Der Gültigkeitsbegriff

Sei spec $=$ (sig, $\mathcal{A}x$) eine Spezifikation und φ eine abgeschlossene Formel über sig.

Bezeichnung

φ **gilt** in spec, falls φ in allen Modellen von spec gilt.
Also: φ gilt in spec $\;\mathrm{gdw}\;\; \mathcal{A}x \models \varphi$
$\qquad\qquad\qquad\qquad\;\;\mathrm{gdw}\;\; \mathcal{A} \models \varphi$ für alle $\mathcal{A} \in$ Mod(spec).

Bemerkung

Es gibt verschiedene sinnvolle Gültigkeitsbegriffe zu einer Spezifikation. Wir wählen für diese Vorlesung den obigen Begriff. Er heißt auch „Gültigkeit in PL1".

Beachte

Sei $\mathcal{A}x = \{A_1, \ldots, A_n\}$, $A_i, \varphi \in$ Form(sig).

Es gilt $\mathcal{A}x \models \varphi\;\; \mathrm{gdw}\;\; A \equiv A_1 \wedge \cdots \wedge A_n \wedge \neg\varphi$ ist unerfüllbar.

Nächstes Ziel

Transformation von A in Klausel-Normalform.

2.2 Transformation in Klausel-Normalform

Bezeichnung

Eine Formel φ ist in **Klausel-Normalform (CNF)**, falls sie die Form hat

$$\varphi \equiv \forall^*(C_1 \wedge \cdots \wedge C_n)$$

wobei die C_i Klauseln sind.

Ziel (*)

Transformation von φ in φ' mit

(1) φ' ist in CNF.

(2) φ ist erfüllbar gdw φ' ist erfüllbar.

Dann heißt φ' eine **CNF für** φ.

Beachte

Ist φ' in CNF, so enthält φ' nicht den Quantor $\exists$ und die C_i sind ganz Quantoren-frei.

Zur Erfüllbarkeit (1)

Sei sig $= (F, P, \alpha)$ eine Signatur,
spec $=$ (sig, $\mathcal{A}x$) eine Spezifikation,
$\mathcal{A}x = \{\varphi_1, \ldots, \varphi_n\}$,
φ eine abgeschlossene Formel über sig.

Ziel des Theorembeweisens

Gegeben: $\mathcal{A}x, \varphi$ über sig.

Zeige: $\mathcal{A}x \models \varphi$.

Beachte

$\mathcal{A}x \models \varphi \text{ gdw } \varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\varphi$ ist unerfüllbar.

Also

Ziel (*) ist sinnvoll.

Zur Erfüllbarkeit (2)

Definition 2.6.

Sei $T : \varphi \mapsto \psi$ eine Transformation von Formeln.

T **erhält die Erfüllbarkeit**, falls gilt

$$\varphi \text{ ist erfüllbar gdw } \psi \text{ ist erfüllbar.}$$

T **erhält die Äquivalenz**, falls für jede sig-Algebra $\mathcal{A}$ gilt:

$$\mathcal{A} \models \varphi \text{ gdw } \mathcal{A} \models \psi$$

Beachte

Erhält T die Äquivalenz, so transformiert T sig-Formeln in sig-Formeln.

Die „Skolemisierung“ führt neue Funktionssymbole ein, erweitert also die Signatur. Sie erhält nur die Erfüllbarkeit, nicht die Äquivalenz.

Ziel

Angabe von T mit Transformationsregeln, so dass $T(\varphi)$ eine CNF für φ ist.

Intention

Die meisten Beweissysteme arbeiten auf Formeln in Klausel-Normalform viel effizienter als auf Formeln in beliebiger Form.

Simplifikationsregeln (1)

Bezeichnung

Einführung neuer Symbole $\top$ und $\bot$ (top und bottom).
$\top$ steht für true, $\bot$ steht für false.

Simplifikationsregeln

$\varphi \wedge \varphi$	$\to \varphi$	$\varphi \vee \varphi$	$\to \varphi$
$\varphi \wedge \neg\varphi$	$\to \bot$	$\varphi \vee \neg\varphi$	$\to \top$
$\varphi \Leftrightarrow \varphi$	$\to \top$	$\varphi \Rightarrow \varphi$	$\to \top$
$\neg\top$	$\to \bot$	$\neg\bot$	$\to \top$
$\varphi \wedge \top$	$\to \varphi$	$\varphi \vee \top$	$\to \top$
$\varphi \wedge \bot$	$\to \bot$	$\varphi \vee \bot$	$\to \varphi$
$\varphi \Rightarrow \top$	$\to \top$	$\top \Rightarrow \varphi$	$\to \varphi$
$\varphi \Rightarrow \bot$	$\to \neg\varphi$	$\bot \Rightarrow \varphi$	$\to \top$
$\varphi \Leftrightarrow \top$	$\to \varphi$	$\varphi \Leftrightarrow \bot$	$\to \neg\varphi$
$\forall x.\varphi$	$\to \varphi$	falls $x \notin$ free(φ)	
$\exists x.\varphi$	$\to \varphi$	falls $x \notin$ free(φ)	

Beachte hier und in den folgenden Regeln:

$\wedge$ und $\vee$ sind kommutativ und assoziativ. Also gilt z. B. auch

$$\neg\varphi \wedge \varphi \to \bot \qquad \neg\varphi \vee \varphi \to \top$$

Simplifikationsregeln (2)

Lemma 2.2

Die Anwendung der Regeln terminiert.

Alle Regeln sind Äquivalenz-erhaltend.

Ist φ irreduzibel, so gilt $\varphi \in \{\bot, \top\}$ oder φ enthält $\bot, \top$ nicht.

Beachte

Sei $\varphi \xrightarrow{*} \psi$ mit den Simplifikationsregeln und ψ irreduzibel.

Ist $\psi \in \{\bot, \top\}$ so stoppt die CNF-Transformation.

Ab jetzt: Sei ψ $\{\bot, \top\}$-frei.

Negations-Normal-Form

Bezeichnung

φ ist in **Negations-Normal-Form (NNF)**, falls

(1) φ enthält $\Rightarrow, \Leftrightarrow$ nicht.

(2) φ enthält $\neg$ nur direkt vor Atomen.

NNF-Regeln

$$
\begin{array}{llll}
\neg(\varphi \wedge \psi) & \to \neg\varphi \vee \neg\psi & \neg(\varphi \vee \psi) & \to \neg\varphi \wedge \neg\psi \\
\neg\forall x.\varphi & \to \exists x.\neg\varphi & \neg\exists x.\varphi & \to \forall x.\neg\varphi \\
\varphi \Rightarrow \psi & \to \neg\varphi \vee \psi & \neg\neg\varphi & \to \varphi \\
\varphi \Leftrightarrow \psi & \to (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi) & &
\end{array}
$$

Lemma 2.3

Das bisherige Regelsystem ist terminierend. Alle Regeln sind Äquivalenz-erhaltend. Gilt $\varphi \xrightarrow{*} \psi$ und ist ψ irreduzibel, so ist ψ in NNF.

Miniscoping

Ziel

Die Quantoren möglichst weit nach innen ziehen.

Miniscoping-Regeln

$$
\begin{array}{lll}
\exists x.(\varphi \wedge \psi) & \to \exists x.\varphi \wedge \psi & \text{falls } x \notin \mathsf{free}(\psi) \\
\exists x.(\varphi \vee \psi) & \to \exists x.\varphi \vee \psi & \text{falls } x \notin \mathsf{free}(\psi) \\
\forall x.(\varphi \wedge \psi) & \to \forall x.\varphi \wedge \psi & \text{falls } x \notin \mathsf{free}(\psi) \\
\forall x.(\varphi \vee \psi) & \to \forall x.\varphi \vee \psi & \text{falls } x \notin \mathsf{free}(\psi) \\
\forall x.(\varphi \wedge \psi) & \to \forall x.\varphi \wedge \forall x.\psi & \text{falls } x \in \mathsf{free}(\varphi) \text{ und } x \in \mathsf{free}(\psi) \\
\exists x.(\varphi \vee \psi) & \to \exists x.\varphi \vee \exists x.\psi & \text{falls } x \in \mathsf{free}(\varphi) \text{ und } x \in \mathsf{free}(\psi)
\end{array}
$$

Lemma 2.4

Das bisherige Regelsystem ist terminierend.

Alle Regeln sind Äquivalenz-erhaltend.

Beispiel zum Miniscoping

Beispiel (*)

$\forall x.\exists y.\forall z.\,(r(x,x) \land [p(y) \lor r(x,y) \lor q(z)])$

$\rightarrow \forall x.\exists y.\,(r(x,x) \land \forall z.[p(y) \lor r(x,y) \lor q(z)])$

$\xrightarrow{*} \forall x.\exists y.\,(r(x,x) \land [p(y) \lor r(x,y) \lor \forall z.q(z)])$

$\rightarrow \forall x.\,(r(x,x) \land \exists y.[p(y) \lor r(x,y) \lor \forall z.q(z)])$

$\xrightarrow{*} \forall x.\,(r(x,x) \land [\exists y.p(y) \lor \exists y.r(x,y) \lor \forall z.q(z)])$

$\rightarrow \forall x.r(x,x) \land \forall x.[\exists y.p(y) \lor \exists y.r(x,y) \lor \forall z.q(z)]$

$\rightarrow \forall x.r(x,x) \land [\exists y.p(y) \lor \forall x.(\exists y.r(x,y) \lor \forall z.q(z))]$

$\rightarrow \forall x.r(x,x) \land [\exists y.p(y) \lor \forall x.\exists y.r(x,y) \lor \forall z.q(z)]$

Beachte

Es gilt nicht

$\forall x.(p(x) \lor q(x)) \rightarrow \forall x.p(x) \lor \forall x.q(x)$

$\exists x.(p(x) \land q(x)) \rightarrow \exists x.p(x) \land \exists x.q(x)$

Diese Regeln sind weder Äquivalenz- noch Erfüllbarkeit-erhaltend.

Variablen-Umbenennung

Regel

Benenne in φ die gebundenen Variablen so um, dass keine Variable mehrfach gebunden ist und keine gebundene Variable frei in φ vorkommt.

Beispiele

$\forall x.p(x) \lor \forall x.q(x) \rightarrow \forall x.p(x) \lor \forall x'.q(x')$

$\forall x.(\exists y.r(x,y) \land \exists y.r(y,x)) \rightarrow \forall x.(\exists y.r(x,y) \land \exists y'.r(y',x))$

$\forall x.(\forall y.r(x,y) \land \forall y.r(y,x)) \rightarrow \forall x.(\forall y.r(x,y) \land \forall y'.r(y',x))$

Beispiel (*) liefert

$\forall x.r(x,x) \land [\exists y.p(y) \lor \forall x.\exists y.r(x,y) \lor \forall z.q(z)]$

$\xrightarrow{*} \forall y_1.r(y_1,y_1) \land [\exists y_2.p(y_2) \lor \forall y_3.\exists y_4.r(y_3,y_4) \lor \forall y_5.q(y_5)]$

Lemma 2.5

Die Regel ist Äquivalenz-erhaltend.

Skolemisierung (1)

Ziel

Elimination der Existenz-Quantoren $\exists$.

Beachte zur Motivation

$\mathcal{A}$ erfüllt $\forall x.\exists y.r(x,y)$ $\;\mathrm{gdw}$

$\mathcal{A}' = \mathcal{A}$ erweitert um das „neue g“ erfüllt $\forall x.r(x, g(x))$.
(Man muss das g in $\mathcal{A}'$ geeignet interpretieren.)

Allgemeiner: Sei g neues Funktionssymbol.

$\varphi \wedge \forall x_1, \ldots, x_n.\exists y.r(x_1, \ldots, x_n, y)$ ist erfüllbar gdw

$\varphi \wedge \forall x_1, \ldots, x_n.r(x_1, \ldots, x_n, g(x_1, \ldots, x_n))$ ist erfüllbar.

Beachte zur $\exists$-Elimination.

1. $\exists$-Elimination „von außen nach innen“, d. h. $\exists y.$ steht nicht im Wirkungsbereich eines anderen $\exists$-Quantors.
2. Jeder Eliminationsschritt führt neues Funktionssymbol ein.
   Also: Signatur-Erweiterung.

Skolemisierung (2)

Regel

$$\varphi[\exists x.\psi]_\pi \to \varphi[\pi \leftarrow \psi[x \mapsto g(y_1, \ldots, y_n)]]$$

falls free$(\exists x.\psi) = \{y_1, \ldots, y_n\}$, g neu,
π minimale Position in φ mit $\varphi/\pi \equiv \exists x.\psi$.

Lemma 2.6

Das bisherige Regelsystem ist terminierend.

Diese Regel ist Erfüllbarkeit-erhaltend.

Also sind alle Regeln Erfüllbarkeit-erhaltend.

Beispiel (*) liefert

$\forall y_1.r(y_1, y_1) \wedge [\exists y_2.p(y_2) \vee \forall y_3.\exists y_4.r(y_3, y_4) \vee \forall y_5.q(y_5)]$

$\xrightarrow{*} \forall y_1.r(y_1, y_1) \wedge [p(a) \vee \forall y_3.r(y_3, g(y_3)) \vee \forall y_5.q(y_5)]$

mit a, g neu.

Erstellung der CNF (1)

Beachte

Sei $\varphi \xrightarrow{*} \psi$ und ψ irreduzibel. Dann gilt:

ψ enthält nur noch $\neg, \vee, \wedge$ und den Quantor $\forall$.

Es steht $\neg$ nur direkt vor Atomen.

Jetzt

a) Alle $\forall$-Quantoren ganz nach vorn ziehen.
   Also: $\psi \xrightarrow{*} \forall^* \psi_0$, die „Matrix" ψ_0 ist $\forall$-frei.

b) Transformation von ψ_0 in Konjunktion von Disjunktionen.

Beachte

Jede Variable in ψ ist nur einmal gebunden.

Also ist die Regel $\forall x.\psi' \vee \psi'' \to \forall x.(\psi' \vee \psi'')$,
angewandt auf ψ, Äquivalenz-erhaltend, da $x \notin$ free(ψ'').

Erstellung der CNF (2)

Finale Regeln

$\forall x.\varphi \wedge \psi \to \forall x.(\varphi \wedge \psi)$ falls $x \notin$ free(ψ)
$\forall x.\varphi \vee \psi \to \forall x.(\varphi \vee \psi)$ falls $x \notin$ free(ψ)
$\varphi \vee (\psi_1 \wedge \psi_2) \to (\varphi \vee \psi_1) \wedge (\varphi \vee \psi_2)$

Beispiel (*) liefert

$\forall y_1.r(y_1, y_1) \wedge [p(a) \vee \forall y_3.r(y_3, g(y_3)) \vee \forall y_5.q(y_5)]$
$\xrightarrow{*} \forall y_1, y_3, y_5.(r(y_1, y_1) \wedge [p(a) \vee r(y_3, g(y_3)) \vee q(y_5)])$

Dies ist in Klausel-Normalform. Die Klauselmenge ist

$$S = \{r(y_1, y_1), p(a) \vee r(y_3, g(y_3)) \vee q(y_5)\}$$

Beispiele

1. Beispiel $\forall$-$\exists$-Vertauschung

a) $\varphi \equiv \forall x.\exists y.\ x \cdot y = 1$
Skolemisierung liefert $\forall x.\ x \cdot g(x) = 1$.

b) $\varphi \equiv \exists y.\forall x.\ x^{-1} \cdot x = y$
Skolemisierung liefert $\forall x.\ x^{-1} \cdot x = a$.

c) $\varphi \equiv \neg\exists y.\forall x.\ x^{-1} \cdot x = y$
Es ergibt sich φ
$\varphi \xrightarrow{*} \forall y.\exists x.\ \neg x^{-1} \cdot x = y \to \forall y.\ \neg g(y)^{-1} \cdot g(y) = y$.

2. Beispiel Quacksalber

$\varphi \equiv \exists x.(\mathsf{pat}(x) \wedge \forall y.(\mathsf{doc}(y) \Rightarrow \mathsf{mag}(x,y)))$

$\to \exists x.(\mathsf{pat}(x) \wedge \forall y.(\neg\mathsf{doc}(y) \vee \mathsf{mag}(x,y)))$

$\to (\mathsf{pat}(a) \wedge \forall y.(\neg\mathsf{doc}(y) \vee \mathsf{mag}(a,y)))$

$\to \forall y.(\mathsf{pat}(a) \wedge (\neg\mathsf{doc}(y) \vee \mathsf{mag}(a,y)))$

Die Klauselmenge zu φ ist also
$S = \{\mathsf{pat}(a), \neg\mathsf{doc}(y) \vee \mathsf{mag}(a,y)\}$.

Erstellung der CNF (3)

Sei T die durch die Regeln beschriebene Transformation:

Erst die Simplifikationsregeln anwenden, dann die Regeln bis zur Skolemisierung anwenden, dann die finalen Regeln anwenden (jeweils so lange wie möglich).

Satz 2.2.

Die Transformation $T : \varphi \mapsto \psi$ terminiert. Sie ist Erfüllbarkeit-erhaltend. Es ist $\psi = T(\varphi)$ eine CNF für φ.

Bezeichnung

Die Klauselmenge $S = \{C_1, \ldots, C_n\}$ ist eine **Klauselmenge zu** φ, falls gilt:

φ ist erfüllbar gdw $\forall^*(C_1 \wedge \cdots \wedge C_n)$ ist erfüllbar.

Ergebnis zur CNF

Satz 2.3.

Sei $\mathcal{A}x = \{\varphi_1, \ldots, \varphi_n\}$ und ψ eine CNF zu $\varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\varphi$.

Es gilt

$\mathcal{A}x \models \varphi \;\; \mathrm{gdw} \;\; \psi$ ist unerfüllbar.

Beweis:

Zusammenfassung der bisherigen Überlegungen.

Frage also

Wie zeigt man, dass eine Formel ψ in CNF unerfüllbar ist?

Beachte

Die CNF zu $\varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\varphi$ ist nicht eindeutig. Es ist also vermutlich ratsam, eine kleine CNF (also eine kleine Klauselmenge S) zu berechnen.

Zur Berechnung einer Klauselmenge

Sei

$\mathcal{A}x = \{\varphi_1, \ldots, \varphi_n\}$ und φ gegeben.

Berechne Klauselmengen $S_1, \ldots, S_n, S_0$ zu $\varphi_1, \ldots, \varphi_n, \neg\varphi$. Benutze dabei keine Skolemfunktion mehrfach.

Dann ist $S = S_1 \cup \cdots \cup S_n \cup S_0$ eine Klauselmenge zu $\varphi_1 \wedge \cdots \wedge \varphi_n \wedge \neg\varphi$.

Also

$\varphi_1 \wedge \cdots \wedge \varphi_n \models \varphi \; \mathrm{gdw} \; S$ ist unerfüllbar.

Beachte

Sei $\{x_1, \ldots, x_n\} = \mathsf{Var}(D)$ und $\sigma = \{x_i \leftarrow y_i \mid i = 1, \ldots, n\}$.

Dann sind $\forall x_1, \ldots, x_n.(C \wedge D)$ und $\forall x_1, \ldots, x_n.C \wedge \forall y_1, \ldots, y_n.\sigma(D)$ logisch äquivalent.

Also

Man kann durch Variablenumbenennung erreichen, dass in $S = \{C_1, \ldots, C_n\}$ die Klauseln C_i paarweise Variablen-disjunkt sind.

Beispiel Quacksalber

Sei

$\varphi_1 \equiv \exists x.(\mathsf{pat}(x) \wedge \forall y.(\mathsf{doc}(y) \Rightarrow \mathsf{mag}(x,y)))$

$\varphi_2 \equiv \forall x.(\mathsf{pat}(x) \Rightarrow \forall y.(\mathsf{qua}(y) \Rightarrow \neg\mathsf{mag}(x,y)))$

$\varphi \equiv \forall x.(\mathsf{doc}(x) \Rightarrow \neg\mathsf{qua}(x))$

Transformation in CNF liefert

$\varphi_1 \overset{*}{\to} \forall y.(\mathsf{pat}(a) \wedge (\neg\mathsf{doc}(y) \vee \mathsf{mag}(a,y)))$

$\varphi_2 \overset{*}{\to} \forall x,y.(\neg\mathsf{pat}(x) \vee \neg\mathsf{qua}(y) \vee \neg\mathsf{mag}(x,y))$

$\neg\varphi \overset{*}{\to} \mathsf{doc}(b) \wedge \mathsf{qua}(b)$

Also

$\varphi_1 \wedge \varphi_2 \models \varphi$ gdw S ist unerfüllbar.

S:	$\mathsf{pat}(a)$, $\neg\mathsf{doc}(y) \vee \mathsf{mag}(a,y)$	aus φ_1
	$\neg\mathsf{pat}(x) \vee \neg\mathsf{qua}(y') \vee \neg\mathsf{mag}(x,y')$	aus φ_2
	$\mathsf{doc}(b)$, $\mathsf{qua}(b)$	aus $\neg\varphi$

Bemerkungen

1. Bemerkung

Es gibt viele Möglichkeiten, Klauselmengen zu vereinfachen.

Beispiel

Klausel C ist eine **Tautologie**, falls sie komplementäre Literale A und $\neg A$ enthält (A ein Atom).

Ist $S \cup \{C\}$ eine Klauselmenge zu φ und C eine Tautologie, so ist auch S eine Klauselmenge zu φ.

2. Bemerkung

Beschrieben ist eine Standardtransformation in CNF. Sie liefert i. Allg. eine zu große Klauselmenge.

Für Verbesserungen siehe
A. Nonnengart + C. Weidenbach.
Computing small clause normal forms
in: Robinson + Voronkov (Eds.): Handbook of Automated Reasoning, Elsevier, 2001.