1. Übungsblatt (mit Lösungen)

3.0 VU Formale Modellierung

Marion Brandsteidl, Gernot Salzer

7. November 2013

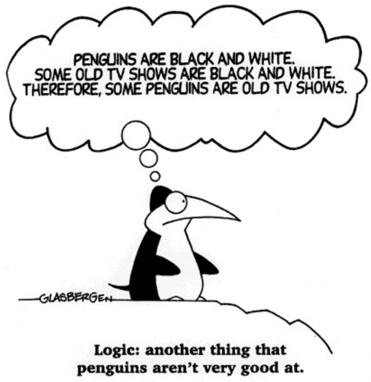
Aufgabe 1 (0.3 Punkte)

Geben Sie für die folgenden Schlussfolgerungen die zugrundeliegende Inferenzregel an und stellen Sie fest, ob diese gültig ist. Wenn ja, geben Sie unter Verwendung von Alltagsbegriffen eine weitere Schlussfolgerung an, die derselben Regel folgt. Wenn nein, modifizieren Sie die Inferenzregel möglichst geringfügig, um eine gültige Regel zu erhalten, und geben Sie dann eine konkrete Schlussfolgerung mit Alltagsbegriffen an, die dieser Regel entspricht.

(a) Kängurus sind Beuteltiere. Kängurus können hüpfen. Daher können alle Beuteltiere hüpfen.

(b) Katzen haben keine Flügel. Ratten sind Katzen. Daher haben Ratten keine Flügel.

(c) Siehe Denkblase rechts.[1]



Lösung

Die Reparatur von falschen Inferenzregeln ist nicht eindeutig. Es kann daher sein, dass Sie eine andere korrekte Lösung gefunden haben.

(a) Alle x sind y.
Alle x können z.
―――――――――
Alle y können z.

Diese Inferenzregel ist nicht gültig. Vertauscht man die zweite Prämisse mit der Konklusion, erhält man eine gültige Inferenzregel:

[1] Geklaut von www.glasbergen.com; Dank an Lara Spendier für den Hinweis.

Alle x sind y.
Alle y können z.
Alle x können z.

Unabhängig davon, ob tatsächlich alle Beuteltiere hüpfen können, folgt aus „Kängurus sind Beuteltiere“ und „Alle Beuteltiere können hüpfen“ zwingend, dass Kängurus hüpfen können. Ein anderes Beispiel, dem dieselbe Inferenzregel zugrunde liegt:

(Alle) Haie sind Fische.
(Alle) Fische können schwimmen.
(Alle) Haie können schwimmen.

(b) Alle x haben keine y.
Alle z sind x.
Alle z haben keine y.

Diese Inferenzregel ist gültig. Ein anderes Beispiel für diese Inferenzregel:

(Alle) Tische haben keine Fenster.
(Alle) Schreibtische sind Tische.
(Alle) Schreibtische haben keine Fenster.

(c) Alle x sind y.
Manche z sind y.
Manche x sind z.

Diese Inferenzregel ist nicht gültig und lässt sich nur mit mehreren Änderungen reparieren, etwa so:

Manche x sind z.
Alle z sind y.
Manche x sind y.

Falls manche Pinguine alte Fernsehshows sind und alle alten Fernsehshows schwarz/weiß sind, dann folgt zwingend, dass manche Pinguine scharz/weiß sind. Ein anderes Beispiel, dem dieselbe Inferenzregel zugrunde liegt:

Manche Wissenschaftler sind Nobelpreisträger.
Alle Nobelpreisträger sind berühmt.
Manche Wissenschaftler sind berühmt.

Aufgabe 2 (0.4 Punkte)

Analysieren Sie die folgenden Sätze und identifizieren Sie ihre logische Struktur sowie die Elementaraussagen.

(a) Ich mag Pizza aber keinen Käse.

(b) Ich fahre nur dann auf Urlaub, wenn ich einen Sommerjob habe.

(c) Das Auto war entweder blau oder anthrazit. (Einfärbig!)

(d) Wenn jemand Langstrecke fliegt, hat er keine Flugangst.

(e) Tom kauft einen Fernseher oder einen Beamer.

(f) Ich gehe schlafen, sobald ich müde bin.

(g) Anna darf nur dann bei der Matura antreten, wenn sie weder in Mathe noch in Deutsch einen Fünfer hat.

(h) Superman und Clark Kent werden nie gleichzeitig gesehen.

Lösung

(a) A ... Ich mag Pizza.
B ... Ich mag Käse.
Struktur: A und nicht B.
Formel: $A \wedge \neg B$

(b) A ... Ich fahre auf Urlaub.
B ... Ich habe einen Sommerjob.
Struktur: A nur dann, wenn B.
Oder: Wenn A, dann B.
Formel: $A \supset B$

(c) A ... Das Auto war blau.
B ... Das Auto war anthrazit.
Struktur: Entweder A oder B. (ausschließendes „oder“)
Formel: $A \not\equiv B$

(d) A ... Jemand fliegt Langstrecke.
B ... Jemand hat Flugangst.
Struktur: Wenn A, dann nicht B.
Formel: $A \supset \neg B$

(e) A ... Tom kauft einen Fernseher.
B ... Tom kauft einen Beamer.
Struktur: A oder B.
Formel: $A \vee B$ oder $A \not\equiv B$

(f) A ... Ich gehe schlafen.
B ... Ich bin müde.
Struktur: A, wenn B.
Formel: $A \subset B$

(g) A ... Anna darf bei der Matura antreten.
B ... Anna hat einen Fünfer in Mathe.
C ... Anna hat einen Fünfer in Deutsch.

Struktur: A nur dann, wenn nicht B und nicht C.
Oder: Wenn A, dann nicht B und nicht C.
Formel: $A \supset (\neg B \wedge \neg C)$ oder $A \supset \neg(B \vee C)$

(h) A ... Superman wird gesehen.
B ... Clark Kent wird gesehen.
Struktur: A nicht gleichzeitig mit B.
Oder: Nicht A oder nicht B.
Formel: $\neg(A \wedge B)$ oder $\neg A \vee \neg B$

Aufgabe 3 (0.4 Punkte)

(a) Zeigen Sie, dass die Menge $\{\mathsf{implies}, \mathsf{false}\}$ vollständig ist für die Klasse der aussagenlogischen Funktionen.

(b) Zeigen Sie, dass die Menge $\{\mathsf{implies}\}$ nicht vollständig ist.

Lösung

(a) Aus der Vorlesung wissen wir, dass die Menge $\{\mathsf{nand}\}$ funktional vollständig ist. Für die funktionale Vollständigkeit von $\{\mathsf{implies}, \mathsf{false}\}$ genügt es daher zu zeigen, dass nand durch die beiden Funktionen implies und false dargestellt werden kann. Um diese Darstellung zu finden, greifen wir auf die in der Vorlesung besprochenen Äquivalenzen zurück:

$$A \uparrow B = \neg A \vee \neg B = A \supset \neg B = A \supset (B \supset \bot)$$

Es gilt also

$$x \mathbin{\mathsf{nand}} y = x \mathbin{\mathsf{implies}} (y \mathbin{\mathsf{implies}} \mathsf{false}) \;,$$

was sich auch unabhängig von den Äquivalenzen durch Überprüfung aller Wahrheitsbelegungen zeigen lässt:

x	y	x nand y	$=$	x implies	(y implies	false)
0	0	**1**	✓	**1**	1	0
0	1	**1**	✓	**1**	0	0
1	0	**1**	✓	**1**	1	0
1	1	**0**	✓	**0**	0	0

Anstelle von $\{\mathsf{nand}\}$ kann jede andere als funktional vollständig bekannte Menge herangezogen werden.

(b) Es genügt von einer einzigen Funktion zu zeigen, dass sie nicht durch implies darstellbar ist. Wir untersuchen, welche einstelligen Funktionen darstellbar sind, indem wir vom einzigen Argument x ausgehend implies mehrfach anwenden. Wir stellen zunächst fest, dass x implies $x = 1$ gilt. Unter Berücksichtigung von 1 als weiterem Argument erhalten wir zusätzlich noch 1 implies $x = x$, x implies $1 = 1$ und 1 implies $1 = 1$. Wir stellen daher folgende Behauptung auf.

Induktionsbehauptung: Jeder Ausdruck bestehend aus implies und x ist äquivalent zu x oder 1.

Der Beweis erfolgt induktiv nach der Anzahl n der Anwendungen der Funktion implies.

Induktionsanfang $n = 0$: Der einzige Ausdruck ohne Anwendung von implies ist x selber, daher gilt unsere Behauptung für $n = 0$.

Induktionshypothese: Unsere Behauptung gelte für alle Ausdrücke mit n oder weniger Anwendungen von implies.

Induktionsschritt: Wir zeigen, dass unter der Annahme, dass die Induktionshypothese zutrifft, unsere Behauptung auch für Ausdrücke mit $n + 1$ Anwendungen von implies gilt. Wir betrachten also einen Ausdruck implies$(f(x), g(x))$, bei dem sowohl f als auch g mit n oder weniger Anwendungen von implies definiert sind. Laut Hypothese ist jeder der beiden Ausdrücke äquivalent zu x oder 1. Wie wir oben aber festgestellt haben, liefert implies mit den Argumenten x bzw. 1 wieder nur x oder 1. Damit gilt die Behauptung auch für den Fall $n + 1$.

Da somit die einzigen darstellbaren einstelligen Funktionen äquivalent zu x bzw. 1 sind, ist z.B. die einstellige Funktion not nicht darstellbar. Die Menge {implies} ist daher nicht funktional vollständig.

Anmerkung für Puristen: Die Argumentation ist ein wenig schlampig, da sie Funktionen und Werte mischt. Genau genommen wollen wir nachweisen, dass ausgehend von der identischen Abbildung id (definiert durch id$(x) = x$ für alle x) durch Anwendung von implies wieder nur id oder die einstellige Funktion one (definiert durch one$(x) = 1$ für alle x) entstehen kann. Mehr dazu unter dem Stichwort „Klon“ (engl. *clone*) im Internet oder in Büchern zur Algebra.

Aufgabe 4 **(0.3 Punkte)**

Sei $\mathcal{M}$ die kleinste Menge mit folgenden Eigenschaften:

(m1) $\{\mathtt{a}, \mathtt{b}\} \subseteq \mathcal{M}$

(m2) Wenn $x \in \mathcal{M}$, dann auch x+$x \in \mathcal{M}$.

(m3) Wenn $x, y \in \mathcal{M}$, dann auch *x*$y \in \mathcal{M}$.

(a) Geben Sie die Mengen $\mathcal{M}_0$, $\mathcal{M}_1$ und $\mathcal{M}_2$ der stufenweise Konstruktion von $\mathcal{M}$ an. Die Menge $\mathcal{M}_2$ enthält bereits mehr als 60 Elemente; es genügt, 10 typische Elemente anzugeben.

(b) Zeigen Sie, dass die Zeichenkette `**a*b+b+**a*b+b*a` in der Menge $\mathcal{M}$ liegt.

(c) Erklären Sie, warum die Zeichenkette `*b*a*b+*b*a*b` nicht in der Menge $\mathcal{M}$ liegen kann.

Lösung

(a) $\mathcal{M}_0 = \{$`a`, `b`$\}$,
$\mathcal{M}_1 = \mathcal{M}_0 \cup \{$`a+a`, `b+b`, `*a*a`, `*a*b`, `*b*a`, `*b*b`$\}$,
$\mathcal{M}_2 = \mathcal{M}_1 \cup \{$`a+a+a+a`, `b+b+b+b`, `*a*a+*a*a`, `*a*b+*a*b`, `*b*a+*b*a`,
`*b*b+*b*b`, `*a*a+a`, `*a*b+b`, `*a**a*a`, `*a**a*b`,
`*a**b*a`, `*a**b*b`, `*b*a+a`, `*b*b+b`, `*b**a*a`,
`*b**a*b`, `*b**b*a`, `*b**b*b`, `*a+a*a`, `*a+a*b`,
`*a+a*a+a`, `*a+a*b+b`, `*a+a**a*a`, `*a+a**a*b`, `*a+a**b*a`,
`*a+a**b*b`, `*b+b*a`, `*b+b*b`, `*b+b*a+a`, `*b+b*b+b`,
`*b+b**a*a`, `*b+b**a*b`, `*b+b**b*a`, `*b+b**b*b`, `**a*a*a`,
`**a*a*b`, `**a*a*a+a`, `**a*a*b+b`, `**a*a**a*a`, `**a*a**a*b`,
`**a*a**b*a`, `**a*a**b*b`, `**a*b*a`, `**a*b*b`, `**a*b*a+a`,
`**a*b*b+b`, `**a*b**a*a`, `**a*b**a*b`, `**a*b**b*a`, `**a*b**b*b`,
`**b*a*a`, `**b*a*b`, `**b*a*a+a`, `**b*a*b+b`, `**b*a**a*a`,
`**b*a**a*b`, `**b*a**b*a`, `**b*a**b*b`, `**b*b*a`, `**b*b*b`,
`**b*b*a+a`, `**b*b*b+b`, `**b*b**a*a`, `**b*b**a*b`, `**b*b**b*a`,
`**b*b**b*b`$\}$.

(b)

i. `b` $\in \mathcal{M}$	Eigenschaft m1
ii. Wegen `b` $\in \mathcal{M}$ (i) gilt `b+b` $\in \mathcal{M}$.	Eigenschaft m2
iii. `a` $\in \mathcal{M}$	Eigenschaft m1
iv. Wegen `a` $\in \mathcal{M}$ (iii) und `b+b` $\in \mathcal{M}$ (ii) gilt `*a*b+b` $\in \mathcal{M}$.	Eigenschaft m3
v. Wegen `*a*b+b` $\in \mathcal{M}$ (iv) gilt `*a*b+b+*a*b+b` $\in \mathcal{M}$.	Eigenschaft m2
vi. Wegen `*a*b+b+*a*b+b` $\in \mathcal{M}$ (v) und `a` $\in \mathcal{M}$ (iii) gilt `**a*b+b+*a*b+b*a` $\in \mathcal{M}$.	Eigenschaft m3

(c) Um zu zeigen, dass ein bestimmtes Wort nicht in der definierten Menge liegt, muss man eine Eigenschaft finden, die alle Wörter in der Menge besitzen, nicht aber das bestimmte Wort. Für die Wahl dieser Eigenschaft gibt es verschiedene Möglichkeiten; hier zwei davon.

Argumentation über die Länge der Wörter. Bei der stufenweisen Konstruktion der gegebenen Menge $\mathcal{M}$ besitzen jene Wörter in $\mathcal{M}_i$, die neu hinzukommen (die also noch nicht in $\mathcal{M}_{i-1}$ vorhanden sind), mindestens die Länge $3i$; das lässt sich mit einem kurzen Induktionsbeweis zeigen. Da das Wort `*b*a*b+*b*a*b` die Länge 13 besitzt, müsste es spätestens ab $i = 4$ in den stufenweise konstruierten Mengen $\mathcal{M}_i$ liegen. Wenn man also $\mathcal{M}_4$ konstruiert und feststellt, dass das gesuchte Wort nicht vorkommt, liegt es nicht in $\mathcal{M}$.

Diese Art von Argumentation lässt sich immer dann verwenden, wenn alle Abschlusseigenschaften der induktiven Definition zu längeren Wörtern führen. Allerdings kann es aufwändig sein, alle Wörter bis zur benötigten Länge tatsächlich zu generieren. Etwa besitzt $\mathcal{M}_4$ in diesem Beispiel bereits 30 830 258 Elemente.

Argumentation über die besondere Form der `+`- und `*`-Wörter. Wir stellen zuerst fest, dass das Symbol `*` in jedem Wort der Menge $\mathcal{M}$ in einer geraden Anzahl vorkommt (lässt sich leicht induktiv zeigen). Das Zeichen `+` kann nur durch die Eigenschaft m2 in ein Wort gelangen; in diesem Fall steht links und rechts davon dasselbe Wort $x \in \mathcal{M}$. Für x gibt es in dieser Aufgabe nur zwei Möglichkeiten, nämlich `*b` oder `*b*a*b`, da alle anderen Zeichenfolgen vor `+` verschieden von den Zeichen danach sind. Jedes dieser beiden Wörter enthält aber eine ungerade Zahl von Sternen, es kann daher nicht aus $\mathcal{M}$ stammen.

Diese Argumentation ist kurz, allerdings benützt sie spezielle Eigenschaft der vorliegenden Menge $\mathcal{M}$ und ist nicht auf andere Fälle übertragbar; dort muss eine andere spezifische Eigenschaft gefunden werden.

Aufgabe 5 (0.3 Punkte)

Sei F die Formel $(((A \land B) \supset (C \lor A)) \land \neg B)$.

(a) Zeigen Sie, dass F syntaktisch korrekt ist.

(b) Werten Sie $\mathsf{val}_I(F)$ für $I(A) = 1$, $I(B) = 1$ und $I(C) = 0$ schrittweise aus.

(c) Verwenden Sie eine Wahrheitstafel um festzustellen, ob die Formel F gültig, erfüllbar, widerlegbar und/oder unerfüllbar ist.

Lösung

(a) Die Menge $\mathcal{A}$ der aussagenlogischen Formeln ist die kleinste Menge, für die gilt:

(a1) $\mathcal{V} = \{A, B, C, \dots\} \subseteq \mathcal{A}$

(a2) $\{\top, \bot\} \subseteq \mathcal{A}$

(a3) $\neg F \in \mathcal{A}$, wenn $F \in \mathcal{A}$.

(a4) $(F * G) \in \mathcal{A}$, wenn $F, G \in \mathcal{A}$ und $* \in \{\land, \uparrow, \lor, \downarrow, \equiv, \not\equiv, \supset, \subset\}$.

Wir zeigen, dass $(((A \land B) \supset (C \lor A)) \land \neg B)$ eine aussagenlogische Formel gemäß dieser Definition ist.

- Die Variablen A, B und C sind Formeln (a1).
- Da A und B Formeln sind, ist auch $(A \land B)$ eine Formel (a4).
- Da C und A Formeln sind, ist auch $(C \lor A)$ eine Formel (a4).
- Da $(A \land B)$ und $(C \lor A)$ Formeln sind, ist auch $((A \land B) \supset (C \lor A))$ eine Formel (a4).
- $\neg B$ ist eine Formel, da B eine Formel ist (a3).

- Da $((A \wedge B) \supset (C \vee A))$ und $\neg B$ Formeln sind, ist auch $(((A \wedge B) \supset (C \vee A)) \wedge \neg B)$ eine Formel (a4).

(b) $\mathsf{val}_I(((A \wedge B) \supset (C \vee A)) \wedge \neg B)$

$$
\begin{aligned}
&= \mathsf{val}_I((A \wedge B) \supset (C \vee A)) \text{ and } \mathsf{val}_I(\neg B) \\
&= (\mathsf{val}_I(A \wedge B) \text{ implies } \mathsf{val}_I(C \vee A)) \text{ and not } \mathsf{val}_I(B) \\
&= ((\mathsf{val}_I(A) \text{ and } \mathsf{val}_I(B)) \text{ implies } (\mathsf{val}_I(C) \text{ or } \mathsf{val}_I(A))) \text{ and not } I(B) \\
&= ((I(A) \text{ and } I(B)) \text{ implies } (I(C) \text{ or } I(A))) \text{ and not } 1) \\
&= ((1 \text{ and } 1) \text{ implies } (0 \text{ or } 1)) \text{ and } 0 \\
&= (1 \text{ implies } 1) \text{ and } 0 \\
&= 1 \text{ and } 0 \\
&= 0
\end{aligned}
$$

(c) Die Formel F ist erfüllbar (und daher nicht unerfüllbar), da es eine Interpretation I gibt, in der sie wahr ist, etwa $I(A) = I(B) = I(C) = 0$. Sie ist auch widerlegbar (und daher nicht gültig), da es eine Interpretation I gibt, in der sie falsch ist, etwa $I(A) = I(B) = I(C) = 1$. Diese Interpretationen lassen sich mittels der Wahrheitstafel systematisch finden:

A	B	C	$((A \wedge B)$	$\supset$	$(C \vee A))$	$\wedge$	$\neg B$
0	0	0	0	1	0	**1**	1
0	0	1	0	1	1	**1**	1
0	1	0	0	1	0	**0**	0
0	1	1	0	1	1	**0**	0
1	0	0	0	1	1	**1**	1
1	0	1	0	1	1	**1**	1
1	1	0	1	1	1	**0**	0
1	1	1	1	1	1	**0**	0

Aufgabe 6 (0.3 Punkte)

Zeigen Sie, dass die beiden Formeln $\neg B \wedge ((C \supset A) \supset (A \wedge \neg C))$ und $\neg(B \vee (C \equiv A))$ äquivalent sind

(a) mithilfe einer Wahrheitstafel;

(b) durch algebraische Umformungen.

Lösung

(a) Wahrheitstafel:

A	B	C	$\neg$	$B \wedge$	$((C \supset A)$	$\supset$	$(A \wedge$	$\neg C))$	$\neg$	$(B \vee$	$(C \equiv A))$
0	0	0	1	**0**	1	0	0	1	**0**	1	1
0	0	1	1	**1**	0	1	0	0	**1**	0	0
0	1	0	0	**0**	1	0	0	1	**0**	1	1
0	1	1	0	**0**	0	1	0	0	**0**	1	0
1	0	0	1	**1**	1	1	1	1	**1**	0	0
1	0	1	1	**0**	1	0	0	0	**0**	1	1
1	1	0	0	**0**	1	1	1	1	**0**	1	0
1	1	1	0	**0**	1	0	0	0	**0**	1	1

Da beide Formeln in sämtlichen Wahrheitsbelegungen denselben Wert liefern, sind sie äquivalent.

(b) Wir zeigen die Äquivalenz, indem wir beide Formeln in dieselbe DNF umformen.

$$
\begin{aligned}
&\neg B \wedge ((C \supset A) \supset (A \wedge \neg C)) && \\
&= \neg B \wedge (\neg(\neg C \vee A) \vee (A \wedge \neg C)) && \text{Ersetzen von } \supset \text{ durch } \neg, \vee \\
&= \neg B \wedge ((C \wedge \neg A) \vee (A \wedge \neg C)) && \text{De Morgan} \\
&= (\neg B \wedge C \wedge \neg A) \vee (\neg B \wedge A \wedge \neg C) && \text{Distributivität}
\end{aligned}
$$

$$
\begin{aligned}
&\neg(B \vee (C \equiv A)) && \\
&= \neg(B \vee ((\neg C \vee A) \wedge (\neg A \vee C))) && \text{Ersetzen von } \equiv \text{ durch } \neg, \vee \\
&= (\neg B \wedge C \wedge \neg A) \vee (\neg B \wedge A \wedge \neg C) && \text{De Morgan}
\end{aligned}
$$

Aufgabe 7 (0.3 Punkte)

Untersuchen Sie, ob die Formel $G := \neg C$ eine logische Konsequenz der drei Formeln $F_1 := A \vee B$, $F_2 := \neg B$ und $F_3 := C \supset \neg A$ ist. Geben Sie eine Formel H an, die genau dann in einer Interpretation I wahr ist, wenn $F_1, F_2, F_3 \models_I G$ gilt.

Lösung

G ist eine Konsequenz der Formeln F_1, F_2 und F_3, wenn die Beziehung $F_1, F_2, F_3 \models_I G$ für alle Interpretationen I gilt.

$I(A)$	$I(B)$	$I(C)$	$A \vee B,$	$\neg B,$	$C \supset \neg A$	$\models_I$	$\neg C$
0	0	0	0	1	1	✓	1
0	0	1	0	1	1	✓	0
0	1	0	1	0	1	✓	1
0	1	1	1	0	1	✓	0
1	0	0	1	1	1	✓	1
1	0	1	1	1	0	✓	0
1	1	0	1	0	1	✓	1
1	1	1	1	0	0	✓	0

Die Formel $\neg C$ ist somit eine logische Konsequenz der Prämissen.
Arbeitsvereinfachung: Ist in einer Interpretation eine der Prämissen falsch oder die Konklusion wahr, müssen die übrigen Formeln nicht mehr ausgewertet werden, da die Beziehung $\models_I$ dann bereits erfüllt ist. Weiters kann man die Erstellung der Tabelle abbrechen, sobald man eine Interpretation I findet, für die $\models_I$ nicht gilt. Wertet man in diesem Beispiel die Formeln von links nach rechts aus, ergibt sich folgende vereinfachte Tabelle:

$I(A)$	$I(B)$	$I(C)$	$A \vee B,$	$\neg B,$	$C \supset \neg A$	$\models_I$	$\neg C$
0	0	0	0			✓	
0	0	1	0			✓	
0	1	0	1	0		✓	
0	1	1	1	0		✓	
1	0	0	1	1	1	✓	1
1	0	1	1	1	0	✓	
1	1	0	1	0		✓	
1	1	1	1	0		✓	

Formel zur Konsequenzbeziehung: $\neg C$ ist genau dann eine logische Konsequenz der drei Formeln $A \vee B$, $\neg B$ und $C \supset \neg A$, wenn die Formel

$$((A \vee B) \wedge \neg B \wedge (C \supset \neg A)) \supset \neg C$$

gültig ist.

Aufgabe 8 (0.3 Punkte)

Sei f folgende dreistellige Funktion.

x	y	z	$f(x,y,z)$
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	0
0	0	0	1

Stellen Sie f durch eine Formel in

(a) disjunktiver

(b) konjunktiver

Normalform dar.

Lösung

(a) $(A_1 \wedge A_2 \wedge A_3) \vee (\neg A_1 \wedge A_2 \wedge A_3) \vee (\neg A_1 \wedge \neg A_2 \wedge \neg A_3)$

(b) $(\neg A_1 \vee \neg A_2 \vee A_3) \wedge (\neg A_1 \vee A_2 \vee \neg A_3) \wedge (\neg A_1 \vee A_2 \vee A_3) \wedge (A_1 \vee \neg A_2 \vee A_3) \wedge (A_1 \vee A_2 \vee \neg A_3)$

Aufgabe 9 (0.4 Punkte)

Sei F die Formel $((C \supset B) \uparrow A) \vee \neg(A \vee (B \equiv A))$.

(a) Bestimmen Sie eine zu F äquivalente Formel in konjunktiver Normalform. Verwenden Sie die semantische Methode.

(b) Bestimmen Sie eine zu F äquivalente Formel in disjunktiver Normalform. Verwenden Sie die algebraische Methode.

Lösung

(a) KNF mittels semantischer Methode:

A	B	C	$((C \supset$	$B) \uparrow$	$A) \lor$	$\neg(A$	$\lor\, (B$	$\equiv A))$
0	0	0	1	1	**1**	0	1	1
0	0	1	0	1	**1**	0	1	1
0	1	0	1	1	**1**	1	0	0
0	1	1	1	1	**1**	1	0	0
1	0	0	1	0	**0**	0	1	0
1	0	1	0	1	**1**	0	1	0
1	1	0	1	0	**0**	0	1	1
1	1	1	1	0	**0**	0	1	1

Aus dieser Tafel lässt sich folgende KNF ablesen:

$$(\neg A \lor B \lor C) \land (\neg A \lor \neg B \lor C) \land (\neg A \lor \neg B \lor \neg C)$$

(b) DNF mittels algebraischer Methode:

$$
\begin{array}{ll}
((C \supset B) \uparrow A) \lor \neg(A \lor (B \equiv A)) & \\
= ((\neg C \lor B) \uparrow A) \lor \neg(A \lor (B \equiv A)) & X \supset Y = \neg X \lor Y \\
= \neg(\neg C \lor B) \lor \neg A \lor \neg(A \lor (B \equiv A)) & X \uparrow Y = \neg X \lor \neg Y \\
= \neg(\neg C \lor B) \lor \neg A \lor \neg(A \lor (B \land A) \lor (\neg B \land \neg A)) & X \equiv Y = (X \land Y) \lor (\neg X \land \neg Y) \\
= \neg(\neg C \lor B) \lor \neg A \lor \neg(A \lor (\neg B \land \neg A)) & \text{Absorption} \\
= (\neg\neg C \land \neg B) \lor \neg A \lor (\neg A \land (\neg\neg B \lor \neg\neg A)) & \text{De Morgan} \\
= (\neg\neg C \land \neg B) \lor \neg A & \text{Absorption} \\
= (C \land \neg B) \lor \neg A & \neg\neg X = X
\end{array}
$$

Aufgabe 10 (0.4 Punkte)

Sechs Mitglieder der berüchtigten Panzerknacker-Bande verbüßen nach einem Raubzug eine Haftstrafe. *Babyface Knack* war nicht untätig und hat unter seinem Bett einen Fluchttunnel gegraben. Nun überlegt er, wen er mitnehmen soll.
„Ich muss mindestens eine Person mitnehmen, die mir bei der Flucht hilft. *Opa Knack* wird bereits nächste Woche entlassen, er wird daher den Ausbruch nicht riskieren. *Megabyte Knack* und *Karlchen Knack* streiten ständig miteinander, ich nehme sicher nicht beide zusammen mit. *Oma Knack* ist nicht gut zu Fuß, daher wird sie auf *Karlchen Knack* als Stütze bestehen, wenn sie mitkommt. *Schlabber Knack* tut nie das Gleiche wie *Karlchen Knack*; wenn Karlchen mitkommt, bleibt er sicher da. *Megabyte Knack* kommt dann und nur dann mit, wenn *Oma Knack* oder *Karlchen Knack* mitkommen.“

(a) Formalisieren Sie die beschriebene Situation inklusive aller Anhaltspunkte mittels aussagenlogischer Formeln. Geben Sie die Bedeutung jeder Aussagenvariablen an.

(b) Wer begleitet Babyface bei seinem Ausbruchsversuch? Begründen Sie Ihre Antwort mit Hilfe Ihrer aussagenlogischen Modellierung.

Lösung

(a) Aussagenvariablen und ihre Bedeutung:

K ... Karlchen Knack kommt mit.
M ... Megabyte Knack kommt mit.
O ... Oma Knack kommt mit.
P ... Opa Knack kommt mit.
S ... Schlabber Knack kommt mit.

Aussagenlogische Formeln:

	$F_0 := K \vee M \vee O \vee P \vee S$	mindestens 1 Komplize
	$F_1 := \neg P$	Opa kommt nicht mit
	$F_2 := \neg(M \wedge K)$	Megabyte und Karlchen nicht gemeinsam
	$F_3 := O \supset K$	wenn Oma, dann Karlchen
	$F_4 := K \supset \neg S$	wenn Karlchen, dann nicht Schlabber
oder	$F_4 := K \not\equiv S$	Karlchen und Schlabber verhalten sich gegenteilig.
	$F_5 := M \equiv (O \vee K)$	Megabyte genau dann, wenn Oma oder Karlchen

(b) Wir suchen alle Wahrheitsbelegungen für die Variablen K, M, O, P und S, in denen die Formeln $F_0, \ldots, F_5$ wahr sind. Wegen Formel F_1 betrachten wir nur Interpretationen, in denen P falsch ist. Für eine vollständige Argumentation muss für jede Interpretation entweder gezeigt werden, dass eine Formel falsch ist, oder aber, dass alle Formeln wahr sind. Wenn wir also in einer Zeile einen 0-Eintrag gefunden haben, müssen wir den Rest der Zeile nicht mehr betrachten. Die folgende Tabelle enthält alle 0-Einträge.

K	M	O	P	S	F_0	$\neg P$	$\neg(M \wedge K)$	$O \supset K$	$K \supset \neg S$	$M \equiv (O \vee K)$	
0	0	0	0	0	0						
0	0	0	0	1	1	1	1	1	1	1	✓
0	0	1	0	0				0		0	
0	0	1	0	1				0		0	
0	1	0	0	0						0	
0	1	0	0	1						0	
0	1	1	0	0				0			
0	1	1	0	1				0			
1	0	0	0	0						0	
1	0	0	0	1					0	0	
1	0	1	0	0						0	
1	0	1	0	1					0	0	
1	1	0	0	0			0				
1	1	0	0	1			0		0		
1	1	1	0	0			0				
1	1	1	0	1			0		0		

Die einzige Interpretation, in der alle Formeln erfüllt sind, ist jene mit $I(K) = I(M) = I(O) = I(P) = 0$ und $I(S) = 1$. Babyface wird daher von Schlabber begleitet.

Aufgabe 11 (0.4 Punkte)

SAT-Solver sind Programme, die aussagenlogische Formeln auf Erfüllbarkeit testen. Typische SAT-Solver erhalten als Eingabe eine Formel in konjunktiver Normalform und liefern die Antwort „erfüllbar“ bzw. „unerfüllbar“. Im ersten Fall wird eine erfüllende Variablenbelegung als Nachweis für die Erfüllbarkeit ausgegeben.
Beispiel: Für die konjunktive Normalform $F = (A \vee \neg B) \wedge (\neg A \vee B)$ liefern SAT-Solver die Antwort „erfüllbar„ und eine der Variablenbelegungen $I_1(A) = I_1(B) = 1$ oder $I_2(A) = I_2(B) = 0$.

(a) Wie lässt sich eine Konsequenzbeziehung $F_1, \ldots, F_n \models G$ mit Hilfe eines derartigen SAT-Solvers überprüfen? Begründen Sie Ihre Antwort.

(b) SAT-Solver liefern in der Regel nur eine einzige erfüllende Variablenbelegung. Was muss man tun, um auch die anderen mit Hilfe des SAT-Solvers berechnen zu können? Beschreiben Sie Ihre Methode und erläutern Sie sie an Hand des oben angeführten Beispiels. Begründen Sie die Korrektheit Ihrer Methode.

Hinweis: Überlegen Sie, wie die Eingabeformel abgeändert werden muss, um andere Variablenbelegungen als die bereits berechnete zu erhalten.

Lösung

(a) Wir wissen aus der Vorlesung:

- $F_1, \dots, F_n \models G$ gilt genau dann, wenn die Formel $(F_1 \wedge \cdots \wedge F_n) \supset G$ gültig ist (Deduktionstheorem).

Die Frage nach der Gültigkeit einer Formel lässt sich durch Negieren der Formel in ein Erfüllbarkeitsproblem umwandeln.

- F ist gültig genau dann, wenn $\neg F$ unerfüllbar ist.

Wir erhalten somit:

- $F_1, \dots, F_n \models G$ gilt genau dann, wenn die Formel $\neg((F_1 \wedge \cdots \wedge F_n) \supset G)$ unerfüllbar ist.

SAT-Solver benötigen als Eingabe Formeln in KNF. Da die negierte Implikation äquivalent zur Formel $F_1 \wedge \cdots \wedge F_n \wedge \neg G$ ist, genügt es, jede der Formeln $F_1, \dots, F_n$ und $\neg G$ gesondert in konjunktive Normalformen umzuwandeln. In Summe erhalten wir folgendes Verfahren:

1. Wandle jede der Formeln $F_1, \dots, F_n$ und $\neg G$ in KNF um; das Ergebnis sei $F'_1, \dots, F'_n$ und G'.
2. Starte den SAT-Solver mit der Eingabe $F'_1 \wedge \cdots \wedge F'_n \wedge G'$. Lautet die Antwort „erfüllbar“, gilt die Konsequenzbeziehung $F_1, \dots, F_n \models G$ nicht. Lautet die Antwort hingegen „unerfüllbar“, dann gilt sie.

(b) Sei F eine erfüllbare Formel mit den Variablen $A_1, \dots, A_n$, und sei I die vom SAT-Solver berechnete erfüllende Variablenbelegung. Diese Variablenbelegung lässt sich durch eine Disjunktion D_I beschreiben, die für jede wahre Variable A (d.h. $I(A) = 1$) das Literal $\neg A$ und für jede falsche Variable A (d.h. $I(A) = 0$) das Literal A enthält:

$$D_I = \bigvee\{\, \neg A_i \mid I(A_i) = 1 \,\} \vee \bigvee\{\, A_i \mid I(A_i) = 0 \,\}$$

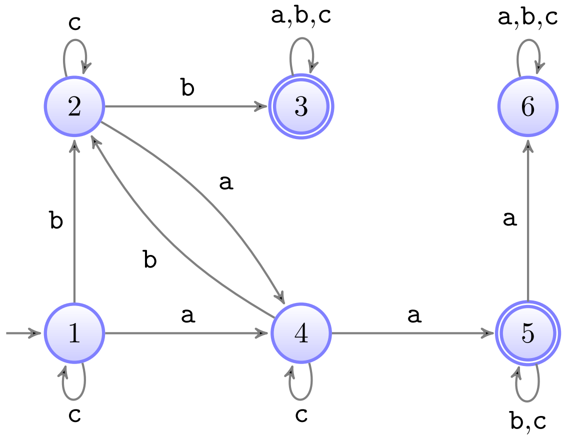
Die Disjunktion D_I hat die Eigenschaft, dass sie in der Interpretation I falsch ist und in allen anderen wahr.

Betrachten wir nun die Formel $F' = F \wedge D_I$. In allen Interpretationen, in denen D_I wahr ist, also in allen außer I, verhält sich F' wie F. In der Interpretation I hingegen liefert F' falsch, da D_I falsch ist. Um also eine weitere erfüllende Interpretation für F zu erhalten, können wir die Formel F' als Eingabe für den SAT-Solver verwenden. War I die letzte erfüllende Interpretation, ist F' unerfüllbar.

Für die Beispielformel $F = (A \vee \neg B) \wedge (\neg A \vee B)$ liefert ein SAT-Solver etwa die Variablenbelegung $I_1(A) = I_1(B) = 1$. Um diese Interpretation beim nächsten Durchlauf nicht mehr zu erhalten, bilden wir die Formel $F_1 = F \wedge D_{I_1}$, wobei D_{I_1} die Disjunktion $\neg A \vee \neg B$ ist. Für diese neue Formel liefert der SAT-Solver die Belegung $I_2(A) = I_2(B) = 0$. Wenn wir die Formel nochmals erweitern zu $F_2 = F_1 \wedge D_{I_2}$ mit $D_{I_2} = A \vee B$, erhalten wir eine unerfüllbare Formel; somit sind I_1 und I_2 die einzigen erfüllenden Interpretationen.

Aufgabe 12 (0.4 Punkte)

Sei $\mathcal{A}$ der folgende endliche Automat.



(a) Geben Sie 5 Wörter an, die von $\mathcal{A}$ akzeptiert werden.

(b) Geben Sie an, welche der folgenden Wörter der Automat akzeptiert: ε, `bba`, `abba`, `aaa`, `aabc`.

(c) Berechnen Sie schrittweise $\delta^*(1, \texttt{babcb})$.

(d) Beschreiben Sie $\mathcal{L}(\mathcal{A})$, die von $\mathcal{A}$ akzeptierte Sprache.

(e) Spezifizieren Sie $\mathcal{A}$ in tabellarischer Form. Handelt es sich bei $\mathcal{A}$ um einen deterministischen oder indeterministischen Automaten?

Lösung

(a) $\mathcal{A}$ akzeptiert zum Beispiel `bb`, `baca`, `accca`, `abb` und `bcbc`.

(b) $\mathcal{A}$ akzeptiert `bba`, `abba` und `aabc`, nicht aber ε und `aaa`.

(c)
$$
\begin{aligned}
\delta^*(1, \texttt{babcb}) &= \delta^*(\delta(1, \texttt{b}), \texttt{abcb}) \\
&= \delta^*(2, \texttt{abcb}) \\
&= \delta^*(\delta(2, \texttt{a}), \texttt{bcb}) \\
&= \delta^*(4, \texttt{bcb}) \\
&= \delta^*(\delta(4, \texttt{b}), \texttt{cb}) \\
&= \delta^*(2, \texttt{cb}) \\
&= \delta^*(\delta(2, \texttt{c}), \texttt{b}) \\
&= \delta^*(2, \texttt{b}) \\
&= \delta^*(\delta(2, \texttt{b}), \varepsilon) \\
&= \delta^*(3, \varepsilon) \\
&= 3
\end{aligned}
$$

(d) Wir betrachten zunächst den Automaten $\mathcal{A}'$, der aus $\mathcal{A}$ durch Streichen aller `c`-Übergänge entsteht. Die Wörter in $\mathcal{L}(\mathcal{A}')$ lassen sich folgendermaßen beschreiben: Ist der erste Doppelbuchstabe im Wort `aa`, dann dürfen nur noch `b`'s folgen; ist der ersten Doppelbuchstabe hingegen `bb`, dann dürfen danach `a`'s und `b`'s in beliebiger Reihenfolge auftreten. Die Sprache $\mathcal{L}(\mathcal{A})$ des ursprünglichen Automaten erhält man aus den Wörtern in $\mathcal{L}(\mathcal{A}')$, indem man an beliebigen Stellen beliebig viele `c`'s einfügt.

(e) $\mathcal{A} = \langle\{1,2,3,4,5,6\}, \{\mathtt{a},\mathtt{b},\mathtt{c}\}, \delta, 1, \{3,5\}\rangle$, wobei die Übergangsfunktion δ durch folgende Tabelle definiert wird:

δ	`a`	`b`	`c`
1	4	2	1
2	4	3	2
3	3	3	3
4	5	2	4
5	6	5	5
6	6	6	6

$\mathcal{A}$ ist ein deterministischer Automat, da der momentane Zustand und die nächste Eingabe immer eindeutig den Folgezustand bestimmen. Das äußert sich in der Tabelle dadurch, dass jeder Eintrag genau einen Zustand enthält.

Aufgabe 13 (0.4 Punkte)

Ein elektronisches Tresorschloss besteht aus einem zweistelligen Display sowie den Tasten `+`, `L`, `R`, `Ok` und `Reset`. Jede Stelle des Displays kann eine der drei Ziffern 0, 1 oder 2 anzeigen. Mit jedem Drücken der `+`-Taste ändert sich die Anzeige der aktiven Stelle von 0 auf 1, von 1 auf 2 bzw. von 2 auf 0. Welche der beiden Stellen aktiv ist, lässt sich durch die `L`- und `R`-Taste kontrollieren: Ein- oder mehrmaliges Drücken der `L`- bzw. `R`-Taste aktiviert die linke bzw. rechte Stelle. Im Anfangszustand zeigt das Display die Zahl 00 an und die linke Stelle ist aktiviert. Wird die Zahl 21 eingestellt und anschließend die `Ok`-Taste gedrückt, öffnet das Schloss; bei jeder anderen Zahl geht das Schloss in einen Fehlerzustand. Sowohl im geöffneten Zustand als auch im Fehlerzustand werden alle weiteren Tasten ausgenommen `Reset` ignoriert, d.h., sie beeinflussen den Zustand des Schlosses nicht. Wird zu einem beliebigen Zeitpunkt die `Reset`-Taste gedrückt, geht das Schloss wieder in den Anfangszustand über. Das Schloss lässt sich also z.B. mit folgenden Tastenfolgen öffnen:

`+` – `+` – `R` – `+` – `Ok`
`+` – `Reset` – `+` – `L` – `R` – `+` – `L` – `+` – `Ok` – `Ok`

- Überlegen Sie, welche Informationen notwendig sind, um den Zustand des Schlosses zu beschreiben. Wieviele Zustände kann das Schloss annehmen? Wieviele Zustände sind es im Allgemeinen, wenn das Schloss n Ziffern (statt 3) pro Stelle sowie k Stellen (statt 2) besitzt?

- Legen Sie die möglichen Aktionen fest, die zu einem Zustandswechsel führen.
- Geben Sie einen endlichen Automaten an, der das Verhalten des beschriebenen Schlosses vollständig beschreibt. Der Endzustand ist erreicht, wenn das Schloss öffnet. Spezifizieren Sie die Übergangsfunktion des Automaten mittels einer Tabelle.

Lösung

Der Zustand des Schlosses wird durch die angezeigten Ziffern sowie durch die Position der Aktivierung eindeutig festgelegt. Daher besitzt das Schloss $3^2 \cdot 2 + 2 = 20$ Zustände, im Allgemeinen sind es $n^k \cdot k + 2$ Zustände. (Die beiden Extrazustände sind der Fehlerzustand und das geöffnete Schloss.) Die Aktionen, die zu Zustandswechseln führen (können), sind die möglichen Tastendrücke, also `+`, `L`, `R`, `Ok` und `Reset`. Als Zustandsbezeichnung wählen wir $\underline{a}b$ bzw. $a\underline{b}$, wobei ab den angezeigten Ziffern entspricht und die Unterstreichung die aktivierte Stelle markiert.
Das Verhalten des Schlosses wird durch den Automaten

$$\langle\{\underline{0}0,\dots,2\underline{2},\mathit{Fehler},\mathit{Offen}\},\{\texttt{+},\texttt{L},\texttt{R},\texttt{Ok},\texttt{Reset}\},\delta,\underline{0}0,\{\mathit{Offen}\}\rangle$$

beschrieben, wobei die Übergangsfunktion δ durch folgende Tabelle definiert wird.

δ	`+`	`L`	`R`	`Ok`	`Reset`
$\underline{0}0$	$\underline{1}0$	$\underline{0}0$	$0\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{1}0$	$\underline{2}0$	$\underline{1}0$	$1\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{2}0$	$\underline{0}0$	$\underline{2}0$	$2\underline{0}$	*Fehler*	$\underline{0}0$
$\underline{0}1$	$\underline{1}1$	$\underline{0}1$	$0\underline{1}$	*Fehler*	$\underline{0}0$
$\underline{1}1$	$\underline{2}1$	$\underline{1}1$	$1\underline{1}$	*Fehler*	$\underline{0}0$
$\underline{2}1$	$\underline{0}1$	$\underline{2}1$	$2\underline{1}$	*Offen*	$\underline{0}0$
$\underline{0}2$	$\underline{1}2$	$\underline{0}2$	$0\underline{2}$	*Fehler*	$\underline{0}0$
$\underline{1}2$	$\underline{2}2$	$\underline{1}2$	$1\underline{2}$	*Fehler*	$\underline{0}0$
$\underline{2}2$	$\underline{0}2$	$\underline{2}2$	$2\underline{2}$	*Fehler*	$\underline{0}0$
$0\underline{0}$	$0\underline{1}$	$\underline{0}0$	$0\underline{0}$	*Fehler*	$\underline{0}0$
$1\underline{0}$	$1\underline{1}$	$\underline{1}0$	$1\underline{0}$	*Fehler*	$\underline{0}0$
$2\underline{0}$	$2\underline{1}$	$\underline{2}0$	$2\underline{0}$	*Fehler*	$\underline{0}0$
$0\underline{1}$	$0\underline{2}$	$\underline{0}1$	$0\underline{1}$	*Fehler*	$\underline{0}0$
$1\underline{1}$	$1\underline{2}$	$\underline{1}1$	$1\underline{1}$	*Fehler*	$\underline{0}0$
$2\underline{1}$	$2\underline{2}$	$\underline{2}1$	$2\underline{1}$	*Offen*	$\underline{0}0$
$0\underline{2}$	$0\underline{0}$	$\underline{0}2$	$0\underline{2}$	*Fehler*	$\underline{0}0$
$1\underline{2}$	$1\underline{0}$	$\underline{1}2$	$1\underline{2}$	*Fehler*	$\underline{0}0$
$2\underline{2}$	$2\underline{0}$	$\underline{2}2$	$2\underline{2}$	*Fehler*	$\underline{0}0$
Fehler	*Fehler*	*Fehler*	*Fehler*	*Fehler*	$\underline{0}0$
Offen	*Offen*	*Offen*	*Offen*	*Offen*	$\underline{0}0$

Aufgabe 14 **(0.4 Punkte)**

Es stehen drei Wasserkrüge mit einem Fassungsvermögen von drei, fünf bzw. sieben Litern zur Verfügung, von denen zu Beginn der kleinste und der größte Krug vollständig gefüllt sind und der mittlere leer ist. Die Krüge sollen nun ohne Wasserverlust so umgefüllt werden, dass sich am Ende in einem der Krüge genau ein Liter befindet.
Modellieren Sie das Rätsel mit Hife eines endlichen Automaten. Beginnen Sie mit folgenden Fragen.

- Wodurch werden die Zustände des Systems charakterisiert, wie lassen sie sich eindeutig beschreiben? Was ist der Startzustand, was sind die Endzustände? Schätzen Sie die Zahl der benötigten Zustände möglichst genau ab.

  Hinweis: Durch das Umfüllen geht kein Wasser verloren. Weiters ist nach jedem Umfüllvorgang einer der beteiligten Krüge leer oder voll, da sich andernfalls die Wassermengen in den Krügen nicht exakt bestimmt wären.

- Wie lassen sich die Zustandsübergänge in diesem System beschreiben? Legen Sie das Alphabet des Automaten fest. Welche Bedeutung besitzt die zum Automaten gehörige Sprache, d.h., was geben die Wörter in dieser Sprache an?

  Wählen Sie das Alphabet möglichst klein, aber groß genug, sodass sich aus den Wörtern über diesem Alphabet die Abläufe im System rekonstruieren lassen. Sie sollten also weder jedem Übergang zwischen zwei Zuständen ein eigenes Symbol zuordnen (das Symbol repräsentiert dann nur genau diesen einen Übergang) noch sollten Sie alle Übergänge mit ein und demselben Symbol beschriften, da sich damit die Abläufe im System nicht beschreiben lassen.

Lösung

Das System besteht aus drei Krügen. Der momentane Zustand wird durch ihren Füllstand beschrieben, also durch ein Zahlentripel ijk, wobei $i \in \{0, \ldots, 3\}$, $j \in \{0, \ldots, 5\}$ und $k \in \{0, \ldots, 7\}$ gilt. Als obere Schranke gibt es daher maximal $4 \times 6 \times 8 = 192$ Zustände. Da kein Wasser verloren geht, muss zusätzlich $i + j + k = 10$ gelten, womit nur noch 18 Zustände in Frage kommen: 037, 046, 055, 127, 136, 145, 154, 217, 226, 235, 244, 253, 307, 316, 325, 334, 343 und 352. Da wegen der exakten Messung nach jedem Umfüllvorgang einer der Krüge voll oder leer sein muss, fallen die Zustände 136, 145, 226, 235 und 244 weg, der Automat besteht also aus höchstens 13 Zuständen.
Startzustand ist laut Angabe 307 (der erste und dritte Krug sind vollständig gefüllt, der zweite ist leer). Endzustände sind alle Zustände, die die Ziffer 1 enthalten, also 127, 136, 145, 154, 217 und 316.
Die Zustände ändern sich durch Umfüllvorgänge. Es gibt dafür sechs Möglichkeiten, je nachdem, von welchem Krug in welchen umgefüllt wird. Für die Umfüllvorgänge wählen wir die Bezeichnungen $\genfrac{}{}{0pt}{}{3}{5}$, $\genfrac{}{}{0pt}{}{3}{7}$, $\genfrac{}{}{0pt}{}{5}{3}$, $\genfrac{}{}{0pt}{}{5}{7}$, $\genfrac{}{}{0pt}{}{7}{3}$ und $\genfrac{}{}{0pt}{}{7}{5}$ mit der Bedeutung, dass vom oberen in den unteren Krug umgefüllt wird; das Symbol $\genfrac{}{}{0pt}{}{5}{3}$ steht zum Beispiel für das Umfüllen vom 5- in den 3-Liter-Krug. Diese sechs Symbole bilden das Alphabet der vom Automaten akzeptierten

Sprache. Jedes Wort über dem Alphabet beschreibt eine Folge von Umfüllvorgängen. Die vom Automaten akzeptierten Worte sind genau jene Folgen, die die Aufgabe lösen, die also von der Anfangsbefüllung in einen Zustand führen, in dem in einem Krug genau ein Liter Wasser ist. Das kürzeste derartige Wort ist $\substack{3\\5}\substack{7\\3}\substack{3\\5}$, das vom Startzustand 307 in den Endzustand 154 führt.
Den Automaten selber erhält man, indem man vom Anfangszustand ausgehend systematisch alle möglichen Umfüllungvorgänge betrachten. Wir sehen, dass von den 13 in Frage kommenden Zuständen nur 8 benötigt werden.