Unit Propagation and Variable Ordering in MiniSAT

Daniel Geschwender

CSCE 235H Introduction to Discrete Structures
Spring 2018
URL: cse.unl.edu/~cse235h
All questions: Piazza

Unit Propagation (1): In a clause

- If a literal is true, the clause is true and can be removed
  - If $a \leftarrow true$, remove all clauses where it is positive

$$a \vee b \vee c$$

  - If $a \leftarrow false$, remove all clauses where it is negative

$$\neg a \vee b \vee c$$

- If a literal is false, it can be removed from the clause
  - If $a \leftarrow false$, remove it from all clauses where it is positive

$$a \vee b \vee c \qquad\qquad b \vee c$$

  - If $a \leftarrow true$, remove all clauses where it is negative

$$\neg a \vee b \vee c \qquad\qquad b \vee c$$

Unit Propagation (2)

- When a literal in a clause is true, the entire clause is true and can be removed

After assignment: $a \leftarrow false$

$$
\begin{aligned}
& \cdots \wedge C_i \wedge (\neg a \vee b \vee \neg c) \wedge C_{i+2} \wedge \ldots \\
\equiv\ & \cdots \wedge C_i \wedge (true \vee b \vee \neg c) \wedge C_{i+2} \wedge \ldots \\
\equiv\ & \cdots \wedge C_i \wedge true \wedge C_{i+2} \wedge \ldots \\
\equiv\ & \cdots \wedge C_i \wedge C_{i+2} \wedge \ldots
\end{aligned}
$$

Unit Propagation (3)

- When a literal in a clause is false, the literal may be removed from the clause

After assignment: $a \leftarrow true$

$$(\neg a \vee b \vee \neg c)$$

$$\equiv (false \vee b \vee \neg c)$$

$$\equiv (b \vee \neg c)$$

Unit Propagation (4)

- Assignments may result in unit clauses

$$a \leftarrow \mathit{false} \text{ and } a \vee b \text{ yield } b$$

- Unit clauses immediately force an assignment

$$b \leftarrow \mathit{true}$$

- This can lead to a chain reaction as new assignments ‘propagate’ throughout the clauses

Unit Propagation: Example

- SAT formula with $1{,}000$ variables
- Cycle of implications

$$v_1 \rightarrow v_2$$
$$v_2 \rightarrow v_3$$
$$\vdots$$

- A single assignment results in unit propagation to solve the entire problem

$$(\neg v_1 \vee v_2)$$
$$\wedge\, (\neg v_2 \vee v_3)$$
$$\vdots$$
$$\wedge\, (\neg v_{999} \vee v_{1000})$$
$$\wedge\, (\neg v_{1000} \vee v_1)$$

Unit Resolution

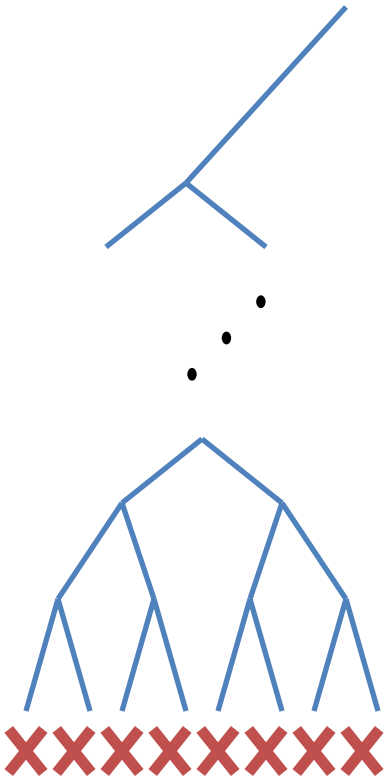
- Unit resolution or disjunctive syllogism

$$\frac{a, \neg a \vee b}{b}$$

- Unit propagation has similarities to unit resolution
  - Assignments add unit clauses to the formula
  - When a new unit clause is added, it can be resolved with other clauses in the formula and the result added to the formula

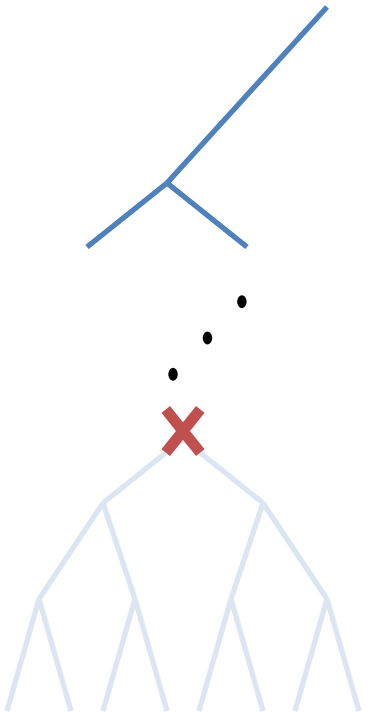
$$(a) \wedge (\neg a \vee b \vee \neg c) \rightarrow (b \vee \neg c)$$

Thrashing in Search



- Repeatedly hitting conflict within a subtree of the search tree
- Thrashing is costly

Fail-First Principle (FFP)



- It is better to fail early than to waste time exploring a subtree with no solutions
- Want to handle potential conflicts earlier

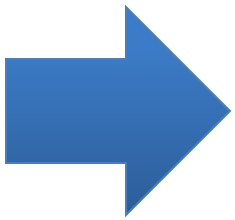
Variable Ordering Heuristics

- The order in which variables are assigned
- Heuristic: good performance in general, not guaranteed to be optimal
- Static versus dynamic
  - Static: Assignment order is decided before search and maintained fixed throughout search
  - Dynamic: Variable ordering is adjusted during the course of the search

Activity-Based Heuristics

- Variables are assigned an ‘activity’ value
- Variables involved in a conflict have their activities increased
- Activities exponentially decay
- Select most ‘active’ variable

$$(\neg a \vee \neg c \vee d)$$

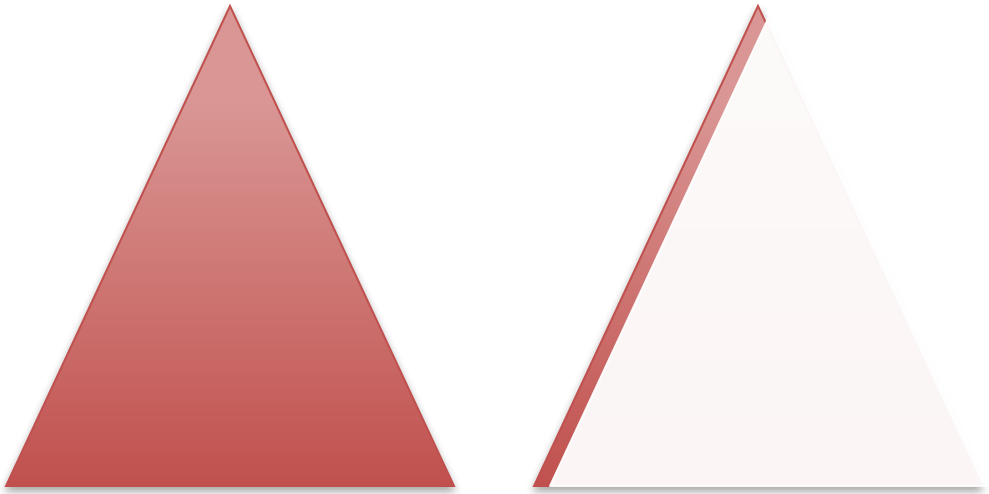


$$Activity(a)+ = 1$$
$$Activity(c)+ = 1$$
$$Activity(d)+ = 1$$

Activity-Based Heuristics Example

- UNSAT formula with 1000 variables
- Conflict occurs between v_{999} and v_{1000}

$$\vdots$$
$$\wedge \left(v_{999} \vee v_{1000}\right)$$
$$\wedge \left(\neg v_{999} \vee v_{1000}\right)$$
$$\wedge \left(v_{999} \vee \neg v_{1000}\right)$$
$$\wedge \left(\neg v_{999} \vee \neg v_{1000}\right)$$



Lexicographic ordering

Activity-based ordering