

Unequality constraints to propositional logic

Matouš Mařík

Suppose we are given a CSP over some unified domain $D : [d_1 \dots d_2]$ where all constraints are of the form $v_i \leq v_j$, $v_i - v_j \leq c$ or $v_i = v_j + c$ for some constant c . For example

$$((v_2 \leq v_3) \vee (v_4 \leq v_1)) \wedge (v_2 = v_1 + 4) \wedge v_4 = v_3 + 3$$

for $v_1, v_2, v_3, v_4 \in [0 \dots 7]$ is a formula belonging to this fragment. This formula is satisfied by one of two solutions: $(v_1 \mapsto 0, v_2 \mapsto 4, v_3 \mapsto 4, v_4 \mapsto 7)$, or $(v_3 \mapsto 0, v_1 \mapsto 3, v_4 \mapsto 3, v_2 \mapsto 7)$. Show a reduction of such formulas to propositional logic.

Hint: an encoding which requires $|V| \cdot |D|$ propositional variables, where $|V|$ is the number of variables and $|D|$ is the size of the domain, is given by introducing a propositional variable b_{ij} for each variable $v_i \in V$ and $j \in D$, which indicates that $v_i \leq j$ is true.

Pozn.: Výsledná formule bude konjunkce jednotlivých omezení.

Nechť $b_{i,j}$ jsou proměnné zakódované podle nápovědy (s indexováním V a D od 1). Je třeba zajistit konzistenci výsledného ohodnocení přidáním podmínky:

$$\forall i \in 1..|V| : \bigwedge_{d_1, d_2 \in D, d_1 \leq d_2} (\neg b_{i,d_1} \vee b_{i,d_2})$$

- díky této podmínce se v kódování nemůže stát, že by hodnota proměnné nebyla menší, nebo rovna nějakému j a zároveň by hodnota proměnné byla menší, nebo rovna nějakému k , kde $j \leq k$, tedy zajišťuje tranzitivitu \leq .

Kódování podmínek

Nechť je doména D uspořádaná podle " \leq ", tedy

$$\forall i, j \in 1..|D|, i \leq j : d_i \leq d_j$$

jinak pracujeme s D' , která obsahuje všechny prvky D a navíc zachovává uspořádání.

Podmínka $v_i \leq v_j$

Pro každou možnou hodnotu proměnné v_i je vytvořena podmínka na hodnotu proměnné v_j , tak aby splňovala " \leq ":

$$\bigwedge_{d \in D} (\neg b_{j,d} \vee b_{i,d})$$

- to odpovídá $\forall d \in D : v_j \leq d \implies v_i \leq d$

Podmínka $v_i - v_j \leq c$

- podmínka je nejdříve převedena do tvaru $v_i \leq v_j + c$ (plus definováno, díky výskytu další podmínky)

$$\forall d_1 \in D : \bigwedge_{d_2 \in D, d_2 + c \leq d_1} (\neg b_{j,d_2} \vee b_{i,d_1})$$

Podmínka $v_i = v_j + c$

$$\forall d_1 \in D : \bigwedge_{d_2 \in D, d_1 \leq d_2 + c, d_2 + c \leq d_1} (\neg b_{j,d_2} \vee b_{i,d_1})$$

Pozn.: pokud neexistuje žádné d_2 splňující podmínku, pak je přidána prázdná klauzule, která značí spor.

Pozn2.: výsledná formule by šla zkrátit pouze na omezování "sousedních" hodnot d_1, d_2 , kde d_2 by byly nejmenší prvky z množiny $\{d | d_1 < d\}$.

Bonus

In the previous example, try to describe an encoding with only a $O(\log_2 |D|)$ propositional variable for each variable v_i .

Celá myšlenka je založena na indexování domény pomocí binární reprezentace indexu.

Každé hodnotě z D je přiřazen unikátní index (počínající od 0). Výsledná formule je potom vytvořena stejně jako v předchozím případě, akorát každá proměnná $b_{i,j}$ je nahrazena klauzulí s $\lceil \log_2 |D| \rceil$ proměnnými tak, že pro hodnotu j je vybrán odpovídající binární zápis B , který lze reprezentovat jako

$b_{\lceil \log_2 |D| \rceil}, b_{\lceil \log_2 |D| \rceil - 1}, \dots, b_1, b_0$, kde $b_i \in \{0, 1\}$, a výsledná kombinace pro $b_{i,j}$ z předchozího řešení vypadá takto:

$$\bigwedge_{k \in 0.. \lceil \log_2 |D| \rceil} \text{bin_repr}(i, j, k)$$

kde $\text{bin_repr}(i, j, k)$ je buď " $\neg b_{i,k}$ " pokud b_k v binárním zápisu j je 0, nebo " $b_{i,k}$ ", pokud je totéž rovno 1.

- Například pro 3. hodnotu z domény $D - d_3$ (počítáno od 1). Je odpovídající index 2, jehož zápis je 0..010, tedy výsledná klauzule odpovídající proměnné b_{i,d_3} z předchozího řešení bude $(\dots \wedge \neg b_{i,2} \wedge b_{i,1} \wedge \neg b_{i,0})$.

Zároveň je třeba zakázat všechny kombinace které neodpovídají validním indexům.

$$\forall v_i \in V, \forall d \in |D| \dots \lceil \log_2 |D| \rceil : \neg \bigwedge_{k \in 0.. \lceil \log_2 |D| \rceil} \text{bin_repr}(i, d, k)$$