

Protokol k semestrální práci

Téma: Implementace algoritmů pro řešení Sudoku pomocí hill climbing

Úvod:

Cílem této semestrální práce bylo implementovat řešení Sudoku pomocí hill climbing. Dle zadání byly implementovány 2 metody měnění kandidáta na řešení. První algoritmus je klasický hill climbing, který v každém kroku vybírá jeden z nejlepších možných tahů. Druhý algoritmus je hill climbing s náhodným výběrem, který vybírá náhodně z tahů, které přináší alspůň nějaké zlepšení a ne vždy nejlepší možný tah.

Implementace:

Pro implementaci algoritmů byl použit jazyk Python. Kód obsahuje třídu `SudokuSolver`, která přijímá vstupní Sudoku matici a obsahuje metody pro vyplnění a vizualizaci Sudoku, výpočet objektivní funkce a samotné algoritmy hill climbing. Na počátku dostaneme, matici kde jsou některé hodnoty vyplněné, ty si uložíme protože s nimi nemůžeme hýbat. Poté víme že v každém sudoku by mělo být $N \times N$ čísel a z toho N jedniček, N dvojek a tak dále. Takže do sudoku doplníme čísla tak aby toto platilo. Jediné co potom může být špatně jsou jednotlivé podmínky sudoku. A tedy dokud nebude vše splněno budeme prohazovat čísla dle daného typu hill climbingu.

Resety

Pro únik z lokálního extrému byly použity resety. Pro dosažení větší efektivity jsem použil 2 typy. Soft reset, který pouze prohodí hodnoty na 2 náhodných pozicích a hard reset, který prohodí větší počet pozic. Hard reset se použije pokud jsme už například použili 10 soft resetů.

Experimenty:

Experimenty byly provedeny na různých typech Sudoku matic. Byly použity jak standardní 9x9 Sudoku, tak i Hexadoku. Pro klasické 9*9 Sudoku byl průměrný čas pro klasický hill climb mezi 1 - 10 sekundami. Pro druhý způsob zhruba o 5-10 sekund více. Pro Hexadoku byl čas okolo jedné minuty a to použité haxadoku nebylo nejtěžší.

Výsledky:

Z experimentů vyplývá, že klasický hill climbing dosahuje v průměru rychlejšího nalezení řešení. Ale během hledání řešení je potřeba víc restartů pro únik z lokálních extrémů. Zato druhý způsob vyžadoval méně restartů protože ne vždy bral optimální řešení.

Závěr:

Na základě provedených experimentů lze konstatovat, že výběr vhodného algoritmu pro řešení Sudoku závisí na složitosti matice a požadavcích na rychlost řešení. Oba dva typy hill climbingu jsou vhodné pro 9*9 Sudoku, ale pro Hexadoku by byl za potřebí efektivnější algoritmus, například backtracking který zkouší postupně dosazovat čísla a pokud se mu to nepovede vrátí se a zkusí to jinak.