

# Report - CDCL solver

David Surma

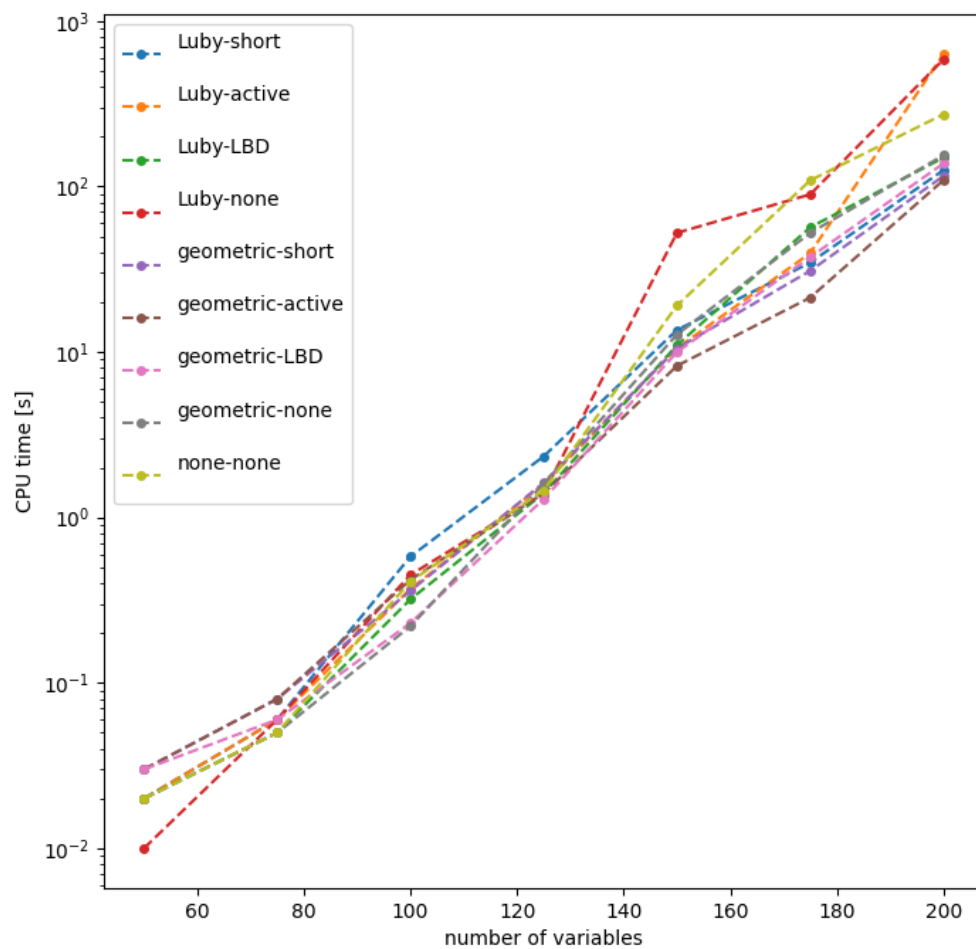
Implementoval jsem klasický CDCL solver, který se učí klauzule obsahující 1-UIP. Volitelnými parametry je typ restartů, mazání klauzulí (pouze při restartu) a heuristiky pro výběr dalšího literálu:

- Pokud je zadán parametr "restart", dochází k restartům vždy po dosažení určitého množství konfliktů, na výběr jsou možnosti "Luby" a "geometric".
- Při restartu může docházet k odstranění některých naučených klauzulí. Implementoval jsem mazání na základě délky klauzulí (argument "short"), počtu různých decision levelů v klauzuli ("LBD") a aktivity měřené počtem jednotkových propagací vycházejících z dané klauzule ("active").
- Kromě náhodného výběru dalšího literálu jsem implementoval heuristiky Jeroslow-Wang, VSIDS a jednoduchou heuristiku vybírající literál obsažený v největším počtu klauzulí ("most common").

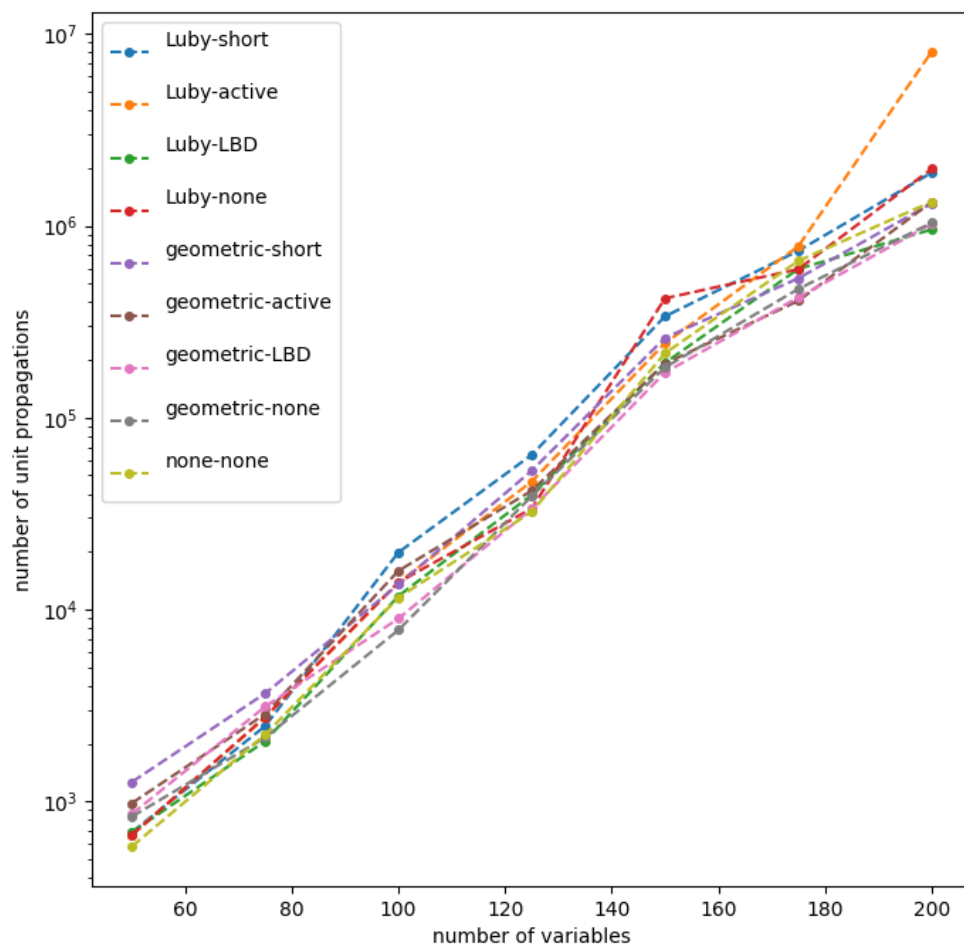
Nejprve jsem experimentoval s různými parametry pro restarty a mazání klauzulí, kvůli urychlení výpočtu jsem při tom používal heuristiku "most common". Některé kombinace parametrů se ukázali jako nevhodné, jiné naopak porazili defaultní běh bez parametrů na takřka všech příkladech, viz. Obrázek 1.

Na zmíněném grafu vidíme, že většina experimentů dopadla podobně, některé kombinace parametrů dopadly lépe na menších instancích, některé naopak. Za povšimnutí stojí výsledky experimentu "Luby-none", který na velkých instancích běžel déle než základní experiment bez parametrů. To může být způsobeno faktem, že nedochází k mazání žádných klauzulí a jejich počet se při častých restartech zvyšuje příliš rychle. Výsledky geometrických restartů dopadly obecně lépe než Luby. Na druhou stranu nelze tyto výsledky přeceňovat, protože experimenty byly provedeny na malém množství instancí a navíc hyperparametry u restartů a mazání (parametry geometrické řady, konstanta pro posloupnost Luby, parametry pro mazání klauzulí) nejsou příliš optimalizovány.

Obrázek 2 ještě zachycuje výsledky stejných experimentů, avšak porovnává počty jednotkových propagací. Oba grafy jsou však hodně podobné.



Obrázek 1: Graf závislosti doby běhu na počtu proměnných v nesplnitelné formuli. Legenda obsahuje použitý typ restartů a mazání klauzulí.

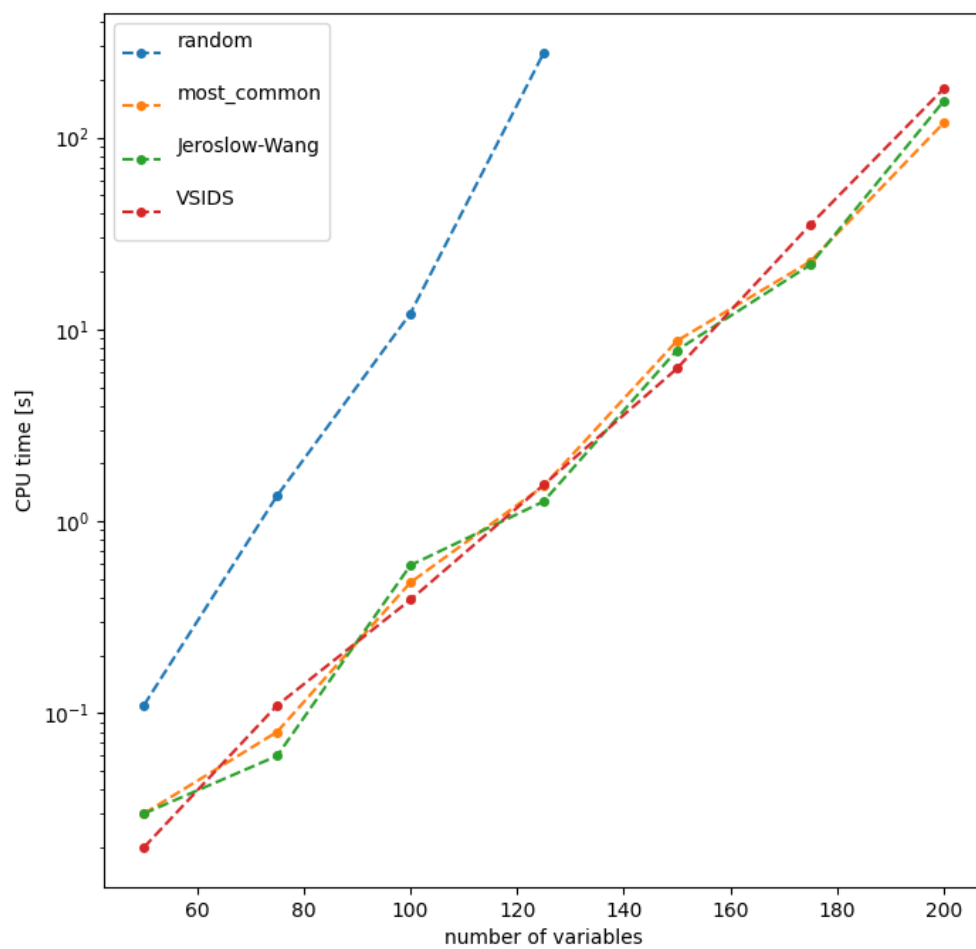


Obrázek 2: Graf závislosti počtu provedených jednotkových propagací na počtu proměnných.

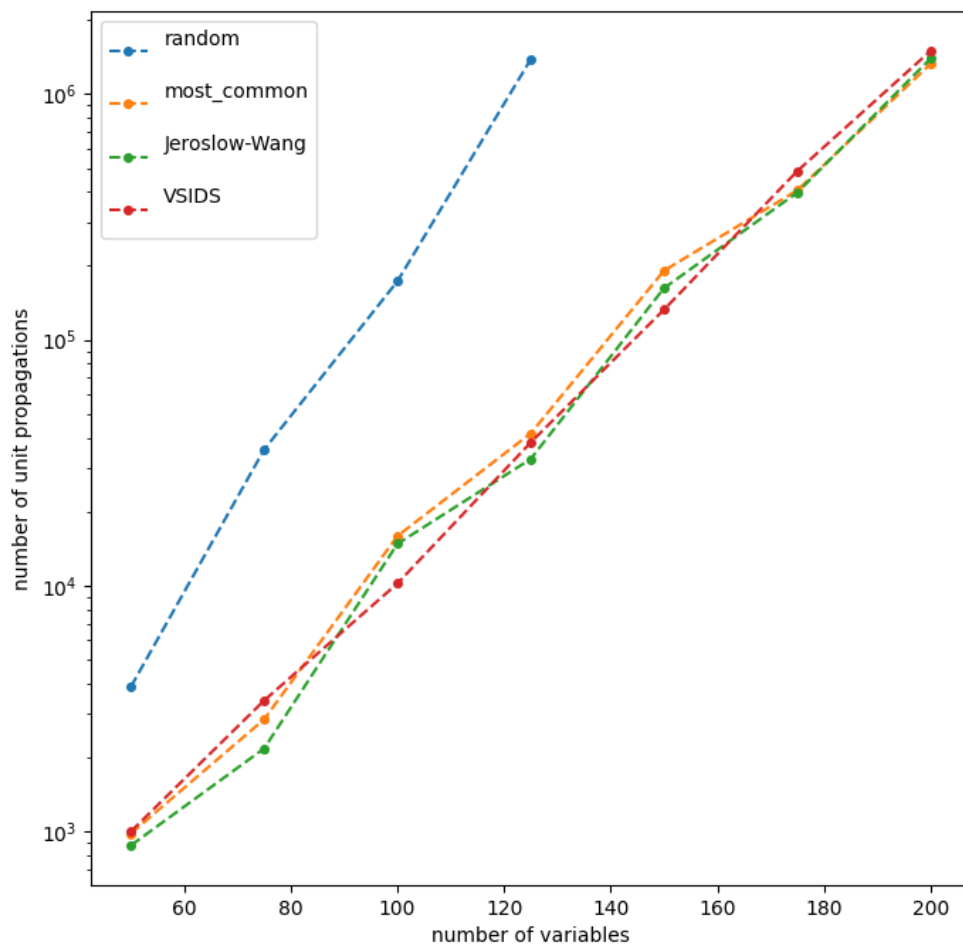
Jako nejlepší parametry pro restarty a mazání se z provedených experimentů jeví kombinace "geometric-active", která dopadla na větších instancích nejlépe co se týče doby běhu a patřila k nejlepším i v počtu provedených jednotkových propagací. Tuto dvojici parametrů jsem poté zvolil jako defaultní pro porovnávání různých rozhodovacích heuristik. Výsledky doby běhů těchto experimentů jsou zachyceny na Obrázku 3.

Na zmíněném grafu vidíme očekávaný výsledek, že náhodný výběr literálu je mnohem horší, než všechny ostatní sofistikovanější přístupy, jejichž výsledky jsou naopak překvapivě hodně podobné. Kupodivu jednoduchá heuristika "most common" dopadla na zkoušených instancích ne hůře než zbývající dvě složitější heuristiky. Pro lepší porovnání heuristik by bylo pravděpodobně potřeba spustit řešiče na větším množství velkých instancí.

Pokud bych měl tedy vybrat nejlepší parametry pro svůj CDCL solver, použil bych geometrické restarty s mazáním naučených klauzulí podle jejich aktivity a k tomu bych přidal heuristiku "Jeroslaw-Wang", která je de facto vylepšením heuristiky "most common", přestože to výsledky experimentů příliš nepotvrdily.



Obrázek 3: Porovnání doby běhu solverů používajících různé rozhodovací heuristiky.



Obrázek 4: Porovnání různých rozhodovacích heuristik z hlediska počtu provedených jednotkových propagací.