CSE 599S Proof Complexity & Applications

Lecture 11 4 Nov 2020

Conflict-Directed Clause-Learning (CDCL) Solvers

Recall: DPLL(F, A)

while F contains a clause x of size 1 do
F ← F|x=1 ; A ← (A, x)

if F is empty then
halt and output set not A

if F contains the empty clause ⊥ then return add clause

else
choose unset literal x
DPLL(F|x=1, (A, x))
DPLL(F|x=0, (A, x̄))

Call DPPL(F, nil)

Modified form: at each leaf (conflict) add clause with negation of all decisions made to get to leaf.
move up to 1st level where this unit propagates.
replaces last decision with unit prop.

We can learn better clauses than negation of decision

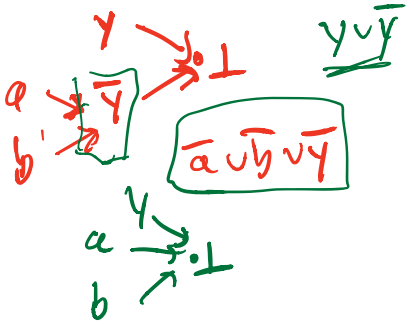
Conflict Graph: built on original F

Decision literals sources

Add a node at every unit prop.

$C = (x \vee \bar{y} \vee z)$ — unit x

$y = 1$, $z = 0$

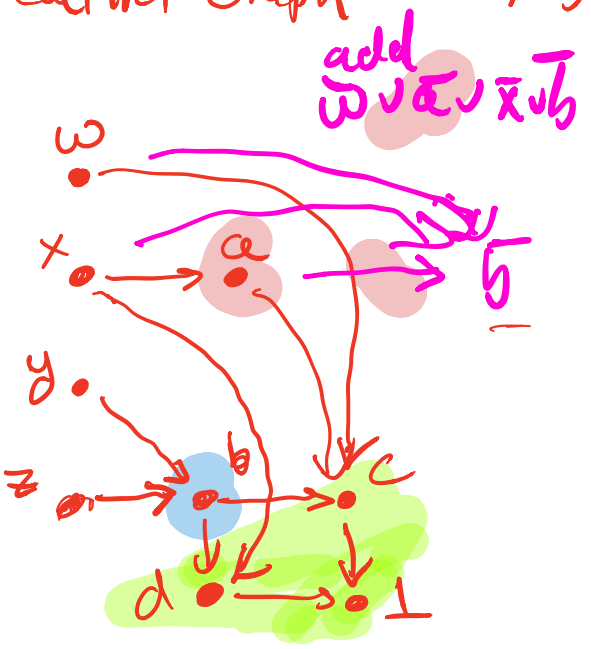


Example next page

$F: \bar{w} \vee \bar{a} \vee \bar{b} \vee c \quad x \vee a, \quad \bar{x} \vee \bar{b} \vee d$
$\bar{y} \vee \bar{z} \vee b, \quad \bar{c} \vee \bar{d}, \quad w \vee \bar{x} \vee b$

Decision Level	Decision
0	$\emptyset$
1	w
2	x
3	y
4	z

Conflict Graph $\bar{w} \vee \bar{x} \vee b$



any cut in conflict graph with all decision literals on one side
empty clause on other
that is cleared
corresponds to clause that could be added to F

$\bar{x} \vee b \vee d \quad \bar{c} \vee \bar{d}$

$\bar{x} \vee b \vee \bar{c}$

$\bar{w} \vee \bar{a} \vee b \vee c$

$\bar{w} \vee \bar{a} \vee \bar{x} \vee b$

Want to choose a clause that unit propagates at a higher level
- just want 1 literal at this level

* asserting clause"

1 UIP

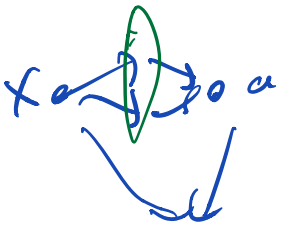
"1st unique implication point"

$\bar{w} \vee \bar{a} \vee x \vee \bar{x} \vee b$

GRASP

working backwards from 1

Clause associated with cut just after 1st node separating last decision from 1



Learned Clause Minimization (MiniSAT)
allows removal of some literals
from the learned

$\bar{w} \vee \bar{a} \vee \bar{x} \vee b \quad \bar{x} \vee a$

$\bar{w} \vee \bar{x} \vee b$

The Resolution symbolic CDCL

Implementation: learned clauses
- big. sometimes
- many of them

Lazy evaluation: Don't keep resolved formula
chaff 2000-2001 Only keep 2 watched literals per clause

If a watched literal is set to false
search for another literal to watch
If none available: unit propagate.

2 literals / clause in cache.

DIMACS var names: integers ± sign
Format

$-1\ 2\ 3 \qquad \bar{x}_1 \vee x_2 \vee x_3$

Assignment A: "trail"
is also in cache

Decision heuristic:

VSIDS

variable decaying sum

priority queue

count for each var

+1 every time a var appears in a learned clause

periodically divide all counts by 2 (every 256 conflicts) changed

multiplicate weight updates

Choose "phase" of the variable

X

- phase saving
  - set var to the sign it was last set

Learned clause deletion:

If a clause hasn't been touched recently throw it away

Periodically cut set of learned clauses by 1/2

cut long clauses

Restarts very common in many backtracking
& learned clauses
restart at level 0.
eg. learned unit clause x

Glucose "literal block distance" heuristic

Thm CDCL + phase saving + nondet branching can simulate resolution
$O(n^3)$ restarts per resolution step

Thm CDCL is not proper under restrictions
$F|_\alpha$ may be harder for CDCL than F is
↑ truth assn

Unless . . .

Run $F + G$ ↑ new variables SAT. doesn't depend on F

No restarts

CDCL simulates resolution $O(n^2)$ steps per resolution clause

In practice CDCL still has major problems on Tseitin examples even with polynice proofs