Towards a Complexity-theoretic Understanding of Restarts in SAT solvers

Chunxiao Li[1], Noah Fleming[2], Marc Vinyals[3], Toniann Pitassi[2] and Vijay Ganesh[1]

[1] University of Waterloo, Canada

[2] University of Toronto, Canada

[3] Technion, Israel

PART 1

Context and Motivation

Backjumping

Variable selection

And a few more…

CDCL SAT solver

Conflict analysis

Restarts

Clause deletion

Value selection

What is restart?

- History of restarts
  - Restarts have been studied extensively in the context of search and optimization problems.
    - Escape local minima
- Restarts in DPLL:
  - Upon invocation, erase the trail (partial assignment)
  - Heavy-tailed phenomenon [Gomes and Selman. 2000]
- Restarts in CDCL solvers:
  - Upon invocation, erase the trail while keeping other information
    - Learnt clauses
    - Activities in VSIDS branching
    - Phase-saving values.
- Are restarts really useful for SAT solvers? How do we prove it theoretically?

Motivation to study restarts in the context of SAT solvers

- Empirical:
  - Solvers with restarts outperform solvers without restarts

- Theoretical:
  - CDCL with non-deterministic branching and restarts (after every conflict) is p-equivalent to general resolution [Pipatsrisawat and Darwiche *2011, Atserias et al. 2011*]

  - Unclear if the equivalence with resolution still holds for CDCL solvers without restarts

Previous work on the power of restarts

- Empirical:
  - Heavy-tailed explanation
    - “Heavy-Tailed Phenomena in Satisability and Constraint Satisfaction Problems” [Gomes and Selman 2000]
  - Restarts compact assignment trail
    - “ManySAT: a Parallel SAT solver” [Hamadi et al. 2008]
    - “Machine Learning-based Restart Policy for CDCL SAT Solvers” [Liang et al. 2018]
- Theoretical:
  - Pool resolution [Van Gelder 2005] and regWRTI [Buss et al. 2008]
  - Common consensus: CDCL solvers without restarts are weaker than general resolution

Main Results

- Separation result: drunk CDCL
  - For satisfiable formulas
  - backtracking + non-deterministic variable selection + random value selection
    - Inspired by the drunk model [Alekhnovich et al. 2004]
- Separation result: VSIDS
  - For unsatisfiable formulas
  - backjumping + VSIDS variable selection + phase-saving value selection
- A total of 4 separation results and 2 equivalence results

Our approach to study the power of restarts

	Previous theoretical approach	Our approach
Type of formulas	Unsatisfiable	Unsatisfiable + satisfiable
Type of heuristics	Non-deterministic	Weakened variable selection Weakened value selection Backtracking/backjumping

- Why weakened heuristics?
  - Proving separation/equivalence results seems to be quite challenging when all heuristics are non-deterministic
  - The power of restarts is subtle:
    - Subtle interplay between solver heuristics and the power of restarts
    - The power of restarts becomes more apparent when certain heuristics are weaker than non-deterministic

PART 2

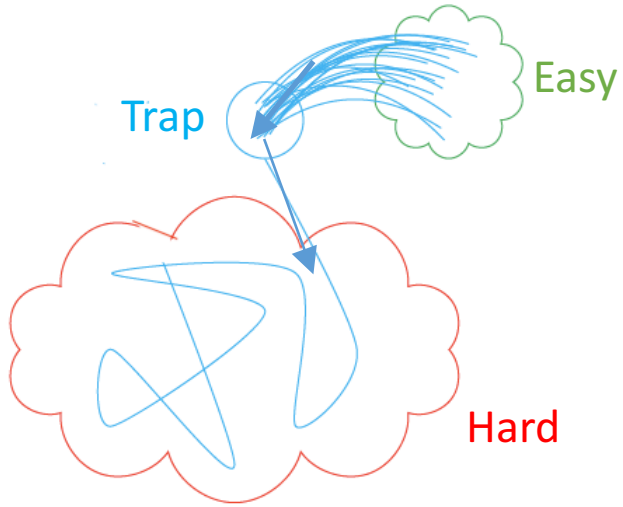
Results

Main Results

- Separation result: drunk CDCL
  - For satisfiable formulas
  - backtracking + non-deterministic variable selection + random value selection
    - Inspired by the drunk model [Alekhnovich et al. 2004]
- Separation result: VSIDS
  - For unsatisfiable formulas
  - backjumping + VSIDS variable selection + phase-saving value selection
- A total of 4 separation results and 2 equivalence results

Proof methodology – Pitfall formulas

- The pitfall formulas have three components:
  - Hard formula for resolution
  - Trap – Tricks the solver into focusing on the hard formula
  - Easy formula – a small backdoor
    - (weak backdoor in the satisfiable case, and strong backdoor for unsatisfiable formulas)



- Lower bound argument:
  - Without restarts, w.h.p. the solver will fall into the trap, and needs to refute the hard formula.
- Upper bound argument:
  - Solvers with restarts can exploit the small backdoor
    - Finding the backdoor variables for the strong backdoor
    - Finding the desired assignment to the backdoor variables for the weak backdoor

Separation result: drunk CDCL

- Model:
  - Backtracking: undo the most recent decision on the trail after learning a conflict
  - Non-deterministic variable selection: non-deterministically returns an unassigned variable upon invocation.
  - Random value selection: returns a truth value uniformly at random
- New formula: Ladder_n
  - Satisfiable formula
  - log(n) size weak backdoor
  - All but one assignment to the weak backdoor variables implies getting trapped
    - No restarts: Hard to assign the backdoor variables correctly with random value selection, branching on other variables also implies the trap w.h.p.
    - Restarts: Keep querying the backdoor variables until assigning them correctly

Separation result: VSIDS

- Model
  - Backjumping: after learning a conflict clause, undo decisions with decision level higher than the second highest decision level in the learnt clause.
  - VSIDS variable selection: returns the variable with highest activity, with random tie breaking. We consider a version of restarts that also resets activities
  - Phase-saving value selection: returns “true” if the input variable x was assigned “true” when the last time x was on the trail, else return “false”. If a variable has not been assigned, then return “false”.
- Formula [Vinyals 2020]:
  - Unsatisfiable formula
  - Constant size strong backdoor
    - No restarts: w.h.p. first conflict bumps activities of variables in the hard formula [Vinyals 2020]
    - Restarts: restart to reset the activities, and use random tie breaking to exploit the constant size backdoor

Other results

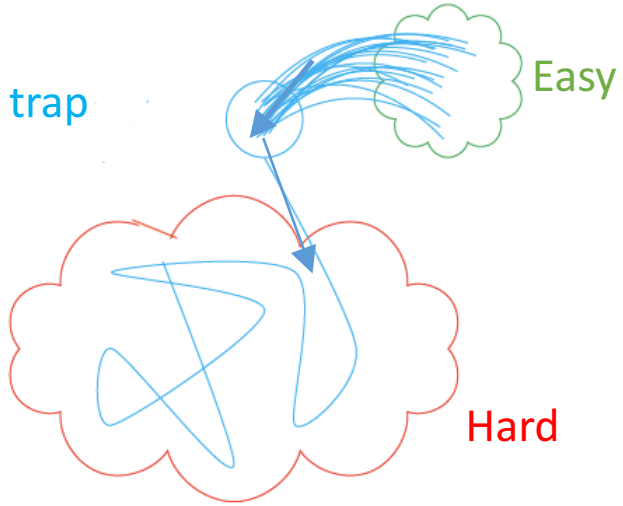
- Equivalence result: static CDCL
  - For satisfiable and unsatisfiable formulas
  - backjumping + static variable selection + static value selection
- Equivalence result: non-deterministic DPLL
  - For unsatisfiable formulas
  - backtracking + non-deterministic variable selection + non-deterministic value selection
- Separation result: drunk DPLL
  - For satisfiable formulas
  - backtracking + non-deterministic variable selection + random value selection
- Separation result: weak decision learning scheme CDCL
  - For unsatisfiable formulas
  - backjumping + non-deterministic variable selection + non-deterministic value selection

PART 3

Insights and Takeaway

Insights that enabled us to prove our results

- Heuristics that are weaker than non-deterministic ones
  - Proving separation/equivalence results seems to be quite challenging when all heuristics are non-determinisitic
  - The power of restarts is subtle:
    - Subtle interplay between solver heuristics and the power of restarts
    - The power of restarts becomes more apparent when certain heuristics are weaker than non-deterministic
- Satisfiable vs unsatisfiable formulas
- Pitfall formulas



Future work

- Equivalence/separation between CDCL + non-deterministic variable and value selection + backjumping with and without restarts remains open

- Plethora of solver configurations with non-deterministic and realistic heuristics (with and without restarts)

Takeaway

- Established 6 equivalence and separation results between SAT solver with and without restarts
  - 4 separation results
  - 2 equivalence results

- Key insights
  - Considering heuristics that are weaker than non-deterministic ones
  - Pitfall formulas