



Heuristické optimalizačné procesy



DPLL rodina algoritmov

Marian.Mach@tuke.sk

<http://neuron.tuke.sk/~machm>

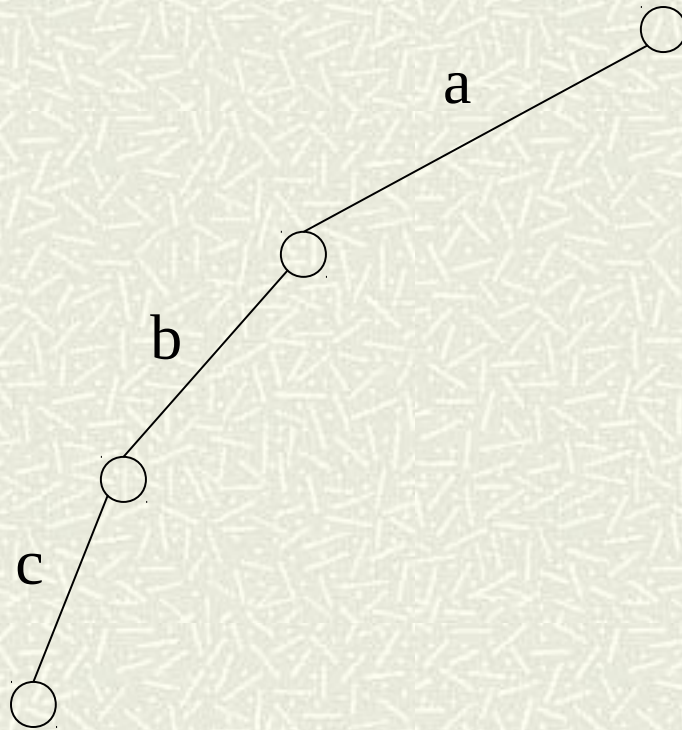
Február, 2013

DPLL algoritmus

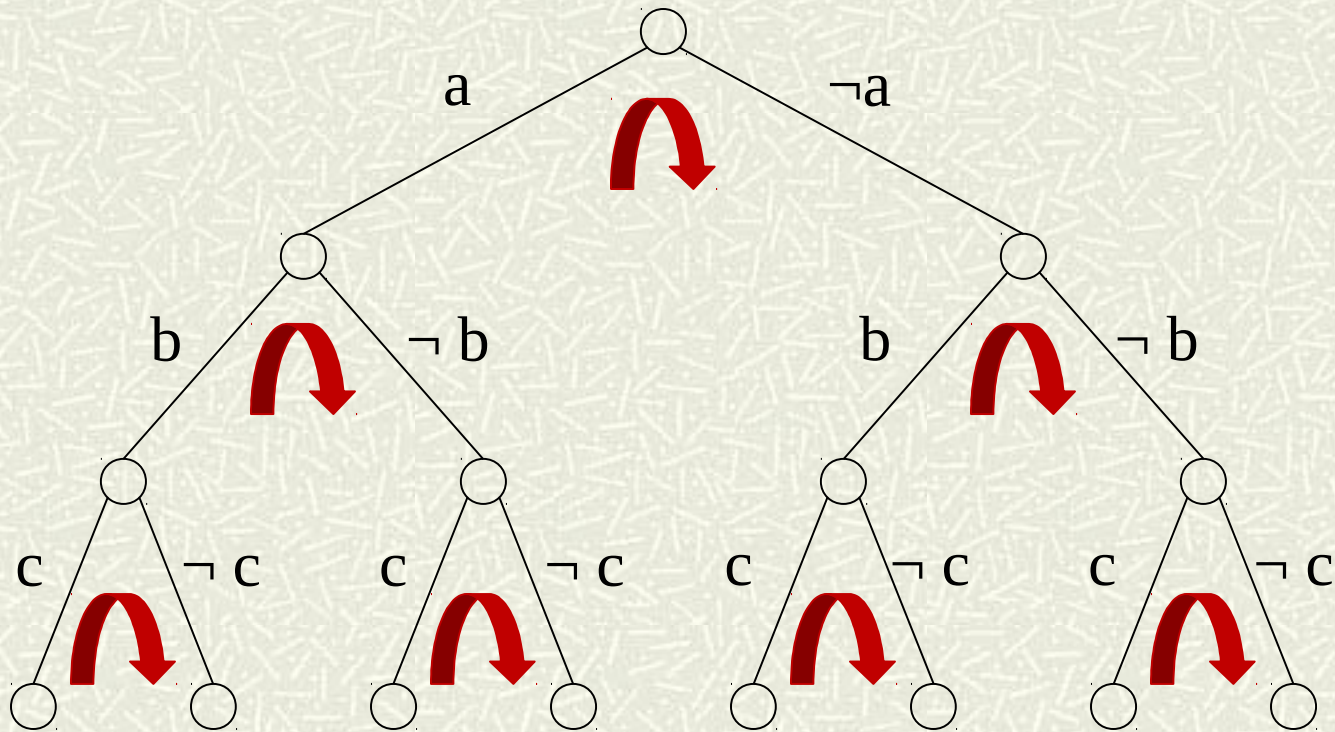
- Vznik
 - 1960 : Davis – Putnam
 - 1962 : Davis – Logemann – Loveland
- zisťuje či CNF je splniteľná a ak áno, tak aký je jej model
- systematický konštrukčný algoritmus

- veľa dnešných SAT solverov je založených na tomto algoritme (s pridanými rozšíreniami)
 - SATO, POSIT, NTAB, MiniMAX

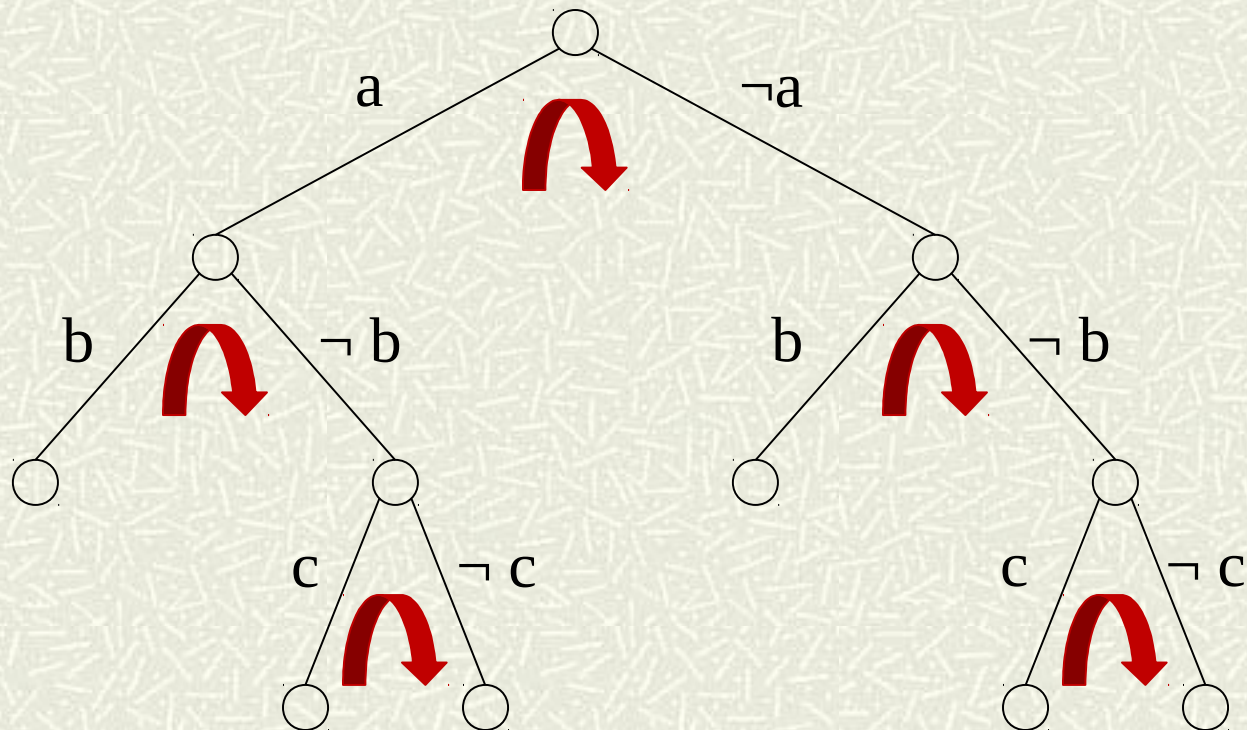
Konštrukcia riešenia



Konštrukcia + navracanie



Konštrukcia + navracanie + orezanie



Štruktúra DPLL algoritmu

- $s = \{ \}$
- while(true)
- $s' = \text{add}(s, \pi)$
- if($s' = s$) then return s
- else
- $\pi' = \text{prune}(\pi)$
- if($\text{conflict}(s', \pi')$)
- then
- $(s, \pi) = \text{backtrack}(s', \pi')$
- if($s = \{ \}$) then return $\{ \}$
- else
- $(s, \pi) = (s', \pi')$
- endif
- endif
- endwhile

Orezávanie – jednotková propagácia

- jednotkový literál – sám vytvára klauzulu
 - umožňuje určiť hodnotu premennej (aby literál bol pravdivý a tým pádom aj klauzula)
 - negácia literálu je nesplnená
- orezávanie
 - každá klauzula, v ktorej vystupuje pravdivý literál, je splnená a možno ju odstrániť
 - nepravdivý literál môže byť odstránený z klauzuly
- ukončenie
 - prázdna klauzula >> nespĺniteľnosť
 - prázdna celková fomula >> splniteľnosť
- príklad: $\{ \neg p \vee q, \neg p \vee \neg q \vee r, p, \neg r \}$

Orezávanie – eliminácia literálu

- čistý literál
 - literál, ktorý v skupine klauzúl má iba jeden tvar (buď priamy alebo negovaný)
 - je možné určiť hodnotu príslušnej premennej (aby literál bol pravdivý)
- orezávanie
 - každá klauzula, v ktorej tento literál vystupuje, bude splnená a možno ju odstrániť
- príklad: $\{ \neg p \vee q, \neg p \vee \neg q \vee r, \neg p, \neg r \}$

DPLL príklad

- 1) $p \vee q \vee r$
- 2) $p \vee \neg q \vee \neg r$
- 3) $p \vee \neg w$
- 4) $\neg q \vee \neg r \vee \neg w$
- 5) $\neg p \vee \neg q \vee r$
- 6) $u \vee x$
- 7) $u \vee \neg x$
- 8) $q \vee \neg u$
- 9) $\neg r \vee \neg u$

Výberová heuristika

- RAND
 - náhodný výber premennej aj hodnoty
- DLIS (*Dynamic Largest Individual Sum*)
 - vyberie sa premenná, ktorej literál má najväčšiu frekvenciu v zostávajúcich klauzulách
 - podľa literálu sa priradí hodnota
- DLCS (*Dynamic Largest Combined Sum*)
 - vyberie sa premenná, ktorá má najväčšiu frekvenciu v zostávajúcich klauzulách
 - hodnota sa priradí podľa toho, v akej forme sa tá premenná častejšie vyskytuje

Výberová heuristika II

- MOM (*Maximum Occurrences on clauses of Minimum size*)

- uvažujú sa iba najkratšie zostávajúce klauzuly
- vyberá sa premenná x maximalizujúca

$$[\#(x) + \#(\neg x)] * 2^k + \#(x) * \#(\neg x)$$

kde $\#(x)$ je počet výskytov premennej v priamej podobe

- hodnota sa priradí podľa toho, či premenná sa vyskytuje častejšie v priamej alebo negovanej podobe

Analýza konfliktov

- pri výskyte konfliktu následkom jednotkovej propagácie sa vykoná analýza tohto konfliktu
 - analyzuje sa štruktúra vykonanej propagácie
 - štartuje sa od nesplnenej klauzuly
 - postupuje sa spätne až po okamihy priradenia hodnôt premenným
 - identifikujú sa nové klauzuly, ktoré v budúcnosti dokážu orezať priestor ešte viac
 - určuje sa bod návratu pre procedúru navracania
 - používajú sa naučené klauzuly

Učenie nových klauzúl

▪ 1. $x_1 \vee x_4$

priradenie: $x_2 = \perp$

▪ 2. $\neg x_4 \vee \neg x_5$

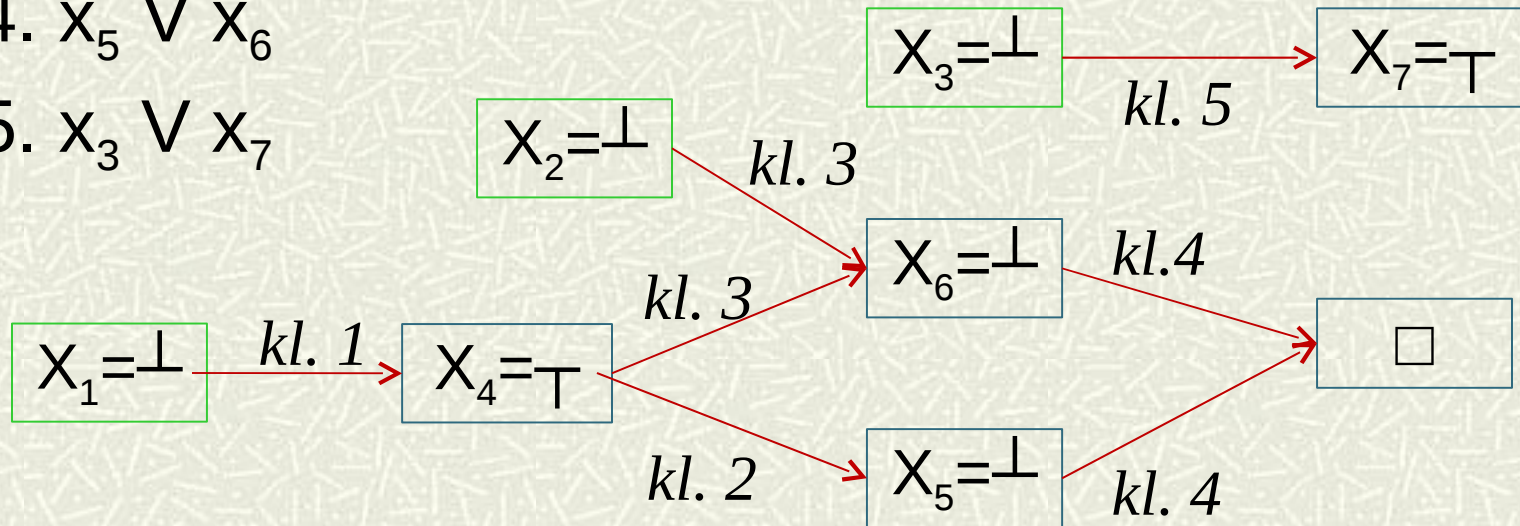
$x_3 = \perp$

▪ 3. $x_2 \vee \neg x_4 \vee \neg x_6$

$x_1 = \perp$

▪ 4. $x_5 \vee x_6$

▪ 5. $x_3 \vee x_7$



Určenie bodu návratu

