

Entscheidbarkeit der linearen Arithmetik über \mathbb{Q}

Tobias Nipkow

L^AT_EXifiziert von Johannes Hölzl

2006-06-09

1 Logik

Terme bestehen aus Variablen und Funktionssymbolen. (*lineare Arithmetik*: $0, 1, +, -, c \cdot -$ ($c \in \mathbb{Q}$)).

Atomare Formeln bestehen aus Termen und Prädikat Symbolen (*lineare Arithmetik*: $=, <, \leq$),

Formeln sind der Abschluss der Atome unter $\neg, \vee, \wedge, \rightarrow, \forall, \exists$.

FV: Die Menge der freien Variablen in einer Formel, d.h. alle Variablen die nicht durch einen Quantor wie \forall oder \exists gebunden wurden.

geschlossener Term / Formel: Ein Term ist *geschlossen* falls $FV(\cdot) = \emptyset$.

Satz ist eine geschlossene Formel.

Struktur ist eine Menge zusammen mit einer Interpretation der Funktionssymbole durch Funktionen und der Prädikatssymbole durch Relationen.

$M \models \phi$: Die Formel ϕ *gilt* in einer Struktur M , d.h. M ist ein *Modell* von ϕ . Dies muss für alle Belegungen der freien Variablen von ϕ gelten.

\mathbb{Q} sei die bekannte Struktur der rationalen Zahlen.

Th(M) ist die *Theorie* von M , d.h. die Menge der in M gültigen Sätze.

$\phi_1 \equiv \phi_2$ genau dann wenn für alle M gilt, dass $M \models \phi_1$ gdw $M \models \phi_2$.

Thm 1 (Pränex-Normalform). *Zu jedem ϕ gibt es ein $\phi' \equiv \phi$, so dass ϕ' in Pränex-Normalform ist:*

$$\phi' = Q_1 x_1 Q_2 x_2 \dots Q_n x_n \cdot \phi_0$$

wobei $Q_i \in \{\forall, \exists\}$ und ϕ_0 quantorenfrei ist.

Beweis.

$$\begin{aligned}(\forall x.P) \wedge Q &\equiv \forall x.(P \wedge Q) \\(\forall x.P) \vee Q &\equiv \forall x.(P \vee Q) \\ \neg(\forall x.P) &\equiv \exists x.\neg P\end{aligned}$$

Dies gilt dual für den Existenzquantor. □

Thm 2 (Negations-Normalform). *Zu jedem ϕ gibt es ein $\phi' \equiv \phi$, so dass ϕ' in Negations-Normalform ist: In ϕ' wird \neg nur auf Atome angewandt.*

Beweis.

$$\begin{aligned} \neg\neg P &\equiv P \\ \neg(\phi_1 \wedge \phi_2) &\equiv \neg\phi_1 \vee \neg\phi_2 \\ \neg(\phi_1 \vee \phi_2) &\equiv \neg\phi_1 \wedge \neg\phi_2 \\ \neg(\forall x.P) &\equiv \exists x.\neg P \\ \neg(\exists x.P) &\equiv \forall x.\neg P \end{aligned}$$

□

2 Quantorenelimination

Eine QE-Prozedur (für eine Struktur M) ist eine Funktion, die eine Formel $\exists x.\phi$, in der ϕ quantorenfrei ist, abbildet auf eine quantorenfreie Formel ϕ' mit $M \models \exists x.\phi$ gwd $M \models \phi$ und mit $FV(\phi') \subseteq FV(\exists x.\phi)$

Beispiel 1. *In QBF (quantifizierten booleschen Formeln) gilt*

$$\exists x.\phi \equiv \phi[0/x] \vee \phi[1/x]$$

Thm 3. *Seien alle geschlossenen Atome entscheidbar, d.h. $M \models A$ ist für alle geschlossenen Atome A entscheidbar. Gibt es eine QE-Prozedur für M , so ist $Th(M)$ entscheidbar.*

Beweis. Um $M \models \phi$ zu entscheiden verfähre wie folgt:

1. Bringe ϕ in Pränex-Normalform.
2. Eliminiere die Quantoren von innen nach außen, wobei Allquantoren jeweils vorher in Existenzquantoren umgewandelt werden:

$$\forall x.\phi \equiv \neg\exists x.\neg\phi$$

3. Entscheide dir resultierende boolesche Kombination von geschlossenen Atomen.

□

3 Fourier-Motzkin Elimination

Wir erlauben nur die Prädikate $<$ und $=$.

Gegeben: $\exists x.\phi$

1. Wandle ϕ in Negations-Normalform um und eliminire \neg :

$$\begin{aligned} \neg(t_1 = t_2) &\equiv t_1 < t_2 \vee t_2 < t_1 \\ \neg(t_1 < t_2) &\equiv t_2 < t_1 \vee t_2 = t_1 \end{aligned}$$

2. Wandle in DNF um und schiebe $\exists x$ durch \vee nach innen:

$$\exists x.(\phi_1 \vee \dots \vee \phi_n) \equiv (\exists x.\phi_1) \vee \dots \vee (\exists x.\phi_n)$$

3. QE in $\exists x.\phi$ mit $\phi = A_1 \wedge \dots \wedge A_m$.

(a) Miniscoping (betrachte nur die A_i die x enthalten):

$$\exists x.(\psi_1 \wedge \psi_2) \equiv (\exists x.\psi_1) \wedge \psi_2$$

falls $x \notin FV(\psi_2)$. Daher können wir obdA annehmen, dass $x \in FV(A_i)$ für $i = 1, \dots, m$.

(b) Isoliere x in A_i . Da die A_i lineare Formeln sind (Multiplikation nur mit Konstanten!), können wir obdA annehmen, dass jedes A_i von der Form $x = t_i$, $x < t_i$ oder $t_i < x$ ist (bzw sich in diese Form bringen lässt).

(c) Wir unterscheiden 2 Fälle:

i. Es gibt k mit $A_k = (x = t_k)$

$$\Rightarrow \exists x.\phi \equiv \phi[t_k/x]$$

ii. Falls nur $<$ in ϕ vorkommt unterteilen with ϕ :

$$\phi \equiv \bigwedge_{i=1}^l L_i \wedge \bigwedge_{j=1}^u U_j$$

mit $L_i = (l_i < x)$ und $U_i = (x < u_i)$.

A. $l = 0$ (oder $p = 0$), d.h. es gibt nur obere (oder untere) Schranken.

$$\Rightarrow \exists x.\phi \equiv \exists x. \bigwedge_{j=1}^u U_j \equiv True$$

Zeuge: $x = \min u_j - 1$, es gibt in \mathbb{Q} immer eine kleinere Zahl.

B. $l > 0 \wedge u > 0$, d.h. es gibt sowohl obere als auch untere Schranken.

$$\Rightarrow \exists x.\phi \equiv \bigwedge_{i,j} l_i < u_j$$

Alle oberen Schranken müssen größer sein als alle unteren Schranken.

Beispiel 2.

$$\begin{aligned} & \forall yx. \quad x < y \rightarrow x < 2y \\ \equiv & \neg \exists yx. \quad x < y \wedge 2y \leq x \\ \equiv & \neg \exists y. \quad 2y < y \\ \equiv & \neg \exists y. \quad y < 0 \\ \equiv & \neg True \\ \equiv & False \end{aligned}$$

Lemma 1. *Eine Konjunktion von Ungleichungen ($<$ und \leq) ist in \mathbb{Q} unerfüllbar genau dann, wenn es eine Linearkombination der Ungleichungen gibt, die falsch ist, d.h. äquivalent zu „ $c < d$ “ (oder „ $c \leq d$ “) ist, wobei der Wert von c größer-gleich (oder größer) als der von d ist.*

Zum Beweis überlege man sich, dass im entscheidenden Schritt in der Fourier-Motzkin Elimination $l_i < x$ und $x < u_j$ addiert werden um $(l_i + x < x + u_j) \equiv (l_i < u_j)$ zu erhalten, und dass die Addition zweier Linearkombinationen wieder eine Linearkombination ist.

Beispiel 3.

$$\begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \\ -1 & 1 & 2 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

ist unerfüllbar da

$$(1, 0, 1, 2) \begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \\ -1 & 1 & 2 \\ 0 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \not\leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$