Grundlagen der Programmierung

Vorlesung 2

Sebastian Iwanowski
FH Wedel

Beispiel für eine Programmverifikation

Gegeben sei folgender Algorithmus:

```
if (x>0) ∨ ((y+x)≤0)
  then
    z := x • y
  else
    z := x / y
```

Behauptung: **Dieser Algorithmus ist für alle x,y ∈ R ausführbar**

Beweis ?

Frage: **Ist der Algorithmus auch korrekt ?**

Ausblick auf die nächsten Vorlesungen

Wir haben in diesem Beispiel gesehen:

➔ **Das Gebiet der Programmverifikation erfordert einen sicheren Umgang mit formalen logischen Schlüssen**

Es gilt allgemein für <u>alle</u> Aspekte der Programmierung:

! **Grundkenntnisse der Aussagen- und Prädikatenlogik sind unentbehrlich.** **!**

Daher befassen sich die nächsten Vorlesungen mit dem Thema:

Logik

➔ **Details in Kapitel 2 <u>jetzt</u>**

Grundlagen der Programmierung

1. Einführung
   - Grundlegende Eigenschaften von Algorithmen und Programmen
2. Logik
   - ➔ Aussagenlogik
   - Prädikatenlogik
3. Programmentwicklung und –verifikation
   - Grundlagen der Programmverifikation
   - Verbundanweisungen
   - Verzweigungen
   - Schleifen
   - Modularisierung
   - Rekursion
4. Algorithmen
   - Entwurf von Algorithmen
   - Bewertung von Algorithmen

Aussagenlogik

Wdh.: Operatoren zwischen Aussagen

Durch Operatoren werden aus alten Aussagen neue Aussagen geschaffen:

Einstelliger Operator:

- **Negation (¬)**

Zweistellige Operatoren:

- **Konjunktion (∧)**
- **Disjunktion (∨)**
- **Implikation (→)**
- **Äquivalenz (↔)**

Wahrheitswerte für die neuen Aussagen:

p	q	¬p	p ∧ q	p ∨ q	p → q	p ↔ q
w	w	f	w	w	w	w
w	f	f	f	w	f	f
f	w	w	f	w	w	f
f	f	w	f	f	w	w

Aussagenlogik

Achtung: Warum einfach, wenn es auch kompliziert geht ?

Folgende Notationen sind in der Literatur ebenfalls gebräuchlich:

für logische Schlussregeln: $\dfrac{p\,,\,q}{r}$ *ist dasselbe wie* $p \wedge q \Rightarrow r$

In der Logik benutzt man unterschiedliche Symbole für Schlussfolgerungen und Äquivalenzen je nach Anwendungsgebiet:

$\vdash\,,\ \models\,,\ \prec\,,\ \rightarrow$ *entsprechen der logischen Implikation:* $\Rightarrow$

$\dashv\vdash\,,\ =\,,\ \equiv\,,\ \prec\succ\,,\ \leftrightarrow$ *entsprechen der logischen Äquivalenz:* $\Leftrightarrow$

Aussagenlogische Formeln

- **Da Aussagen wegen der Unteilbarkeit sinnvollerweise mit Buchstaben abgekürzt werden, nennt man sie auch *Literale***

- **Eine aussagenlogische Formel ist eine Verknüpfung von endlich vielen Literalen mit logischen Operatoren.**

- Die Literale einer Formel entsprechen Variablen, die mit w und f belegt werden können.

- Formeln können auch Konstante enthalten:

  Die Konstante $\top$ ist immer wahr und das neutrale Element bezüglich der Konjunktion.

  Die Konstante $\perp$ ist immer falsch und das neutrale Element bezüglich der Disjunktion.

- **Eine Belegung einer Formel ist eine Zuweisung von Wahrheitswerten an die Literale derart, dass dieselben Literale immer denselben Wahrheitswert erhalten.**

  Die Formel als ganze bekommt durch die Belegung ebenfalls einen Wahrheitswert.

Aussagenlogische Formeln

- **Eine Formel heißt erfüllbar, wenn es eine Belegung gibt derart, dass die Formel den Wahrheitswert w hat.**

  Eine Formel, in der jeder Literal höchstens einmal vorkommt, ist immer erfüllbar!

  Eine Formel, in der keine Negation vorkommt, ist immer erfüllbar!

  $\Rightarrow$ **Nur Formeln, die einen Literal mehrfach und mindestens eine Negation enthalten, könnten unerfüllbar sein.**

- **Eine Formel heißt Tautologie oder gültig, wenn sie bei jeder Belegung den Wahrheitswert w hat.**

- **Eine Formel heißt widersprüchlich, wenn sie bei keiner Belegung den Wahrheitswert w hat.**

Aussagenlogische Formeln

Erfüllbarkeitsproblem (Satisfiability, SAT):

Wie bekommt man heraus, ob eine gegebene Formel erfüllbar ist ?

Normalformen

Definition:

Eine aussagenlogische Formel ist in **konjunktiver Normalform (KNF)**, wenn sie als Konjunktion von Disjunktionen aus Aussagen oder Negationen von Aussagen dargestellt ist.

Etwas langsamer zum Mitdenken:

Ein **Literal** ist eine Aussage oder die Negation einer Aussage.

Beispiele: p $\neg q$ r $\neg r$

Eine **Klausel** ist eine Disjunktion aus Literalen.

Beispiele: $p \lor \neg q \lor r$ $\neg p \lor \neg q \lor \neg r$ $p \lor q$

Eine **Formel in KNF** ist eine Konjunktion von Klauseln.

Beispiel: $(p \lor \neg q \lor r) \land (\neg p \lor \neg q \lor \neg r) \land (p \lor q)$

Normalformen

Definition:

Eine aussagenlogische Formel ist in **konjunktiver Normalform (KNF)**, wenn sie als Konjunktion von Disjunktionen aus Aussagen oder Negationen von Aussagen dargestellt ist.

Spezialfälle:

Eine **Klausel** darf auch nur aus einem Literal bestehen.

Beispiel: $(p \lor \neg q \lor r) \land \neg p \land (p \lor q)$ ist auch in KNF

Eine **Formel** darf auch nur aus einer Klausel bestehen.

Beispiel: $p \lor \neg q \lor r$ ist auch in KNF

Eine **leere Klausel** entspricht dem neutralen Element der Disjunktion: $\bot$

Beispiel: $(p \lor \neg q \lor r) \land \bot \land (p \lor q)$ ist auch in KNF

Eine **leere Formel** entspricht dem neutralen Element der Konjunktion: $\top$

Beispiel: $\top$ ist auch in KNF

Normalformen

Welche Formeln sind eigentlich nicht in KNF ?

- Formeln, die andere als die 3 Booleschen Operatoren enthalten

  Beispiel: $((p \lor \neg q \lor r) \rightarrow (\neg p \lor \neg q \lor \neg r)) \leftrightarrow (p \lor q)$

- Formeln, in denen andere Terme als atomare Aussagen negiert werden

  Beispiel: $\neg(p \lor \neg q \lor r) \land (\neg p \lor \neg q \lor \neg r) \land \neg(p \lor q)$

- Formeln, in denen Konjunktionen und Disjunktionen wild durcheinander sind

  Beispiel: $(p \land (\neg q \lor r)) \land (\neg p \lor \neg q \lor \neg r) \lor p \land q$

  Beispiel: $p \land (\neg q \lor r \land \neg p) \lor (\neg q \land \neg r) \lor p \land q$

- Formeln, in denen eine tiefere Klammerschachtelungstiefe vorliegt

  Beispiel: $p \land (\neg q \lor (r \land q))) \land (\neg p \lor \neg q \lor \neg r) \land p \lor q$

Satz: **Jede aussagenlogische Formel lässt sich durch endlich viele äquivalente Umformungen in KNF bringen.**

Normalformen

Warum wollen wir Formeln in KNF bringen ?

- **Übersichtlichere Auswertung beim Erfüllbarkeitstest:**

  Eine Formel in KNF ist erfüllbar.
  ⇔ Eine Belegung enthält für jede Klausel wenigstens einen wahren Literal.

  Mögliche Belegungsstrategie:

  - Gehe die Klauseln nacheinander durch:
  - Belege genau einen noch nicht festgelegten Literal mit **t**

    (dadurch werden gleiche Literale oder deren Negationen in anderen Klauseln festgelegt)
  - Wenn es keine Möglichkeit mehr gibt, springe zurück zur vorigen Klausel und nimm eine andere Belegung

Vorsicht vor Illusionen:

Im schlechtesten Fall bringt das keinen Zeitgewinn verglichen mit purem Ausprobieren !

Normalformen

Warum wollen wir Formeln in KNF bringen ?

- **Kompakte Darstellbarkeit im Computer:**

Stelle Klauseln als Mengen von Literalen dar:

$\{p, \neg q, r\}$	entspricht	$(p \lor \neg q \lor r)$	**als Klausel**

Stelle Formeln als Mengen von Klauseln dar:

$\{\{p, \neg q, r\}, \{\neg p, \neg q, \neg r\}, \{p, q\}\}$	entspricht	$(p \lor \neg q \lor r) \land (\neg p \lor \neg q \lor \neg r) \land (p \lor q)$

Das funktioniert sogar für die Spezialfälle:

$\{\{p, \neg q, r\}\}$	entspricht	$(p \lor \neg q \lor r)$	**als Formel**
$\{\{p\}, \{\neg q\}, \{r\}\}$	entspricht	$(p \land \neg q \land r)$	**als Formel**
$\{\}$	entspricht	$\top$	**als Formel**
$\{\{\}\}$	entspricht	$\bot$	**als Formel**

Warnung:

Was dem Computer glasklar ist, kann für den Menschen höchst verwirrend sein:

- Das Trennzeichen (,) in **inneren** Klammern (Klauseln) entspricht einer Disjunktion ($\lor$)
- Das Trennzeichen (,) in **äußeren** Klammern (Formeln) entspricht einer Konjunktion ($\land$)

Normalformen

Algorithmus zum Umwandeln einer Formel in die KNF:

1. Eliminiere alle Operatoren der Form $\leftrightarrow$ und $\rightarrow$ mit den Ersetzungsregeln durch $\wedge$ und $\vee$!

2. Ziehe alle Negationszeichen vor Klammern in die Klammern hinein mit den deMorganschen Regeln !

3. Wende die Distributivgesetze so lange an, bis auf oberster Ebene nur noch Konjunktionen und darunter Disjunktionen sind !

Das funktioniert immer !

Andere Normalformen

Definition:

Eine aussagenlogische Formel ist in **disjunktiver Normalform (DNF)**, wenn sie als Disjunktion von Konjunktionen aus Aussagen oder Negationen von Aussagen dargestellt ist.

Jetzt fällt das Mitdenken schon leichter:

Was ist in DNF ein **Literal** ?

Was ist in DNF eine **Klausel** ?

Was ist in DNF eine **Formel** ?

Beispiele ?

Warum wollen wir Formeln in DNF bringen ?

wollen wir nicht: Eine Normalform reicht uns aus !

Beim nächsten Mal: Prädikatenlogik