Prof. Dr. M. Grohe E. Fluck, L. Härtel, T. Novotny

Übungsblatt 1 mit Lösungen

Abgabetermin: Montag, der 29. April 2024 um 14:30

Hausaufgabe 3 (3-Colouring) **1+4=5 Punkte**

3-Colouring

Eingabe: Graph $G = (V, E)$ mit Knoten $V = \{0, \ldots n-1\}$

Problem: Gibt es eine Funktion $c\colon V \to [3]$, sodass für alle $uv \in E$ gilt dass $c(u) \neq c(v)$.

Konstruieren Sie zu einem beliebigen aber festen endlichen Graphen G eine Formel φ_C, deren Modelle gereade den korrekten Lösungen von Colouring entsprichen. Gehen Sie dabei wie folgt vor:

a) Definieren Sie eine geeignete Symbolmenge und deren intendierte Semantik.

b) Nennen Sie geeignete Bedingungen und formalisieren Sie diese in der Aussagenlogik. Erklären Sie ihre Formeln kurz und begründen Sie deren Korrektheit in 1-2 Sätzen.

Lösung:

a) Für jeden Knoten $v \in V$ und jede Farbe $c \in [3]$, definieren wir ein Symbol $P_{v,c}$. Dieses steht für die Aussage: “Der Knoten v erhält Farbe c.”

b) Wir definieren einzelne Formeln für die folgenden Bedingungen:

(i) Jeder Knoten $v \in V$ erhält mindestens eine Farbe:

$$\varphi_{v,\min} := P_{v,1} \vee P_{v,2} \vee P_{v,3}.$$

Die oben angegebene Formel ist genau dann erfüllt, wenn mindestens ein Symbole mit 1 interpretiert wird.

(ii) Jeder Knoten $v \in V$ erhält maximal eine Farbe:

$$\varphi_{v,\max} := \neg \left(\bigvee_{\substack{i,j \in [3] \\ i \neq j}} P_{v,i} \wedge P_{v,j} \right).$$

Die oben angegebene Formel ist genau dann erfüllt, wenn für den entsprechende Knoten kein disjunktes Paar an Symbolen mit 1 interpretiert wird.

(iii) Jede Kante $uv \in E$ hat unterschiedlich gefärbte Endpunkte:

$$\varphi_{uv} := \neg \left((P_{v,1} \wedge P_{u,1}) \vee (P_{v,2} \wedge P_{u,2}) \vee (P_{v,3} \wedge P_{u,3}) \right).$$

Die oben angegebene Formel ist genau dann erfüllt, wenn für keine Farbe die Symbole der entsprechenden Knoten beide mit 1 interpretiert werden.

Die finale Formel ist

$$\varphi_C \coloneqq \bigwedge_{v \in V} (\varphi_{v,\min} \wedge \varphi_{v,\max}) \wedge \bigwedge_{uv \in E} \varphi_{uv}.$$

Die Korrektheit der Formeln folgt aus der Korrektheit der einzelnen Teilformeln.

Hausaufgabe 4 (Äquivalenz) 3+3=6 Punkte

Beweisen Sie die folgenden Aussagen. Geben Sie bei jedem Schritt an, welche Regeln Sie verwendet haben (sei es eine Aussage oder Umformungsregel aus der Vorlesung).

a) Für alle Formeln $\varphi, \psi \in \mathrm{AL}$ gilt:

$$\varphi \not\equiv \psi \Leftrightarrow (\varphi \vee \psi) \wedge (\neg\varphi \vee \neg\psi) \text{ ist erfüllbar.}$$

b) Für alle endlichen Formelmengen $\Phi \subseteq \mathrm{AL}$ gilt:

$$\Phi \text{ ist unerfüllbar } \Leftrightarrow \bigvee_{\varphi\in\Phi} \neg\varphi \equiv \bigwedge_{\varphi\in\Phi} (\varphi \to \varphi).$$

Lösung:

a) Aus Beobachtung 1.20 wissen wir dass

$$\varphi \equiv \psi \Leftrightarrow (\varphi \leftrightarrow \psi) \text{ ist allgemeingültig,}$$

also auch

$$\varphi \not\equiv \psi \Leftrightarrow (\varphi \leftrightarrow \psi) \text{ ist nicht allgemeingültig}$$

und mit Benutzung von Doppelter Negation

$$\varphi \not\equiv \psi \Leftrightarrow \neg\neg(\varphi \leftrightarrow \psi) \text{ ist nicht allgemeingültig.}$$

Aus Beobachtung 1.13 folgt dann

$$\varphi \not\equiv \psi \Leftrightarrow \neg\neg(\varphi \leftrightarrow \psi) \text{ ist nicht allgemeingültig} \Leftrightarrow \neg(\varphi \leftrightarrow \psi) \text{ ist nicht unerfüllbar.}$$

Es reicht also aus zu zeigen dass $\neg(\varphi \leftrightarrow \psi) \equiv (\varphi \vee \psi) \wedge (\neg\varphi \vee \neg\psi)$.

$$\begin{aligned}
\neg(\varphi \leftrightarrow \psi) &\overset{\text{Definition}\leftrightarrow}{\equiv} \neg((\varphi \to \psi) \wedge (\psi \to \varphi)) \\
&\overset{\text{Elimination der Implikation}}{\equiv} \neg((\neg\varphi \vee \psi) \wedge (\neg\psi \vee \varphi)) \\
&\overset{\text{De Morgansche Regel}}{\equiv} (\neg(\neg\varphi \vee \psi) \vee \neg(\neg\psi \vee \varphi)) \\
&\overset{\text{De Morgansche Regel}}{\equiv} ((\neg\neg\varphi \wedge \neg\psi) \vee (\neg\neg\psi \wedge \neg\varphi)) \\
&\overset{\text{Doppelte Negation}}{\equiv} ((\varphi \wedge \neg\psi) \vee (\psi \wedge \neg\varphi)) \\
&\overset{\text{Distributivität}}{\equiv} ((\varphi \wedge \neg\psi) \vee \psi) \wedge ((\varphi \wedge \neg\psi) \vee \neg\varphi)) \\
&\overset{\text{Distributivität}}{\equiv} (\varphi \vee \psi) \wedge (\neg\psi \vee \psi) \wedge (\varphi \vee \neg\varphi)) \wedge (\neg\psi \vee \neg\varphi)) \\
&\overset{\text{Tertium Non Datur}}{\equiv} (\varphi \vee \psi) \wedge \top \wedge \top \wedge (\neg\psi \vee \neg\varphi)) \\
&\overset{\text{Regeln für boolsche Konstanten}}{\equiv} (\varphi \vee \psi) \wedge (\neg\psi \vee \neg\varphi))\,.
\end{aligned}$$

Alles in allem haben wir gezeigt

$$\varphi \not\equiv \psi \Leftrightarrow (\varphi \vee \psi) \wedge (\neg\psi \vee \neg\varphi)) \text{ ist erfüllbar.}$$

b) Aus Definition 1.11 wissen wir

$$\Phi \text{ ist unerfüllbar } \Leftrightarrow \bigwedge_{\varphi\in\Phi} \varphi \text{ ist unerfüllbar.}$$

Nach Beobachtung 1.13 gilt

$$\bigwedge_{\varphi\in\Phi} \varphi \text{ ist unerfüllbar} \Leftrightarrow \neg\left(\bigwedge_{\varphi\in\Phi} \varphi\right) \text{ ist allgemeingültig.}$$

Kombiniert man Definition 1.12 mit Definition 1.17 erhält man, dass für alle Formeln $\varphi \in \mathrm{AL}$ gilt

$$\psi \text{ ist allgemeingültig} \Leftrightarrow \psi \equiv \top,$$

also erhalten wir

$$\neg\left(\bigwedge_{\varphi\in\Phi} \varphi\right) \text{ ist allgemeingültig} \Leftrightarrow \neg\left(\bigwedge_{\varphi\in\Phi} \varphi\right) \equiv \top.$$

Unter Anwendung der Regeln für boolsche Konstanten erhalten wir $\top \equiv \bigwedge_{\varphi\in\Phi} \top$ und mit Hilfe von Tertium Non Datur erhalten wir daraus $\top \equiv \bigwedge_{\varphi\in\Phi} (\varphi \vee \neg\varphi)$ und aus Kommutativität und der Elimination der Implikation

$$\top \equiv \bigwedge_{\varphi\in\Phi} (\varphi \to \varphi)$$

Aus den De Morganschen Regeln folgt

$$\neg\left(\bigwedge_{\varphi\in\Phi} \varphi\right) \equiv \bigvee_{\varphi\in\Phi} \neg\varphi.$$

Alles in allem erhalten wir also

$$\Phi \text{ ist unerfüllbar } \Leftrightarrow \bigvee_{\varphi\in\Phi} \neg\varphi \equiv \bigwedge_{\varphi\in\Phi} (\varphi \to \varphi).$$

Hausaufgabe 5 (Funktional vollständige Junktorenmengen) 2+2=4 Punkte

Den 3-stelligen Junktor majority: $\{0,1\}^3 \to \{0,1\}$ definieren wir wie folgt:

$$\text{majority}(x_1, x_2, x_3) = 1 \Leftrightarrow |\{i \in [3] \mid x_i = 1\}| \geq 2,$$

Zeigen oder wiederlegen Sie die folgenden Aussagen.

a) Die Junktorenmenge $\{\text{majority}, \top\}$ ist funktional vollständig.

b) Die Junktorenmenge $\{\text{minority}, \top\}$ ist funktional vollständig.

Hinweis: Sie dürfen ohne Beweis verwenden, dass die Verkettung monotoner boolscher Funktionen wieder monoton ist.

Lösung:

a) Die Aussage ist falsch. Die Junktoren $\neg, \to$ lassen sich nicht ausdrücken.

Wir zeigen die folgende Aussage:

Claim: Die Funktionen $\{\text{majority}, F_\top\}$ sind monoton.

Bevor wir den Claim beweisen, zeigen wir wie daraus die Aufgabe folgt. Die Funktion $F_\neg$ ist nicht monoton, da $F_\neg(0) = 1$ aber $F_\neg(1) = 0$, ebenso die Funktion $F_\to$, da $F_\to(0,0) = 1$, aber $F_\to(1,0) = 0$. Allerdings wissen wir aus dem Hinweis und dem Claim dass alle Funktionen die sich mit Hilfe von $\{\text{majority}, F_\top\}$ darstellen lassen monoton sind.

Beweis des Claim: $F_\top$ ist monoton, weil die Funktion constant ist. majority ist monoton, da, für alle $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^3$, aus $x_i \leq y_i$, für alle $i \in [3]$, auch folgt dass $|\{i \mid x_i = 1\}| \leq |\{i \mid y_i = 1\}|$ und somit $\text{majority}(x_1, x_2, x_3) \leq \text{majority}(y_1, y_2, y_3)$.

b) Die Aussage ist korrekt. Wir zeigen dass man $\{\wedge, \neg\}$ ausdrücken kann.

$\neg$ Für alle aussagenlogischen Formeln $\varphi \in \text{AL}$ gilt, dass $\text{minority}(\varphi, \varphi, \varphi) \equiv \neg\varphi$, da $\text{minority}(0,0,0) = 1$ und $\text{minority}(1,1,1) = 0$.

$\vee$ Für alle aussagenlogischen Formeln $\varphi, \psi \in \text{AL}$ gilt, $\neg\,\text{minority}(\varphi, \psi, \top) \equiv \varphi \vee \psi$, da $\text{minority}(x, y, \top) = 1$ genau dann wenn $x = y = 0$. Da wir bereits gezeigt haben dass sich $\neg$ mit Hilfe von minority ausdrücken lässt, haben wir gezeigt dass sich $\vee$ mit Hilfe von minority und $\top$ ausdrücken lässt.

Da $\{\wedge, \neg\}$ nach Vorlesung funtional vollständig ist, folgt dass $\{\text{minority}, \top\}$ funktional vollständig ist.

Programmieraufgabe 6 (Sudoku Modellierung) 5 Punkte

- Die Abgabe der Programmieraufgabe erfolgt über **Speichern** oder **Abgabe** in VPL. Bis zur Abgabefrist könnt ihr so oft abgeben, wie ihr wollt. Wir bewerten nur die aktuellste Abgabe.
- Ihr könnt in **assignment.py** euren eigenen Code schreiben und dabei die von uns zur Verfügung gestellten Bibliotheken benutzen. Achtet allerdings darauf, keine Dateien zu löschen und die Header der Funktionen unverändert zu lassen.
- Nicht alle Importe sind möglich, manche Bibliotheken werden also einen Fehler wie z.B. `Module assignment tries to import numpy, which does not exist` liefern, wenn ihr versucht diese zu verwenden.
- Wir empfehlen, den Code mindestens einmal zu testen, mit **Ausführen** oder Strg+F11. Dies kann einige Sekunden dauern.
- Punkte und Code sind automatisch mit eurer Abgabegruppe synchronisiert.

In der Vorlesung wurde gezeigt, dass man zu jedem (9×9) Sudoku Rätsel S eine Formelmenge Φ_S konstruieren kann, die genau dann erfüllbar ist wenn S eine Lösung hat.

Alternativ kann man eine Formelmenge Φ'_S erzeugen, die nur noch disjunktive Klauseln enthält, und ebenfalls genau dann erfüllbar ist wenn S eine Lösung hat. So können z. B. $(\neg X \vee Y), \neg X, Y$ in Φ'_S vorkommen, aber $(X \wedge Y)$, $\neg(X \vee Y)$ nicht.

Schreiben Sie die Funktion `generate_sudoku_cnf(sudoku: Sudoku) -> Iterable(Formula)`, welche als Eingabe ein Objekt der Klasse `Sudoku` nimmt und ein `Iterable` Ihrer Wahl, das Objekte der Klasse `Formula` enthält, ausgibt. Die Konjunktion der `Formula`-Objekte soll erfüllbar sein genau dann wenn das Sudoku eine Lösung hat.

Achten Sie darauf, dass wir nur Lösungen auswerten können, deren `Formula`-Objekte ausschließlich disjunktive Klauseln sind.

Lösung:

Wir orientieren uns an der aussagenlogischen Modellierung von Sudoku auf Seite 1.29 und 1.30 der Vorlesung. Durch Umformen der Bedingung “Auf jedem Feld steht höchstens eine Zahl” in $\neg P_{i,j,k} \vee \neg P_{i,j,l}$ sind alle Formeln in konjunktiver Normalform, deren Klauseln sich z.B. mit `for`-Schleifen und `yield` relativ leicht produzieren lassen.