Advanced consistency methods
Chapter 8

ICS-275
Spring 10
Relational consistency (Chapter 8)

- Relational arc-consistency
- Relational path-consistency
- Relational m-consistency

Relational consistency for Boolean and linear constraints:
- Unit-resolution is relational-arc-consistency
- Pair-wise resolution is relational path-consistency
Example

Consider a constraint network over five integer domains, where the constraints take the form of linear equations and the domains are integers bounded by

- \( D_x \) in \([-2,3]\)
- \( D_y \) in \([-5,7]\)
- \( R_{xyz} := x + y = z \)
- \( R_{ztl} := z + t = l \)
- From \( D_x \) and \( R_{xyz} \) infer \( z - y \) in \([-2,3]\) from this and \( D_y \) we can infer \( z \) in \([-7,10]\)
Relational arc-consistency

Let $R$ be a constraint network, $X= \{x_1, \ldots, x_n\}$, $D_1, \ldots, D_n$, $R_S$ a relation.

$R_S$ in $R$ is relational-arc-consistent relative to $x$ in $S$, iff any consistent instantiation of the variables in $S - \{x\}$ has an extension to a value in $D_x$ that satisfies $R_S$. Namely,

$$\rho(S - x) \subseteq \pi_{S - x} R_S \otimes D_x$$
Enforcing relational arc-consistency

- If arc-consistency is not satisfied add:

\[ R_{S-x} \leftarrow R_{S-x} \cap \pi_{S-x} R_S \otimes D_S \]
Example

- \( R_{\{xyz\}} = \{(a,a,a),(a,b,c),(b,b,c)\} \).
- This relation is not relational arc-consistent, but if we add the projection:
  \[ R_{\{xy\}} = \{(a,a),(a,b),(b,b)\} , \]
  then \( R_{\{xyz\}} \) will be relational arc-consistent relative to \( \{z\} \).
- To make this network relational-arc-consistent, we would have to add all the projections of \( R_{\{xyz\}} \) with respect to all subsets of its variables.
Relational path-cosistency

- Let $R_S$ and $R_T$ be two constraints in a network.
- $R_S$ and $R_T$ are relational-path-consistent relative to a variable $x$ in $S \cup T$ iff any consistent instantiation of variables in $S \cap T - \{x\}$ has an extension to in the domain $D_x$, s.t. $R_S$ and $R_T$ simultaneously;

$$\rho(A) \subseteq \pi_A R_S \otimes R_T \otimes D_x$$

$$A = S \cup T - x$$

- A pair of relations $R_S$ and $R_T$ is relational-path-consistent iff it is relational-path-consistent relative to every variable in $S \cap T$. A network is relational-path-consistent iff every pair of its relations is relational-path-consistent.
\[D_x \text{ in } [-2,3]\]
\[D_y \text{ in } [-5,7]\]

\[R_{xyz} := x + y = z\]
\[R_{ztl} := z + t = l\]

- We can assign to \( x, y, l \) and \( t \) values that are consistent relative to the relational-arc-consistent network generated in earlier. For example, the assignment

- \((x=2, y= -5, t=3, l=15)\) is consistent, since only domain restrictions are applicable, but no value of \( z \) that satisfies \( x+y = z \) and \( z+t = l \).

- To make the two constraints relational path-consistent relative to \( z \) add: \( x+y+t = l \).
Enforcing relational arc, path and m-consistency

- If arc-consistency is not satisfied add:

\[ R_{S-x} \leftarrow R_{S-x} \cap \pi_{S-x} R_{S} \otimes D_{S} \]

\[ \rho(A) \subseteq \pi_{A} R_{S} \otimes R_{T} \otimes D_{x} \]

\[ A = S \cup T - x \]

\[ \rho(A) \subseteq \pi_{A} \otimes_{i=1,m} R_{S_{i}} \otimes D_{x} \]

\[ A = S_{1} \cup \ldots S_{m} - x \]
Extended composition

- The extended composition of relation $R_{\{S_1\}}, \ldots, R_{\{S_m\}}$ relative to $A$ is defined by
  
  \[
  EC_A(R_1, \ldots, R_m) = \pi_A(R_1 \otimes R_2 \otimes \ldots, \otimes R_m)
  \]

- If the projection operation is restricted to subsets of size $i$, it is called extended $(i,m)$-composition.

- Special cases: domain propagation and relational arc-consistency

\[
D_x \leftarrow D_x \cap \pi_x R_S \otimes D_S
\]

\[
R_{S-x} \leftarrow R_{S-x} \cap \pi_{S-x} R_S \otimes D_S
\]
Directional relational consistency

- Given an ordering $d$, a constraint network $R$ is $m$-directionally relationally consistent r.t $d$ iff for every subset $R_1, \ldots, R_m$ whose latest variable is $x_l$, for every $A$ in $\{x_1, \ldots, x_{l-1}\}$, every consistent assignment to $A$ can be extended to $x_l$ simultaneously satisfying all these constraints.
Summary: directional i-consistency

E: \( E \neq D, E \neq C, E \neq B \)
D: \( D \neq C, D \neq A \)
C: \( C \neq B \)
B: \( A \neq B \)
A: 

Adaptive

\[ R_{DCB} \]

d-path

\[ R_{DC}, R_{DB}, R_{CB} \]

d-arc

\[ R_D, R_C, R_D \]
Example: crossword puzzle

\[ R_{1,2,3,4,5} = \{(H,O,S,E,S), (L,A,S,E,R), (S,H,E,E,T), \]
\[ (S,N,A,I,L), (S,T,E,E,R)\} \]
\[ R_{3,6,9,12} = \{(H,I,K,E), (A,R,O,N), (K,E,E,T), (E,A,R,N), \]
\[ (S,A,M,E)\} \]
\[ R_{5,7,11} = \{(R,U,N), (S,U,N), (L,E,T), (Y,E,S), (E,A,T), (T,E,N)\} \]
\[ R_{8,9,10,11} = R_{3,6,9,12} \]
\[ R_{10,13} = \{(N,O), (B,E), (U,S), (I,T)\} \]
\[ R_{12,13} = R_{10,13} \]
Example: crossword puzzle, DRC_2

\[
\begin{align*}
\text{bucket}(x_1) & \rightarrow R_{1,2,3,4,5} \\
\text{bucket}(x_2) & \rightarrow H_{2,3,4,5} \\
\text{bucket}(x_3) & \rightarrow R_{3,6,9,12} \\
& \quad \rightarrow H_{3,4,5} \\
\text{bucket}(x_4) & \rightarrow H_{4,5,6,9,12} \\
\text{bucket}(x_5) & \rightarrow R_{5,7,11} \\
& \quad \rightarrow H_{5,6,9,12} \\
\text{bucket}(x_6) & \rightarrow H_{6,7,9,11,12} \\
\text{bucket}(x_7) & \rightarrow H_{7,9,11,12} \\
\text{bucket}(x_8) & \rightarrow R_{8,9,10,11} \\
& \quad \rightarrow H_{9,10,11,12} \\
\text{bucket}(x_9) & \rightarrow H_{9,10,11,12} \\
\text{bucket}(x_{10}) & \rightarrow R_{10,13} \\
& \quad \rightarrow H_{10,11,12} \\
\text{bucket}(x_{11}) & \rightarrow \text{Empty relation . . . exit.} \\
\text{bucket}(x_{12}) & \rightarrow R_{12,13} \\
\text{bucket}(x_{13}) &
\end{align*}
\]
Complexity

- Even DRC_2 is exponential in the induced-width.

- Crossword puzzles can be made directional backtrack-free by DRC_2.
Domain tightness

- **Theorem:** A strong relational 2-consistent constraint network over bi-valued domains is globally consistent.

- **Theorem:** A strong relational k-consistent constraint network with at most k values is globally consistent.
Inference for Boolean theories

- Resolution is identical to Extended 2 decomposition
- Boolean theories have domain size 2
- Therefore DRC_2 makes a cnf globally consistent.
- DRC_2 expressed on cnfs is directional resolution
**Directional-resolution**

**Directional-Resolution**

**Input:** A CNF theory $\varphi$, an ordering $d = Q_1, \ldots, Q_n$ of its variables.

**Output:** A decision of whether $\varphi$ is satisfiable. If it is, a theory $E_d(\varphi)$, equivalent to $\varphi$, else an empty directional extension.

1. **Initialize:** generate an ordered partition of clauses into buckets. $\text{bucket}_1, \ldots, \text{bucket}_n$, where $\text{bucket}_i$ contains all clauses whose highest literal is $Q_i$.

2. **for** $i \leftarrow n$ **down to** 1 process $\text{bucket}_i$:

3. **if** there is a unit clause **then** (the instantiation step)
   - apply unit-resolution in $\text{bucket}_i$ and place the resolvents in their right buckets.
   - **if** the empty clause was generated, **theory is not satisfiable**.

4. **else** resolve each pair $\{(\alpha \lor Q_i), (\beta \lor \neg Q_i)\} \subseteq \text{bucket}_i$.
   - **if** $\gamma = \alpha \lor \beta$ is empty, return $E_d(\varphi) = \{\}$, **theory is not satisfiable**
   - **else** determine the index of $\gamma$ and add it to the appropriate bucket.

5. return $E_d(\varphi) \leftarrow \bigcup_i \text{bucket}_i$

Figure 4.20: Directional-resolution
DR resolution = adaptive-consistency=directional relational path-consistency

\[ |bucket_i| = O(\exp(w^*)) \]

DR time and space: \( O(n \exp(w^*)) \)
Directional Resolution ⇔ Adaptive Consistency

Knowledge compilation

Bucket A

\[ A \lor B \lor C \land \neg A \lor B \lor E \]

Bucket B

\[ \neg B \lor C \lor D \land B \lor C \lor E \]

Bucket C

\[ \neg C \land C \lor D \lor E \]

Bucket D

\[ D \lor E \]

Bucket E

Directional Extension

Model generation

Input

\[ A = 0 \]

\[ B = 1 \]

\[ C = 0 \]

\[ D = 1 \]

\[ E = 0 \]
History

- 1960 – resolution-based Davis-Putnam algorithm
- 1962 – resolution step replaced by conditioning (Davis, Logemann and Loveland, 1962) to avoid memory explosion, resulting into a backtracking search algorithm known as Davis-Putnam (DP), or DPLL procedure.
- The dependency on induced width was not known in 1960.
- 1994 – Directional Resolution (DR), a rediscovery of the original Davis-Putnam, identification of tractable classes (Dechter and Rish, 1994).
Complexity of DR

Theorem 4.7.6 (complexity of DR)
Given a theory $\varphi$ and an ordering of its variables $o$, the time complexity of algorithm DR along $o$ is $O(n \cdot 9^{w^*})$, and $E_o(\varphi)$ contains at most $n \cdot 3^{w^* + 1}$ clauses, where $w^*_o$ is the induced width of $\varphi$'s interaction graph along $o$. $\square$

- 2-cnfs and Horn theories

Theorem 4.7.7 Given a 2-cnf theory $\varphi$, its directional extension $E_o(\varphi)$ along any ordering $o$ is of size $O(n \cdot w^*_o^2)$, and can be generated in $O(n \cdot w^*_o^2)$ time.

Theorem 4.7.8 The consistency of Horn theories can be determined by unit propagation. If the empty clause is not generated, the theory is satisfiable. $\square$
Linear inequalities

- Consider r-ary constraints over a subset of variables $x_1, \ldots, x_r$ of the form
  
  $$a_1 x_1 + \ldots + a_r x_r \leq c,$$
  
  Where $a_i$ are rational constants. The r-ary inequalities define corresponding r-ary relations that are row convex.

- Since r-ary linear inequalities that are closed under relational path-consistency are row-convex, relative to any set of integer domains (using the natural ordering).

- **Proposition:** A set of linear inequalities that is closed under RC_2 is globally consistent.
Linear inequalities

- Gaussian elimination with domain constraint is relational-arc-consistency
- Gaussian elimination of 2 inequalities is relational path-consistency

**Theorem:** directional relational path-consistency is complete for CNFs and for linear inequalities
**Directional-Linear-Elimination** $(\varphi, d)$

**Input:** A set of linear inequalities $\varphi$, an ordering $d = x_1, \ldots, x_n$.

**Output:** A decision of whether $\varphi$ is satisfiable. If it is, a backtrack-free theory $E_d(\varphi)$.

1. **Initialize:** Partition inequalities into ordered buckets.

2. **for** $i \leftarrow n$ **downto** 1 **do**

3.  **if** $x_i$ has one value in its domain **then**

   substitute the value into each inequality in the bucket and put the resulting inequality in the right bucket.

4.  **else,** for each pair $\{\alpha, \beta\} \subseteq \text{bucket}_i$, compute $\gamma = \text{elim}_i(\alpha, \beta)$

   if $\gamma$ has no solutions, return $E_d(\varphi) = \{\}$, “inconsistency”

   else add $\gamma$ to the appropriate lower bucket.

5. **return** $E_d(\varphi) \leftarrow \bigcup_i \text{bucket}_i$

---

Figure 4.22: Fourier Elimination; DLE
Directional linear elimination, DLE: generates a backtrack-free representation

Theorem 4.8.3 Given a set of linear inequalities $\varphi$, algorithm DLE (Fourier elimination) decides the consistency of $\varphi$ over the Rationals and the Reals, and it generates an equivalent backtrack-free representation. □
Example

\[ \text{bucket}_4 : \quad 5x_4 + 3x_2 - x_1 \leq 5, \quad x_4 + x_1 \leq 2, \quad -x_4 \leq 0, \]
\[ \text{bucket}_3 : \quad x_3 \leq 5, \quad x_1 + x_2 - x_3 \leq -10 \]
\[ \text{bucket}_2 : \quad x_1 + 2x_2 \leq 0. \]
\[ \text{bucket}_1 : \]

Figure 4.23: initial buckets

\[ \text{bucket}_4 : \quad 5x_4 + 3x_2 - x_1 \leq 5, \quad x_4 + x_1 \leq 2, \quad -x_4 \leq 0, \]
\[ \text{bucket}_3 : \quad x_3 \leq 5, \quad x_1 + x_2 - x_3 \leq -10 \]
\[ \text{bucket}_2 : \quad x_1 + 2x_2 \leq 0 \parallel 3x_2 - x_1 \leq 5, \quad x_1 + x_2 \leq -5 \]
\[ \text{bucket}_1 : \quad \parallel x_1 \leq 2. \]

Figure 4.24: final buckets