Class discussion on Temporal Constraint Networks (Chapter 12)

Presentation by Chao Chen
Scribe: Tibor Moldovan

Overview

Temporal constraint networks are used for representing and processing temporal information. Two approaches for this are presented:

1. qualitative (point and interval) and
2. quantitative.

The basic algorithms and concepts for their processing are presented and analyzed.

The CSP properties that are highly desirable in this context are **minimality** and **decomposability** (i.e., global consistency).

- **Minimality**: a network is minimal iff:
  - its domains are minimal: Given two values for two variables, if they are consistent, then they appear in at least one solution, in other words, removing a value out of a minimal network removes a possible solution
  - constraints are minimal: Tightest possible binary constraints yield the minimal network
- **Decomposability**: a network is decomposable if every consistent assignment of values to a set of variables S can be extended to a solution.

Vocabulary: The vocabulary consists of:

1. Temporal objects
2. Constraints
3. Domains

Temporal Objects:

- *Intervals*: time period during which events occur or propositions hold (i.e., during class, am, pm)
- *Points*: beginning and ending of some events (i.e., R-, R+ for started reading, finished reading respectively)

Constraints:

- *Qualitative*: Relation between time point and interval
  - Extensional (enumerated), atomic relations
  - interval algebra: before, during, starts, etc.
  - point algebra: <, =, >
- *Quantitative/metric*: Duration of an event in a numerical fashion. Typically expressed in intension (i.e., declared as a mathematical function) such as constraints of bounded differences

Domain**:**

- Continuous intervals in R representing time.
- Integral values, rational or irrational values

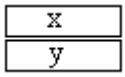
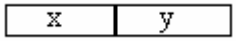
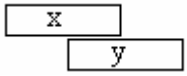
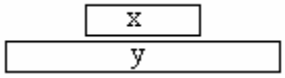
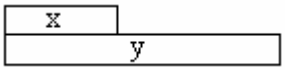
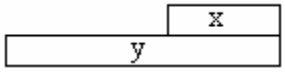
12.1 Qualitative Temporal Networks

Two approaches are presented for solving qualitative temporal networks:

1. Interval algebra, and
2. Point algebra.

12.1.1 The Interval Algebra

Interval Algebra formulates temporal knowledge in terms of qualitative statements regarding the relative locations of paired intervals; in other words, it describes relations between intervals. There are seven basic relations: `before`, `meets`, `overlaps`, `starts`, `during`, `finishes` and `equal`. Furthermore, each one of these is associated with an inverse relation. Since `equal` is its own inverse there are thirteen relations in total:

Relation	Symbol	Inverse	Example
X before Y	b	bi	
X equal Y	=	=	
X meets Y	m	mi	
X overlaps Y	o	oi	
X during Y	d	di	
X starts Y	s	si	
X finishes Y	f	fi	

12.1.2 Path Consistency in Interval Algebra

Two main binary operations on constraints are used for interval algebra: *intersection* and *composition*. Algorithm QPC-1 (qualitative Path-consistency) achieves path consistency by first applying the intersection operator, followed by composition, tightening every pair of constraints, until nothing changes or inconsistency is detected.

In some cases PC algorithms are exact; that is, they are guaranteed to generate the minimal network and sometimes decide consistency. However to solve the general IA network and generate a solution we need to use backtracking search. Even when the minimal network is available it is not guaranteed to be globally consistent to allow backtrack-free search.

12.1.3 The Point Algebra

Point algebra, a less computationally expensive model than interval algebra, expresses information by means of constraints on points. There are three possible basic relations that can hold between a pair of points P and Q: P<Q, P=Q, and P>Q. The elements of the PA are all 2^3 subsets of the basic relations $\{<, =, >\}$.

Definition 12.1.6 (point algebra networks -- PANs) A point algebra network involves a set of variables $\{x_1, \ldots, x_n\}$, where each variable represents a time point. The domain of each variable is the set of real numbers R, standing for the set of time points the variable may assume. The constraints are given as PA elements, having their algebraic meaning over the real numbers.

Path consistent point algebra networks imply that the network is minimal and decomposable; in other words globally consistent network. Reasoning tasks in point algebra are polynomial-time, $O(n^3)$.

- Maddux and Ladkin proved that a given PA network is consistent if and only if after executing path consistency the resulting network is nonempty. Thus, deciding the consistency of a PA network is $O(n^3)$. Van Beek proposed an $O(n^2)$ approximation algorithm.
- However, path consistency is not exact in the *full* PA (where $\neq$ relation is allowed). In order to compute the minimal network for a PA network, we need to enforce 4-consistency, which requires $O(n^4)$ time.

A special class of PANs are convex PA networks (CPANs). In these networks, the constraints are taken from the set $\{<, \leq, =, \geq, >\}$. In other words, the $\neq$ relation is excluded. Path consistency is exact for CPA and computing the minimal network of a CPA has time-complexity $O(n^3)$.

It is worth remembering:

- Consistency and solution generation of PAN can be done in $O(n^2)$.
- The minimal network of a consistent PAN can be obtained using 4-consistency in $O(n^4)$ steps
- For CPA, path consistency $\Rightarrow$ minimality

Note that, in some cases PA cannot fully express the constraints that are expressed by interval algebra. PA methods are most useful if an IA problem can successfully be translated into a PA form (which has $O(n^2)$ complexity), due to the better tractability of PA over IA.

This brings us to Section 12.2, Quantitative Temporal Networks, covered in the next lecture.

References:

Chao Chen, *Lecture Slides: Temporal Constraint Networks*, 2-26-2003;
Rina Dechter, *Constraint Processing*, 10-3-2002;
R.D. Maddux, P.B. Ladkin: *On binary constraint networks,* 1988
http://www.math.iastate.edu/maddux/papers