Sudoku SAT solver

Theofilos Papapanagiotou, Hung Nguyen

UvA/VU

October 3, 2017

1. Hypothesis
   - Statement
   - Methodology
2. Data
   - Dataset
   - Demo
3. Metrics
   - CNF
   - Demo
   - SAT Statistics
   - Demo
4. Results
   - Conflicts
   - Restarts
   - Decisions
   - Propagations
   - Conflict Literals

Additional material

- Presentation video
- Code repository

Statement

- Does the performance of the SAT solver improve when we enrich the encoding of the sudoku?

Methodology

- CNF encoding
- SAT solver
- Hypothesis evaluation

Dataset

- Source
  - 20 hard instances
  - https://arxiv.org/pdf/1208.0370.pdf
  - https://github.com/attractivechaos/plb/blob/master/sudoku/sudoku.
- Quality
  - Hard instances
  - 17 givens
  - 9x9 puzzles
  - uniquely completable

Demo

Dataset demo

CNF encoding

- Constraints
  - Uniqueness constraint: Every cell must have at most one number, 1-9
  - Definedness constraint: Every cell must have at least one number, 1-9
- Field
  - Cell
  - Row
  - Column
  - Block
- Optimization
  - Eliminate literals
  - Eliminate clauses

Demo

Converter and CNF demo

SAT Statistics

- minisat
- Useful statistics during DPLL runtime
    - Restarts
    - Conflicts
    - Propagations
    - Decisions
    - Conflict literals

Demo

minisat demo

Conflicts

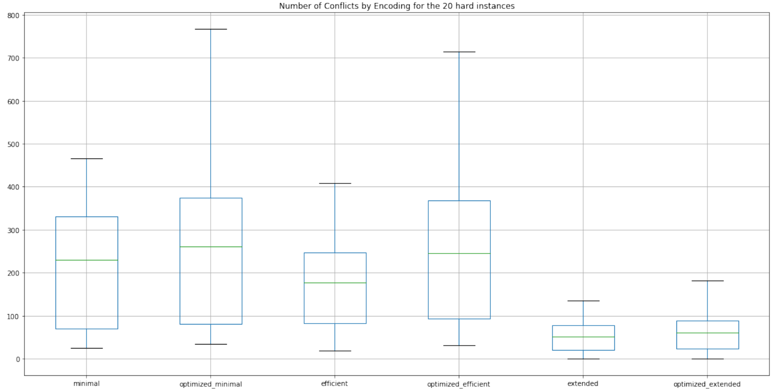

Figure: Graph of the conflicts

Restarts

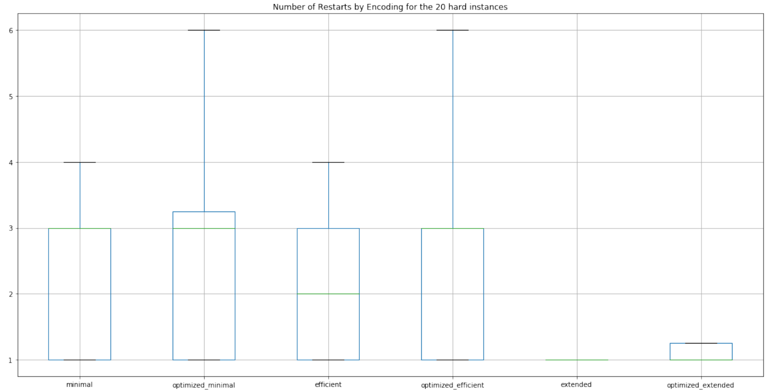

Figure: Graph of the restarts

Decisions

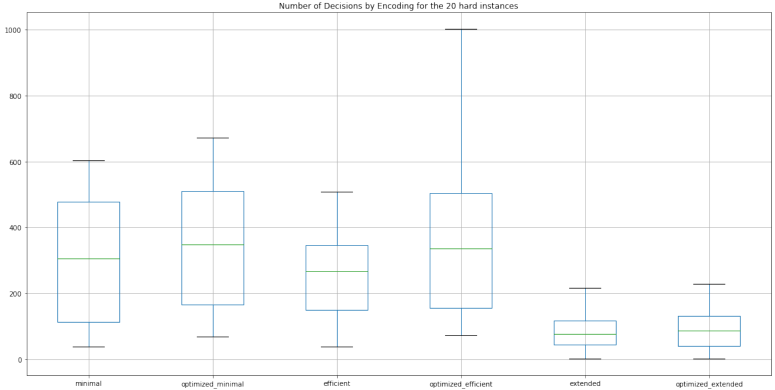

Figure: Graph of the decisions

Propagations

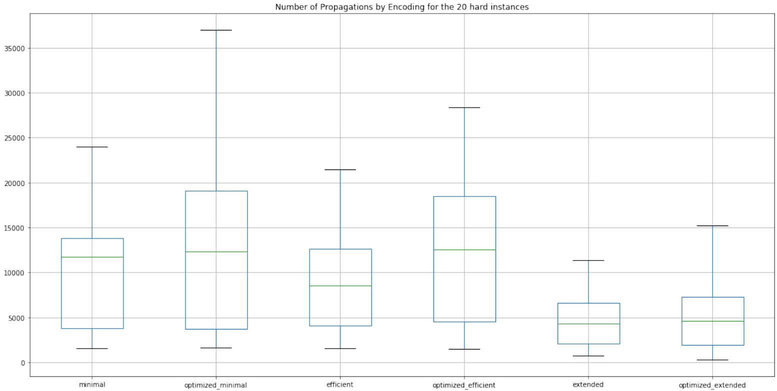

Figure: Graph of the propagations

Conflict Literals

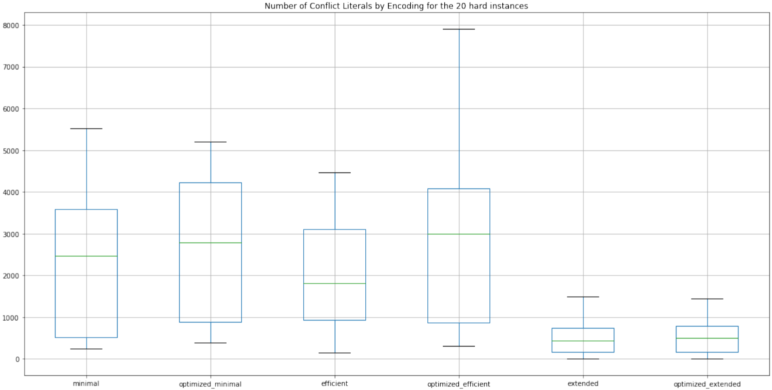

Figure: Graph of the conflict literals