Modeling in Knowledge Representation: The Parthood Relation

Stefano Borgo LOA, ISTC, CNR
Carola Eschenbach Hamburg University
Laure Vieu IRIT, CNRS & LOA, ISTC, CNR

Doctorate Course

2006-2007

5-Reasoning

Doctorate Course

Modeling in Knowledge Representation: The Parthood Relation

2006-2007

Structure

Problem statement, RSAT, decision procedure

Constraint Satisfaction Problems (CSP)

- RSAT as CSP

Systems of Binary Relations

- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

Constraint-based Reasoning

- Constraint propagation (based on composition tables)
- Backtracking-propagation

How to Evaluate the Result?

- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency

- When is constraint propagation sufficient to decide RSAT?

Problem Statement

Systems of Binary Relations

- Provided by ontologies of time and space
- Specified by first-order axiomatic theories

Inference Tasks

- Is a set of formulae consistent / satisfiable?
- Does a formula follow from a set of formulae?
- …

The general problems are not decidable

- By restricting problems to specific types of formulae one can derive decidability.

Decision procedure

Definition

- Given a formal language $\mathcal{L}$ and let $\mathcal{F}$ be a set of formulae of $\mathcal{L}$. An algorithm is a *decision procedure for $\mathcal{F}$* if given an arbitrary formula $A \in \mathcal{L}$, it terminates and returns the answer 'yes' if $A \in \mathcal{F}$ and the answer 'no' if $A \notin \mathcal{F}$.

Generalization

- Represent (finite) sets of formulae by single formulae to gain decision procedures for sets of sets of formulae.

Example

- Propositional logic: There are decision procedures for the sets of valid / satisfiable / contradictory formulae
- Predicate logic: There are no such decision procedures.

RSAT

Given

- A specification of a finite system of binary relations $\mathcal{R}$
- A finite set of variables $\{X_1, \ldots, X_n\}$
- A (finite) set $\mathcal{S}$ of formulae of the form $R(X_i, X_j)$, where $R \in \mathcal{R}$.

Problem

- Is $\mathcal{S}$ satisfiable?

Representation

- n*n matrix M of relations with M(i, i) = identity, $M(i, j) = (M(j, i))^{-1}$
- complete directed graph with n nodes and edge labels from $2^{\mathcal{R}}$

Decidability / Complexity

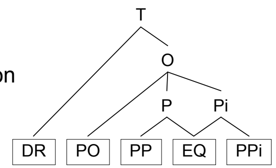
- depends on $\mathcal{R}$
- 13 Allen-relations: decidable and NP-complete
- RCC: decidable and NP-complete

Group discussion

Example RSAT

Relations

P: Part O: Overlap DR: disjoint
EQ: Equivalence/Identity T: universal relation



Variables

- $\{X_1, \ldots, X_5\}$

Two sets of formulae

- $\{P(X_1, X_2), O(X_1, X_3), O(X_3, X_4), O(X_4, X_1), DR(X_4, X_5), P(X_2, X_5)\}$
- $\{P(X_1, X_2), O(X_1, X_3), O(X_3, X_4), O(X_4, X_1), DR(X_5, X_4), P(X_5, X_2)\}$

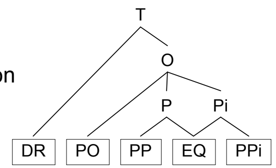
Is one of these sets satisfiable?

- How to prove it?

Example RSAT

Relations

P: Part O: Overlap DR: disjoint
EQ: Equivalence/Identity T: universal relation



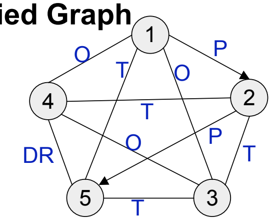
Variables

- $\{X_1, \ldots, X_5\}$

The set of formulae

- $\{P(X_1, X_2), O(X_1, X_3), O(X_3, X_4), O(X_4, X_1), DR(X_4, X_5), P(X_2, X_5)\}$

Simplified Graph



Matrix Representation

	1	2	3	4	5
1	EQ	P	O	O	T
2	Pi	EQ	T	T	P
3	O	T	EQ	O	T
4	O	T	O	EQ	DR
5	T	Pi	T	DR	EQ

Structure

✓**Problem statement, RSAT, decision procedure**

Constraint Satisfaction Problems (CSP)
- RSAT as CSP

Systems of Binary Relations
- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

Constraint-based Reasoning
- Constraint propagation (based on composition tables)
- Backtracking-propagation

How to Evaluate the Result?
- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency
- When is constraint propagation sufficient to decide RSAT?

Constraint Satisfaction Problems (CSP)

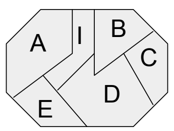
Given
- A (finite) set of variables, $\{X_1, \ldots, X_n\}$.
- Each variable X_i has a domain D_i (possible values).
- A (finite) set of constraints $\{C_1, \ldots, C_m\}$.
- Each constraint C_i involves some variables and specifies allowable combinations of values for these variables.

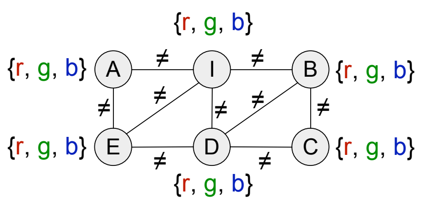
Problem
- Is there an assignment of all the variables to values that satisfies all given constraints?
- Determine all assignment of the variables to values that satisfies all given constraints.

Simple Example for CSP

Map-coloring
- Variables: States (in a spatial configuration)
- Values: Set of colors
- General constraint: Adjacent states need different colors

Representation: Graph
- Nodes: Variables; Node-Labels: Sets of Values
- Edges: Constraints





Constraint Propagation

Basic Idea
- Given: a set of constraints
- Goal: find out, whether there is a solution; find a/all solution/s
- Method: filter algorithms
  - change the problem description
  - without changing the set of solutions
  - by making implicit restrictions explicit
  - where implicit restrictions derive from the interaction of explicitly given constraints

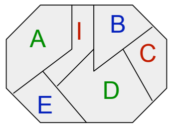
Example for Solution of Simple CSP

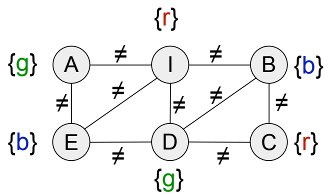
Solution

- Assignment of states to colors such that adjacent states have different colors

Representation of a solution

- Graph with unique label at each node such that connected nodes have different labels.

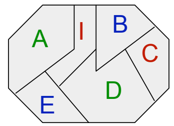


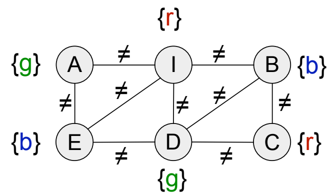


Simple Example for CSP

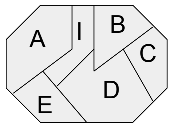
This is simple (partly) because

- the domains are finite
- constraints are (at most) binary
- only same/different in domain relevant
- unique labelling guarantees solution
- (but still NP-complete)





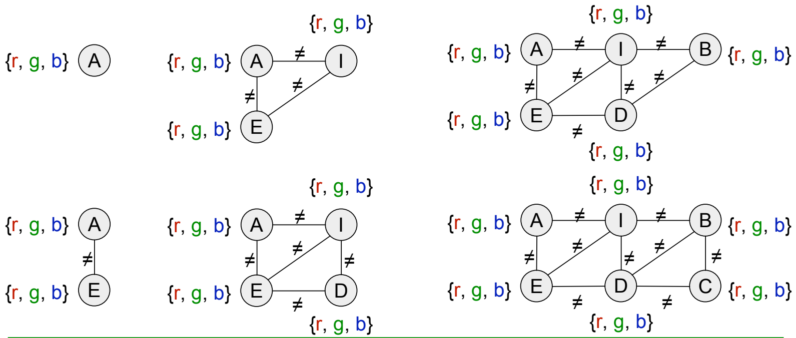
Example: Sub-Problems of CSP



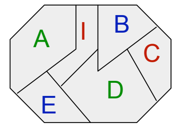
Variables: {A, B, C, D, E, I}

Series of Sub-Problems as Simplified Graphs

- develop solutions from solutions of sub-problems



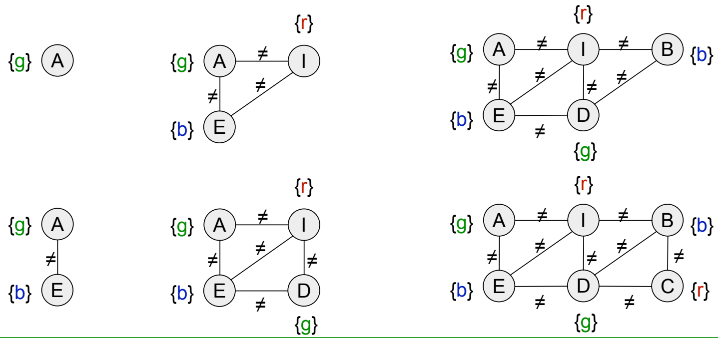
Example: Sub-Problems of CSP



Variables: {A, B, C, D, E, I}

Series of Sub-Problems as Simplified Graphs

- develop solutions from solutions of sub-problems



Sub-problem of a CSP

For sub-sets of variables

- Let C be a CSP and V be a sub-set of the variables of C.
- C_V is the *sub-problem of C restricted to V* iff
  - V are the variables of C_V,
  - the domains for the variables are the same in C and C_V and
  - the constraints of C_V are exactly those constraints of C that involve only variables from V.

The sub-problems of a CSP depend on its representation (not on its solutions).

Types of Consistency

A CSP C is

- *consistent* / satisfiable
  - iff C has a solution.
- *k-consistent*
  - iff for every set V of k variables and every X ∈ V every solution for $C_{V\setminus\{X\}}$ can be extended to a solution for C_V.
- *strongly k-consistent*
  - iff it is i-consistent for every i ≤ k.
- *path-consistent*
  - iff the (explicit) constraint between any two variables is at least as restrictive as every path in the constraint graph between the same two variables.

Observations on Consistency

In complete graphs,

- path consistency is equivalent to
- path consistency for all length-2 paths,
- which is the same as 3-consistency.

A CSP C with n variables is

- *globally consistent*, iff it is strongly n-consistent.

For a globally consistent CSP

- a solution can be constructed incrementally
- without backtracking.

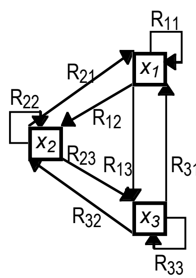
RSAT as CSP

- Domain can hold infinitely many possible values
- Constraint: Binary relations between variables

Representation: Complete directed graph

- Nodes: Variables
- Edges: Relation between two variables
- Edge-Labels: Sets of base-relation symbols

$$R_{11}(x_1, x_1) \wedge R_{12}(x_1, x_2) \wedge R_{21}(x_2, x_1) \wedge$$
$$R_{22}(x_2, x_2) \wedge R_{13}(x_1, x_3) \wedge R_{31}(x_3, x_1) \wedge$$
$$R_{23}(x_2, x_3) \wedge R_{32}(x_3, x_2) \wedge R_{33}(x_3, x_3) \wedge \ldots$$



Structure

✓**Problem statement, RSAT, decision procedure**
✓**Constraint Satisfaction Problems (CSP)**
- RSAT as CSP

Systems of Binary Relations
- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

Constraint-based Reasoning
- Constraint propagation (based on composition tables)
- Backtracking-propagation

How to Evaluate the Result?
- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency
- When is constraint propagation sufficient to decide RSAT?

System of Binary Relations

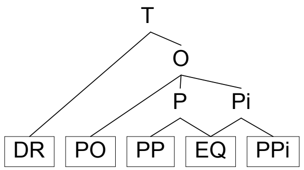
Base relations
- finitely many
- jointly exhaustive (JE)
- pairwise exclusive / disjoint (PD)

➔The selection of the set of base relations is formally restricted but not determined.

General relations
- are disjunctions of base relations
- form an atomic Boolean algebra

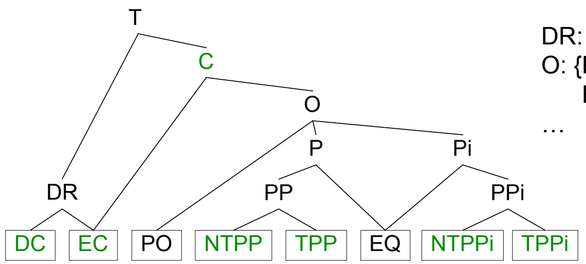
Standard representation of relations
- sets of base relations

Base Relations



← Mereological base relations

P: {EQ, PP}
Pi: {EQ, PPi}
O: {PO, PP, EQ, PPi}



DR: {DC, EC}
O: {PO, NTPP, TPP, EQ, NTPPi, TPPI}
…

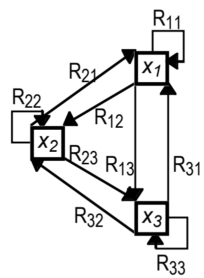
Mereotopological ←base relations

CSP for Binary Relation Systems

Solution
- Assignment of variables to values such that the base relation between two assigned values is included in the label at the edge connecting the two variables

Representation of a solution
- Irreducible graph with unique base-relation label at each edge

Problems
- How to derive such a graph?
- When does the graph guarantee a solution?



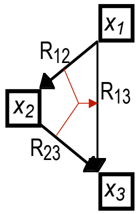
CSP for Binary Relation Systems

Solution

- Assignment of variables (X_i) to values (a_i) such that the base relation between two assigned values (r_{ij}) is included in the label at the edge connecting the two variables (R_{ij}).

Observation regarding solutions

- If a_i, a_j, a_k are part of a solution then $r_{ij} \in R_{ij}$ and $r_{ij} \subseteq r_{ik} \circ r_{kj}$



Derived ternary constrains

- on edge labels
- from composition of binary relations

Revise labelling

- $R'_{ij} = R_{ij} \cap R_{ik} \circ R_{kj}$

Relation Algebras

- the mathematical (algebraic) approach

Background (Motivation of Tarski)

- Boolean Algebras form adequate models for propositional logic and monadic predicate logic
- Which type of algebra does the same for logics with binary relations?
- Predecessors: DeMorgan, Peirce, Schröder

Goal

- Finite system of operators on binary relations
- Axiomatic specification of their interactions

(Concrete) Relation Algebras

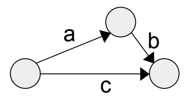
Definition

- The **complete relation algebra** for a set U is the structure $\mathcal{R}el(U) = \langle 2^V, \cup, \cap, -, \emptyset, V, \circ, \breve{}, Id \rangle$, where
  - $V = U \times U$,
  - $-$ is complement formation,
  - $\circ$ is relation composition and
  - $\breve{}$ is converse formation,
  - Id is the identity relation for U.
- Each sub-set of 2^V, that is closed under $\cup$, $\cap$, $-$, $\circ$, $\breve{}$ and that includes the constants $\emptyset$, V, and Id, is a **relation algebra**.

Example: Concrete Relation Algebra IdDiff_2

Universe with two entities: $U_2 = \{T, F\}$

- $\mathit{IdDiff}_2 = \{V_2, \emptyset, Id_2, Diff_2\}$
  - $V_2 = U_2 \times U_2 = \{(T, T), (T, F), (F, T), (F, F)\}$
  - $\emptyset = \{\} = -V_2$
  - $Id_2 = \{(T, T), (F, F)\}$
  - $Diff_2 = -Id = \{(T, F), (F, T)\}$
- IdDiff_2 is closed under $-$, $\cup$, $\cap$
- IdDiff_2 is closed under $\breve{}$
  - V_2, $\emptyset$, Id_2 and $Diff_2$ are symmetric
  - $\breve{V_2} = V_2$; $\breve{\emptyset} = \emptyset$; $\breve{Id_2} = Id_2$; $\breve{Diff_2} = Diff_2$
- IdDiff_2 is closed under $\circ$
  - $X \in \mathit{IdDiff}_2 : \emptyset \circ X = X \circ \emptyset = \emptyset$
  - $X \in \mathit{IdDiff}_2 : Id_2 \circ X = X \circ Id_2 = X$
  - $X \in \{V_2, Id_2, Diff_2\}: V_2 \circ X = X \circ V_2 = V_2$
  - $Diff_2 \circ Diff_2 = Id_2$



Tabular Presentation $\mathcal{IdDiff}_2$

Universe with two entities: U_2 = {T, F}

- $\mathcal{IdDiff}_2$ = {V_2, ∅, Id_2, $Diff_2$}

		–	˘
{(T, T), (T, F), (F, T), (F, F)}	**V_2**	∅	V_2
{}	∅	V_2	∅
{(T, T), (F, F)}	**Id_2**	$Diff_2$	Id_2
{(T, F), (F, T)}	**$Diff_2$**	Id_2	$Diff_2$

∘	V_2	∅	Id_2	$Diff_2$
V_2	V_2	∅	V_2	V_2
∅	∅	∅	∅	∅
Id_2	V_2	∅	Id_2	$Diff_2$
$Diff_2$	V_2	∅	$Diff_2$	Id_2

∪	V_2	∅	Id_2	$Diff_2$
V_2	V_2	V_2	V_2	V_2
∅	V_2	∅	Id_2	$Diff_2$
Id_2	V_2	Id_2	Id_2	V_2
$Diff_2$	V_2	$Diff_2$	V_2	$Diff_2$

∩	V_2	∅	Id_2	$Diff_2$
V_2	V_2	∅	Id_2	$Diff_2$
∅	∅	∅	∅	∅
Id_2	Id_2	∅	Id_2	∅
$Diff_2$	$Diff_2$	∅	∅	$Diff_2$

Reading (Strong) Composition Tables

∘	…	…	R_2	…
…	…	…	V_2	…
R_1	…	…	R_3	…
…	…	…	…	…
…	…	…	…	…

Relation Algebraic Composition Tables

$R_1 \circ R_2 = R_3$

Formulated in Predicate Logic

$\forall x\ y\ z\ [R_1(x, y) \wedge R_2(y, z) \Leftrightarrow R_3(x, z)]$

Tabular Presentation $\mathcal{IdDiff}_3$

Universe with three entities: U_3 = {r, g, b}

- $\mathcal{IdDiff}_3$ = {V_3, ∅, Id_3, $Diff_3$}

		–	˘
{(r, r), (g, g), (b, b), (r, g), (g, r), (r, b), (b, r), (g, b), (b, g)}	**V_3**	∅	V_3
{}	∅	V_3	∅
{(r, r), (g, g), (b, b)}	**Id_3**	$Diff_3$	Id_3
{(r, g), (g, r), (r, b), (b, r), (g, b), (b, g)}	**$Diff_3$**	Id_3	$Diff_3$

∘	V_3	∅	Id_3	$Diff_3$
V_3	V_3	∅	V_3	V_3
∅	∅	∅	∅	∅
Id_3	V_3	∅	Id_3	$Diff_3$
$Diff_3$	V_3	∅	$Diff_3$	V_3

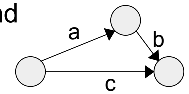
∪	V_3	∅	Id_3	$Diff_3$
V_3	V_3	V_3	V_3	V_3
∅	V_3	∅	Id_3	$Diff_3$
Id_3	V_3	Id_3	Id_3	V_3
$Diff_3$	V_3	$Diff_3$	V_3	$Diff_3$

∩	V_3	∅	Id_3	$Diff_3$
V_3	V_3	∅	Id_3	$Diff_3$
∅	∅	∅	∅	∅
Id_3	Id_3	∅	Id_3	∅
$Diff_3$	$Diff_3$	∅	∅	$Diff_3$

Relation Algebra

Definition

- An **abstract relation algebra** is a structure $\langle A, +, \bullet, -, 0, 1, \circ, \breve{}, 1' \rangle$ of the type $\langle 2, 2, 1, 0, 0, 2, 1, 0 \rangle$, where
  - $\langle A, +, \bullet, -, 0, 1 \rangle$ is a Boolean algebra and
  - $\langle A, \circ, 1' \rangle$ is a monoid
  - $\breve{\breve{a}} = a$ and $(a \circ b)\breve{} = \breve{b} \circ \breve{a}$, for a, b ∈ A, and
  - $(a \circ b) \bullet c = 0 \Leftrightarrow (\breve{a} \circ c) \bullet b = 0$
  - $(a \circ b) \bullet c = 0 \Leftrightarrow (c \circ \breve{b}) \bullet a = 0$



➢ Each concrete relation algebra is an abstract relation algebra.

➢ There are abstract relation algebras that are not isomorphic to any concrete relation algebra.

Atomic Relation Algebras

Definitions

Let $\mathcal{R} = \langle A, +, \bullet, -, 0, 1, \circ, \breve{}, 1' \rangle$ be a relation algebra.

$\mathcal{R}$ is *atomic*, iff the Boolean Algebra $\langle A, +, \bullet, -, 0, 1 \rangle$ is atomic.

- The set of atomic relations in an atomic relation algebra is jointly exclusive and pairwise disjoint (JEPD).
- In atomic relation algebras the behaviour of $\circ$ and $\breve{}$ on the atoms uniquely determine their complete behaviour.
- If the set of atoms is finite, then the operators $\circ$ and $\breve{}$ can be represented by tables.

Atomic relation algebra for a set of base relations

- Is there an atomic relation algebra where the base relations are the atoms?

Generated Algebras

Definitions

Let $\mathcal{R} = \langle A, +, \bullet, -, 0, 1, \circ, \breve{}, 1' \rangle$ be a relation algebra.

The relation algebra $\mathcal{R}' = \langle A', +', \bullet', -', 0, 1, \circ', \breve{}', 1' \rangle$ is a *sub-algebra of $\mathcal{R}$*, iff $A' \subseteq A$, $+' \subseteq +$, $\bullet' \subseteq \bullet$, $-' \subseteq -$, $\circ' \subseteq \circ$, $\breve{}' \subseteq \breve{}$.

Let $B \subseteq A$. The *sub-algebra of $\mathcal{R}$ generated by B* is the smallest sub-algebra $\mathcal{R}' = \langle A', +', \bullet', -', 0, 1, \circ', \breve{}', 1' \rangle$ of $\mathcal{R}$ with $B \subseteq A'$.

Let B be a set of binary relations over domain U. The *algebra generated by B* is the sub-algebra of $\mathcal{R}el(U)$ generated by B.

Atomic relation algebra for a set of base relations

- Even if B is a finite JEPD set of relations, the relation algebra generated by B need not be atomic.
- Even if it is atomic, the elements of B need not be its atoms.

Example: Concrete Relation Algebra $\mathcal{IdDiff}_3$

Universe with three entities: U_3 = {r, g, b}

$\mathcal{IdDiff}_3 = \{V_3, \emptyset, Id_3, Diff_3\}$

- is an atomic relation algebra.
- The atoms of $\mathcal{IdDiff}_3$ are Id_3, $Diff_3$.

$\mathcal{IdDiff}_3$ is generated by $\{Id_3\}$, $\{Diff_3\}$.

- $\breve{}$ and $\circ$ are completely determined by their behaviour on the atoms.

	$\breve{}$
Id_3	Id_3
$Diff_3$	$Diff_3$

$\circ$	Id_3	$Diff_3$
Id_3	Id_3	$Diff_3$
$Diff_3$	$Diff_3$	V_3

Example: Concrete Relation Algebra $\mathcal{Ord}_{\mathbb{Q}}$

Rational Numbers: $\mathbb{Q}$

- $V_{\mathbb{Q}} = \mathbb{Q} \times \mathbb{Q}$

$\mathcal{Ord}_{\mathbb{Q}} = \{V_{\mathbb{Q}}, \emptyset, =_{\mathbb{Q}}, \neq_{\mathbb{Q}}, <_{\mathbb{Q}}, >_{\mathbb{Q}}, \leq_{\mathbb{Q}}, \geq_{\mathbb{Q}}\}$

- is an atomic relation algebra.
- Atoms: $=_{\mathbb{Q}}$, $<_{\mathbb{Q}}$, $>_{\mathbb{Q}}$.

$\mathcal{Ord}_{\mathbb{Q}}$ is generated by $\{<_{\mathbb{Q}}\}$, $\{>_{\mathbb{Q}}\}$, $\{\leq_{\mathbb{Q}}\}$, $\{\geq_{\mathbb{Q}}\}$.

- (The simplicity of this relation algebra depends on the denseness and unboundedness of the order.)

	$\breve{}$
$=_{\mathbb{Q}}$	$=_{\mathbb{Q}}$
$<_{\mathbb{Q}}$	$>_{\mathbb{Q}}$
$>_{\mathbb{Q}}$	$<_{\mathbb{Q}}$

$\circ$	$=_{\mathbb{Q}}$	$<_{\mathbb{Q}}$	$>_{\mathbb{Q}}$
$=_{\mathbb{Q}}$	$=_{\mathbb{Q}}$	$<_{\mathbb{Q}}$	$>_{\mathbb{Q}}$
$<_{\mathbb{Q}}$	$<_{\mathbb{Q}}$	$<_{\mathbb{Q}}$	$V_{\mathbb{Q}}$
$>_{\mathbb{Q}}$	$>_{\mathbb{Q}}$	$V_{\mathbb{Q}}$	$>_{\mathbb{Q}}$

Structure

- ✓**Problem statement, RSAT, decision procedure**
- ✓**Constraint Satisfaction Problems (CSP)**
  - RSAT as CSP
- ✓**Systems of Binary Relations**
  - Base relations
  - (Atomic) Concrete/abstract relation algebras
  - Composition tables

Constraint-based Reasoning
- Constraint propagation (based on composition tables)
- Backtracking-propagation

How to Evaluate the Result?
- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency
- When is constraint propagation sufficient to decide RSAT?

How to proceed

First
- Consider the simple case
  - the set of base relations generates an atomic relation algebra
  - where the base relations are the atoms.
  - ➔Composition can be represented by a finite table: The composition table (CT).
- Study techniques to determine satisfiability.

Then
- Consider in which additional cases the techniques can yield sound and complete decision procedures.

Composition Table (CT)

CT: Specifies composition rules for base relations

$R_k := CT(r_i, r_j)$ $(= r_i \circ r_j)$

$\forall x\, z\, [\exists y\, [r_i(x, y) \wedge r_j(y, z)] \leftrightarrow R_k(x, z)]$

Composition for disjunctions (CT*)

from composition of base relations (CT)

$CT^*(R_1, R_2) = \cup\, \{CT(r_1, r_2) \mid r_1 \in R_1, r_2 \in R_2\}$

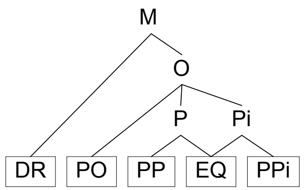
$(A \leftrightarrow C) \wedge (B \leftrightarrow D) \rightarrow ((A \vee B) \leftrightarrow (C \vee D))$

$\circ$	V_3	$\emptyset$	Id_3	$Diff_3$
V_3	V_3	$\emptyset$	V_3	V_3
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
Id_3	V_3	$\emptyset$	Id_3	$Diff_3$
$Diff_3$	V_3	$\emptyset$	$Diff_3$	V_3

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

Example: Mereological Relations (RCC-5)

∨	DR	PO	PPi	EQ	PP
DR	M	~Pi	DR	DR	~Pi
PO	~P	M	~P	PO	PO PP
PP	DR	~Pi	M	PP	PP
EQ	DR	PO	PPi	EQ	PP
PPi	~P	PO PPi	PPi	PPi	O



P: Part
Pi: has_Part
PP: proper part
O: Overlap
DR: disjoint
PO: proper/partial overlap
EQ: Equivalence/Identity

Taxonomy based presentation

Disjunctive form

$\forall x\ y\ z\ [PO(x, y) \wedge PP(y, z) \rightarrow (PO(x, z) \vee PP(x, z))]$

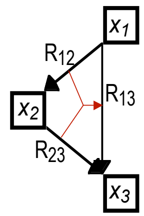
CSP for Binary Relation Systems

Solution

- Assignment of variables (X_i) to values (a_i) such that the base relation between two assigned values (r_{ij}) is included in the label at the edge connecting the two variables (R_{ij}).

Observation regarding solutions

- If a_i, a_j, a_k are part of a solution then $r_{ij} \in R_{ij}$ and $r_{ij} \subseteq r_{ik} \circ r_{kj}$



Derived ternary constrains

- on edge labels
- from composition of binary relations

Revise labelling

- $R'_{ij} = R_{ij} \cap R_{ik} \circ R_{kj}$

Constraint-Propagation Algorithm (Allen 1983)

- N(i, j): Relations between nodes i and j (old / known)
- R(i, j): Relations between nodes i and j (new / to be integrated)
- ToDo: List of edges

PROCEDURE Propagate R(a, b) into N

```
N(a, b) ← N(a, b) ∩ R(a, b);
Add edge ⟨a, b⟩ to ToDo;
WHILE ToDo is not empty DO
  Get next edge ⟨i, j⟩ from ToDo;
  FOR EACH node k DO
      R(k, j) ← N(k, j) ∩ CT*(N(k, i), N(i, j));
      IF R(k, j) ≠ N(k, j) THEN add ⟨k, j⟩ to ToDo; N(k, j) ← R(k, j) FI;
      R(i, k) ← N(i, k) ∩ CT*(N(i, j), N(j, k));
      IF R(i, k) ≠ N(i, k) THEN add ⟨i, k⟩ to ToDo; N(i, k) ← R(i, k) FI;
  OD;
OD;
```


Irreducible regarding CT

Definition

- Let B be a set of base relations, CT be a composition table for B and C be a CSP based on B.
- *C is irreducible regarding CT* iff
  - for every triangle in the graph representation
    $X_k\ R_{kj}\ X_j,\ X_k\ R_{ki}\ X_i,\ X_i\ R_{ij}\ X_j \in C$: $\quad R_{kj} \subseteq CT^*(R_{ki}, R_{ij})$

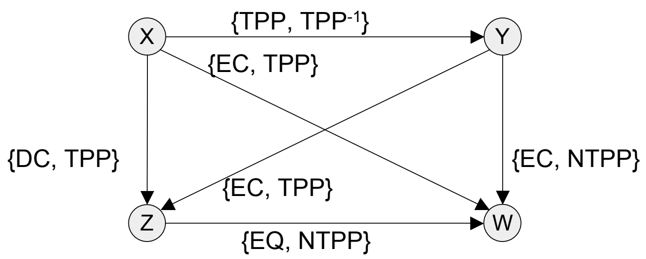
Observation

- The constraint propagation algorithm terminates with a CSP graph that is irreducible regarding CT.

Propagation is not Sufficient

Irreducibility does not imply consistency

- and does not imply unique labels



Renz & Nebel (1998)

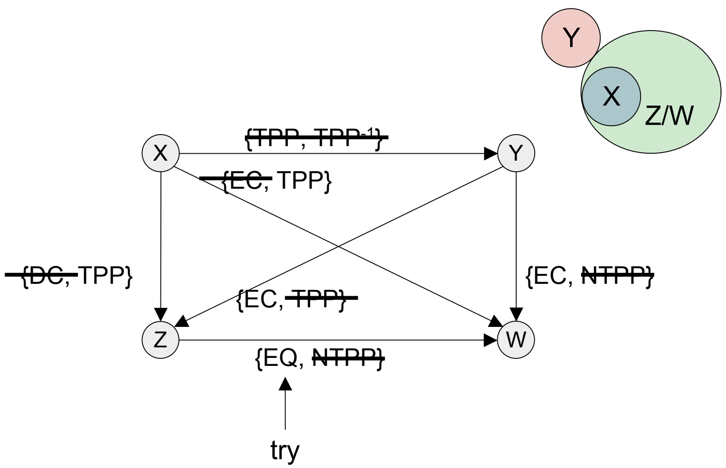
Enforce unique Labels: Backtracking

- Inspect non-unique labels by backtracking with unique labels

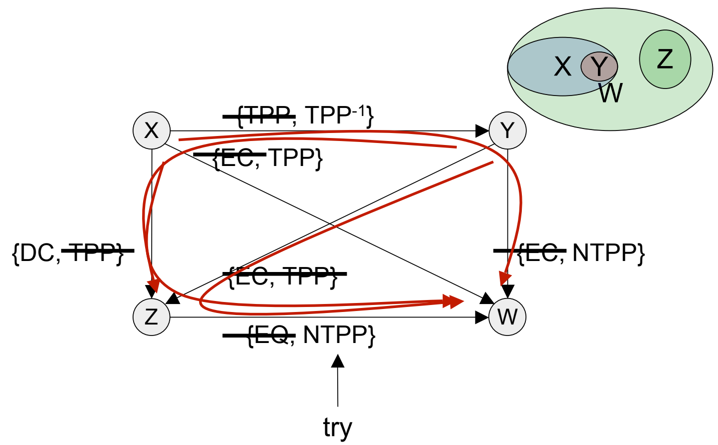
PROCEDURE BacktrackingPropagation R(a, b) into N

Propagate R(a, b) into N;
IF all edges have a unique label in N
THEN RETURN consistent(N);
Get an edge ⟨c, d⟩ with ambiguous label;
FOR EACH base label L of ⟨c, d⟩ DO
  IF BacktrackingPropagation L(c, d) into a copy of N yields consistent(N')
  THEN RETURN consistent(N');
OD;
RETURN inconsistent;

Backtracking Propagation



Backtracking Propagation



CSP for Binary Relation Systems

Solution

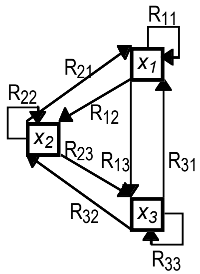
- Assignment of variables to values such that the base relation between two assigned values is included in the label at the edge connecting the two variables

Representation of a solution

- Irreducible graph with unique base-relation label at each edge

Problems

- How to derive such a graph?
- When does the graph guarantee a solution?



When does irreducibility guarantee consistency?

Let B be a set of base relations, CT be a composition table for B that represents (strong) composition, and C be a CSP based on B.

- If in a complete graph-representation of C all edge-labels are unique and irreducible regarding CT, then C is path-consistent.
- If for the set B path-consistency with unique labels guarantees consistency, then C is consistent.

Structure

✓**Problem statement, RSAT, decision procedure**

✓**Constraint Satisfaction Problems (CSP)**

- RSAT as CSP

✓**Systems of Binary Relations**

- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

✓**Constraint-based Reasoning**

- Constraint propagation (based on composition tables)
- Backtracking-propagation

How to Evaluate the Result?

- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency

- When is constraint propagation sufficient to decide RSAT?

Strong and Weak Composition

The basic problem

- Assume a set B of JEPD (jointly exhaustive and pairwise disjoint) binary base relations (such as RCC8) over a domain D, such that Id(D) $\in$ B and for every relation in B, B also contains its converse relation.
- The subsets of B represent (binary) relations (disjunctions of base relations).
- 2^B forms a Boolean Algebra (2^B contains $\emptyset$, B, {Id(D)} and 2^B is closed under $\cup$, $\cap$, $-$, $\breve{}$).
- However, 2^B need not be closed under $\circ$.
- Example: In atomic domains, PP $\circ$ PP $\neq$ PP.

➔(Strong) composition cannot be represented by 2^B.

Strong and Weak Composition (2)

Weak composition

- Let B be a set of JEPD base relations over a domain D, including Id(D) and closed under $\breve{}$.
- Let weak composition $\diamond$ be a binary operator mapping two relations $R_1, R_2 \in 2^B$ to the strongest relation in 2^B that includes $R_1 \circ R_2$ (strong/proper composition).

➔ Composition tables (CT) for weak composition can be finite and are easy to compute.

➔ A CSP graph that is irreducible regarding CT need not be 3-consistent / path-consistent.

Reading Weak Composition Tables

$\diamond$	…	…	R_2	…
…	…	…	V_2	…
R_1	…	…	R_3	…
…	…	…	…	…
…	…	…	…	…

Weak Composition Tables

$R_1 \circ R_2 \subseteq R_3$

Formulated in Predicate Logic

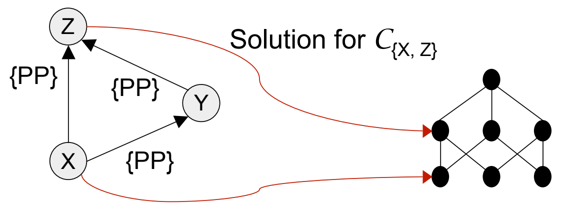
$\forall x\ y\ z\ [R_1(x, y) \wedge R_2(y, z) \Rightarrow R_3(x, z)]$

Most composition tables you will find are weak composition tables !

Graph of Base Relations is not Sufficient

A partial solution that cannot be extended

Constraint graph for C Domain D



The underlying problem

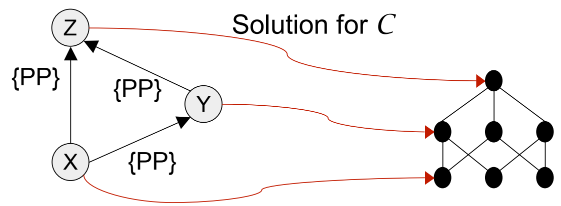
- Restrictions between two variables enforced by larger graphs cannot be completely expressed by the edge label.

Why Weak Compostion Need not be a Problem

3-consistency is too strong

- if our goal is to check consistency.
- The example graph has a solution ! (even within D)

Constraint graph for C Domain D

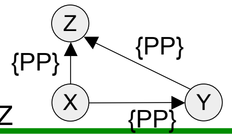


Refinement to Subatomic Relations

Let C be a consistent base-relation CSP over a domain D and X_i R $X_j \in C$ be a constraint. Let R' be the set of all pairs (a_i, a_j) that appear in any solution of C as values for X_i and X_j. Thus $\emptyset \neq$ R' $\subseteq$ R. If R' $\subset$ R, then *C refines R to the subatomic relation R'*.

If C refines R to a subatomic relation R' ,

- then at least one solution for X_i R X_j cannot be extended to a solution for C
- and C cannot be globally consistent.



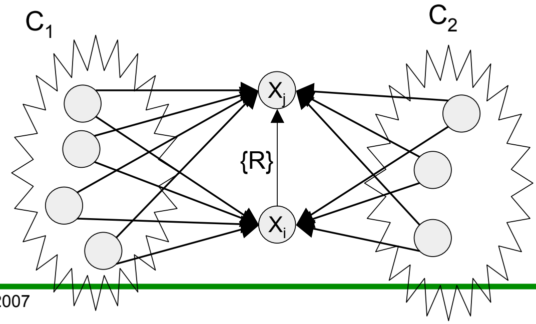
C refines X PP Z

Renz & Ligozat (2005)

The Problem of Refinement

Conflicting refinement can yield inconsistency

- If different CSPs can refine a base relation R to exclusive subatomic relations, then there can be inconsistent CSPs with unique labels that are irreducible regarding CT.



Closed under Constraints

Let B be a set of base relations in a domain D. *B is closed under constraints* iff for each base relation R $\in$ B all subatomic relations R', R'' $\subset$ R to which R can be refined have a nonempty intersection.

Theorem

- Let D be a domain, B be a finite set of base relations over D, and CT a (weak) composition table for B. Then every B-CSP that is irreducible regarding CT is consistent, iff B is closed under constraints.

Renz & Ligozat (2005)

First Summary: RSAT as CSP

RSAT

- Given: a set S of formulae of the form R(X_i, X_j), where R $\in \mathcal{R}$.
- Problem: Is S satisfiable?
- Can be considered as CSP with infinite domain.

Representation of a solution

- Irreducible graph with unique base-relation label at each edge

Constraint-based reasoning techniques

- Constraint propagation
- Backtracking-propagation

System of base relations

- Reducibility: Composition tables
  - Specify operation of composition in a strong or weak manner
- Closed under constraints
  - Inconsistencies are guarateed to show up in solution graphs

Summary: Reasoning

- Sound inference procedures can easily be define.
- General inference problems are not decidable
  - (as usual with first-order logic)
- RSAT: The restricted problem is decidable for mereological, mereotopological, and temporal relations
  - … but in general not tractable (NP-hard)
- Restricting the set of allowed relations (to well-behaved sets)
  - can yield tractability
- Using well-behaved sets in backtracking-search
  - can yield algorithms that can answer many questions

Good News & Bad News

Good news

- Closed under constraints
  - Mereological base relation {DR, PO, PP, EQ, PPi}
  - RCC-8 Mereotopological base relations {DC, EC, PO, NTPP, TPP, EQ, NTPPi, TPPI}

Bad news (?)

- To apply constraint-based reasoning
  - for other (refined) sets of relations
  - and rely on positive outcomes
- one (only) has to check (prove)
  - that the new set is closed under constraints.

Structure

✓**Problem statement, RSAT, decision procedure**

✓**Constraint Satisfaction Problems (CSP)**
- RSAT as CSP

✓**Systems of Binary Relations**
- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

✓**Constraint-based Reasoning**
- Constraint propagation (based on composition tables)
- Backtracking-propagation

✓**How to Evaluate the Result?**
- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

Efficiency
- When is constraint propagation sufficient to decide RSAT?

Efficiency

Constraint-based reasoning algorithms

- Backtracking (depth-) search
  - intractable for large sets of variables: (NP-hard)
- Constraint propagation
  - tractable: $O(n^3)$

Efficient procedures exist

- For RSAT-problems where irreducibility regarding CT is sufficient for satisfiability
- Restrict the set of available relations (do not require closedness under –, ∪)

Role of Constraint Propagation

Preprocessing

- for backtracking search
- producing irreducible graphs

Forward checking

- as part of backtracking

Complete procedure for restricted languages

- Result of Nebel & Renz (1999): There are sub-sets $B \subset \hat{}B \subset \hat{}H_8$ of the 256 RCC8-relations such that if all constraints of the initial CSP are from these sub-sets, then irreducibility regarding CT_{RCC} guarantees consistency. ($\hat{}H_8$ has 148 relations.)
- Similar results for time; Nebel & Bürckert (1995)

Backtracking

- is computationally costly

Split-set B

- branches according to division of labels into labels from B
- enforces unique labels
- average branching factor (RCC-8): 4.0

Split-set $\hat{}H_8$

- branches according to division of labels into label from $\hat{}H_8$
- enforces labels from $\hat{}H_8$
- average branching factor (RCC-8): 1.4375

Complete procedures: Backtracking

- Inspect non-unique labels by backtracking with labels from F

PROCEDURE BacktrackingPropagation2 R(a, b) into N using F
Propagate R(a, b) into N;
IF all edges in N have a label belonging to F
THEN RETURN consistent(N);
Get an edge ⟨c, d⟩ with label X not in F;
Get a small set S of relations from F such X is the disjunction of S;
FOR EACH member L of S DO
IF BacktrackingPropagation2 L(c, d) into a copy of N using F yields consistent(N')
THEN RETURN consistent(N');
OD;
RETURN inconsistent;

Structure

✓**Problem statement, RSAT, decision procedure**

✓**Constraint Satisfaction Problems (CSP)**
- RSAT as CSP

✓**Systems of Binary Relations**
- Base relations
- (Atomic) Concrete/abstract relation algebras
- Composition tables

✓**Constraint-based Reasoning**
- Constraint propagation (based on composition tables)
- Backtracking-propagation

✓**How to Evaluate the Result?**
- Strong and weak composition
- When does constraint-based reasoning decide RSAT?

✓**Efficiency**
- When is constraint propagation sufficient to decide RSAT?