ECE570 Lecture 12: Allen's Temporal Logic

Jeffrey Mark Siskind

School of Electrical and Computer Engineering

Fall 2013

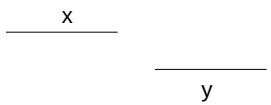
PURDUE UNIVERSITY

Motivation

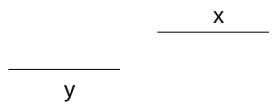
Alice came to the party with Bob
Bob left the party with Carol
David met Bob at the party
David was not at the party when Alice was there
David was not at the party when Carol was there

Alice left before Carol arrived

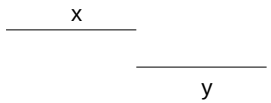
The Thirteen Relations Between Two Intervals



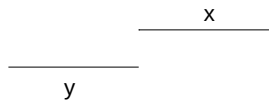
x < y



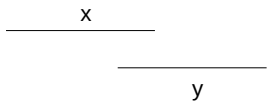
x > y



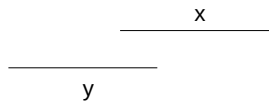
x m y



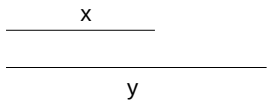
x mi y



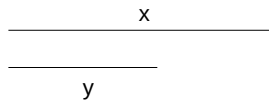
x o y



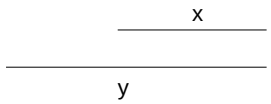
x oi y



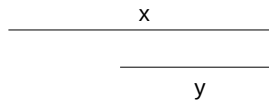
x s y



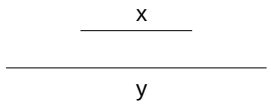
x si y



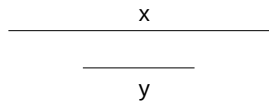
x f y



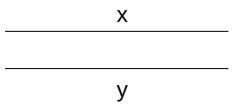
x fi y



x d y



x di y



x = y

Events as Intervals

$$A \triangleq \textit{Alice at party}$$

$$B \triangleq \textit{Bob at party}$$

$$C \triangleq \textit{Carol at party}$$

$$D \triangleq \textit{David at party}$$

Statements as Relations Between Intervals—I

Alice came to the party with Bob

$$A \ \mathsf{s} \ B \vee A = B \vee A \ \mathsf{si} \ B$$

Bob left the party with Carol

$$B \ \mathsf{f} \ C \vee B = C \vee B \ \mathsf{fi} \ C$$

David met Bob at the party

$$D = B \vee D \ \mathsf{s} \ B \vee D \ \mathsf{si} \ B \vee D \ \mathsf{f} \ B \vee D \ \mathsf{fi} \ B \vee D \ \mathsf{o} \ B$$
$$\vee D \ \mathsf{oi} \ B \vee D \ \mathsf{d} \ B \vee D \ \mathsf{di} \ B$$

David was not at the party when Alice was there

$$D < A \vee D \ \mathsf{m} \ A \vee D \ \mathsf{mi} \ A \vee D > A$$

David was not at the party when Carol was there

$$D < C \vee D \ \mathsf{m} \ C \vee D \ \mathsf{mi} \ C \vee D > C$$

Statements as Relations Between Intervals—II

Alice came to the party with Bob

$$A \ \{\mathsf{s}, =, \mathsf{si}\} \ B$$

Bob left the party with Carol

$$B \ \{\mathsf{f}, =, \mathsf{fi}\} \ C$$

David met Bob at the party

$$D \ \{=, \mathsf{s}, \mathsf{si}, \mathsf{f}, \mathsf{fi}, \mathsf{o}, \mathsf{oi}, \mathsf{d}, \mathsf{di}\} \ B$$

David was not at the party when Alice was there

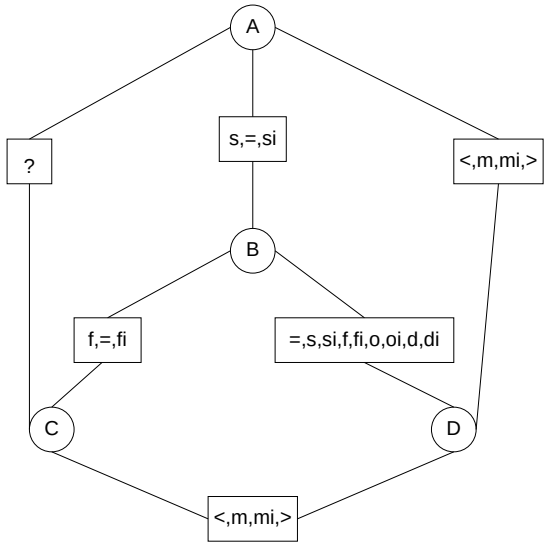
$$D \ \{<, \mathsf{m}, \mathsf{mi}, >\} \ A$$

David was not at the party when Carol was there

$$D \ \{<, \mathsf{m}, \mathsf{mi}, >\} \ C$$

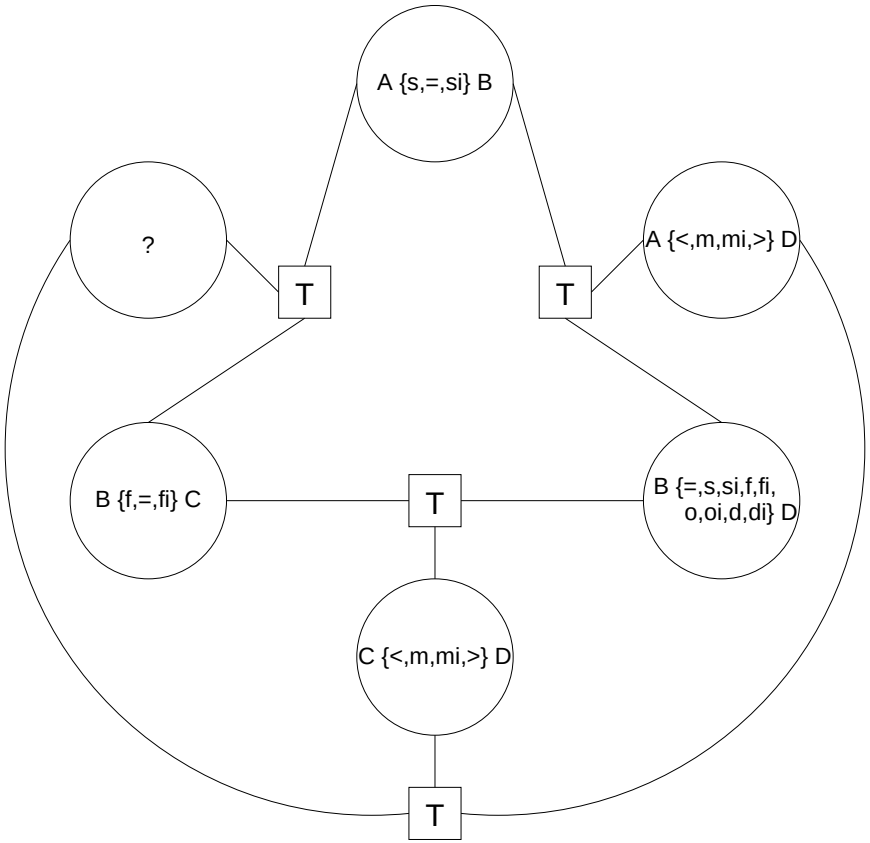
Temporal Relations as a CSP—First Try

Intervals as variables
Relations as constraints



Problems: Infinite Domains
Asking for possible values of a constraint, not a variable

Temporal Relations as a CSP—Second Try



Transitivity Table—I

	$<$	$>$	d	di	o	oi	m	mi	s	si	f	fi
$<$	$<$	$< > \text{ m mi o oi s si f fi d di } =$	$< \text{ o m d s}$	$<$	$<$	$< \text{ o m d s}$	$<$	$< \text{ o m d s}$	$<$	$<$	$< \text{ o m d s}$	$<$
$>$	$< > \text{ m mi o oi s si f fi d di } =$	$>$	$> \text{ oi mi d f}$	$>$	$> \text{ oi mi d f}$	$>$	$> \text{ oi mi d f}$	$>$	$> \text{ oi mi d f}$	$>$	$>$	$>$
d	$<$	$>$	d	$< > \text{ m mi o oi s si f fi d di } =$	$< \text{ o m d s}$	$> \text{ oi mi d f}$	$<$	$>$	d	$> \text{ oi mi d f}$	d	$< \text{ o m d s}$
di	$< \text{ o m di fi}$	$> \text{ oi di mi si}$	$\text{ o oi s si f fi d di } =$	di	o di fi	oi di si	o di fi	oi di si	o di fi	di	oi di si	di

Transitivity Table—II

	$<$	$>$	d	di	o	oi	m	mi	s	si	f	fi
o	$<$	$>$ oi di mi si	o d s	$<$ o m di fi	$<$ o m	o oi s si f fi d di $=$	$<$	oi di si	o	o di fi	o d s	$<$ o m
oi	$<$ o m di fi	$>$	oi d f	$>$ oi di mi si	o oi s si f fi d di $=$	$>$ oi mi	o di fi	$>$	oi d f	$>$ oi mi	oi	oi di si
m	$<$	$>$ oi di mi si	o d s	$<$	$<$	o d s	$<$	f fi $=$	m	m	o d s	$<$
mi	$<$ o m di fi	$>$	oi d f	$>$	oi d f	$>$	s si $=$	$>$	oi d f	$>$	mi	mi
s	$<$	$>$	d	$<$ o m di fi	$<$ o m	oi d f	$<$	mi	s	s si $=$	d	$<$ o m
si	$<$ o m di fi	$>$	oi d f	di	o di fi	oi	o di fi	mi	s si $=$	si	oi	di
f	$<$	$>$	d	$>$ oi di mi si	o d s	$>$ oi mi	m	$>$	d	$>$ oi mi	f	f fi $=$
fi	$<$	$>$ oi di mi si	o d s	di	o	oi di si	m	oi di si	o	di	f fi $=$	fi

Encoding in Scheme—I

```
(define *transitivity-table*
 '((< (< <)
      (> = < > d di o oi m mi s si f fi) ; no info
      (d < o m d s)
      (di <)
      (o <)
      (oi < o m d s)
      (m <)
      (mi < o m d s)
      (s <)
      (si <)
      (f < o m d s)
      (fi <))
   (> (< = < > d di o oi m mi s si f fi) ; no info
      (> >)
      (d > oi mi d f)
      (di >)
      (o > oi mi d f)
      (oi >)
      (m > oi mi d f)
      (mi >)
      (s > oi mi d f)
      (si >)
      (f >)
      (fi >))
   ...))
```

Encoding in Scheme—II

```
(define (transitive? x y z)
 (memq z (rest (assq y (rest (assq x *transitivity-table*))))))
```

Encoding in Scheme—III

```
(define (party)
 (let ((ab (create-domain-variable '(s = si)))
       (bc (create-domain-variable '(f = fi)))
       (bd (create-domain-variable
            '(= s si f fi o oi d di)))
       (ad (create-domain-variable '(< m mi >)))
       (cd (create-domain-variable '(< m mi >)))
       (ac (create-domain-variable
            '(= < > m mi o oi s si f fi d di))))
  (assert-constraint! transitive? (list ab bc ac))
  (assert-constraint! transitive? (list ab bd ad))
  (assert-constraint! transitive? (list bc cd bd))
  (assert-constraint! transitive? (list ac cd ad))
  (write (domain-variable-domain ac))
  (newline)))
```

Incompleteness of Arc Consistency for Allen's Temporal Logic