Logik und Logikprogrammierung
8. Vorlesung

Dietrich Kuske

FG Automaten und Logik, TU Ilmenau

Wintersemester 2022/23

Tseitin-Konstruktion

Eine Formel ist in 3-KNF, wenn sie Konjunktion von Klauseln ist, die jeweils aus höchstens drei Literalen bestehen (z.B. $(p \vee \bot \vee q) \wedge (p \vee \neg q) \wedge (\neg p \vee q \vee \neg r) \wedge (\neg q \vee r)$).

Satz (Grigori Tseitin 1936-)

Aus einer Formel γ kann in polynomieller Zeit eine Formel γ' in 3-KNF berechnet werden, so daß für jede Belegung $\mathcal{B}$ gilt: $\mathcal{B}(\gamma) = 1$ genau dann, wenn es eine Belegung $\mathcal{B}'$ gibt mit

- $\mathcal{B}'(\gamma') = 1$ und
- $\mathcal{B}(p) = \mathcal{B}'(p)$ für alle atomaren Formeln p aus γ.

Insbesondere: Ist Γ endliche Formelmenge, so kann in polynomieller Zeit Klauselmenge Γ' berechnet werden mit

$$\Gamma \text{ unerfüllbar} \iff \Gamma' \text{ unerfüllbar} \iff \Gamma' \vdash_{\mathrm{Res}} \Box\,.$$

Beweis: Wir zeigen die sog. Tseitin-Konstruktion am Beispiel

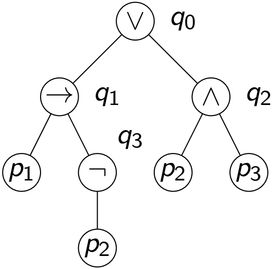
$$\gamma = (p_1 \to \neg p_2) \vee (p_2 \wedge p_3)\,.$$

1. Schritt:
Betrachte die Formel γ als Baum, dessen innere Knoten mit den Operatoren $\wedge$, $\vee$, $\to$, $\leftrightarrow$ und $\neg$ und dessen Blätter mit atomaren Formeln p_i beschriftet sind.

Beispiel

$$
\begin{forest}
[$\vee$ [$\to$ [p_1] [$\neg$ [p_2]]] [$\wedge$ [p_2] [p_3]]]
\end{forest}
$$

2. Schritt:
Ordne jedem inneren Knoten eine neue atomare Formeln $q_0, q_1, q_2, \ldots$ zu.
Der Wurzel wird q_0 zugeordnet.



3. Schritt:
für jeden inneren Knoten bilden wir eine Formel und betrachten deren Konjunktion (zusammen mit der Formel q_0):

- ist q_i die Beschriftung eines inneren Knotens mit Operator $\neg$ und u die des Kindes, so betrachte Formel $q_i \leftrightarrow \neg u$
- ist q_i die Beschriftung eines inneren Knotens mit binärem Operator $\circ$ und sind u, v die Beschriftungen der Kinder, so betrachte Formel $q_i \leftrightarrow u \circ v$

In unserem Beispiel ergibt sich:

$$\big(q_0 \leftrightarrow (q_1 \vee q_2)\big) \wedge \big(q_1 \leftrightarrow (p_1 \to q_3)\big) \wedge \big(q_2 \leftrightarrow (p_2 \wedge p_3)\big) \wedge \big(q_3 \leftrightarrow \neg p_2\big) \wedge q_0$$

Beachte: die so konstruierte Formel ist erfüllbar gdw. γ erfüllbar

4. Schritt:
Forme diese Formel in die verlangte 3-KNF γ' um:

$$
\begin{aligned}
q \leftrightarrow (u \vee v) &\equiv (\neg q \vee u \vee v) \wedge (\neg(u \vee v) \vee q) \\
&\equiv (\neg q \vee u \vee v) \wedge \big((\neg u \wedge \neg v) \vee q\big) \\
&\equiv (\neg q \vee u \vee v) \wedge (\neg u \vee q) \wedge (\neg v \vee q) \\
&\qquad\qquad \text{(Distributivgesetz!)}
\end{aligned}
$$

Analog: $q \leftrightarrow (u \wedge v) \equiv (\neg q \vee u) \wedge (\neg q \vee v) \wedge (\neg u \vee \neg v \vee q)$,
$q \leftrightarrow (u \rightarrow v) \equiv (\neg q \vee \neg u \vee v) \wedge (q \vee u) \wedge (q \vee \neg v)$ und
$q \leftrightarrow \neg z \equiv (\neg q \vee \neg z) \wedge (q \vee z)$

In unserem Beispiel ergibt sich

$$
\begin{aligned}
\gamma' = \quad & (\neg q_0 \vee q_1 \vee q_2) \wedge (\neg q_1 \vee q_0) \wedge (\neg q_2 \vee q_0) \\
\wedge \quad & (\neg q_1 \vee \neg p_1 \vee q_3) \wedge (q_1 \vee p_1) \wedge (q_1 \vee \neg q_3) \\
\wedge \quad & (\neg q_2 \vee p_2) \wedge (\neg q_2 \vee p_3) \wedge (\neg p_2 \vee \neg p_3 \vee q_2) \\
\wedge \quad & (\neg q_3 \vee \neg p_2) \wedge (q_3 \vee p_2) \\
\wedge \quad & q_0
\end{aligned}
$$

Damit erhält man die gesuchte Formel γ'. Außerdem wurden alle Umformungsschritte mit nur polynomialem Aufwand durchgeführt. $\square$

Die Anwendung der Tseitin-Konstruktion auf die Formelmenge Γ von Folie 6.21 ergibt die Menge der folgenden Klauseln:

- $\{AK, BK\}$ entsteht aus $AK \vee BK$ (ohne Tseitin)
- aus $AK \to BK$ wird die 3-KNF

  $$q_0 \wedge (\neg q_0 \vee \neg AK \vee BK) \wedge (q_0 \vee AK) \wedge (q_0 \vee \neg BK)$$

  und damit die Klauseln

  $$\{q_0\},\ \{\neg q_0, \neg AK, BK\}, \{q_0, AK\}, \{q_0, \neg BK\}$$

- aus $(BK \wedge RL) \to \neg AK$ entstehen die Klauseln $\{q_3\}$, $\{\neg q_0, \neg q_1, q_2\}, \{q_0, q_1\}, \{q_0, \neg q_2\}$, $\{\neg q_1, BK\}, \{\neg q_1, RL\}, \{\neg BK, \neg RL, q_1\}$, $\{\neg q_2, \neg AK\}, \{q_2, AK\}$
- aus RL wird die Klausel $\{RL\}$
- aus $\neg\neg AK$ werden die Klauseln $\{q_5\}$, $\{\neg q_5, \neg q_4\}, \{q_5, q_4\}$, $\{\neg q_4, \neg AK\}, \{q_4, AK\}$

Eingabe für logictools.org:

```
  (AK V BK)
& q0 & (-q0 V -AK V BK) & (q0 V AK) & (q0 V -BK)
& q3 & (-q0 V -q1 V q2) & (q0 V q1) & (q0 V -q2)
     & (-q1 V BK) & (-q1 V RL) & (-BK V -RL V q1)
     & (-q2 V -AK) & (q2 V AK)
& RL
& q5 & (-q5 V -q4) & (q5 V q4) & (-q4 V -AK) & (q4 V AK)
```

Aufruf von `Solve using Resolution: better showing html trace` liefert $\Gamma' \vdash_{\text{Res}} \Box$.

Daher ist Γ' (und somit Γ) unerfüllbar. Also haben wir erneut (und jetzt wirklich zum letzten Mal) nachgewiesen, daß unter den genannten Bedingungen das Teil A heil ist.

Zusammenfassung 8. Vorlesung

in dieser Vorlesung neu

- Tseitin-Konstruktion: schnelle Berechnung einer Formel γ' in KNF aus einer Formel γ, so daß γ' erfüllbar gdw. γ erfüllbar (verwendet für Anwendung der Resolution zur Entscheidung „Gilt $\Gamma \models \varphi$?“ bzw. „Ist Γ unerfüllbar?“)

kommende Vorlesung

- Prädikatenlogik, um über „Strukturen“ (z.B. relationale Datenbanken) reden zu können.

Zusammenfassung Aussagenlogik

- aussagenlogische Formeln formalisieren aus atomaren Aussagen gebildete zusammengesetzte Aussagen
- die Regeln des natürlichen Schließens formalisieren die üblichen Argumentationsmuster
- das natürliche Schließen ist vollständig und korrekt für die Feststellung, ob eine Folgerung gilt.
- Endlichkeitssatz
- das Tableau-Verfahren ist vollständig und korrekt für die Feststellung, ob eine Formel erfüllbar ist
- die Resolution ist vollständig und korrekt für die Feststellung, ob eine Menge von Klauseln erfüllbar ist
- die Horn-Resolution ist vollständig und korrekt für die Feststellung, ob eine Menge von Horn-Klauseln erfüllbar ist, ihre Einschränkung auf die SLD-Resolution bildet die Grundlage der logischen Programmierung