2 Resolution

Zu grundlegenden Begriffen und Definitionen siehe auch Kapitel 2 (Aussagenlogik) und Abschnitt 4.1 (Kalküle) im Skript der VL "Logik in der Informatik".

- Ein Kalkül über einer Menge M ist eine Menge von Ableitungsregeln $\dfrac{a_1 \quad \cdots \quad a_n}{b}$

- Eine <u>(markierte)</u> Ableitung von a aus $V \subseteq M$ ist eine Folge $((a_1, A_1), \dots, (a_\ell, A_\ell)) \in (M \times 2^M)^\ell$ (← Potenzmenge) mit $a_\ell = a$, $A_i \subseteq \{a_j \mid 1 \leq j \leq i-1\}$ für $1 \leq i \leq \ell$ und für alle $1 \leq i \leq \ell$:

    - $a_i \in V$ und $A_i = \emptyset$ oder
    - $\dfrac{}{a_i}$ ist Axiom und $A_i = \emptyset$ oder
    - $\dfrac{a_{j_1} \cdots a_{j_k}}{a_i}$ ist Ableitungsregel und $A_i = \{a_{j_1}, \dots, a_{j_k}\}$.

▶ Der Ableitungsgraph einer Ableitung $\Gamma = ((a_1, A_1), \ldots, (a_\ell, A_\ell))$ ist ein azyklischer gerichteter Graph G mit $V(G) := \{a_1, \ldots, a_\ell\}$ und

$$E(G) := \{ (a_i, a_j) \mid (a_i, A_i) \in \Gamma, a_j \in A_i \}.$$

- x, y, z bezeichnen aussagenlogische Variablen
- ein Literal λ ist eine Variable x oder deren Negat $\overline{x}$
- eine Klausel C ist eine Menge (Disjunktion) von Literalen
- eine Formel F in konjunktiver Normalform (KNF) ist eine Menge (Konjunktion) von Klauseln

▶ Der Resolutionskalkül über der Menge aller Klauseln besteht nur aus der Resolutionsregel

$$\frac{C \cup \{x\} \qquad D \cup \{\overline{x}\}}{C \cup D}$$

Hierbei ist $C \cup D$ die Resolvente von $C \cup \{x\}$ und $D \cup \{\overline{x}\}$ auf x.

- Eine Resolutionswiderlegung von F ist eine Resolutionsableitung der leeren Klausel $\emptyset$ aus F.

$\boxed{\text{Satz 2.1}}$

$$R_{res} := \{(F, \Gamma) \mid \Gamma \text{ ist Resolutionswiderlegung von } F\}$$

ist ein Beweissystem für UNSAT nach Definition 1.1.

<u>Beweis:</u> Folgt direkt aus Vollständigkeit und Korrektheit (siehe Satz 2.60 in (LogInf)). $\square$

<u>Übung:</u> Geben Sie eine Resolutionswiderlegung von

$$F = \{ \{x_1, x_2\}, \{\overline{x}_1, \overline{x}_2\}, \{x_1, \overline{x}_2\}, \{\overline{x}_1, x_2, y_1, y_2\}, \{\overline{x}_1, x_2, \overline{y}_1, \overline{y}_2\}, \{\overline{y}_1 \vee y_2\}, \{y_1 \vee \overline{y}_2\}, \}$$

an!

Komplexitätsmaße für Resolution

- Die Weite $w(C)$ einer Klausel $C = \{\lambda_1, \ldots, \lambda_k\}$ ist k.
- Die Weite einer KNF-Formel F ist $w(F) := \max_{C \in F} w(C)$.
- Eine k-KNF-Formel F hat $w(F) \leq k$.

Sei $\Gamma = ((C_1, M_1), \ldots, (C_\ell, M_\ell))$ eine Resolutionsableitung.

- Die Weite von Γ ist $w(\Gamma) := \max_{i \in [\ell]} w(C_i)$.
- Die Länge von Γ ist $L(\Gamma) := \ell$. $[\ell] := \{1, \ldots, \ell\}$
- Die Größe von Γ ist $S(\Gamma) := \sum_{i \in [\ell]} w(C_i)$ und proportional zu der Länge einer natürlichen Kodierung von Γ.
- Die Tiefe von Γ ($D(\Gamma)$) ist die Anzahl der Kanten auf dem längsten gerichteten Pfad im Ableitungsgraphen von Γ.

- Γ ist baumartig, wenn der Eingangsgrad aller Resolventen im Ableitungsgraphen höchstens 1 ist. (D.h. der Ableitungsgraph ist ein gerichteter Baum, in dem einige Blätter "zusammengeklebt" wurden.)

Sei $X \in \{w, S, L, D\}$ ein Komplexitätsmaß und F eine unerfüllbare KNF.

Dann ist $X(F \vdash \emptyset) := \min\{X(\Gamma) \mid \Gamma \text{ ist Resolutionswiderlegung von } F\}$

und $X_T(F \vdash \emptyset) := \min\{X(\Gamma) \mid \Gamma \text{ ist baumartige Resolutionswiderlegung von } F\}$.

Lemma 2.2 Für jede unerfüllbare Klauselmenge F mit n Variablen gibt es eine baumartige Resolutionswiderlegung Γ der Weite $w(\Gamma) \leq n$, Tiefe $D(\Gamma) \leq n$ und Länge $L(\Gamma) \leq 2^{n+1} - 1$.

Beweis: Übungsaufgabe.

▶ Oft ist es nützlich die Weakening-Regel $\dfrac{C}{C \cup D}$ dem Resolutionskalkül hinzuzufügen. Das macht Resolution allerdings nicht stärker (Beweis ist Übungsaufgabe):

Lemma 2.3 Für jede Resolutionswiderlegung Γ von F mit Weakening gibt es eine Widerlegung Γ' von F ohne Weakening, sodass

- $L(\Gamma') = L^*(\Gamma) \leq L(\Gamma)$
- $D(\Gamma') = D^*(\Gamma) \leq D(\Gamma)$
- $w(\Gamma') \leq w(\Gamma)$
- wenn Γ baumartig ist, dann auch Γ'

hierbei ist

$L^*(\Gamma) :=$ # Resolventen und Axiome in Γ

$D^*(\Gamma) :=$ maximale Anzahl von Resolventen auf gerichtetem Pfad im Ableitungsgraphen.

2.1. DPLL und baumartige Resolution

- Eine <u>partielle Belegung</u> ist eine partielle Abbildung $\alpha : VAR \to \{0,1\}$ mit Definitionsbereich $dom(\alpha) \subseteq VAR$.
  Für negierte Literale $\overline{x}$ schreiben wir $\alpha(\overline{x})$ als Abkürzung für $1-\alpha(x)$ und $\overline{x} \mapsto b$ für $x \mapsto 1-b$
- Für eine aussagenlogische Formel F und eine partielle Belegung α ist die <u>Einschränkung $F|_\alpha$</u> eine aussagenlogische Formel, die aus F entsteht, indem jede Variable x mit $\alpha(x)=0$ durch $\mathbf{0}$ und jede Variable x mit $\alpha(x)=1$ durch $\mathbf{1}$ ersetzt wird.
- Für Klauselmengen F heißt das:

$$F|_{x=1} := \{ C \setminus \{\overline{x}\} \mid C \in F,\ x \notin C \}$$

$$F|_{x=0} := \{ C \setminus \{x\} \mid C \in F,\ \overline{x} \notin C \}$$

Für $\alpha = \{x_1 \mapsto a_1, \ldots, x_k \mapsto a_k\}$ rekursiv:

$$F|_\alpha := \left(F|_{\alpha \setminus \{x_k \mapsto a_k\}} \right) \Big|_{x_k = a_k}$$

Der DPLL-Algorithmus

- Ein "unit clause" $\{\lambda\}$ ist eine Klausel mit nur einem Literal.
- Ein "pure literal" ist ein Literal, das in einer Formel nur negiert oder nur positiv vorkommt.

DPLL (F, α)

% F ... Formel in KNF (Mengenschreibweise)
% α ... partielle Belegung
% Die Prozedur testet, ob $F|_\alpha$ erfüllbar ist

Falls $\emptyset \in F|_\alpha$:
  return FALSE

Falls $F|_\alpha = \emptyset$ ($\hat{=}$ α ist erfüllende Belegung für F):
  return TRUE

Falls λ pure literal in $F|_\alpha$:
  return DPLL$(F, \alpha \cup \{\lambda \mapsto 1\})$

Falls $\{\lambda\}$ unit clause in $F|_\alpha$:
  return DPLL$(F, \alpha \cup \{\lambda \mapsto 1\})$

Wähle Variable $x \in \mathrm{Vars}(F|_\alpha)$
return DPLL$(F, \alpha \cup \{x \mapsto 0\})$ or DPLL$(F, \alpha \cup \{x \mapsto 1\})$

- Für eine partielle Belegung α sei $C(\alpha)$ die eindeutige Klausel C mit $vars(C) = dom(\alpha)$ und $\alpha \not\models C$.

<u>Beispiel:</u> $C(\{x \mapsto 1, y \mapsto 0, z \mapsto 1\}) = \{\bar{x}, y, \bar{z}\}$

Wir zeigen nun, dass die Ausführungszeit des DPLL-Algorithmus auf einer unerfüllbaren KNF mindestens so groß ist, wie die kleinste baumartige Resolutionswiderlegung.

Lemma 2.4

Wenn DPLL(F,α) nach n rekursiven Aufrufen FALSE zurückgibt, dann gibt es eine baumartige Resolutionsableitung von $C(\alpha)$ aus F mit weakening und höchstens n Resolutionsschritten.

Korollar 2.5

Die Laufzeit von DPLL auf unerfüllbaren Instanzen F (d.h. DPLL$(F,\emptyset)$) ist mindestens

$$L_T(F \vdash \emptyset) := \min\{L(\Gamma) \mid \Gamma \text{ ist baumartige Widerlegung von } F\}.$$

(Folgt direkt aus Lemma 2.4. und Lemma 2.3.)