SS 2011 25. Mai 2011

Übungen zur Vorlesung Logik
Blatt 6

Prof. Dr. Klaus Madlener Abgabe bis 1. Juni 2011 10:00 Uhr

1. Aufgabe: [Negationsnormalform, Übung]
Bringen Sie die folgenden Formeln in Negationsnormalform:

1. $A_1 \equiv p \wedge ((\neg q \to r) \leftrightarrow (\neg r \vee p))$
2. $A_2 \equiv (p \to (q \to r)) \to ((p \to q) \to (p \to r))$
3. $A_3 \equiv (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee (p_1 \to p_3)$
4. $A_4 \equiv \neg(p_1 \to (\neg p_2 \wedge \neg p_3 \wedge \neg p_5)) \wedge (p_2 \to (p_4 \to p_3)) \wedge \neg((p_2 \wedge p_4) \vee (\neg p_2 \wedge p_5))$

2. Aufgabe: [Davis-Putnam, Übung]
Testen Sie die Formeln aus Aufgabe 1 mit dem Davis-Putnam-Verfahren auf Erfüllbarkeit.

3. Aufgabe: [Duale Formeln, 5P]
Es sei $A \in F(\{\neg, \vee, \wedge\})$ und $d(A)$ die duale Formel von A. Ferner sei φ eine Bewertung und φ' die durch $\varphi'(p) := 1 - \varphi(p)$ für alle $p \in V$ definierte Bewertung. Zeigen Sie $\varphi'(d(A)) = 1 - \varphi(A)$.

4. Aufgabe: [Davis-Putnam, 4P]
Zeigen Sie mit dem Davis-Putnam-Verfahren:

1. $p \wedge q, q \to r \models r$
2. $p \to r, q \to s, p \vee q \models r \vee s$
3. $\neg q, p \to q \models \neg p$
4. $\models \neg(p \to q) \to (q \to p)$

5. Aufgabe: [Pure-Literal-Regel, 8P]

1. Sei A eine Formel in Negationsnormalform, in der p nur positiv vorkommt. Zeigen Sie mit struktureller Induktion:
$$A[p/0] \models A[p/1].$$
2. Folgern Sie daraus, dass A erfüllbarkeitsäquivalent zu $A[p/1]$ ist.
3. Geben Sie eine Formel $A \in F(\{\neg, \wedge, \vee\})$ an, auf die die Regel anwendbar ist und für die $A[p/1]$ bzw. $A[p/0]$ nicht erfüllbarkeitsäquivalent zu A sind.

Abgabe: bis 1. Juni 2011 10:00 Uhr im Kasten neben Raum 34-401.4

zu **Aufgabe 1:**

1. $A_1 \equiv p \wedge ((\neg q \to r) \leftrightarrow (\neg r \vee p))$:

$$\begin{aligned}
& p \wedge ((\neg q \to r) \leftrightarrow (\neg r \vee p)) \\
\models\!\!\!=\!\!\!\mid\; & p \wedge ((q \vee r) \leftrightarrow (\neg r \vee p)) \\
\models\!\!\!=\!\!\!\mid\; & p \wedge (((q \vee r) \wedge (\neg r \vee p)) \vee (\neg(q \vee r) \wedge \neg(\neg r \vee p))) \\
\models\!\!\!=\!\!\!\mid\; & p \wedge (((q \vee r) \wedge (\neg r \vee p)) \vee ((\neg q \wedge \neg r) \wedge (r \wedge \neg p))) \\
\models\!\!\!=\!\!\!\mid\; & p \wedge (((q \vee r) \wedge (\neg r \vee p)) \vee (\neg q \wedge \neg r \wedge r \wedge \neg p))
\end{aligned}$$

2. $A_2 \equiv (p \to (q \to r)) \to ((p \to q) \to (p \to r))$:

$$\begin{aligned}
& (p \to (q \to r)) \to ((p \to q) \to (p \to r)) \\
\models\!\!\!=\!\!\!\mid\; & \neg(\neg p \vee (\neg q \vee r)) \vee (\neg(\neg p \vee q) \vee (\neg p \vee r)) \\
\models\!\!\!=\!\!\!\mid\; & (p \wedge \neg(\neg q \vee r)) \vee ((p \wedge \neg q) \vee (\neg p \vee r)) \\
\models\!\!\!=\!\!\!\mid\; & (p \wedge (q \wedge \neg r)) \vee ((p \wedge \neg q) \vee (\neg p \vee r)) \\
\models\!\!\!=\!\!\!\mid\; & (p \wedge q \wedge \neg r) \vee (p \wedge \neg q) \vee \neg p \vee r
\end{aligned}$$

3. $A_3 \equiv (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee (p_1 \to p_3)$:

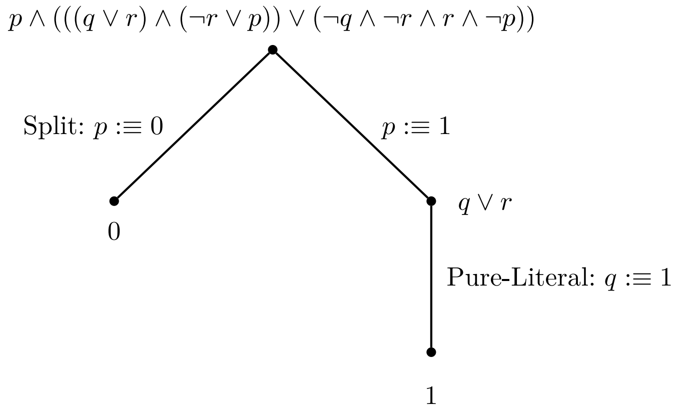
$$\begin{aligned}
& (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee (p_1 \to p_3) \\
\models\!\!\!=\!\!\!\mid\; & (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee (\neg p_1 \vee p_3) \\
\models\!\!\!=\!\!\!\mid\; & (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee \neg p_1 \vee p_3
\end{aligned}$$

4. $\neg(p_1 \wedge \neg p_2 \wedge \neg p_3) \wedge (p_2 \to \neg(p_3 \wedge p_4)) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1$

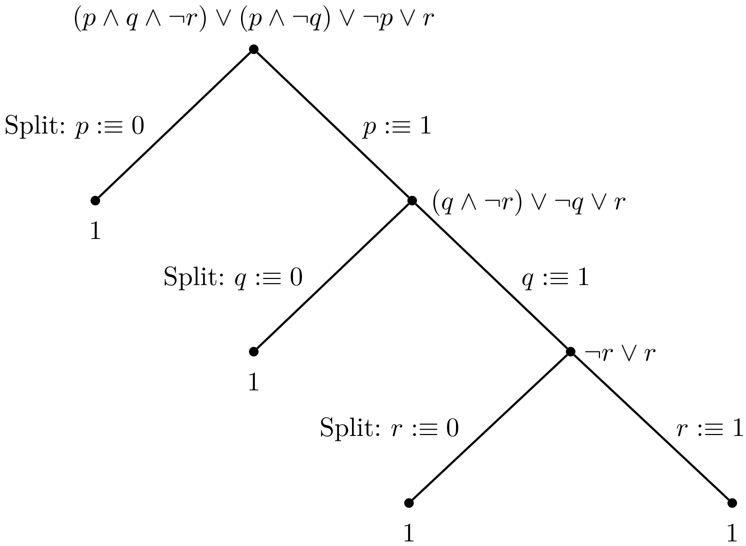
$$\begin{aligned}
& \neg(p_1 \wedge \neg p_2 \wedge \neg p_3) \wedge (p_2 \to \neg(p_3 \wedge p_4)) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1 \\
\models\!\!\!=\!\!\!\mid\; & (\neg p_1 \vee p_2 \vee p_3) \wedge (p_2 \to \neg(p_3 \wedge p_4)) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1 \\
\models\!\!\!=\!\!\!\mid\; & (\neg p_1 \vee p_2 \vee p_3) \wedge (\neg p_2 \vee \neg(p_3 \wedge p_4)) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1 \\
\models\!\!\!=\!\!\!\mid\; & (\neg p_1 \vee p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3 \vee \neg p_4) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1
\end{aligned}$$

zu **Aufgabe 2:**
Wir gehen von den Negationsnormalformen aus der Lösung von Aufgabe 1 aus und konstruieren analog zur Vorlesung (S. 117) einen Baum, der die Formeln beschreibt, die im Laufe des Algorithmus entstehen.

1. $A_1 \equiv p \wedge ((\neg q \to r) \leftrightarrow (\neg r \vee p))$:

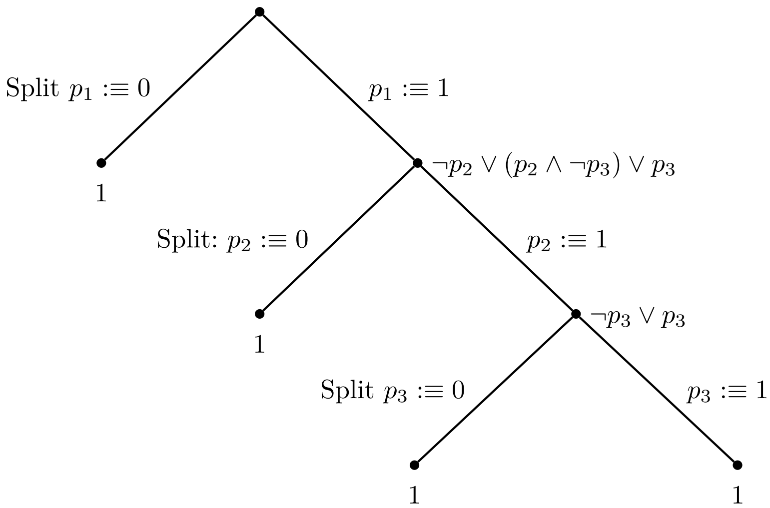


2. $A_2 \equiv (p \to (q \to r)) \to ((p \to q) \to (p \to r))$:



3. $A_3 \equiv (\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee (p_1 \rightarrow p_3)$:

$(\neg p_1 \wedge \neg p_2) \vee (p_1 \wedge \neg p_2) \vee (p_2 \wedge \neg p_3) \vee \neg p_1 \vee p_3$



4. $\neg(p_1 \wedge \neg p_2 \wedge \neg p_3) \wedge (p_2 \to \neg(p_3 \wedge p_4)) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1$:
Diese Formel ist nach der Umformung in Aufgabe 1 nicht nur in Negationsnormalform, sondern sogar in KNF, so dass hier neben der Pure-Literal- und der Splitting-Regel auch noch die Unit- und die Supsumptionsregel angewendet werden kann.
$(\neg p_1 \vee p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3 \vee \neg p_4) \wedge (p_2 \vee p_3 \vee p_4) \wedge p_1$

Unit: $p_1 = 1$

$(p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3 \neg p_4) \wedge (p_2 \vee p_3 \vee p_4)$

Subsumption

$(p_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3 \neg p_4)$

Pure-Literal: $p_4 = 0$

$(p_2 \vee p_3)$

Pure-Literal: $p_2 = 1$

1

Im letzten Schritt könnte man auch die Unit-Regel anwenden, der Algorithmus aus der Vorlesung gibt aber die Reihenfolge der Regeln vor, so dass hier Pure-Literal angewendet werden muss.

zu **Aufgabe 3:**
Der Beweis wird mittels Induktion über den Aufbau von A geführt.

Induktionsanfang: Sei A atomar, also $A \equiv p_i$ für ein $i \in \mathbb{N}$. Nach Definition von φ' und d gilt

$$\varphi'(d(A)) = \varphi'(d(p_i)) = \varphi'(p_i) = 1 - \varphi(p_i) = 1 - \varphi(A).$$

Induktionsschritt: Sei A nicht atomar. Dann hat A die Form $A \equiv \neg B$, $A \equiv B \vee C$ oder $A \equiv B \wedge C$.

Ist $A \equiv \neg B$, so folgt

$$\begin{aligned} \varphi'(d(A)) &= \varphi'(d(\neg B)) = \varphi'(\neg d(B)) = 1 - \varphi'(d(B)) \overset{\text{I.V.}}{=} 1 - (1 - \varphi(B)) \\ &= 1 - \varphi(\neg B) = 1 - \varphi(A). \end{aligned}$$

Ist $A \equiv B \vee C$, so folgt

$$\begin{aligned} \varphi'(d(A)) &= \varphi'(d(B \vee C)) = \varphi'(d(B) \wedge d(C)) = \min\{\varphi'(d(B)), \varphi'(d(C))\} \\ &\overset{\text{I.V.}}{=} \min\{1 - \varphi(B), 1 - \varphi(C)\} = 1 - \max\{\varphi(B), \varphi(C)\} \\ &= 1 - \varphi(B \vee C) = 1 - \varphi(A). \end{aligned}$$

Ist $A \equiv B \wedge C$, so folgt

$$\begin{aligned}\varphi'(d(A)) &= \varphi'(d(B \wedge C)) = \varphi'(d(B) \vee d(C)) = \max\{\varphi'(d(B)), \varphi'(d(C))\} \\ &\overset{\text{I.V.}}{=} \max\{1 - \varphi(B), 1 - \varphi(C)\} = 1 - \min\{\varphi(B), \varphi(C)\} \\ &= 1 - \varphi(B \wedge C) = 1 - \varphi(A).\end{aligned}$$

Damit ist alles gezeigt.

zu **Aufgabe 4:** Um $\Sigma \models A$ zu zeigen, zeigt man, dass $\Sigma \cup \{\neg A\}$ unerfüllbar ist. Dazu bildet man die Konjunktion aller Formeln in $\Sigma \cup \{\neg A\}$, bringt sie in NNF und zeigt, dass es keine erfüllende Belegung gibt. An allen Blättern des Davis-Putnam-Baumes muss also immer 0 stehen.

Man könnte auch auf die Idee kommen, mit dem Deduktionstheorem alle Formeln aus Σ nach rechts zu schieben und zu zeigen, dass an allen Blättern des Baumes 1 steht. Das Problem dabei ist, dass daraus nicht unbedingt folgt, dass die Formel eine Tautologie ist (überlegen Sie sich ein Gegenbeispiel). Man hat dann nur gezeigt, dass sie erfüllbar ist. Diese Methode funktioniert nur, wenn man immer nur die Splitting-Regel benutzt. Das ist aber nicht der Sinn des Verfahrens, denn dies geht nicht schneller, als eine Wertetabelle zu erstellen. Außerdem würde es auch nicht dem Verfahren aus der Vorlesung entsprechen, bei dem die Reihenfolge der Regeln vorgegeben ist.

zu **Aufgabe 5:**

1. **IA:** Formeln in NNF kann man induktiv so definieren, dass die Atome die (positiven oder negativen) Literale sind und die zusammengesetzten Formeln nur noch mit $\wedge$ und $\vee$ gebildet werden. Wir haben also im Induktionsanfang die Fälle $A \equiv p_i$ und $A \equiv \neg p_i$ zu unterscheiden.

   Sei $A \equiv p_i$

   Fall 1: $p_i \equiv p$ Dann ist $A[p/0] = 0$ und $A[p/1] = 1$, also gilt $A[p/0] \models A[p/1]$.

   Fall 2: $p_i \not\equiv p$ Dann ist $A[p/0] = A[p/1]$, also gilt ebenfalls $A[p/0] \models A[p/1]$.

   Sei $A \equiv \neg p_i$ Da p nur positiv in A vorkommt, gilt $p_i \not\equiv p$ und $A[p/0] \models A[p/1]$ wie oben.

   IV: Es gelte $B[p/0] \models B[p/1]$ für alle echten Teilformeln von A.

   IS: $A \equiv B \wedge C$ Nach Induktionsvoraussetzung gilt $B[p/0] \models B[p/1]$ und $C[p/0] \models C[p/1]$. Deshalb erfüllen auch alle Bewertungen, die sowohl $B[p/0]$ als auch $C[p/0]$ erfüllen, sowohl $B[p/1]$ als auch $C[p/1]$. Daher gilt $A[p/0] \equiv B[p/0] \wedge C[p/0] \models B[p/1] \wedge C[p/1] \equiv A[p/1]$.

   $A \equiv B \vee C$ Nach Induktionsvoraussetzung gilt $B[p/0] \models B[p/1]$ und $C[p/0] \models C[p/1]$. Deshalb erfüllen auch alle Bewertungen, die eine der Formeln $B[p/0]$ oder $C[p/0]$ erfüllen, auch eine der Formeln $B[p/1]$ oder $C[p/1]$. Daher gilt $A[p/0] \equiv B[p/0] \vee C[p/0] \models B[p/1] \vee C[p/1] \equiv A[p/1]$.

2. Sei A erfüllbar, also $\varphi(A) = 1$ für eine Bewertung φ. Es kann entweder $\varphi(p) = 0$ oder $\varphi(p) = 1$ sein. Im ersten Fall ist (nach Vorlesung) $\varphi(A) = \varphi(A[p/0])$, im

zweiten Fall $\varphi(A) = \varphi(A[p/1])$. Da aber $A[p/0] \models A[p/1]$ gilt, ist dann in beiden Fällen $\varphi(A[p/1]) = 1$, womit gezeigt ist, dass $A[p/1]$ erfüllbar ist.

Sei nun umgekehrt $A[p/1]$ erfüllbar mit einer Bewertung φ. Wir definieren eine Bewertung $\varphi\prime$, die die Formel A erfüllt. Dazu belegt sie alle Variablen bis auf p genau wie φ, nur p wird auf jeden Fall mit 1 belegt:

$$\varphi\prime(p_i) := \begin{cases} \varphi(p_i) & \text{falls } p_i \neq p \\ 1 & \text{falls } p_i = p \end{cases}$$

Da p in $A[p/1]$ nicht vorkommt und sich ansonsten $\varphi\prime$ genau wie φ verhält, muss $\varphi\prime(A[p/1]) = 1$ gelten. Da $\varphi\prime(p) = 1$ ist, gilt aber (nach Vorlesung) $\varphi\prime(A) = \varphi\prime(A[p/1])$, also $\varphi\prime(A) = 1$.

3. Die Forderung, dass A in Negationsnormalform vorliegt, garantiert, dass A nicht von der Form $\neg(B \wedge C)$ oder $\neg(B \vee C)$ sein kann. Auf solche Formeln könnten nämlich die DeMorgan-Regeln angewendet werden, was dazu führen kann, dass A eine äquivalente Formel in NNF hat, in der p gar nicht nur positiv oder nur negativ vorkommt. Kommen solche Teilformeln vor, so kann man also nicht sicher sein, dass die Pure-Literal-Regel in ihrem eigentlichen Sinne überhaupt anwendbar ist.

   Eine Formel $A \in F(\{\neg, \wedge, \vee\})$, auf die die Regel scheinbar anwendbar ist und für die $A[p/1]$ nicht erfüllbarkeitsäquivalent zu A ist, muss entsprechend obiger Überlegung p nur positiv nthalten und eine Teilformel haben, auf die die DeMorgan-Regeln angewendet werden können, so dass p in einer entsprechenden NNF-Formel eben doch nicht nur positiv vorkommt. Ein Beispiel ist

$$A \equiv \neg(p \vee \neg(p \vee q)).$$

   Diese Formel ist erfüllbar und es gilt

$$A \models\!\!\dashv \neg p \wedge (p \vee q)).$$

   Setzt man jedoch $p = 1$ und vereinfacht die Formel nach den Regeln zur Bildung von $A[p/1]$, so ergibt sich

$$A \rightsquigarrow \neg(1 \vee \neg(1 \vee q)) \rightsquigarrow \neg 1 \rightsquigarrow 0 \equiv A[p/1].$$

   $A[p/1]$ ist also unerfüllbar.