

**AD4M33AU**  
**Automatické uvažování**

**Sémanticky řízená rezoluce**

**Petr Pudlák**

# Problém binární rezoluce

- Při klasické (někdy též **binární**) rezoluci je mnoho možností, jak odvozovat rezolventy.
- Mnoho různých možností přitom nakonec vede ke stejnému výsledku.
- Rezoluci můžeme omezit ve dvou směrech:
  - Výběr literálů – umožníme provádět rezoluci jen přes některé literály.
  - Výběr klauzulí – pro rezoluci umožníme vybírat jen některé klauzule.
- Tím zmenšíme prohledávací prostor.
- Vyžadujeme aby modifikovaný kalkulus zachovával úplnost!

# Uspořádání literálů při rezoluce

- K odvození prázdné klauzule musíme eliminovat všechny literály.
- Můžeme si tedy zvolit pořadí, v jakém se budeme snažit literály eliminovat.
- Tím omezíme množství odvozených klauzulí z dané množiny.

# Uspořádaná (ordered) rezoluce

- Zvolíme si uspořádání  $P$  na predikátových symbolech.
- Při rezoluci dvou klauzulí smíme provést rezoluci pouze přes takové literály, jejichž predikátový symbol je v dané klauzuli největší vzhledem k  $P$ .
- Toto omezení zachovává úplnost rezolučního kalkulu.
- (Příklad.)

# Sémanticky řízená rezoluce

- **Sémanticky řízená rezoluce** je pojem zahrnující dvě často používané metody nazývané **hyperrezoluce** a **set of support**.
- Omezuje klauzule, které smíme vybírat.

# Principy sémantické rezoluce

1. Je-li  $\mathbf{S}$  nespínitelná množina klauzulí pak v každé interpretaci  $M$  je množina
$$\mathbf{S}' = \{ C \in \mathbf{S} \mid M \not\models C \}$$
neprázdná. Výběr klauzulí při sémanticky řízené rezoluci omezíme tak, aby jedna z klauzulí byla vždy z takové množiny.
2. Využijeme také uspořádanou rezoluci – zvolíme si uspořádání na predikátových symbolech a omezíme výběr literálu pro rezoluci podle tohoto uspořádání.

# Sémanticky řízená rezoluce

- Bud'  $I$  interpretace a  $P$  uspořádání na predikátových symbolech.
- Konečnou množinu  $\{E_1, \dots, E_q, N\}$ ,  $q \geq 1$  nazveme **semantic clash** vzhledem k  $P$  a  $I$  (nebo jen **PI-clash**), právě když platí:
  - $I \not\models E_i$ ,  $1 \leq i \leq q$
  - Existuje posloupnost klauzulí  $R_i$ , taková, že  $R_1 = N$  a  $R_{i+1}$  je resolventou  $R_i$  a  $E_i$ .
  - Predikát literálu  $E_i$ , přes který provádíme rezoluci, je největší vzhledem k  $P$ .
  - $I \not\models R_{q+1}$

# Sémanticky řízená rezoluce

- semantic clash  $\{E_1, \dots, E_q, N\}$ :
  - $I \not\models E_i, 1 \leq i \leq q$
  - Existuje posloupnost klauzulí  $R_i$ , taková, že  $R_1 = N$  a  $R_{i+1}$  je resolventou  $R_i$  a  $E_i$ .
  - Predikát literálu  $E_i$ , přes který provádíme rezoluci, je největší vzhledem k  $P$ .
  - $I \not\models R_{q+1}$
- $E_i$  nazýváme **elektrony**,  $N$  **nukleus** a  $R_{q+1}$  **PI-resolventou**. Posloupnost **PI-resolvent** odvozujících klauzuli  $C$  z množiny klauzulí **S** nazveme **PI-dedukcí**  $C$  z **S**.



# Úplnost sémanticky řízené rezoluce

- Bud'  $S$  konečná nespíitelná množina klauzulí,  $P$  uspořádání predikátových symbolů a  $I$  interpretace.
- Pak existuje  $PI$ -dedukce sporu z  $S$ .

# Příklad sémanticky řízené rezoluce

- Bud'  $\mathbf{S} = \{$ 
  - $\{q(a), r(X)\},$
  - $\{\sim q(X), r(X)\},$
  - $\{\sim r(a), \sim s(a)\},$
  - $\{s(X)\}$ $\}.$
- Uspořádejme predikátové symboly  $r < q < s$ .
- Provedme sémanticky řízenou rezoluci
  - $I \models q(X), r(X), s(X)$
  - $I \models \sim q(X), \sim r(X), \sim s(X)$

# Hyperrezoluce

- Při **hyperrezoluci** volíme interpretaci, ve které jsou pravdivé všechny pozitivní nebo naopak všechny negativní literály.
- Pokud volíme interpretaci, kde jsou všechny literály negativní, jsou všechny elektrony a *PI*-resolventy pozitivní klauzule. Nazýváme **pozitivní hyperrezoluce**.
- Pokud volíme interpretaci, kde jsou všechny literály pozitivní, jsou všechny elektrony a *PI*-resolventy negativní klauzule. Nazýváme **negativní hyperrezoluce**.

# Set of support

- Předpokládejme, že se snažíme dokázat spor z  $\mathbf{S}$  a dopředu víme, že  $\mathbf{A} \subseteq \mathbf{S}$  je splnitelná.
- Buď libovolný  $M \models \mathbf{A}$ . Při použití sémanticky řízené rezoluce na  $M$  a libovolné uspořádání  $P$  bude vždy elektron mimo  $\mathbf{A}$ .
- Pokud tedy binární rezoluci omezíme tak, že alespoň jedna z klauzulí není z  $\mathbf{A}$ , věta o úplnosti sémanticky řízené rezoluce zaručuje, že je-li  $\mathbf{S}$  sporná, odvodíme takto spor.

# Hyperparamodulace

- Poznámka: Existují také rozšíření hyperrezoluce pro rovnost. Například podobné rozšíření paramodulace se nazývá **hyperparamodulace**.

# ANL smyčka

- Poprvé použita v dokazovačích Argonne National Laboratory, z nichž nejznámější je Otter.
- Zajišťuje, že prohledáme všechny možné kombinace aplikace rezoluce (nezávisle na zvolené strategii).

# Popis ANL smyčky

- **čekající** := vstupní klauzule
- ***k\_použití*** :=  $\{\}$
- ***while* (čekající  $\neq \{\}$ ):**
  - vyber klauzuli **C** z **čekající**
  - **čekající** := **čekající**  $\setminus \{C\}$
  - **k\_použití** := **k\_použití**  $\cup \{C\}$
  - **odvozeno** :=  
odvod' všechny resolventy z **C** a **k\_použití**
  - ***if* ( $\square \in$  **odvozeno**):**
    - *return* SPOR
  - **čekající** := **čekající**  $\cup$  **odvozeno**
- *return* SPLNITELNÁ

# Možné způsoby prohledávání

- Některé časté způsoby výběru klauzule:
  - Do hloubky: vybereme poslední rezolventu z **čekajících**.
    - Není to úplné prohledávání (může se zacyklit).
    - Pokud najde spor, nemusí být jeho důkaz nejkratší možný.
  - Do šířky: vybereme první rezolventu z **čekajících**.
    - Úplné prohledávání.
    - Najde nejkratší důkaz sporu (pokud existuje).
    - Implementuje hledání „po discích“.
  - Podle „nejlepší“ klauzule: Vybereme v nějakém smyslu nejlepší rezolventu z **čekajících**, např. nejmenší podle nějakého dobrého uspořádání.



# Popis ANL smyčky se subsumpcí

- **čekající** := vstupní klauzule
- ***k\_použití*** :=  $\{\}$
- ***while*** (**čekající**  $\neq \{\}$ ):
  - vyber klauzuli **C** z **čekající**
  - **čekající** := **čekající**  $\setminus \{C\}$
  - ***if*** (**k\_použití**  $\sqsubseteq \{C\}$  || **čekající**  $\sqsubseteq \{C\}$ )
    - *continue*
  - **k\_použití** :=  $\{ D \in \text{k\_použití} \mid C \not\sqsubseteq D \} \cup \{C\}$
  - ***for*** **D in** (všechny resolventy z **C** a **k\_použití**)
    - ***if*** (**D** ==  $\square$ )
      - *return* SPOR
    - ***else if*** (**k\_použití**  $\not\sqsubseteq \{D\}$ )
      - **čekající** := **čekající**  $\cup \{D\}$
- ***return*** SPLNITELNÁ