Reformulations of CSPs

Tero Laitinen

March 23, 2009

Why Reformulations

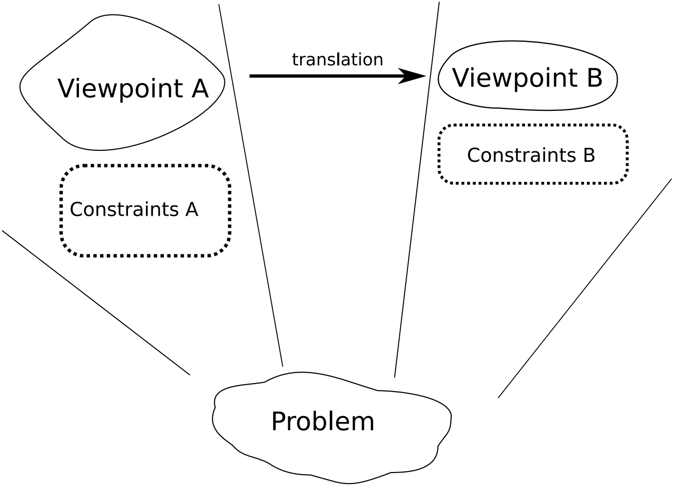
- easier to model
  - less constraints
  - less variables
  - better expressivity
- faster search
  - better propagation
  - smaller search space

Viewpoint

Definition (Viewpoint)

A *viewpoint* is a pair $\langle X, D \rangle$ with respect to some problem P, where $X = \{ x_1, \ldots, x_n \}$ is a set of variables, and D is a set of domains; for each $x_i \in X$, D_i is the set of possible values for x_i. For each viewpoint of problem P, there is a corresponding set of constraints which will define all possible solutions of P.

What Are Translations



Non-Binary to Binary Translations

- non-binary constraints have not been available always
  $\Longrightarrow$ there are standard translations for decomposing such constraints

Hidden Variable Transformation

Definition

The *hidden variable* transformation translates non-binary constraints to binary constraints by adding hidden variables.

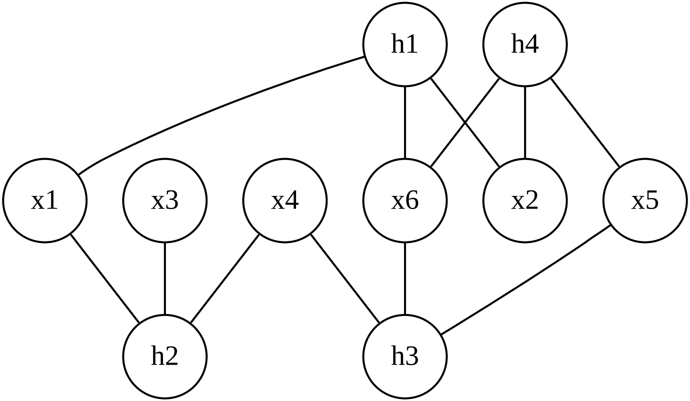
- new variable h_i for each non-binary constraint c_i
- values of h_i are tuples satisfying c_i
- constraint c_i replaced by binary constraints between h_i and the variables in c_i

Example of Hidden Variable Transformation

Example

Consider the 3-SAT problem,
$\psi = (x_1 \vee x_2 \vee x_6) \wedge (\neg x_1 \vee x_3 \vee x_4) \wedge (\neg x_4 \vee \neg x_5 \vee x_6) \wedge (x_2 \vee x_5 \vee \neg x_6)$.
A solution is a truth assignment $\mathcal{A}$ such that $\mathcal{A} \models \psi$, for example, $\{ x_1, x_3, x_5, x_6 \}$

Example of Hidden Variable Transformation



$$(x_1 \vee x_2 \vee x_6) \wedge (\neg x_1 \vee x_3 \vee x_4) \wedge (\neg x_4 \vee \neg x_5 \vee x_6) \wedge (x_2 \vee x_5 \vee \neg x_6)$$

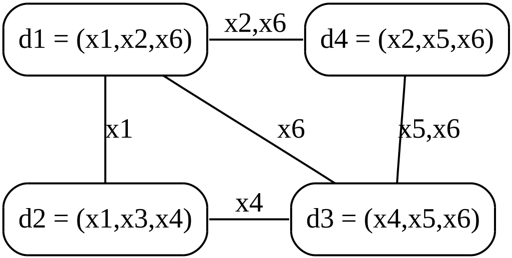
Dual Graph Translation

Definition

The *dual graph* translation replaces the original constraints by new variables.

- dual variable d_i represents constraint c_i
- values of d_i are tuples satisfying c_i
- there is a binary constraint between dual variables d_i and d_j if c_i and c_j have common satisfying tuples

Example of Dual Graph Translation



$$(x_1 \vee x_2 \vee x_6) \wedge (\neg x_1 \vee x_3 \vee x_4) \wedge (\neg x_4 \vee \neg x_5 \vee x_6) \wedge (x_2 \vee x_5 \vee \neg x_6)$$

Example : Covering Array

Example

For a given tuple (t, k, g, b) find a *covering array* $CA(t, k, g)$ of size b or show that none exists. The covering array has b rows, k columns, and in every subset of t columns every possible t-tuple over the alphabet $Z_g = \{0, 1, \ldots, g-1\}$ must occur at least once.

Example : Covering Array (continued)

1	2	3	4	5
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

A covering array $CA(3, 5, 2)$ of size 10.

Example : Covering Array (continued)

Definition

compound variables, analogous to the variables of the dual graph transformation, represent every t-tuple of columns in each row. Each compound variable has domain $\{0, \ldots, 2^t\}$. A global cardinality constraint ensures that every value is assigned at least once. Also, there are binary constraints between compound variables to ensure that they agree on the values of the shared variables.

Summary of Non-Binary to Binary Translations

- in some cases more efficient (even orders of magnitude)
- little used in practice due to wide set of fast non-binary constraint implementations
- useful in rewriting non-binary constraints

Permutation Problems

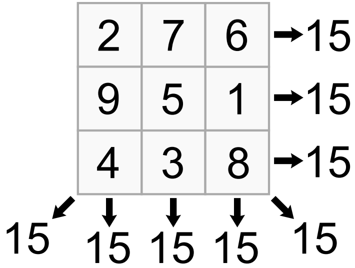
Definition

A CSP is a *permutation problem* if the union of the domains has the same number of elements as there are variables and each variable must be assigned a different value.

Definition

The *dual viewpoint* switches the roles of variables and values.

Permutation Problem : Magic Square



duality : either x_i represents number in the cell i
or x_i is the cell for the number i

Different Perspectives

- sometimes possible to find a new viewpoint
- different insights may express alternative ways to solve the problem
- viewpoints can be combined

Combining Viewpoints

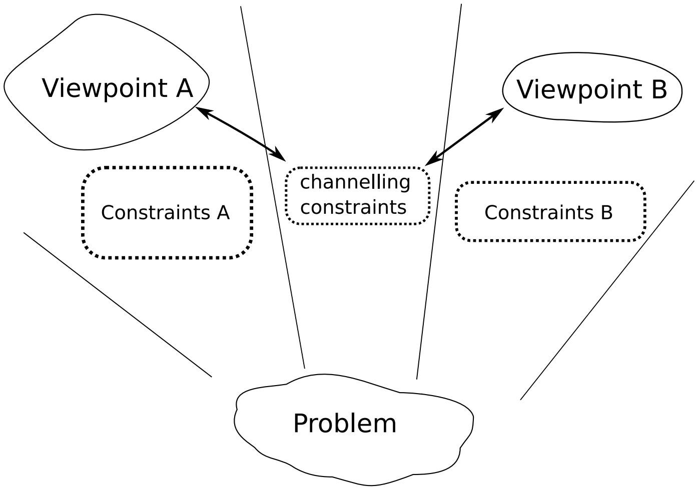
Definition

Given two viewpoints $V_1 = \langle X_1, D_1 \rangle$, $V_2 = \langle X_2, D_2 \rangle$ for the same problem, a complete model can be constructed from each viewpoint: $M_1 = \langle X_1, D_1, C_1 \rangle$, $M_2 = \langle X_2, D_2, C_2 \rangle$. The models are *mutually redundant*.

Definition

The combined model has variables $X_1 \cup X_2$ and constraints $C_1 \cup C_2 \cup C_c$, where C_c is a set of *channelling constraints*. The channelling constraints express the relationship between X_1 and X_2.

Combining Viewpoints



Why Combine Viewpoints

- more effective constraint propagation
- maybe the only feasible way to express the problem
- possible to combine more than two viewpoints

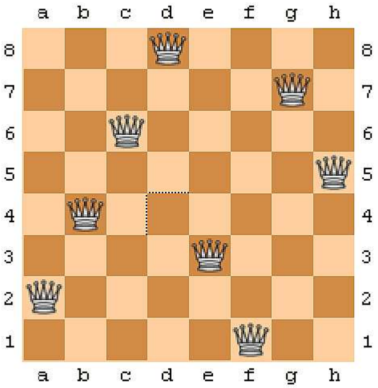
Selecting Constraints

- possibly unnecessary constraints in a combined model (*propagation redundant*)
- faster search by dropping some constraints

Sources of Symmetry

- viewpoint with distinct variables/values for indistinguishable entities
- use of sequences when the order does not matter

Example of Symmetry : N-queens



A bad viewpoint : 1st queen, 2nd queen, etc.

Example of Symmetry : Golfers

Example (Problem)

32 golfers want to play in 8 groups of 4 each week, in such a way that any two golfers play in the same group at most once. How many weeks can they do this for?

Example (Possible viewpoint)

A possible viewpoint has variables x_{ijkl} where $x_{ijkl} = 1$ if player i is the jth player in the kth group in week l and 0 otherwise.
Symmetry : the players within each group, the groups within each week and the weeks can be permuted for more solutions.

Example of Symmetry : Golfers (continued)

Example (Better viewpoint)

The set variable G_{kl} represents the kth group in week l and the value of the variable represents the set of players in the group. Still the weeks remain interchangeable..

Eliminating Symmetry

- fixing entities of the problem to variables in an unambiguous way
  - $\Longrightarrow$ no permutations as solutions
- use of set variables instead of sequences
- symmetry-breaking constraints

Summary

- why reformulations
  - faster search
  - easier to model
- translations are used to improve existing viewpoints
- different viewpoints can be combined
  - faster search
  - maybe the only way to model a problem
- symmetry should be eliminated