

Hladová aproximace SAT

Mějme formuli Φ s n proměnnými, m klauzulemi a celkem $|\Phi|$ literály.

Výsledkem bude ohodnocení proměnných 1 nebo 0 tak, že bude počet splněných klauzulí co možná nejvyšší.

Pro každou proměnnou x , kterých je n , se podíváme, kolikrát se v klauzulích vyskytuje a kolikrát se vyskytuje její negace. Proměnnou pak nastavíme tak, aby splnila co nejvíce klauzulí.

Klauzule, které splníme, pak odebereme (nebudeme je příště procházet, neb jsou již splněné).

Na výstupu dostaneme proměnné ohodnocené tak, že splňují minimálně $\frac{1}{2}$ klauzulí.

Korektnost

Jelikož při každém rozhodování získáme minimálně polovinu klauzulí s danou proměnnou která je vyhodnocena jako pravdivá a ta druhá „menší“ polovina bude vyhodnocována později a má šanci že bude vyhodnocena při rozhodování u jiné proměnné.

V případě, že má klauzule 3 rozdílné literály, pak se formule bude vyhodnocovat 3-krát a tudíž má průměrnou šanci na správné vyhodnocení $1 - 2^{-t}$ kde t je počet literálu v jedné klauzuli = 3/4

Příklad minimální klauzule ($\frac{1}{2}$ resp. $\frac{3}{4}$ násobek optima)

$(a \vee \neg b) \wedge (\neg a)$

Kdy se nejdříve vyhodnotí proměnná a , které se přiřadí *True* ().

Poté se vyhodnotí proměnná b , která ale není zastoupena, tak dostane defaultní *True*.

Pak hladový algoritmus bude mít pouze poloviční úspěšnost.

Časová náročnost

Pro každou proměnnou musíme projít celou formuli a spočítat kolikrát se tam vyskytuje.

Takže procházíme každý literál $|\Phi|$ n -krát $\rightarrow O(n \cdot |\Phi|)$

Paměťová náročnost

Stačí si uchovávat formuli a pro každou proměnnou její hodnotu.

Takže celkem $O(n + |\Phi|)$

Optimalizace

Chtějme vylepšit časovou náročnost. Problém je, že formuli procházíme vícekrát než je třeba.

Takže ji projdeme pouze jednou a pro každou proměnnou si uložíme její výskyty ve formuli. A poté si budeme pouze sahat na tyto odkazy.

Časovou náročnost tedy dokážeme optimalizovat na $O(|\Phi|)$