Class Discussion on Advanced Consistency Methods (Sections 8.1-8.3)
by Yaling Zheng (March 24-April 2)
Scribe: Nurzhan Ustemirov

1 Overview

Until this class we have seen variable-based consistency algorithms. In this chapter, Yaling discussed relation-based consistency enforcing algorithms.

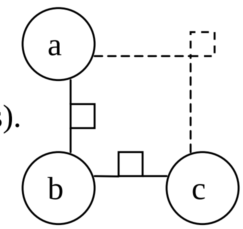
8.1 Relational consistency

Relational-arc-consistency *(RAC):* given a network ***R*** (CSP) and a relation R_S over a set S, R_S is said to be *RAC relative to a variable* $x \in S$ iff any instantiation of variables in S-{x} could be extended to a value in D_x that satisfies R_S:

$$\rho(S-\{x\}) \subseteq \pi_{S-\{x\}}(R_S >< D_x)$$

A relation *R_S is RAC* if it is *RAC relative* to every variable in S.
Finally, a CSP is RAC iff every constraint in CSP is RAC.

Example: {a, b, c} are variables, R_{ab} and R_{bc} are relations (constraints). And newly formed constraint (shaded rectangle with shaded edges) R_{ac} is RAC relative to ‘b’. Where, ‘b’ is a common variable.

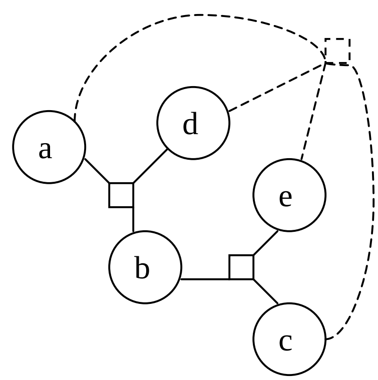


Relational-path-consistency *(RPC):* given a network ***R*** (CSP) and relations R_S and R_D, R_S and R_D are said to be *RPC relative to a variable* $x \in S \cap T$ iff any instantiation of variables in
$A = S \cup T - \{x\}$ could be extended to a value in D_x that satisfies both R_S and R_D:

$$\rho(A) \subseteq \pi_A(R_S >< R_T >< D_x)$$

A pair of relations *R_S and R_D is RPC* iff it is *RPC relative* to every variable in $S \cap T$.
Finally, a CSP is RPC iff every pair of its relations is RPC.

Example: {a, b, c, d, e} are variables, R_{abd} and R_{bce} are relations (constraints). And newly formed constraint R_{adce} is RPC relative to ‘b’, where ‘b’ is a common variable.



From class discussion:

- If a graph is complete or triangulated RPC will guarantee global consistency.
- We are inferring a new relation between variables that are not related.
- In binary-CSP we use relations to update domains, while in relational-CSP we use a domain of one variable to update domain of other variables.

Relational-m-consistency *(R-m-C):* given a network ***R*** (CSP) and relations $R_{S1}, \ldots R_{Sm}$, $R_{S1}, \ldots R_{Sm}$ are said to be *R-m-C relative to a variable* $x \in \bigcap_{i=1}^{m} S_i$ iff any instantiation of variables in $A = \bigcup_{i=1}^{m} S_i - \{x\}$ has an extension to a value in D_x that satisfies all $R_{S1}, \ldots R_{Sm}$ constraints:

$$\rho(A) \subseteq \pi_A(\bowtie_{i=1}^{m} R_{Si} \bowtie D_x)$$

Set of relations $R_{S1}, \ldots R_{Sm}$ is *R-m-C* iff it is *R-m-C relative* to every variable in $\bigcap_{i=1}^{m} S_i$. Finally, a CSP is R-m-C iff every set of m relations is R-*m*-C, a CSP is strong R-m-C if it is R-*i*-C for every $i \leq m$.

8.1.1 Space-bound vs time-bounds

Relational (i, m)-consistency *(R(i,m)-C): a set of relations $\{R_{S1}, \ldots R_{Sm}\}$ is R(i,m)-C iff for every subset (A) of size i $A \subseteq \bigcup_{j=1}^{m} S_j$, any consistent assignment to A could be extended to $\bigcup_{j=1}^{m} S_j - A$ that satisfies all relation in the set $\{R_{S1}, \ldots R_{Sm}\}$. Similarly the CSP would be R(i,m)-C iff every set of size m relations is R(i,m)-C. In addition, a CSP is strong R(i,m)-C iff it is R(j,m)-C for every $j \leq i$.*

From class discussion:

- Inverse arc consistency and arc consistency are identical
  (relational (1,1)-consistency and AC)
- Neighborhood inverse consistency was discussed
  (for those who are interested briefly[4], advanced paper [5])

8.2 Directional-relational-consistency

Directional relational consistency (DRC)*:* given a network ***R*** (CSP) and ordering $d=(x_1, \ldots x_n)$, ***R*** is m-directionally relationally consistent iff

- for every subset of constraints $\{R_{S1}, \ldots, R_{Sm}\}$ where the latest variable in any of the constraints' scopes is x_l
- for every *A*, $A \subseteq \{x_1, \ldots, x_{l-1}\}$, every consistent assignment to *A* can be extended to x_l
- simultaneously satisfies all relevant constraints in $\{R_{S1}, \ldots, R_{Sm}\}$.

Complexity of DRC*:*

- time complexity of $\boldsymbol{DRC_m}$ along ordering ***d*** is $O\left(nm \cdot \left(2^m k^2\right)^{(w^*(d)+1)}\right)$, *w*(d)* is width of a directional graph, ***n*** is number of variables, and ***k*** bounds the domain size
- time complexity of $\boldsymbol{DRC_2}$ is $O\left(n\left(4k^2\right)^{(w^*(d)+1)}\right)$
- complexity of $\boldsymbol{DRC_2}$ is $O\left(n \cdot e^2 \cdot t^2\right)$ where ***e*** is the number of input relations and ***t*** bounds the number of tuples in each relation

8.3 Domain and constraint tightness

Domain tightness:

- *A strong relational 2-consistent network over bi-valued domains is globally consistent*
- *A strong relational k-consistent constraint network R=(X,D,C) over domains with at most k values is globally consistent*

Constraint tightness:

- *A constraint relation R_S of arity r is called m-tight if, for any variable $x_i \in S$ and any instantiation $\bar{a}$ of the remaining r-1 variables in S-{x_i}, either there are at most m extensions of $\bar{a}$ to x_i that satisfy R_S, or there are exactly $|D_S|$ such extensions*
- *A strongly relationally (m+1)-consistent constraint network with m-tightness is globally consistent*

8.4 Inference for Boolean theories

Resolution operation over two clauses $(\alpha \vee Q)$ and $(\alpha \vee \neg Q)$ returns

$$(\alpha \vee \beta) = EC_{Q'}(\mathrm{mod}\,els(\alpha \vee Q), \mathrm{mod}\,els(\beta \vee \neg Q)),$$

where *Q'* is the union of scopes of both clauses excluding Q (there are typos in the slide #44, AND operators should be changed to OR in the second clause, and it is "and" not OR between two clauses).

The *resolution operation* could be generalized extending literals α and β to multiple *conjunctions* of literals as long as both clauses contain a common literal (should be negated in either clause).

Relevance to the discussion: "resolution operation" could modify ***RC₂*** to solve *CNF* theories. Extended 2-composition from ***RC₂*** is replaced with resolution, and applied until no new resolvents generated.

Similarly "resolution operation" could be used to modify **DRC₂** into Directional-resolution (***DR***). In the ***DR*** algorithm extended 2-composition replaced with resolution and instantiation is replaced with unit resolution.

Input: a Boolean network φ in CNF form and an ordering ***d.***

Output: A decision whether φ is satisfiable or not.

Reference:

[1] Yaling Zheng, *Lecture Slides*

[2] Rina Dechter, *Constraint Processing*

[3] Class, *Class discussions*

[4] http://ic.arc.nasa.gov/ic/projects/mba/cs329a/handouts/handout-8.pdf

[5] ftp://ftp.cs.unh.edu/pub/csp/Papers/aaai96-nic-cde-ecf.ps.gz