Werbung2: Berechenbarkeitstheorie

Das Halteproblem ist unentscheidbar:
Terminiert ein gegebener Algorithmus auf Eingabe x?

Weitere unentscheidbare Probleme der Informatik:
• Posts Korrespondenzproblem • Hilberts 10. Problem

Theorem: Für $f:\subseteq \mathbb{N}^d \to \mathbb{N}$ sind äquivalent:
• f ist berechenbar durch eine Turingmaschine
• f ist berechenbar durch ein `WHILE`-Programm
• f läßt sich im λ-Kalkül ausdrücken
• f ist μ-rekursiv

Rekursionstheorem/Quines: In jeder Turing-vollständigen Programmiersprache gibt es ein Programm, das sich selbst reproduziert.

Einige weitere unentscheidbare Probleme …

… die nicht Eigenschaften von DTMs testen.

Diophantische Gleichungen:

Gegeben ein multivariates Polynom p
mit ganzzahl. Koeffizienten,
besitzt p eine ganzzahlige Nullstelle?

$$\{ \langle p \rangle : p \in \mathbb{Z}[X_1,\ldots,X_n] \,,\, n \in \mathbb{N},\ \exists\, \underline{x} \in \mathbb{Z}^n : p(\underline{x}) = 0 \}$$

semi-entscheidbar √ unentscheidbar:

Arithmetik: → Hilberts zehntes Problem (1900)

Gegeben eine arithmetische Aussage (Variablen über $\mathbb{Z}$, $+$, $-$, $\times$, Quantoren, Vergleiche, logische Verknüpfungen): Ist sie wahr?

Beispiel: $p := X^7 + Y^7 - Z^7$

Beispiel: $\forall n\, \exists x \geq n$: $A(x) \wedge A(x+2)$,
$A(x) :=$ „$\forall y,z$: $x = y \cdot z \Rightarrow y = 1 \vee z = 1$”

Presburger Arithmetik
(wie oben, aber ohne Multiplikation) ist entscheidbar !!

Posts Korrespondenzproblem

Satz: Die folgende Sprache ist semi-, unentscheidbar:

$$\mathrm{PKP} := \{ (\underline{x}_1, \underline{y}_1), \ldots, (\underline{x}_k, \underline{y}_k) \mid k \in \mathbb{N},\ \underline{x}_i, \underline{y}_i \in \Sigma^*,$$
$$\exists\, n \in \mathbb{N}, i_1, \ldots, i_n \in \{1, \ldots, k\}:\ \underline{x}_{i_1}\underline{x}_{i_2}\ldots\underline{x}_{i_n} = \underline{y}_{i_1}\underline{y}_{i_2}\ldots\underline{y}_{i_n} \}$$

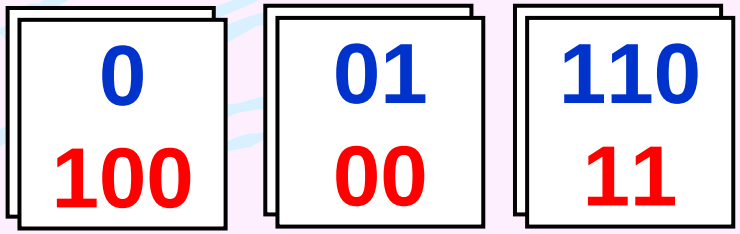
Bsp: k=3, $\underline{x}_1$=0, $\underline{x}_2$=01, $\underline{x}_3$=110, $\underline{y}_1$=100, $\underline{y}_2$=00, $\underline{y}_3$=11

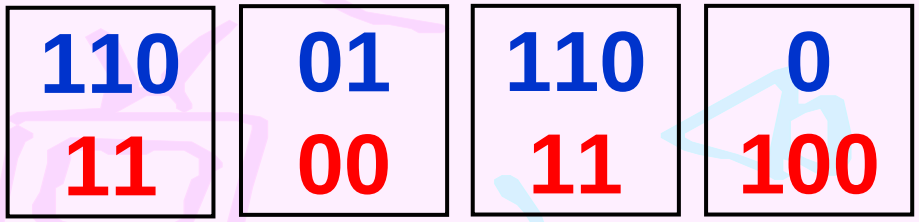
z.B. n=4, i_1=3, i_2=2, i_3=3, i_4=1: 1 1 0 0 1 1 1 0 0

Bsp: k=2, $\underline{x}_1$=01, $\underline{x}_2$=110, $\underline{y}_1$=00, $\underline{y}_2$=11

keine Lösung , da letzte Ziffern von $\underline{x}_{i_n}$ und $\underline{y}_{i_n}$ ungleich

PKP = Spiel mit Dominos:





Werbung3: Komplexitätstheorie

Erfüllbarkeitsproblem der Aussagenlogik:
zB Ausprobieren aller 2^n Boole'schen Belegungen
Geht es schneller? $\rightarrow$ $\mathcal{P}$ versus $\mathcal{NP}$ (1Mio$)

- asymptotische Laufzeiten / Speicherverbrauch
- Algorithmen-Entwicklung und –Analyse: obere Schranken
- Komplexitätstheorie: untere Schranken
- Gibt es Sprachen, die sich algorithmisch z.B. in Zeit $O(2^n)$ aber nicht in $O(n^3)$ entscheiden lassen?

Ja!

- Vergleich und Klassifikation von Problemen
- kryptographische Komplexität, Randomisierung, Kommunikationskomplexität (Parallelcomputing)…

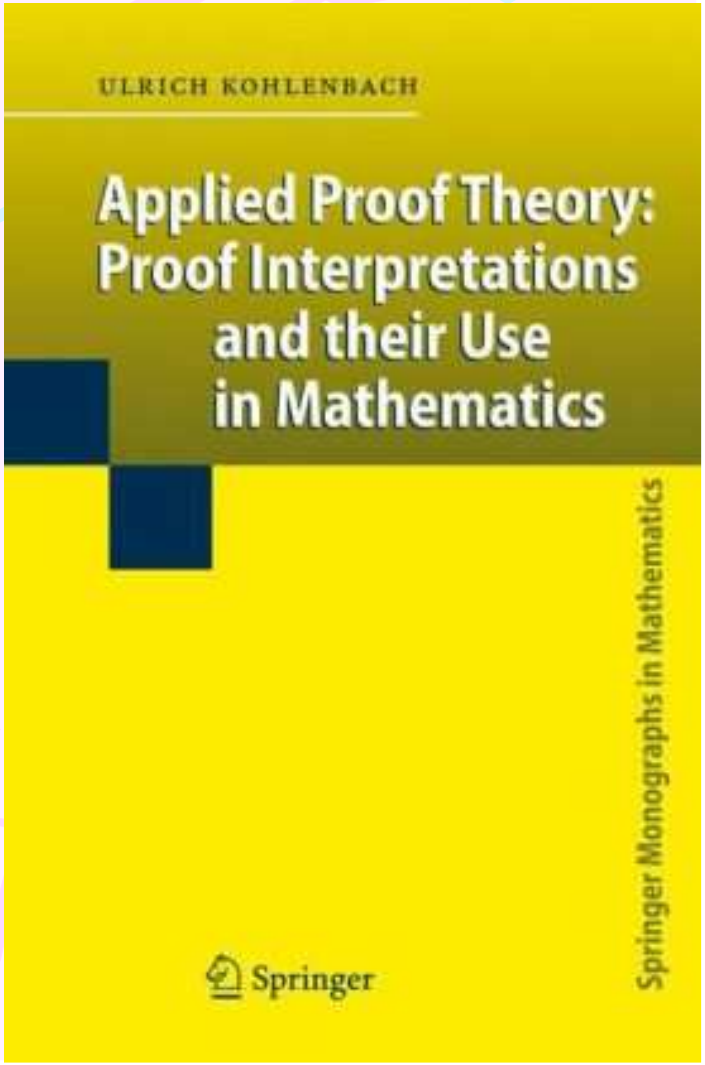
Werbung4: Beweistheorie

$\forall x\, \exists y \;\; x^2<y$ | $\forall f{:}\{1,\ldots,n\}\to\{1,\ldots,n\text{-}1\}\; \exists x\neq y{:}\; f(x)=f(y)$

Kann man für gegebenes x
so ein y auch berechnen?

Kann man für gegebenes f
so ein (x,y) berechnen?

Idee: Der Existenzbeweis
einer Formel enthält implizit
einen Algorithmus.
→ Proof mining



Der erfolgreiche Blick des Logikers

Ulrich Kohlenbach erhält den Forschungspreis der Kurt-Gödel-Gesellschaft

Ein neuer Blickwinkel liefert oft neue Ergebnisse. Das macht sich Ulrich Kohlenbach, Professor am Fachbereich Mathematik der TU Darmstadt, erfolgreich zu nutze. Er formuliert mathematische Beweise aus der Sicht des Logikers um und legt so neue, stärkere Aussagen frei. Für seine exzellente Arbeit erhält er heute in Wien den mit 100.000 Euro dotierten Kurt-Gödel-Forschungspreis.

„Viele Beweise in der Mathematik verwenden komplizierte Prinzipien. Oft ist es unmöglich, effektive Schranken oder andere Daten direkt aus dem Beweis abzulesen“, sagt Professor Ulrich Kohlenbach vom Fachbereich Mathematik der TU Darmstadt. Um neue Informationen aus den Beweisen zu gewinnen, nutzt er Beweisinterpretationen. „Ich formuliere den Beweis so um, dass der endliche kombinatorische Kern freigelegt wird“, erläutert Kohlenbach. „Proof Mining“ heißt das Verfahren.

Am Ende der Arbeit steht der Beweis einer neuen, stärkeren Aussage. Dieser Beweis ist auch ohne Kenntnis der angewandten logischen Methoden verständlich. Er kann daher in den einschlägigen Zeitschriften des jeweiligen Anwendungsgebiets veröffentlicht werden. Kohlenbach und seine Mitarbeiter haben schon zahlreiche Arbeiten in mathematischen Zeitschriften wie „Nonlinear Analysis“ oder „Ergodic Theory and Dynamical Systems“ veröffentlicht, die normalerweise keine Ergebnisse der Logik drucken.

Kohlenbachs Forschung zielt auf Anwendungen der Logik innerhalb der Mathematik. „Es geht mir um Erkenntnisgewinn bezüglich grundlagentheoretischer Fragen wie der Anwendbarkeit infinitärer mengentheoretischer Prinzipien zum Beweis finiter kombinatorischer Aussagen“, sagt er. Die Logik spielt auch in der Informatik eine große Rolle, etwa in der künstlichen Intelligenz, bei automatischen Beweisen oder in der Semantik funktionaler Programmiersprachen. So wurden schon Beweisinterpretationen implementiert, um Software automatisch zu verifizieren.

Werbung5: mathematische Grundlagen der funktionalen Programmierung

- konstruktive Typtheorie und ihre (kategorielle) Semantik
- denotationale Semantik von funktionalen Programmiersprachen
- synthetische Domaintheorie und ihre Axiomatisierung in Typtheorie
- Untersuchung voll abstrakter Modelle für sequentielle Sprachen
- Herleitung abstrakter Environment Maschinen aus Continuationsemantik
- semantische Beweise von Normalisierung.