



MiniSAT: World fastest SAT solver (2005 and 2006)

Niklas Sorensson, Niklas Een and
Armin Biere

Some slides are from Niklas Sorensson presentation

Presented by: Tamir Heyman (2006), Himanshu Jain (2007)



DPLL SAT Solving

- Branching
- Unit propagation
- Backtracking
- Learning
 - New feature: Conflict clause minimization
- Pre-processing



Learning in SAT Solver

- A conflict happens when one clause is falsified by unit propagation
- Analyze the **conflicting clause** to infer a clause
- This clause is a logical consequence of the problem
- The inferred clause is a new knowledge



Conflict/Learned Clause

- Inferred by conflict analysis
- Helps prune future parts of the search space
- Actually drives backtracking



Conflict Clause Requirements

- Consequence of the clause set
- Falsified by current assignment
- Contains exactly one literal implied by last assumption (**Asserting clause**)

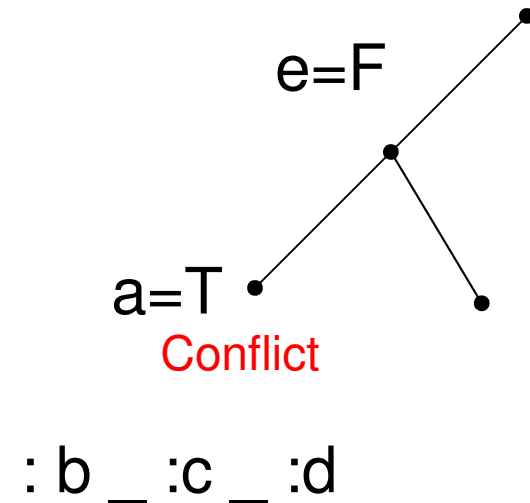


Conflict Analysis Algorithm

- Begin with conflicting clause
- Resolve on the most recently propagated literal
 - Using the antecedent as side clause
- Repeat until the clause contains exactly one literal from the last assumption

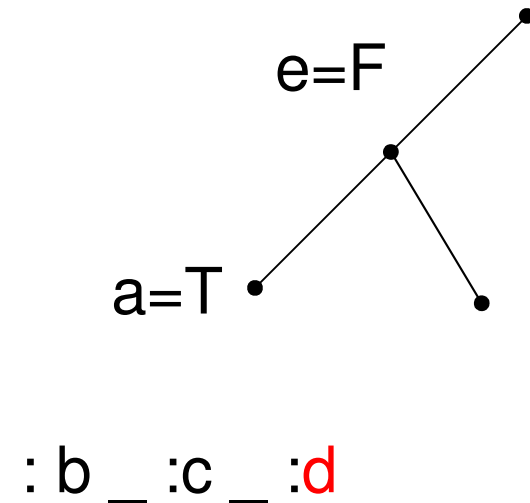
Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f _ e$
$g=F$	$\neg g _ f$
$h=F$	$\neg h _ g$
$a=T$	n
$b=T$	$b _ \neg a _ e$
$c=T$	$c _ e _ f _ \neg b$
$d=T$	$d _ \neg b _ h$



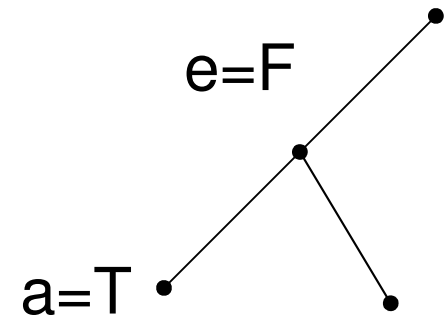
Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f _ e$
$g=F$	$\neg g _ f$
$h=F$	$\neg h _ g$
$a=T$	n
$b=T$	$b _ \neg a _ e$
$c=T$	$c _ e _ f _ \neg b$
$d=T$	$d _ \neg b _ h$



Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f_e$
$g=F$	$\neg g_f$
$h=F$	$\neg h_g$
$a=T$	n
$b=T$	$b_ \neg a_e$
$c=T$	$c_e_f_ \neg b$
$d=T$	$d_ \neg b_h$

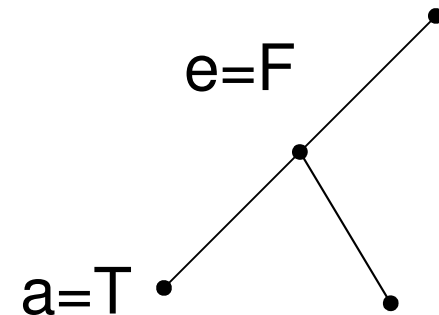


: b _ :c _ :d

: b _ :c _ h

Example

Assignment	assumption
$\epsilon = F$	n
$f = F$	$\neg f_e$
$g = F$	$\neg g_f$
$h = F$	$\neg h_g$
$a = T$	n
$b = T$	$b_ \neg a_e$
$c = T$	$c_e_f_ \neg b$
$d = T$	$d_ \neg b_h$

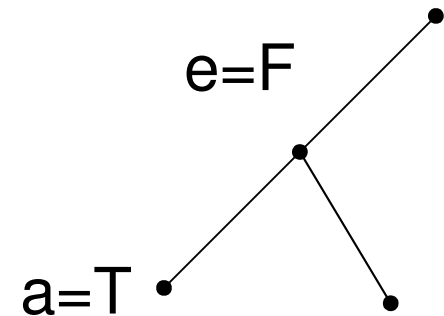


: b _ : c _ : d

: b _ : **c** _ h

Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f_e$
$g=F$	$\neg g_f$
$h=F$	$\neg h_g$
$a=T$	n
$b=T$	$b_ \neg a_e$
$c=T$	$c_e_f_ \neg b$
$d=T$	$d_ \neg b_h$



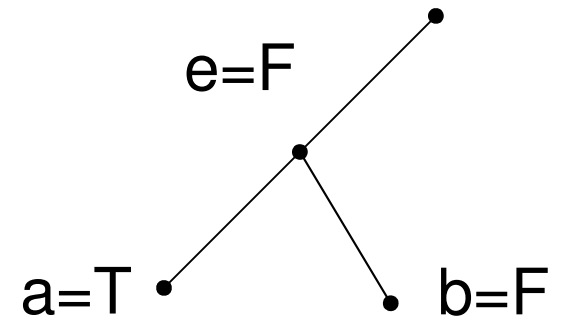
: b _ :c _ :d

: b _ :c _ h

: b _ e _ f _ h

Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f_e$
$g=F$	$\neg g_f$
$h=F$	$\neg h_g$
$a=T$	n
$b=T$	$b_ \neg a_e$
$c=T$	$c_e_f_ \neg b$
$d=T$	$d_ \neg b_h$



: b _ : c _ : d

: b _ : c _ h

: b _ e _ f _ h



Conflict Minimizing

- Traditional Conflict Analysis is minimal in the number of derivations
- Balance between time spent and usefulness of the conflict clause
- Is a shorter clause always better?



Basic Conflict Minimizing

- Start from an ordinary conflict clause
- Apply resolution greedily
- Works because there are no cyclic dependencies
- Also uses antecedent clauses from other levels



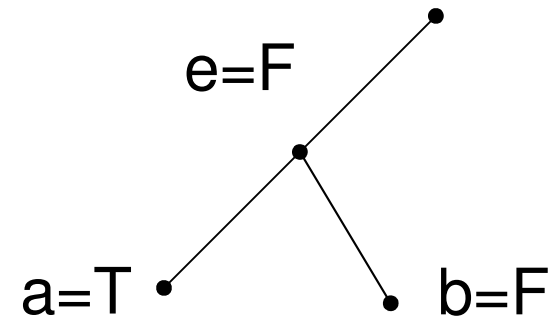
Resolution

$$\frac{x _ A \quad : x _ B}{A _ B}$$

A, B are clauses
x is a variable

Example

Assignment	assumption
$e=F$	n
$f=F$	$\neg f _ e$
$g=F$	$\neg g _ f$
$h=F$	$\neg h _ g$
$a=T$	n
$b=T$	$b _ \neg a _ e$
$c=T$	$c _ e _ f _ \neg b$
$d=T$	$d _ \neg b _ h$

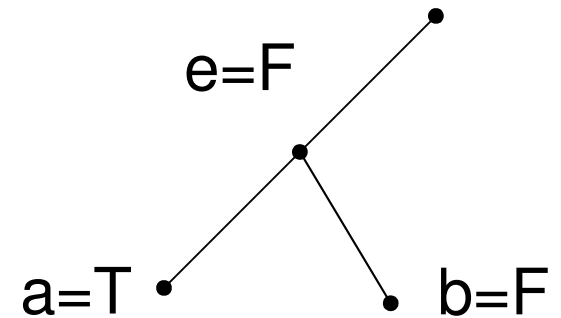


: b _ :c _ :d

: b _ :c _ h

: b _ e _ h _ **f**

Assignment	assumption
$\epsilon = F$	n
$f = F$	$\neg f_e$
$g = F$	$\neg g_f$
$h = F$	$\neg h_g$
$a = T$	n
$b = T$	$b_ \neg a_e$
$c = T$	$c_e_f_ \neg b$
$d = T$	$d_ \neg b_h$



```
: b _ :c _ :d
: b _ :c _ h
: b _ e _ h _ f
: b _ e _ h
```

Conflict clause minimization

Impact



- On average there is 25-50% reduction in the size of learned clauses
- Helps reduce Boolean constraint propagation time

Effective Preprocessing in SAT through Variable and Clause Elimination

Niklas Eén

*Cadence Berkeley Laboratories
Berkeley, USA*

Armin Biere

*Johannes Kepler University
Linz, Austria*

for **SAT 2005**

***The Eighth International Conference on
Theory and Applications of Satisfiability Testing***

TimeLine

