A}$.

The **constraint satisfiability problem** for $\mathcal{A}$

Given a set S of *constraints* of the form iRj, where i and j are interval variables and R is a relation in $\mathcal{A}$, is it possible to associate actual intervals (specified as real-number pairs (x, y)) with the variables appearing in S so that all the constraints are satisfied?

Example of the Constraint Satisfiability Problem

Constraints:

$$a\{\,\mathrm{m}\,,\ \mathrm{o}\,\}b, \qquad b\{\,\mathrm{f}\,, =, \ \mathrm{fi}\,\}c, \qquad c\ \mathrm{mi}\ d, \qquad d < a$$

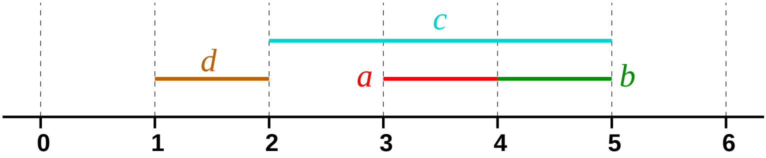
Example of the Constraint Satisfiability Problem

Constraints:

$$a\{\,\mathrm{m}\,,\,\mathrm{o}\,\}b, \qquad b\{\,\mathrm{f}\,, =, \,\mathrm{fi}\,\}c, \qquad c\,\mathrm{mi}\,d, \qquad d < a$$

Sample solution:

$$a = (3,4), \quad b = (4,5), \quad c = (2,5), \quad d = (1,2)$$



Computational considerations

- The constraint satisfiability problem for the Interval Algebra is NP-complete (Vilain and Kautz, 1986)
- Assuming P$\neq$NP, this means that temporal reasoning using the full Interval Algebra is intractable (probably of exponential complexity in the worst case).
- Nebel & Bürckert (1995) and Drakengren & Jonsson (1998) identified *maximal tractable subalgebras* of $\mathcal{A}$.
- Krokhin *et al.* (2003) provided a complete enumeration of all the tractable subalgebras of $\mathcal{A}$.

Analogous work has been done for *spatial* reasoning using the Region Connection Calculus (RCC) — but that is outside the scope of this tutorial.