

Constraint-Satisfaction-Probleme

B. Nebel, S. Wölfl
R. Mattmüller, M. Westphal
Wintersemester 2009/2010

Universität Freiburg
Institut für Informatik

Übungsblatt 3

Abgabe: 11. November 2009

Aufgabe 3.1 (1+1+1 Punkte)

- (a) Beweisen Sie, dass der Algorithmus AC-3 ein zur Eingabe $\mathcal{C} = \langle V, D, C \rangle$ äquivalentes Netzwerk $\mathcal{C}' = \langle V', D', C' \rangle$ zurückliefert.
- (b) Zeigen Sie, dass der Algorithmus AC-3 ein kantenkonsistentes Netzwerk $\mathcal{C}' = \langle V', D', C' \rangle$ liefert.
- (c) Zeigen Sie, dass der Algorithmus AC-4 die Laufzeit $\Theta(e \cdot k^2)$ hat.

Aufgabe 3.2 (1+1 Punkte)

Betrachten Sie das Constraintnetzwerk $\langle V = \{x_1, x_2, x_3\}, D, C = \{R_{12}, R_{23}\} \rangle$:

$$D_1 = \{1, \dots, 10\}$$

$$D_2 = \{5, \dots, 15\}$$

$$D_3 = \{-10, \dots, 10\}$$

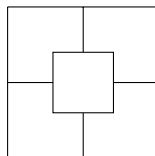
$$R_{12} = \{(x, y) | x + y = 10\}$$

$$R_{23} = \{(x, y) | x + y \leq 3\}$$

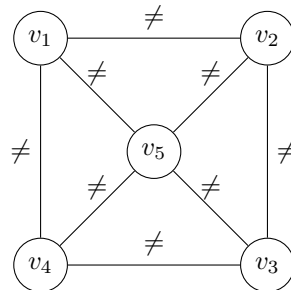
- (a) Verwenden Sie einen der Algorithmen aus der Vorlesung, um ein äquivalentes, kantenkonsistentes Netzwerk zu finden. Geben Sie das resultierende Netzwerk an.
- (b) Verwenden Sie den Algorithmus PC-2 aus der Vorlesung, um zu Ihrem kantenkonsistenten Netzwerk aus Teilaufgabe (a) ein äquivalentes, pfadkonsistentes Netz zu finden. Geben Sie dabei bei jeder Iteration jeweils den Inhalt der Queue und, falls sich ein Constraint geändert hat, auch dessen aktuelle Tupelmeng an.

Aufgabe 3.3 (1+1+1 Punkte)

Betrachten Sie folgendes Färbbarkeitsproblem, bei dem jede der Flächen entweder rot, grün oder blau eingefärbt werden soll, wobei zwei Flächen, die sich an einer Kante berühren, unterschiedliche Farben zugewiesen bekommen sollen.



Dieses Problem lässt sich als Constraintnetz $\mathcal{C} = \langle \{v_1, \dots, v_5\}, \text{dom}, C \rangle$ mit $\text{dom}(v_i) = \{\mathbf{r}, \mathbf{g}, \mathbf{b}\}$ für $1 \leq i \leq 5$ und den folgenden Constraints formalisieren:



- (a) Wieviele Zustände enthält der ungeordnete Suchraum dieses Constraintnetzes? Wieviele Zielzustände gibt es?
- (b) Zeichnen Sie den geordneten Suchraum entlang $\sigma_1 = v_1, v_2, v_3, v_4, v_5$. Da der resultierende Baum sehr groß wird und symmetrisch ist, dürfen Sie die Teilbäume mit $v_1 = \mathbf{b}$ und $v_1 = \mathbf{g}$ nach der ersten Variablenbelegung abbrechen. Kennzeichnen Sie die Sackgassen (dead ends) in Ihrer Zeichnung. Wieviele Zustände hat der *gesamte* geordnete Suchraum (also der Suchraum, der auch die Zustände enthält, die Sie nicht zeichnen mussten)? Wieviele davon sind Sackgassen?
- (c) Zeichnen Sie den geordneten Suchraum entlang $\sigma_2 = v_5, v_1, v_2, v_3, v_4$. Sie können sich wie oben wieder auf einen Teilbaum beschränken. Wieviele Zustände hat der *gesamte* geordnete Suchraum? Wieviele Sackgassen gibt es?