Vorlesung “Logik”

10-201-2108-1

4. Hornformeln und Resolution

Ringo Baumann
Professur für Formale Argumentation
und Logisches Schließen

08. Mai 2025
Leipzig

DRESDEN LEIPZIG

In der letzten Vorlesung

Folgerung

Deduktionstheorem

Semantische Äquivalenz

Ersetzungstheorem

DNF und KNF

Fahrplan für diese Vorlesung

Wiederholung: Erfüllbarkeit

Hornformeln

Resolution

Wiederholung - DNF

- Erfüllbarkeitsproblem für DNF effizient lösbar
- Aber! Konstruktion einer sem. äqu. DNF via Wahrheitstabelle im Zweifel exponentiell (2^n Zeilen/Disjunkte)

A_1	A_2	A_3	$(A_1 \vee A_2) \to A_3$
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$$\begin{aligned}\phi_D = {} & (\neg A_1 \wedge \neg A_2 \wedge \neg A_3) \vee (\neg A_1 \wedge \neg A_2 \wedge A_3) \vee \\ & (\neg A_1 \wedge A_2 \wedge A_3) \vee (A_1 \wedge \neg A_2 \wedge A_3) \vee (A_1 \wedge A_2 \wedge A_3)\end{aligned}$$

Wiederholung - DNF

- Erfüllbarkeitsproblem für DNF effizient lösbar
- Aber! Konstruktion einer sem. äqu. DNF via Wahrheitstabelle im Zweifel exponentiell (2^n Zeilen/Disjunkte)
- Gibt es eine effizientere Konstruktionsmethode?

  Anwort: Nein! (Håstad, 1986)

  Beweis über n-stellige Paritätsfunktion $A_1 \dot{\vee} \dots \dot{\vee} A_n$

  (Verallg. ausschließendes Oder)

  Jede sem. äqu. DNF erfordert exp. Anzahl an Disjunkten.

  (gilt analog für KNF)
- Idee: Semantische Äquivalenz ist eine zu starke Forderung, sogenannte Erfüllbarkeitsäquivalenz reicht aus.

…dazu später mehr

Hornformeln

- benannt nach Alfred Horn (1918 – 2001)
- Teilklasse von Formeln für die Erfüllbarkeitsproblem effizient lösbar

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$, und (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

Drei Fälle:

$\neg A_1 \vee \ldots \vee \neg A_n \vee A_{n+1}$ (genau 1 positives Literal)

A_{n+1} (nur 1 positives Literal)

$\neg A_1 \vee \ldots \vee \neg A_n$ (kein positives Literal)

Hornformeln

- benannt nach Alfred Horn (1918 – 2001)
- Teilklasse von Formeln, für die das Erfüllbarkeitsproblem effizient lösbar ist

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

In Implikationsform:

$$
\begin{aligned}
\neg A_1 \vee \ldots \vee \neg A_n \vee A_{n+1} &\equiv A_1 \wedge \ldots \wedge A_n \to A_{n+1} \\
A_{n+1} &\equiv \top \to A_{n+1} \\
\neg A_1 \vee \ldots \vee \neg A_n &\equiv A_1 \wedge \ldots \wedge A_n \to \bot
\end{aligned}
$$

Hornformeln

- benannt nach Alfred Horn (1918 – 2001)
- Teilklasse von Formeln, für die das Erfüllbarkeitsproblem effizient lösbar ist

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

In Implikationsform (übliche Notation):

$$
\begin{aligned}
\neg A_1 \vee \ldots \vee \neg A_n \vee A_{n+1} &\equiv A_1 \wedge \ldots \wedge A_n \rightarrow A_{n+1} \\
A_{n+1} &\equiv 1 \rightarrow A_{n+1} \\
\neg A_1 \vee \ldots \vee \neg A_n &\equiv A_1 \wedge \ldots \wedge A_n \rightarrow 0
\end{aligned}
$$

Hornformeln

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

Bemerkungen:

- Nicht alle Formeln sind Hornformeln. Welche sind keine?

$$A_1 \land A_2, \quad A_1 \lor A_2, \quad (\neg A_1 \land \neg A_2) \lor A_3$$

Hornformeln

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

Bemerkungen:

- Nicht alle Formeln sind Hornformeln. Welche sind keine?

$$A_1 \wedge A_2, \quad A_1 \vee A_2, \quad \cancel{(\neg A_1 \wedge \neg A_2) \vee A_3}$$

(nicht in KNF)

Hornformeln

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

Bemerkungen:

- Nicht alle Formeln sind Hornformeln. Welche sind keine?

$$A_1 \wedge A_2, \quad \cancel{A_1 \vee A_2}, \quad \cancel{(\neg A_1 \wedge \neg A_2) \vee A_3}$$

(2 pos. Lit.) (nicht in KNF)

Hornformeln

Definition

Eine Formel $\phi \in \mathcal{F}$ ist eine Hornformel, sofern:

1. $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ (in KNF)
2. jedes Konjunkt $\bigvee_{j=1}^{m_i} L_{ij}$ besitzt maximal ein positives Literal

Bemerkungen:

- Nicht alle Formeln sind Hornformeln. Welche sind keine?

$$A_1 \wedge A_2, \quad \cancel{A_1 \vee A_2}, \quad \cancel{(\neg A_1 \wedge \neg A_2) \vee A_3}$$

- Aber! $(\neg A_1 \wedge \neg A_2) \vee A_3 \equiv (\neg A_1 \vee A_3) \wedge (\neg A_2 \vee A_3)$
  (sem. äqu. zu Hornformel)

- Kann $A_1 \vee A_2$ auch transformiert werden?

Schnitteigenschaft der Modelle

Mengenschreibweise von Interpretationen:

Jede Interpretation $I : \mathcal{A} \to \{0, 1\}$ kann eindeutig mit einer Menge $M_I = \{A \in \mathcal{A} \mid I(A) = 1\}$ identifiziert werden.

Definition

Eine Formel $\phi \in \mathcal{F}$ besitzt die Schnitteigenschaft (der Modelle), sofern für alle $M, M' \in Mod(\phi)$ gilt: $M \cap M' \in Mod(\phi)$.

Proposition

Jede Hornformel ϕ besitzt die Schnitteigenschaft.

Beweis: Übung 3

Beispiel: $\phi = A_1 \vee A_2$ Was können wir folgern?

Da $\{A_1\} \cap \{A_2\} = \emptyset \notin Mod(\phi)$ kann ϕ nicht zu einer Hornformel semantisch äquivalent sein.

Schnitteigenschaft der Modelle

Mengenschreibweise von Interpretationen:

Jede Interpretation $I : \mathcal{A} \to \{0, 1\}$ kann eindeutig mit einer Menge $M_I = \{A \in \mathcal{A} \mid I(A) = 1\}$ identifiziert werden.

Definition

Eine Formel $\phi \in \mathcal{F}$ besitzt die Schnitteigenschaft (der Modelle), sofern für alle $M, M' \in Mod(\phi)$ gilt: $M \cap M' \in Mod(\phi)$.

Proposition

Jede Hornformel ϕ besitzt die Schnitteigenschaft.

Theorem (Horn, 1951)

Eine Formel ϕ ist semantisch äquivalent zu einer Hornformel genau dann, wenn ϕ die Schnitteigenschaft besitzt.

Markierungsalgorithmus

ist ein effizienter Erfüllbarkeitstest für Hornformeln.

Eingabe: Hornformel ϕ in Implikationsform

Ausgabe: $\subseteq$-kleinstes Modell von ϕ oder unerfüllbar

Ablauf:

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \rightarrow A_3) \wedge (1 \rightarrow A_2) \wedge (A_2 \wedge A_3 \rightarrow 0) \wedge (A_3 \rightarrow A_4) \wedge (A_2 \rightarrow A_1)$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \rightarrow A_3) \wedge \underline{(1 \rightarrow A_2)} \wedge (A_2 \wedge A_3 \rightarrow 0) \wedge (A_3 \rightarrow A_4) \wedge (A_2 \rightarrow A_1)$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \to A_3) \wedge (1 \to A_2) \wedge (A_2 \wedge A_3 \to 0) \wedge (A_3 \to A_4) \wedge \underline{(A_2 \to A_1)}$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \to A_3) \wedge (1 \to A_2) \wedge (A_2 \wedge A_3 \to 0) \wedge (A_3 \to A_4) \wedge \underline{(A_2 \to A_1)}$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$\underline{(A_1 \wedge A_2 \to A_3)} \wedge (1 \to A_2) \wedge (A_2 \wedge A_3 \to 0) \wedge (A_3 \to A_4) \wedge (A_2 \to A_1)$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$\underline{(A_1 \wedge A_2 \rightarrow A_3)} \wedge (1 \rightarrow A_2) \wedge (A_2 \wedge A_3 \rightarrow 0) \wedge (A_3 \rightarrow A_4) \wedge (A_2 \rightarrow A_1)$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \rightarrow A_3) \wedge (1 \rightarrow A_2) \wedge \underline{(A_2 \wedge A_3 \rightarrow 0)} \wedge (A_3 \rightarrow A_4) \wedge (A_2 \rightarrow A_1)$$

Markierungsalgorithmus

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Beispiel:

$$(A_1 \wedge A_2 \rightarrow A_3) \wedge (1 \rightarrow A_2) \wedge \underline{(A_2 \wedge A_3 \rightarrow 0)} \wedge (A_3 \rightarrow A_4) \wedge (A_2 \rightarrow A_1)$$

unerfüllbar

Hörsaalaufgabe

Überprüfen Sie mithilfe des Markierungsalgorithmus die Erfüllbarkeit der folgenden Hornormel. Bei Fragen konsultieren Sie die Person rechts oder links von Ihnen. (2 Min)

$$(A_1 \wedge A_2 \to A_3) \wedge (1 \to A_1) \wedge (A_2 \wedge A_3 \to 0) \wedge (A_4 \to A_3) \wedge (1 \to A_4)$$

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Hörsaalaufgabe

Überprüfen Sie mithilfe des Markierungsalgorithmus die Erfüllbarkeit der folgenden Hornformel. Bei Fragen konsultieren Sie die Person rechts oder links von Ihnen. (2 Min)

$$(A_1 \wedge A_2 \rightarrow A_3) \wedge (1 \rightarrow A_1) \wedge (A_2 \wedge A_3 \rightarrow 0) \wedge (A_4 \rightarrow A_3) \wedge (1 \rightarrow A_4)$$

$$\{A_1, A_3, A_4\}$$

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Beweis:

- Nach spätestens $|s(\phi)|$-Schritten sind alle Atome markiert.
- Zeige, für jedes $M \in Mod(\phi)$: $A \in M$ für jedes markierte A.
  vollständige Induktion über Anzahl der Markierungsschritte
  - für 0 Schritte ist $A \in M$ für jedes markierte A erfüllt
  - für Schritte der Art 1 werden Atome A mit $1 \rightarrow A$ markiert. Da $M \in Mod(\phi)$ muß auch $M \in Mod(1 \rightarrow A)$, also insbesondere $A \in M$.

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Beweis:

- Nach spätestens $|s(\phi)|$-Schritten sind alle Atome markiert.
- Zeige, für jedes $M \in Mod(\phi)$: $A \in M$ für jedes markierte A. vollständige Induktion über Anzahl der Markierungsschritte
  - für 0 Schritte ist $A \in M$ für jedes markierte A erfüllt
  - für Schritte der Art 2 werden Atome B mit $A_1 \wedge \ldots \wedge A_n \rightarrow B$ markiert, wobei $A_1, \ldots, A_n$ schon markiert. Nach IV gilt $\{A_1, \ldots, A_n\} \subseteq M$ für alle $M \in Mod(\phi)$. Somit nach Semantik der Implikation auch $B \in M$.

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Beweis:

- Nach spätestens $|s(\phi)|$-Schritten sind alle Atome markiert.
- Zeige, für jedes $M \in Mod(\phi)$: $A \in M$ für jedes markierte A.
- Falls Ausgabe unerfüllbar, dann $B = 0$ markiert, für $A_1 \wedge \ldots \wedge A_n \to B$ mit schon markierten $A_1, \ldots, A_n$. Aufgrund obigen Satzes wäre mit $M \in Mod(\phi)$ auch $\{A_1, \ldots, A_n\} \subseteq M$ und somit aber $M \notin Mod(A_1 \wedge \ldots \wedge A_n \to B)$. W!

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \rightarrow A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \rightarrow B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Beweis:

- Nach spätestens $|s(\phi)|$-Schritten sind alle Atome markiert.
- Zeige, für jedes $M \in Mod(\phi)$: $A \in M$ für jedes markierte A.
- Falls Ausgabe M, dann
  - für Vorkommen $1 \rightarrow A$ in ϕ ist nach 1,3 $A \in M$. Also, $M \in Mod(1 \rightarrow A)$
  - für $A_1 \wedge \ldots \wedge A_n \rightarrow 0$ ist nach 2 mindestens ein A_i nicht markiert. Nach 3: $A_i \notin M$, d.h. $M \in Mod(A_1 \wedge \ldots \wedge A_n \rightarrow 0)$

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Beweis:

- Nach spätestens $|s(\phi)|$-Schritten sind alle Atome markiert.
- Zeige, für jedes $M \in Mod(\phi)$: $A \in M$ für jedes markierte A.
- Falls Ausgabe M, dann
  - für Vorkommen $1 \to A$ in ϕ ist nach 1,3 $A \in M$. Also, $M \in Mod(1 \to A)$
  - für $A_1 \wedge \ldots \wedge A_n \to B$ mit $B \neq 0$. Falls $\{A_1, \ldots, A_n\} \subseteq M$, dann nach 2,3 $B \in M$. Somit $M \in Mod(A_1 \wedge \ldots \wedge A_n \to B)$. Falls $\{A_1, \ldots, A_n\} \nsubseteq M$, dann trivialerweise Modell

Eigenschaften

1. Markiere jedes Vorkommen von A für Implikationen $1 \to A$
2. Wiederhole:
   - Markiere jedes Vorkommen von B für Implikationen $A_1 \wedge \ldots \wedge A_n \to B$ wobei $A_1, \ldots, A_n$ schon markiert
   - Falls ein $B = 0$ markiert, gib unerfüllbar aus und stoppe
3. Andernfalls: Gib $M = \{A \mid A \text{ wurde markiert}\}$ aus

Theorem

Markierungsalgorithmus terminiert und ist korrekt.

Anmerkungen:

- bei geeigneten Implementierung läuft Algorithmus in Linearzeit
- Ausgabe M ist sogar $\subseteq$-kleinstes Modell
- Programmiersprache Prolog basiert auf Hornformeln
- Expertensystem MYCIN zur Diagnose und Therapie von Infektionskrankheiten (70er Jahre)

Resolutionsverfahren

- eingeführt 1965 von John Alan Robinson
- Verfahren zum Testen auf Unerfüllbarkeit (bzw. Erfüllbarkeit)
- benötigt KNF als Eingabe
- Idee: Implementiere

$$(\phi \vee A) \wedge (\psi \vee \neg A) \vDash \phi \vee \psi$$

  als rein syntaktische Regel
- Ziel: Ableitung leerer Klausel zum Nachweis der Unerfüllbarkeit

Input KNF

- Herstellung einer semantisch äquivalenten KNF im Allgemeinen nicht effizient machbar (Paritätsfunktion)
- Aber! für Test reicht Erfüllbarkeitsäquivalenz aus

Definition

Zwei Formeln $\phi, \psi \in \mathcal{F}$ sind erfüllbarkeitsäquivalent sofern:

$$\phi \text{ erfüllbar} \quad \text{gdw.} \quad \psi \text{ erfüllbar}$$

Beispiele:

A_1 und $A_2 \wedge A_3$ sind erfüllbarkeitsäquivalent
A_1 und $A_2 \vee \neg A_2$ sind erfüllbarkeitsäquivalent
A_1 und $A_2 \wedge \neg A_2$ sind es nicht

- Frage: Wieviele Äquivalenzklassen gibt es in Bezug auf Semantische Äquivalenz bzw. Erfüllbarkeitsäquivalenz?

Input KNF

- Herstellung einer semantisch äquivalenten KNF i.A. nicht effizient machbar (Paritätsfunktion)
- Aber! für Test reicht Erfüllbarkeitsäquivalenz aus

Definition

Zwei Formeln $\phi, \psi \in \mathcal{F}$ sind erfüllbarkeitsäquivalent sofern:

$$\phi \text{ erfüllbar} \quad \text{gdw.} \quad \psi \text{ erfüllbar}$$

Beispiele:

A_1 und $A_2 \wedge A_3$ sind erfüllbarkeitsäquivalent
A_1 und $A_2 \vee \neg A_2$ sind erfüllbarkeitsäquivalent
A_1 und $A_2 \wedge \neg A_2$ sind es nicht

- zu jeder Formel existiert erfüllbarkeitsäquivalente KNF, die in polynomieller Zeit hergestellt werden kann (Tseitin-Transformation)

Repräsentation der KNF

Definition

- Eine Klausel ist eine endliche Menge von Literalen. Leere Klausel wird mit $\square$ bezeichnet
- Einer KNF $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ wird Klauselmenge $M(\phi) = \{C_1, \ldots, C_n\}$ zugeordnet, wobei $C_i = \{L_{i1}, \ldots, L_{im_i}\}$

Beispiele:

$$\phi = (A_1 \vee \neg A_2) \wedge (\neg A_1 \vee A_3) \wedge (\neg A_2 \vee \neg A_3 \vee A_4)$$

$$M(\phi) = \{\{A_1, \neg A_2\},\ \{\neg A_1, A_3\},\ \{\neg A_2, \neg A_3, A_4\}\}$$

$$\psi = (\neg A_1 \vee A_2 \vee A_4) \wedge (A_2 \vee \neg A_3) \wedge (\neg A_3 \vee A_2)$$

$$M(\psi) = \{\{\neg A_1, A_2, A_4\},\ \{A_2, \neg A_3\}\}$$

Repräsentation der KNF

Definition

- Eine Klausel ist eine endliche Menge von Literalen. Leere Klausel wird mit $\square$ bezeichnet
- Einer KNF $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ wird Klauselmenge $M(\phi) = \{C_1, \ldots, C_n\}$ zugeordnet, wobei $C_i = \{L_{i1}, \ldots, L_{im_i}\}$

Bestimmen Sie $M(\phi)$:

$$\phi = (A_1 \vee A_1) \wedge (\neg A_1 \vee A_3) \wedge (\neg A_1 \vee A_3 \vee A_4)$$

$$M(\phi) = \{\{A_1\},\ \{\neg A_1, A_3\},\ \{\neg A_1, A_3, A_4\}\}$$

Repräsentation der KNF

Definition

- Eine Klausel ist eine endliche Menge von Literalen. Leere Klausel wird mit $\Box$ bezeichnet
- Einer KNF $\phi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} L_{ij} \right)$ wird Klauselmenge $M(\phi) = \{C_1, \ldots, C_n\}$ zugeordnet, wobei $C_i = \{L_{i1}, \ldots, L_{im_i}\}$

- Umgekehrt kann jede Klauselmenge $M = \{C_1, \ldots, C_n\}$ mit einer KNF $\bigwedge_{i=1}^{n} (\bigvee C_i)$ identifiziert werden, und somit übertragen sich semantische Begriffe wie Erfüllbarkeit

Wichtig Grenzfälle:

- leere Klausel $\Box$ führt zur „leeren Disjunktion“ und wird als unerfüllbar gesetzt (Warum sinnvoll?)
- Somit jede Klauselmenge M mit $\Box \in M$ unerfüllbar
- (eher uninteressant, aber vollständigkeitshalber) führt $M = \varnothing$ zur „leeren Konjunktion“ und ist tautologisch

Resolvente

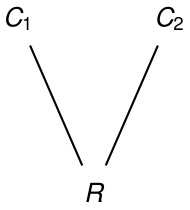
Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \left(C_2 \setminus \{\overline{L}\}\right)$$

(Resolution nach L)

Graphische Darstellung:



Resolvente

Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \left(C_2 \setminus \{\overline{L}\}\right)$$

(Resolution nach L)

Beispiele:

$$\begin{tikzcd}[column sep=small] \{A_1, \neg A_2\} \arrow[dr, dash] & & \{A_2, A_3\} \arrow[dl, dash] \\ & R & \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] C_1 \arrow[dr, dash] & & C_2 \arrow[dl, dash] \\ & R & \end{tikzcd} \qquad \begin{tikzcd}[column sep=small] C_1 \arrow[dr, dash] & & C_2 \arrow[dl, dash] \\ & R & \end{tikzcd}$$

Resolvente

Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \left(C_2 \setminus \{\overline{L}\}\right)$$

(Resolution nach L)

Beispiele:

$\{A_1, \neg A_2\}$ $\{A_2, A_3\}$

$\{A_1, A_3\}$

C_1 C_2

R

C_1 C_2

R

Resolution nach A_2

Resolvente

Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \big(C_2 \setminus \{\overline{L}\}\big)$$

(Resolution nach L)

Beispiele:

$$\begin{array}{ccc} \{A_1, \neg A_2\} & & \{A_2, A_3\} \\ & \{A_1, A_3\} & \end{array} \qquad \begin{array}{ccc} \{A_1, \neg A_2\} & & \{A_2, \neg A_2, \neg A_3\} \\ & R & \end{array} \qquad \begin{array}{ccc} C_1 & & C_2 \\ & R & \end{array}$$

Resolution nach A_2

Resolvente

Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \left(C_2 \setminus \{\overline{L}\}\right)$$

(Resolution nach L)

Beispiele:

$$\begin{tikzcd}[column sep=tiny] \{A_1, \neg A_2\} \arrow[dr, dash] & & \{A_2, A_3\} \arrow[dl, dash] \\ & \{A_1, A_3\} & \end{tikzcd} \qquad \begin{tikzcd}[column sep=tiny] \{A_1, \neg A_2\} \arrow[dr, dash] & & \{A_2, \neg A_2, \neg A_3\} \arrow[dl, dash] \\ & \{A_1, \neg A_2, \neg A_3\} & \end{tikzcd} \qquad \begin{tikzcd}[column sep=tiny] C_1 \arrow[dr, dash] & & C_2 \arrow[dl, dash] \\ & R & \end{tikzcd}$$

Resolution nach A_2

Resolvente

Definition

Seien C_1, C_2 Klauseln. Eine Klausel R heißt Resolvente von C_1 und C_2, falls es ein Literal L gibt, sodass:

$$L \in C_1, \quad \overline{L} \in C_2 \quad \text{und} \quad R = (C_1 \setminus \{L\}) \cup \left(C_2 \setminus \{\overline{L}\}\right)$$

(Resolution nach L)

Beispiele:

$$\begin{tikzcd}[arrows=dash] \{A_1, \neg A_2\} \arrow[dr] & & \{A_2, A_3\} \arrow[dl] \\ & \{A_1, A_3\} & \end{tikzcd}$$

$$\begin{tikzcd}[arrows=dash] \{A_1, \neg A_2\} \arrow[dr] & & \{A_2, \neg A_2, \neg A_3\} \arrow[dl] \\ & \{A_1, \neg A_2, \neg A_3\} & \end{tikzcd}$$

$$\begin{tikzcd}[arrows=dash] \{A_1\} \arrow[dr] & & \{\neg A_1\} \arrow[dl] \\ & \square & \end{tikzcd}$$

Resolution nach A_1

Vorlesung “Logik”

10-201-2108-1

4. Hornformeln und Resolution

Ringo Baumann
Professur für Formale Argumentation
und Logisches Schließen

08. Mai 2025
Leipzig