Technische Universität Braunschweig

TCS

Prof. Dr. Roland Meyer, Dr. Jürgen Koslowski

Einführung in die Logik

Aufgabenblatt 0, 2018-04-07

Präsenzaufgabe 1

[Semantik von Formeln]

(a) Sei φ eine Bewertung mit $\varphi(p) = 1$ und $\varphi(q) = \varphi(r) = 0$. Berechnen Sie den Wert

$$\varphi(\neg(p \wedge q) \to r)$$

schrittweise gemäß der Definition.

(b) Beweisen oder widerlegen Sie, dass $q \to (r \to (p \vee q))$ eine Tautologie ist.

(c) Beweisen oder widerlegen Sie, dass $q \to p \models p \to q$ gilt.

(d) Beweisen oder widerlegen Sie, dass $\neg p \vee \neg q \models\!\!\!\models \neg(p \wedge q)$, wobei $A \models\!\!\!\models B$ abkürzend für $A \models B$ und $B \models A$ steht.

Lösungsvorschlag:

Laut VL erfüllt eine *Bewertung* der aussagenlogischen Formeln, also eine (partielle) Funktion $\varphi : \mathcal{F} \longrightarrow \mathbb{B} := \{0, 1\}$, die Bedingungen

$$\begin{aligned}
\varphi(\neg A) &= 1 - \varphi(A) \\
\varphi(A \vee B) &= \max\{\varphi(A), \varphi(B)\} \\
\varphi(A \wedge B) &= \min\{\varphi(A), \varphi(B)\} \\
\varphi(A \to B) &= \begin{cases} 0 & \text{falls } \varphi(A) = 1 \text{ und } \varphi(B) = 0 \\ 1 & \text{sonst} \end{cases} \\
&= \max\{1 - \varphi(A), \varphi(B)\} \\
\varphi(A \leftrightarrow B) &= \begin{cases} 0 & \text{falls } \varphi(A) \neq \varphi(B) \\ 1 & \text{falls } \varphi(A) = \varphi(B) \end{cases} \\
&= \max\{\varphi(A) \cdot \varphi(B), (1 - \varphi(A)) \cdot (1 - \varphi(b))\}
\end{aligned}$$

wobei die rot markierten Ausdrücke zur Vereinfachung der Rechnung dienen.

(a)

$$\underbrace{\underbrace{\neg(\underbrace{p}_{1} \wedge \underbrace{q}_{0})}_{0} \to \underbrace{r}_{0}}_{1}\Bigg\}_{0}$$

oder alternativ (hier kommen die rot markierten Varianten zum Tragen):

$$\begin{aligned}
\varphi(\neg(p \wedge q) \to r) &= \max\{1 - \varphi(\neg(p \wedge q)), \varphi(r)\} \\
&= \max\{1 - (1 - \varphi(p \wedge q)), 0\} \\
&= 1 - (1 - \varphi(p \wedge q)) \\
&= \varphi(p \wedge q) \\
&= \min\{\varphi(p), \varphi(q)\} \\
&= \min\{1, 0\} = 0
\end{aligned}$$

(b) Wir stellen eine Wahrheitstabelle mit zwei Hilfsspalten auf:

p	q	r	$p \vee q$	$r \to (p \vee q)$	$q \to (r \to (p \vee q))$
0	0	0	0	1	1
0	0	1	0	0	1
0	1	0	1	1	1
0	1	1	1	1	1
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	1	1	1
1	1	1	1	1	1

Da die letzte Spalte nur Einsen enthält, handelt es sich bei $q \to (r \to (p \vee q))$ um eine Tautologie.

(c) Unter den Bewertungen φ, die $\varphi(p \to q) = 1$ oder äquivalent $\varphi(p) \leq \varphi(q)$ erfüllen, existiert (mindestens) eine, für die $\varphi(q \to p) = 1$ bzw. $\varphi(q) \leq \varphi(p)$ nicht gilt: nämlich jede Belegung mit $\varphi(p) = 0$ und $\varphi(q) = 1$. Daher ist die Behauptung falsch.

(d) Da der logische Folgerungsbegriff semantisch definiert ist, lohnt auch hier ein Blick auf die Wahheitstabellen:

p	q	$\neg p \vee \neg q$	$\neg(p \wedge q)$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

Die Wahrheitswerte der Formeln $\neg p \vee \neg q$ und $\neg(p \wedge q)$ stimmen für alle möglichen Belegungen überein. Insbesondere folgt aus der Wahrheit von einer der beiden Formeln die der jeweils anderen. Damit ist die Behauptung korrekt.

(Die Behauptung ist übrigens äquivalent dazu, dass es sich bei $(\neg p \vee \neg q) \leftrightarrow (\neg(p \wedge q)$ um eine Tautologie handelt.)

Präsenzaufgabe 2

[Deduktionstheorem]

(a) Das (semantische) *Deduktionstheorem* besagt, dass für jede Formelmenge Γ und Formeln A und B gilt:

$$\Gamma \models A \to B \qquad \text{gdw} \qquad \Gamma \cup \{A\} \models B$$

Statt $\Gamma \cup \{A\}$ schreibt man an dieser Stelle oft vereinfachend Γ, A.

Beweisen Sie das Deduktionstheorem.

(b) Betrachte aussagenlogische Formeln A_i, $i < n$, und B. Zeigen Sie, dass

$$\{ A_i : i < n\} \models B$$

äquivalent ist zu

$$\models A_0 \to (A_1 \to (\dots (A_{n-2} \to (A_{n-1} \to B)) \dots))$$

(c) Zeigen Sie, dass $\Gamma \models B$ äquivalent ist zur Unerfüllbarlkeit von $\Gamma \cup \{\neg B\}$.

Lösungsvorschlag:

(a) Wir betrachten nur Bewertungen, die auf $\Gamma \cup \{A, B\}$ definiert sind.

Speziell für $\Delta \subseteq \Gamma \cup \{A, B\}$ sei $\mathcal{B}(\Delta)$ die Menge der Bewertungen, die Δ erfüllen.

Die Bedingung $\Gamma \models A \to B$ ist zu $\mathcal{B}(\Gamma) \subseteq \mathcal{B}(A \to B)$ äquivalent, während $\Gamma, A \models B$ gleichbedeutend ist mit $\mathcal{B}(\Gamma \cup \{A\}) \subseteq \mathcal{B}(B)$.

Weiterhin ist $\mathcal{B}(\Gamma \cup \{A\})$ eine Teilmenge von $\mathcal{B}(\Gamma)$, da die erste Menge einer potentiell strengeren Bedingung genügt. Aufgrund der Semantik von $\to$ sind weiterhin $\mathcal{B}(\neg A)$ und $\mathcal{B}(B)$ Teilmengen von $\mathcal{B}(A \to B)$.

Falls also $\mathcal{B}(\Gamma) \subseteq \mathcal{B}(A \to B)$, gilt auch $\mathcal{B}(\Gamma \cup \{A\}) \subseteq \mathcal{B}(A, A \to B) = \mathcal{B}(B)$, was zu $\Gamma, A \models B$ äquivalent ist.

Umgekehrt möge $\mathcal{B}(\Gamma \cup \{A\}) \subseteq \mathcal{B}(B)$ gelten, dann also auch $\mathcal{B}(\Gamma \cup \{A\}) \subseteq \mathcal{B}(A \to B)$. Da $\mathcal{B}(\Gamma)$ die disjunkte Vereinigung von $\mathcal{B}(\Gamma \cup \{A\})$ und $\mathcal{B}(\Gamma \cup \{\neg A\})$ ist, brauchen wir nur noch $\mathcal{B}(\Gamma \cup \{\neg A\}) \subseteq \mathcal{B}(A \to B)$ zu überprüfen. Aber aufgrund der Semantik von $\to$ gilt

$$\mathcal{B}(\Gamma \cup \{\neg A\}) \subseteq \mathcal{B}(\neg A) \subseteq \mathcal{B}(A \to B)$$

Insgesamt gilt also $\mathcal{B}(\Gamma) \subseteq \mathcal{B}(A \to B)$.

(b) Klar durch iterative Anwendung des Deduktionstheorems: Setze $\Gamma_j = \{ A_i : i < j \}$, für $j < n+1$. Speziell also $\Gamma_0 = \emptyset$ und $\Gamma_n = \{ A_i : i < n \}$. Nun gilt

$$\begin{aligned} \Gamma_n \models B \quad & \text{gdw} \quad \Gamma_{n-1} \models A_{n-1} \to B \\ & \text{gdw} \quad \Gamma_{n-2} \models A_{n-2} \to (A_{n-1} \to B) \\ & \qquad \vdots \\ & \text{gdw} \quad \Gamma_0 \models A_0 \to (A_1 \to (\dots (A_{n-2} \to (A_{n-1} \to B)) \dots)) \end{aligned}$$

Natürlich ist die Reihenfolge irrelevant, in der die Formeln A_i auf die andere Seite des $\models$-Symbols transportiert werden.

(c) Die Unerfüllbarkeit von $\Gamma \cup \{\neg B\}$ ist äquivalent zu $\mathcal{B}(\Gamma \cup \{\neg B\}) = \emptyset$, und somit zu $\Gamma \cup \{\neg B\} \models \bot$, wobei $\bot$ die konstant falsche Aussage ist. Nach dem Deduktionstheorem ist dies aber äquivalent zu $\Gamma \models \neg B \to \bot$.

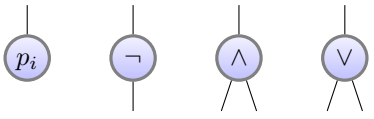
Es genügt also, $B \models\!\!\!\models \neg B \to \bot$ nachzuweisen, was mit einer Wahrheitstabelle trivial ist.

Hausaufgabe 3 [15 PUNKTE]

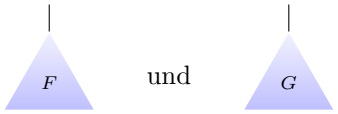
[Strukturelle Rekursion/Induktion]

In der VL wurden aussagenlogische Formeln zunächst als bestimmte 1-dimensionale Wörter (= Tupel) in den atomaren Aussagen p, q, $r \dots$, den Junktoren $\neg$, $\wedge$, $\vee$, $\to$ und $\leftrightarrow$ und den Klammern „(“ sowie „)“ als Hilfssymbolen eingeführt. Die Klammern waren nötig, weil für binäre Junktoren die Infix-Schreibweise $A * B$ verwendet wurde.

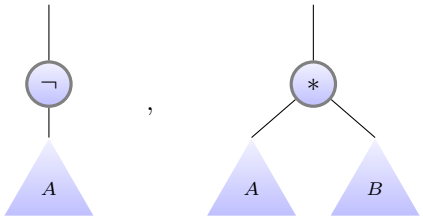
Alternativ kann man Formeln aber auch als 2-dimensionale geordnete markierte Bäume mit Wurzel auffassen, die keine Ausgabe haben: alle Knoten haben eine Eingabekante (oben), zwischen 0 und 2 Ausgabekanten (unten) und tragen Markierungen aus atomaren Aussagen oder Junktoren deren Stelligkeit mit der Anzahl der Ausgaben übereinstimmt.



Sind



Formeln, dann gilt dies auch für



wobei $*$ ein 2-stelliger Junktor ist. Die Baumdarstellung benötigt keine Klammern.

Die *Länge* $|A|$ einer Formel A ist die Anzahl aller Symbole in der obigen 1-dimensionalen Schreibweise einschließlich aller Klammern (keine Vereinfachungen erlaubt); ihre *Größe* $\|A\|$ ist die Anzahl der Knoten der Baumdarstellung, während ihre *Tiefe* gegeben ist durch

- ▷ $t(A) = 0$, falls A atomar ist;
- ▷ $t(\neg B) = t(B) + 1$
- ▷ $t(B * C) = \max\{t(B), t(C)\} + 1$, falls $*$ binär ist.

Beweisen Sie mittels struktureller Induktion über den Aufbau aussagenlogischer Formeln

(a) [5 PUNKTE] In der linearen Tupel-Darstellung jeder korrekten Formel stimmt die Anzahl der öffnenden mit der Anzahl der schließenden Klammern überein.

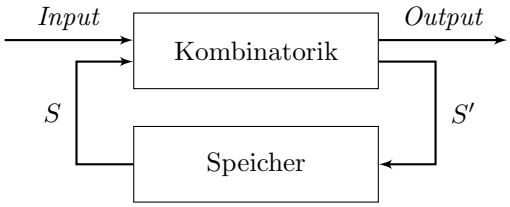
(b) [5 PUNKTE] $|A| \leq 5k + 1$ wobei k die Anzahl der Baumknoten ist, die keine Blätter und daher mit Junktoren markiert sind.

(c) [5 PUNKTE] $|A| \leq 4 \cdot 2^{t(A)} - 3$.

Hausaufgabe 4 [15 PUNKTE]

Iteratives Bounded Model-Checking ist eine Technik zum Auffinden von Fehlern. Sie kann z.B. verwendet werden, um einen getakteten Schaltkreis Sys , daraufhin zu untersuchen, ob er die Eigenschaft $Prop$ hat.

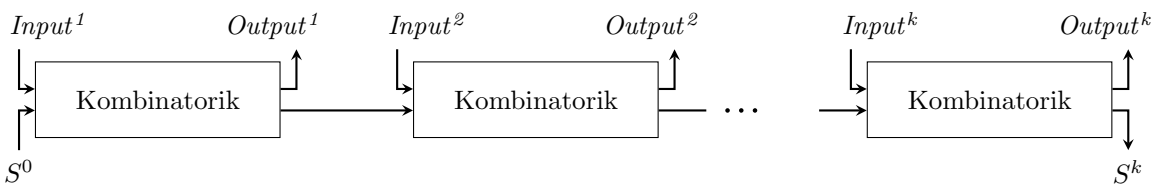
Dabei habe der Schaltkreis Sys folgende allgemeine Form:



Der Algorithmus bildet ausgehend von $k = 1$ iterativ eine logische Formel $BMC(Sys, \neg Prop, k)$ und überprüft diese mechanisch (mit einem SAT-Solver Programm) auf Erfüllbarkeit. Im positiven Fall ist ein Fehler gefunden, andernfalls wird k um 1 erhöht. Wird kein Fehler gefunden, kann $BMC(Sys, \neg Prop, k)$ irgendwann zu groß für den SAT-Solver werden.

Die Funktion BMC operiert in zwei Schritten:

- ▷ Das System Sys wird k-mal entfaltet, dadurch erübrigt sich der Speicher:



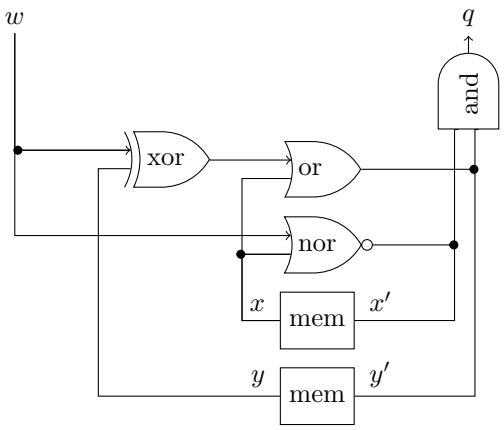
  Hier steht S^0 für den Speicherinhalt zu Beginn, und S^k für den Speicherinhalt nach k Durchläufen.

▷ Das entfaltete System und die Negation der Eigenschaft *Prop* für die aktuelle Entfaltung werden in einer aussagenlogischen Formel $BMC(Sys, \neg Prop, k)$ codiert:

- Jedes Eingabesignal, Ausgabesignal und jeder Speicherwert *pro Takt* entspricht einer aussagenlogischen Variable.
- Der kombinatorische Schaltkreis wird in eine aussagenlogische Formel (in Abhängigkeit der Eingaben) übersetzt.
- Die Negation der Eigenschaft *Prop für die k Entfaltungen* wird per Konjunktion zu obiger Formel hinzugefügt.

Man beachte, dass sich ein Großteil der Formel $BMC(Sys, \neg Prop, k)$, nämlich alles bis auf die Negation von *Prop*, sich in $BMC(Sys, \neg Prop, k+1)$ wiederfindet.

Aufgabe: Betrachte folgendes System *Sys* mit Input w und Output q:



Pro Takt wird der Inhalt x bzw. y der Speicherregister *nach links* weitergeleitet und *von rechts* durch x' bzw. y' ersetzt. Zu Beginn gilt $x = 0$ und $y = 0$.

Die Eigenschaft von Interesse ist, ob q immer den Wert 0 hat. Das kann man zwar durch scharfes Hinschauen direkt überprüfen, aber Ihre Aufgabe ist es, sie mit obigem Verfahren zu prüfen.

Zeichnen Sie hinreichend viele Entfaltungen auf (höchstens vier), und überprüfen Sie jeweils die Formel $BMC(Sys, \neg Prop, k)$ systematisch auf Erfüllbarkeit (Sie sind der SAT-Solver). Sobald $BMC(Sys, \neg Prop, k)$ erfüllbar ist, geben Sie die entsprechende Belegung der Variablen an.

Hinweise: Jede Formel $BMC(Sys, \neg Prop, k)$ muß $\neg x^o \wedge \neg y^0$ enthalten (Anfangsbedingung), zweckmäßigerweise vorne. Die Bedingung für q (hinten) nimmt immer die Form $(q^0 \vee q^1 \vee \cdots \vee q^k)$ an. Dazwischen sind die Werte von x^{i+1}, y^{i+1} und q^i auszudrücken, mit dem Junktor $\leftrightarrow$ und ggf. mit Hilfe der Vorgängerwerte x^i, y^i und w^i.

Abgabe bis Montag, 2018-04-16, 13:15, im Kasten neben IZ 343