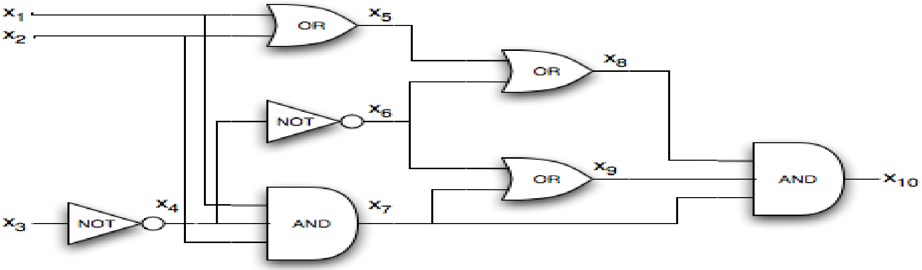
Practical SAT Solving

Lecture 2

Carsten Sinz, Tomáš Balyo | April 30, 2019

INSTITUTE FOR THEORETICAL COMPUTER SCIENCE



SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \lor x_1 \lor x_2 \lor \cdots \lor x_n) \land (\bar{l} \lor y_1 \lor y_2 \lor \cdots \lor y_m)}{(x_1 \lor x_2 \lor \cdots \lor x_n \lor y_1 \lor y_2 \lor \cdots \lor y_m)}$$

The upper two clauses are called *Input Clauses* the bottom clause is called the *Resolvent*

SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \vee x_1 \vee x_2 \vee \cdots \vee x_n) \wedge (\bar{l} \vee y_1 \vee y_2 \vee \cdots \vee y_m)}{(x_1 \vee x_2 \vee \cdots \vee x_n \vee y_1 \vee y_2 \vee \cdots \vee y_m)}$$

The upper two clauses are called *Input Clauses* the bottom clause is called the *Resolvent*

Examples

- $(x_1 \vee x_3 \vee \overline{x_7}) \wedge (\overline{x_1} \vee x_2) \vdash (x_3 \vee \overline{x_7} \vee x_2)$

SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \vee x_1 \vee x_2 \vee \cdots \vee x_n) \wedge (\bar{l} \vee y_1 \vee y_2 \vee \cdots \vee y_m)}{(x_1 \vee x_2 \vee \cdots \vee x_n \vee y_1 \vee y_2 \vee \cdots \vee y_m)}$$

The upper two clauses are called *Input Clauses* the bottom clause is called the *Resolvent*

Examples

- $(x_1 \vee x_3 \vee \overline{x_7}) \wedge (\overline{x_1} \vee x_2) \vdash (x_3 \vee \overline{x_7} \vee x_2)$
- $(x_4 \vee x_5) \wedge (\overline{x_5}) \vdash (x_4)$

SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \vee x_1 \vee x_2 \vee \dots \vee x_n) \wedge (\bar{l} \vee y_1 \vee y_2 \vee \dots \vee y_m)}{(x_1 \vee x_2 \vee \dots \vee x_n \vee y_1 \vee y_2 \vee \dots \vee y_m)}$$

The upper two clauses are called *Input Clauses* the bottom clause is called the *Resolvent*

Examples

- $(x_1 \vee x_3 \vee \overline{x_7}) \wedge (\overline{x_1} \vee x_2) \vdash (x_3 \vee \overline{x_7} \vee x_2)$
- $(x_4 \vee x_5) \wedge (\overline{x_5}) \vdash (x_4)$
- $(x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2}) \vdash$

SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \lor x_1 \lor x_2 \lor \cdots \lor x_n) \land (\bar{l} \lor y_1 \lor y_2 \lor \cdots \lor y_m)}{(x_1 \lor x_2 \lor \cdots \lor x_n \lor y_1 \lor y_2 \lor \cdots \lor y_m)}$$

The upper two clauses are called *Input Clauses* the bottom clause is called the *Resolvent*

Examples

- $(x_1 \lor x_3 \lor \overline{x_7}) \land (\overline{x_1} \lor x_2) \vdash (x_3 \lor \overline{x_7} \lor x_2)$
- $(x_4 \lor x_5) \land (\overline{x_5}) \vdash (x_4)$
- $(x_1 \lor x_2) \land (\overline{x_1} \lor \overline{x_2}) \vdash$
- $(x_1) \land (\overline{x_1}) \vdash$

SAT Tools – Resolution

The Resolution Rule

$$\frac{(l \lor x_1 \lor x_2 \lor \cdots \lor x_n) \land (\bar{l} \lor y_1 \lor y_2 \lor \cdots \lor y_m)}{(x_1 \lor x_2 \lor \cdots \lor x_n \lor y_1 \lor y_2 \lor \cdots \lor y_m)}$$

Special Cases

- Tautological Resolvent $(x_1 \lor x_2) \land (\overline{x_1} \lor \overline{x_2}) \vdash (x_1 \lor \overline{x_1})$
  - Usually forbidden, does no harm, will be useful later
- Empty Clause $(x_1) \land (\overline{x_1}) \vdash \bot$
  - The empty clause a.k.a conflict clause a.k.a "$\bot$" is unsatisfiable

Notation

- $\mathrm{R}((x_1 \lor x_2), (\overline{x_1} \lor x_3)) = (x_2 \lor x_3)$

SAT Tools – Resolution

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \wedge \mathrm{R}(C_1, C_2)$ is satisfiable.

SAT Tools – Resolution

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \wedge \mathrm{R}(C_1, C_2)$ is satisfiable.

Proof:

- If F is not satisfiable then $F \wedge C$ for any C is also not satisfiable.
- If F is satisfiable and ϕ is a satisfying assignment of F then we show that ϕ also satisfies $\mathrm{R}(C_1, C_2)$.
  - If $C_1 = (l \vee P_1)$ and $C_2 = (\bar{l} \vee P_2)$ then $\mathrm{R}(C_1, C_2) = (P_1 \vee P_2)$
  - Since ϕ satisfies both C_1 and C_2 it must satisfy at least one of the literals in P_1 or P_2.
    - if ϕ satisfies l then it satisfies some literal in P_2
    - if ϕ satisfies $\bar{l}$ then it satisfies some literal in P_1

SAT Tools – Resolution

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \wedge \mathrm{R}(C_1, C_2)$ is satisfiable.

Consequences

- If we manage to resolve the empty clause ($\perp$) the original formula is unsatisfiable

SAT Tools – Resolution

Karlsruhe Institute of Technology

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \wedge \mathrm{R}(C_1, C_2)$ is satisfiable.

Consequences

- If we manage to resolve the empty clause ($\perp$) the original formula is unsatisfiable

Usage

- Proof of unsatisfiability – Resolution Proof
  - A resolution proof is a sequence of clauses such that each clause is either a clause of the original formula or a resolvent of two previous clauses ending with $\perp$.

SAT Tools – Resolution

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \wedge \mathrm{R}(C_1, C_2)$ is satisfiable.

Consequences

- If we manage to resolve the empty clause ($\perp$) the original formula is unsatisfiable

Usage

- Proof of unsatisfiability – Resolution Proof
  - A resolution proof is a sequence of clauses such that each clause is either a clause of the original formula or a resolvent of two previous clauses ending with $\perp$.

Example: $(x_1 \vee x_2), (\overline{x_1} \vee x_2), (x_1 \vee \overline{x_2}), (\overline{x_1} \vee \overline{x_2})$

SAT Tools – Resolution

Theorem: Resolution maintains satisfiability

Let F be a CNF formula and C_1 and C_2 two of it's clauses with a pair complementary literals. Then F is satisfiable if and only if $F \land \mathrm{R}(C_1, C_2)$ is satisfiable.

Consequences

- If we manage to resolve the empty clause ($\perp$) the original formula is unsatisfiable

Usage

- Proof of unsatisfiability – Resolution Proof
  - A resolution proof is a sequence of clauses such that each clause is either a clause of the original formula or a resolvent of two previous clauses ending with $\perp$.

Example: $(x_1 \lor x_2), (\overline{x_1} \lor x_2), (x_1 \lor \overline{x_2}), (\overline{x_1} \lor \overline{x_2}), (x_2), (\overline{x_2}), \perp$

SAT Tools – Saturation Algorithm

Saturation Algorithm

- INPUT: CNF formula F
- OUTPUT: $\{SAT, UNSAT\}$

while (true) **do**
  $R = \text{resolveAll}(F)$
  if $(R \cap F \neq R)$ **then** $F = F \cup R$
  else break
if $(\perp \in F)$ **then return** $UNSAT$ **else return** SAT

SAT Tools – Saturation Algorithm

Saturation Algorithm

- INPUT: CNF formula F
- OUTPUT: $\{SAT, UNSAT\}$

while (true) **do**
  $R = \text{resolveAll}(F)$
  if $(R \cap F \neq R)$ **then** $F = F \cup R$
  else break
if $(\bot \in F)$ **then return** $UNSAT$ **else return** SAT

Properties of the saturation algorithm:

- it is sound and complete – always terminates and answers correctly
- has exponential time and space complexity (always for Pigeons)

SAT Tools – Unit Propagation

Unit Resolution

= at least one of the resolved clauses is unit (has one literal).

Example:

- $R((x_1 \vee x_7 \vee \overline{x_2} \vee x_4), (x_2)) = (x_1 \vee x_7 \vee x_4)$

SAT Tools – Unit Propagation

Unit Resolution

= at least one of the resolved clauses is unit (has one literal).

Example:

- $R((x_1 \vee x_7 \vee \overline{x_2} \vee x_4), (x_2)) = (x_1 \vee x_7 \vee x_4)$

Unit Propagation

= a process of applying unit resolution as long as we get new clauses.

Example:

- $(x_1) \wedge (x_7 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3)$
- $(x_1) \wedge (x_7 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3) \wedge (x_3)$
- $(x_1) \wedge (x_7 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3) \wedge (x_3) \wedge (x_7 \vee x_2)$

Easy Cases

SAT is not always hard, in the following cases it is polynomially solvable

- 2-SAT
- Horn-SAT
- Hidden Horn-SAT
- SLUR

2-SAT

2-SAT Formula

= each clause has exactly 2 literals.

Example:

- $(x_1 \vee x_3) \wedge (x_7 \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_3)$
- $(x_1 \vee x_2) \wedge (\overline{x_1} \vee x_2) \wedge (x_1 \vee \overline{x_2}) \wedge (\overline{x_1} \vee \overline{x_2})$

Also called Binary SAT or Quadratic SAT

How to solve 2-SAT?

How to solve 2-SAT?

Saturation Algorithm

The resolution saturation algorithm is polynomial for 2-SAT

Proof:

- Only 2-literal resolvents are possible
- There are only $\mathcal{O}(n^2)$ 2-literal clauses on n variables

Complexity:

- Both time and space $\mathcal{O}(n^2)$
- There exists a linear algorithm! [**?**]

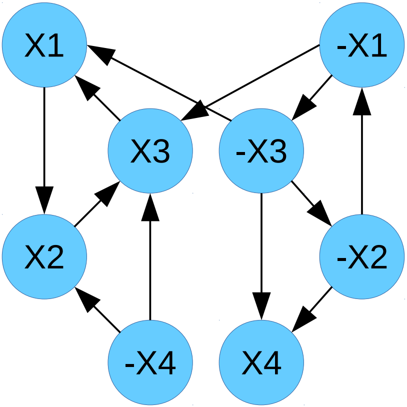
Implication Graph

Implication graph of a formula F is an oriented graph that has:

- a vertex for each literal of F
- 2 edges for each clause $(l_1 \vee l_2)$
  - $\bar{l}_1 \rightarrow l_2$
  - $\bar{l}_2 \rightarrow l_1$

Example:

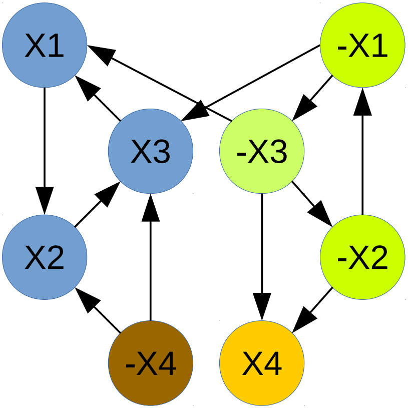
$(\overline{x}_1 \vee x_2) \wedge (\overline{x}_2 \vee x_3) \wedge (\overline{x}_3 \vee x_1) \wedge$
$(x_2 \vee x_4) \wedge (x_3 \vee x_4) \wedge (x_1 \vee x_3)$



Implication Graph

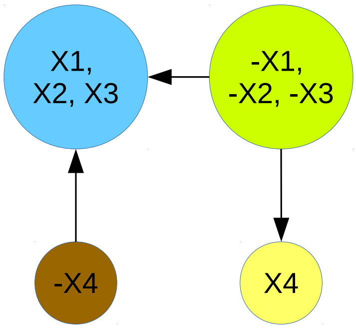
The next step is to analyze the *Strongly Connected Components* of the implication graphs

- SCC = there is a path in each direction between each pair
- Tarjan's algorithm finds SCCs in $\mathcal{O}(|V| + |E|)$
- If any x and $\overline{x}$ literal pair is in the same SCC then the formula is UNSAT
    - All the literals in an SCC must be all True or all False



How to find the solution?

- Construct the *Condensation* of the implication graph
  - contract each SCC into one vertex
- Topologically order the vertices of the condensation
- In reverse topological order, if the variables do not already have truth assignments, set all the terms to true.



Example: $x_1 = x_2 = x_3 = \mathit{True}$, $x_4 = \mathit{True}$, the rest is already assigned.

How to solve 2-SAT?

Linear Algorithm

- Construct the Implication Graph
- Find all the SCCs
- Check if any SCC contains a complementary pair
- Construct a condensation of the implication graph
- Run topological sort on the condensation
- Construct the solution

Complexity:

- All the steps can be done in linear time