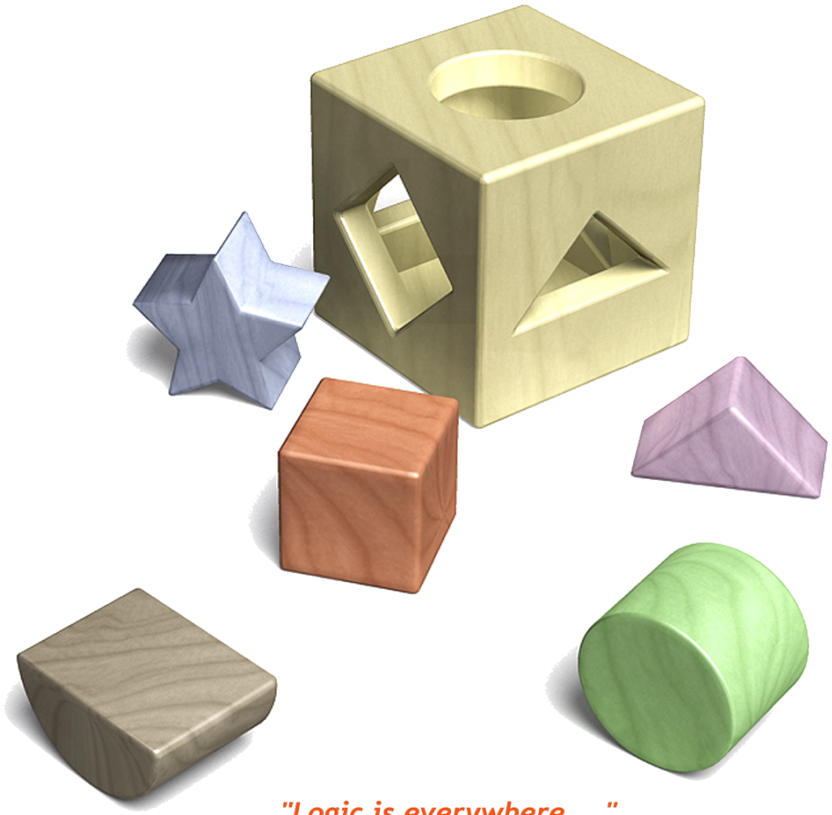
SAT Solving oder Lösen von Erfüllbarkeitsproblemen

Steffen Hölldobler
International Center for Computational Logic
Technische Universität Dresden
Germany

- Aussagenlogik
- Erfüllbarkeitsprobleme
- Anwendungen
- Konjunktive Normalform
- SAT Solver
- Hamiltonkreisprobleme
- Wettbewerb



"Logic is everywhere ..."

Aussagen

- **Atomare Aussagen (atomare Formeln oder Atome)**

  Die Sonne scheint
  Der Knoten mit der Nummer 1 ist grün
  LH1015 fliegt von Dresden nach Frankfurt
  An der 4. Stelle des DNA-Strangs steht das Nukleotid C

- **Abkürzungen für Atome**

  p_1 Die Sonne scheint
  p_2 ...

- **Wahrheitswerte** wahr ($\top$) und falsch ($\bot$)

- **Interpretation** Abbildung von der Menge der Aussagen zu $\{\top, \bot\}$

  $p_1 \mapsto \top$ Die Aussage "die Sonne scheint" ist wahr
  $p_2 \mapsto \bot$ Die Aussage "der Knoten mit der Nummer 1 ist grün" ist falsch

Komplexe Aussagen

- **Komplexe Aussagen (Sätze oder Formeln)**

Die Sonne scheint	p_1
Die Sonne scheint nicht	$\neg p_1$
Die Sonne scheint oder die Sonne scheint nicht	$(p_1 \vee \neg p_1)$
Die Sonne scheint und der Knoten mit der Nummer 1 ist grün	$(p_1 \wedge p_2)$
Wenn die Sonne scheint, dann ist der Knoten mit der Nummer 1 grün	$(p_1 \rightarrow p_2)$

- **Definition** Die Menge der (aussagenlogischen) Formeln ist die kleinste Menge, die die folgenden Bedingungen erfüllt:
  - Alle Atome sind Formeln.
  - Wenn F eine Formel ist, dann ist auch $\neg F$ eine Formel.
  - Wenn F und G Formeln sind, dann sind auch $(F \vee G)$, $(F \wedge G)$ und $(F \rightarrow G)$ Formeln.

- **Beispiel** $\neg(\neg(p_1 \wedge p_2) \rightarrow (p_1 \vee \neg p_1))$
- **Verabredung** Wir lassen "p" weg und notieren Atome als natürliche Zahlen.

Interpretationen und Modelle

- **Wir erinnern uns**
  Interpretationen sind Abbildungen von der Menge der Aussagen zu $\{\top, \bot\}$
- **Wie interpretieren wir komplexe Aussagen?**

F	$\neg F$
$\top$	$\bot$
$\bot$	$\top$

F	G	$(F \vee G)$	$(F \wedge G)$	$(F \rightarrow G)$
$\top$	$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\top$	$\bot$	$\top$
$\top$	$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\bot$	$\bot$	$\top$

- **Gegeben** **Interpretation I mit $1 \mapsto \top$ und $2 \mapsto \bot$; dann**

$$I(\neg(\neg(1 \wedge 2) \rightarrow (1 \vee \neg 1))) = \bot$$

- **Definition** **Eine Interpretation, die eine gegebene Formel F auf $\top$ abbildet, wird Modell für F genannt**

Erfüllbarkeitsprobleme

- **Definition** Eine Formel F ist **erfüllbar**, wenn es eine Interpretation I gibt, die F auf $\top$ abbildet
- **Definition** Ein **Erfüllbarkeitsproblem** besteht aus einer Formel F und ist die Frage, ob F erfüllbar ist
- **Beispiel** Ist $F = \neg(\neg(1 \wedge 2) \rightarrow (1 \vee \neg 1))$ erfüllbar? **Nein**

Interpretation		Formel
1	2	F
$\top$	$\top$	$\bot$
$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\bot$

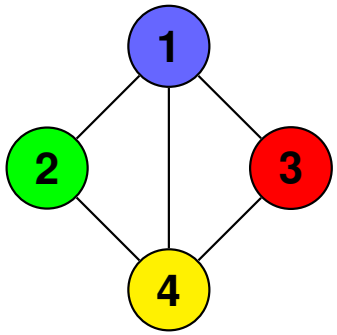
- **Beobachtung** Wenn eine Formel n verschiedene Atome enthält, dann gibt es 2^n verschiedene Interpretationen
- **Bemerkung**
  Die Komplexitätstheorie wurde anhand des Erfüllbarkeitsproblems entwickelt!

Anwendungen

- Terminierung von Programmen
- Planen und Konfigurieren
- Bioinformatik
- Verifikation von Hard- und Software
- Scheduling
- Kryptoanalyse
- Verifikation von Bahnsignalsystemen
- Answer Set Programming

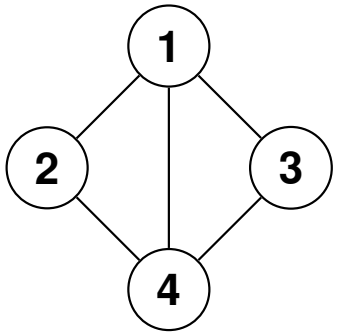
Das Graphfärbungsproblem

- **Gegeben** ein endlicher Graph und eine Menge von Farben
- **Gesucht** eine Zuordnung von Farben zu Knoten,
  so dass alle benachbarte Knoten unterschiedlich Farben haben
- **Formal**
  - Jedem Knoten ist mindestens eine Farbe zugeordnet
  - Jedem Knoten ist höchstens eine Farbe zugeordnet
  - Benachbarte Knoten haben verschiedene Farben



Naive Kodierung des Graphfärbungsproblems (1)

- **Aufgabe** Finde für ein gegebenes Graphfärbungsproblem G eine aussagenlogische Formel F, so dass jedes Modell für F eine Lösung für G kodiert
- **Beispiel** Wir betrachten die Menge $\{1, 2, 3, 4\}$ von Farben und den Graphen



- **Naive Kodierung der Zuordnung von Farben zu Knoten**

11	Knoten 1 ist Farbe 1 zugeordnet
12	Knoten 1 ist Farbe 2 zugeordnet
⋮	⋮
44	Knoten 4 ist Farbe 4 zugeordnet.

Naive Kodierung des Graphfärbungsproblems (2)

- **Jedem Knoten ist mindestens eine Farbe zugeordnet**

$$(11 \vee 12 \vee 13 \vee 14) \wedge \ldots \wedge (41 \vee 42 \vee 43 \vee 44)$$

- **Jedem Knoten ist höchstens eine Farbe zugeordnet**

$$\neg(11 \wedge 12) \wedge \neg(11 \wedge 13) \wedge \neg(11 \wedge 14) \ldots \wedge \neg(43 \wedge 44)$$

- **Benachbarte Knoten haben verschiedene Farben**

$$\neg(11 \wedge 21) \wedge \neg(12 \wedge 22) \wedge \neg(13 \wedge 23) \wedge \ldots \wedge \neg(34 \wedge 44)$$

- **Behauptung** Sei F die Konjunktion der so erhaltenen Formeln. Jedes Modell für F kodiert eine Lösung des Graphfärbungsproblems
- **Beispiel** Die Interpretation, die 11, 22, 33 und 44 auf $\top$ und alle anderen Atome auf $\bot$ abbildet, ist ein Modell für F

Einige Beobachtungen

- Die Formeln werden sehr groß
- 10^7 verschiedene Atome und 10^8 Konjunkte entstehen leicht bei Anwendungen
- Damit ergibt sich ein Suchraum von 2^{10^7} !
- Die Formeln sind nicht unmittelbar zu verstehen
- Wir brauchen maschinelle Unterstützung
- Dies leisten SAT Solver

Eine geschicktere Kodierung des Graphfärbungsproblems (1)

- **Idee** **Wir ordnen die Farben an: $1 < 2 < 3 < 4$**
- **Kodierung der Zuordnung von Farben zu Knoten**

11	**Knoten 1 hat Farbe größer gleich 1**
12	**Knoten 1 hat Farbe größer gleich 2**
$\vdots$	$\vdots$
44	**Knoten 4 hat Farbe größer gleich 4**
$11 \wedge \neg 12$	**Knoten 1 hat die Farbe 1**
$12 \wedge \neg 13$	**Knoten 1 hat die Farbe 2**
$13 \wedge \neg 14$	**Knoten 1 hat die Farbe 3**
14	**Knoten 1 hat die Farbe 4**
$\vdots$	$\vdots$
44	**Knoten 4 hat die Farbe 4**

Eine geschicktere Kodierung des Graphfärbungsproblems (1)

- **Jedem Knoten ist mindestens eine Farbe zugeordnet**

$$11 \wedge 21 \wedge 31 \wedge 41$$

- **Jedem Knoten ist höchstens eine Farbe zugeordnet**

$$(\neg 11 \rightarrow \neg 12) \wedge (\neg 12 \rightarrow \neg 13) \wedge (\neg 13 \rightarrow \neg 14) \wedge \ldots \wedge (\neg 43 \rightarrow \neg 44)$$

- **Benachbarte Knoten haben verschiedene Farben**

$$\neg(11 \wedge \neg 12 \wedge 21 \wedge \neg 22) \wedge \ldots \wedge \neg(34 \wedge 44)$$

- **Behauptung** Sei F die Konjunktion der so erhaltenen Formeln.
  Jedes Modell für F kodiert eine Lösung des Graphfärbungsproblems
- **Beobachtung** In Experimenten verringerte sich die Laufzeit
  bei Verwendung der geschickteren Kodierung durchschnittlich um 10%

Konjunktive Normalform (1)

- SAT-Solver akzeptieren in der Regel Formeln nur in einer bestimmten Form
- **Definition** Ein **Literal** ist ein Atom oder dessen Negation
- **Beispiele** 11, $\neg 12$, $\neg 13$, 14
- **Definition** Eine Formel ist in **konjunktiver Normalform (CNF)**, wenn sie von der Form

$$(L_{11} \vee \ldots \vee L_{1n_1}) \wedge \ldots \wedge (L_{m1} \vee \ldots \vee L_{mn_m})$$

  ist, wobei die L_{ij} Literale sind
- **Definition** Zwei Formeln F und G sind **äquivalent**, wenn für alle Interpretationen I die Gleichung $I(F) = I(G)$ gilt
- **Beispiel** $\neg(11 \wedge 12)$ und $(\neg 11 \vee \neg 12)$ sind äquivalent

Konjunktive Normalform (2)

- **Theorem** **Zu jeder Formel gibt es eine äquivalente Formel in CNF**
- **Beispiel**

$$(\neg 11 \vee \neg 12) \wedge (\neg 11 \vee \neg 13) \wedge (\neg 11 \vee \neg 14) \ldots \wedge (\neg 43 \vee \neg 44)$$

ist eine äquivalente CNF von

$$\neg(11 \wedge 12) \wedge \neg(11 \wedge 13) \wedge \neg(11 \wedge 14) \ldots \wedge \neg(43 \wedge 44)$$

- **Beobachtung**
  Es gibt Algorithmen, die jede gegebene Formel in CNF überführen.

SAT-Solver

- Es gibt eine ganze Anzahl von frei verfügbaren SAT-Solvern, e.g.
  - MiniSAT
  - RSAT
  - riss
- Sie lösen Probleme in CNF mit bis zu zu 10^7 Atomen und 10^8 Konjunktionen
- Sie werden industriell eingesetzt
- Es gibt jährliche internationale Wettbewerbe für die schnellsten SAT-Solver
- SAT-Solver werden laufend verbessert
- Die Verbesserungen umfassen die gesamte Informatik
  von der Theorie und den Anwendungen bis zur Hardware

Lösung des Graphfärbungsproblems mit naiver Kodierung

- **Eingabe an SAT-Solver**

```
p cnf nv nc
11 12 13 14 0
:
41 42 43 44 0
-11 -12 0
:
-43 -44 0
-11 -21 0
:
-34 -44 0
```

wobei nv und nc die Anzahl der Atome (variables) bzw. Konjunkte (clauses) sind

- **Mögliche Ausgabe des SAT-Solvers** 11 22 33 44

Hamiltonkreisprobleme

- Ein **Hamiltonkreis** ist ein geschlossener Pfad in einem Graphen, der jeden Knoten genau einmal erhält
- Das **Hamiltonkreisproblem** ist die Frage, ob ein solcher Kreis in einem gegebenen Graphen existiert
  - Es ist ein fundamentales Problem der Graphentheorie
  - Es ist NP-vollständig

Übungsaufgabe

- Schreiben Sie ein Programm, das nacheinander
  - Hamiltonkreisprobleme von einer Datei einliest,
  - die Frage, ob das Problem eine Lösung hat, in ein SAT-Problem transformiert,
  - wenn das SAT-Problem lösbar ist, dann den gefundenen Zyklus nach stdout schreibt und mit dem Exitcode 10 terminiert,
  - wenn des SAT-Problem unlösbar ist, dann mit dem Exitcode 20 terminiert.
- Dabei können Sie existierende Verfahren für die Transformation in CNF sowie existierende SAT-Solver nutzen
- Eingereichte Lösungen werden automatisch auf ihre Korrektheit überprüft

Wettbewerb

- Eine Folge von Hamiltonkreisproblemen muss in einer bestimmten Zeit betrachtet werden
- Die gemessene Zeit ist die Zeit vom Einlesen des Problems bis zum Schreiben der Lösung bzw. von unerfüllbar
- Für jedes Team t von höchsten 3 Studierenden berechnen wir

  $C(t)$ die Anzahl der korrekt gelösten Aufgaben,

  $T(t)$ die Zeit, die zur Lösung dieser Aufgaben verbraucht wurde, und

  $F(t)$ die Anzahl der eingereichten falschen Lösungen

- Die Teams werden gemäß der folgenden Relation angeordnet:

$$\begin{aligned} s \succ t \quad \text{gdw} \quad & F(s) < F(t) \\ & \vee (F(s) = F(t) \wedge C(s) > C(t)) \\ & \vee (F(s) = F(t) \wedge C(s) = C(t) \wedge T(s) < T(t)) \end{aligned}$$

- Der Wettbewerb findet am 6. Juli 2015 statt
- Das beste Team erhält einen Preis

Sonstige Informationen

- **Die Folien, Infos zum Wettbewerb, etc. finden Sie im Internet unter**
  `http://www.wv.inf.tu-dresden.de/Teaching/SS-2015/`
- **Wir bieten im laufenden Sommersemester eine Vorlesung zu SAT Solving an.**