5CCS2FC2: Foundations of Computing II

Introduction to SAT Solving

Week 5

Dr Christopher Hampson

Department of Informatics

King's College London

Are all SAT problems hard?

Restricted versions of SAT

The Boolean Satisfiability Problem N SAT

Input) A propositional formula F in CNF with *at most* N literals in each clause.

Output) **True** if and only if F is *satisfiable*

Restricted versions of SAT

Theorem We have the following chain of polynomial reductions

$$\mathbf{2\,SAT} \leq_p \mathbf{3\,SAT} \leq_p \mathbf{4\,SAT} \leq_p \ldots \leq_p k\,\mathbf{SAT} \leq_p \ldots \leq_p \mathbf{SAT}$$

Proof: This reduction is **easy** since every CNF formula with **at most** k literals in each clause also has **at most** $(k+1)$ literals in each clause.

Q.E.D.

Some definitions of **N SAT** stipulate **exactly** N literals per clause. But the same chain of reductions can be proved for this alternative definition as well.

Restricted versions of SAT

Theorem We also have the following chain of polynomial reductions

$$\textsf{SAT} \leq_p \ldots \leq_p k\,\textsf{SAT} \leq_p \ldots \leq_p 5\,\textsf{SAT} \leq_p 4\,\textsf{SAT} \leq_p 3\,\textsf{SAT}$$

Proof: This direction is **more surprising** as the direction is reversed.

- It is enough to show that

$$\textsf{SAT} \leq_p 3\,\textsf{SAT}(k+1)\,\textsf{SAT} \leq_p \textsf{SAT} \leq_p 3\,\textsf{SAT} \leq_p k\,\textsf{SAT}$$

Restricted versions of SAT

Step 1) Let F be an arbitrary formula in CNF

$$(P \vee \neg Q \vee R \vee S) \quad \wedge \quad (Q \vee \neg R \vee \neg T)$$

Restricted versions of SAT

Step 2) Find a clause which contains **more than 3 literals**

$$(P \lor \neg Q \lor R \lor S) \quad \land \quad (Q \lor \neg R \lor \neg T)$$

✗ ✓

Restricted versions of SAT

Step 3) Break up the clause into **two pieces** in the middle

$$(P \lor \neg Q \quad\quad R \lor S) \quad \land \quad (Q \lor \neg R \lor \neg T)$$

✗ ✓

Restricted versions of SAT

Step 4) Introduce a fresh propositional variable to **'bridge the gap'**

$$(P \lor \neg Q \lor X) \quad \land \quad (\neg X \lor R \lor S) \quad \land \quad (Q \lor \neg R \lor \neg T)$$

✓ ✓ ✓

Step 5) Repeat until all clauses contain **at most 3 literals**

The resulting formula is *satisfiable* if and only if the with the original formula is satisfiable!

Restricted versions of SAT

Theorem We also have the following chain of polynomial reductions

$$\textbf{SAT} \leq_p \ldots \leq_p k\,\textbf{SAT} \leq_p \ldots \leq_p \textbf{5 SAT} \leq_p \textbf{4 SAT} \leq_p \textbf{3 SAT}$$

$$?? \quad \textbf{3 SAT} \leq_p \textbf{2 SAT} \quad ??$$

Solving the 2 SAT Problem

Solving the 2 SAT Problem

Theorem **2 SAT** is solvable in polynomial time.

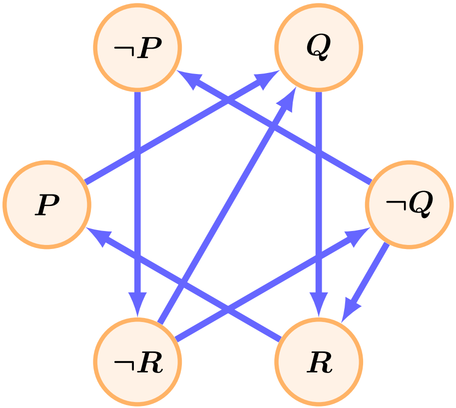
Proof: We can solve **2 SAT** by using our **strongly connected component** algorithm.

Step 1) Write each clause as **two implications**

$$\begin{array}{c} (\neg P \vee Q) \\ (\neg Q \vee R) \\ (P \vee \neg R) \\ (R \vee Q) \end{array} \quad \Longrightarrow \quad \begin{array}{lcl} (P \to Q) & \wedge & (\neg Q \to \neg P) \\ (Q \to R) & \wedge & (\neg R \to \neg Q) \\ (\neg P \to \neg R) & \wedge & (R \to P) \\ (\neg R \to Q) & \wedge & (\neg Q \to R) \end{array}$$

Solving the 2 SAT Problem

Step 2) Construct the **implication graph**



$(P \to Q)$

$(\neg Q \to \neg P)$

$(Q \to R)$

$(\neg R \to \neg Q)$

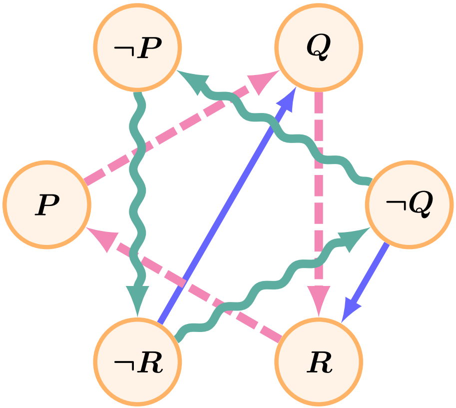
$(\neg P \to \neg R)$

$(R \to P)$

$(\neg R \to Q)$

$(\neg Q \to R)$

Step 3) Compute the **Strongly Connected Components (SCCs)**

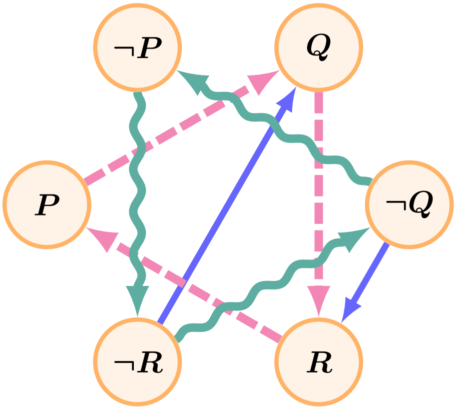


Strongly Connected Components

$\{ P, Q, R \}$

$\{ \neg P, \neg Q, \neg R \}$

Step 4) Check whether any SCC contains a literal and its negation



Strongly Connected Components

$\{ P, Q, R \}$

$\{ \neg P, \neg Q, \neg R \}$

Solving the 2 SAT Problem

Conclusion) Since we can **compute** and **check** each strongly connected component in **polynomial time** we have that

2 SAT is solvable in **polynomial time**

Q.E.D.

Other easy variants of SAT?

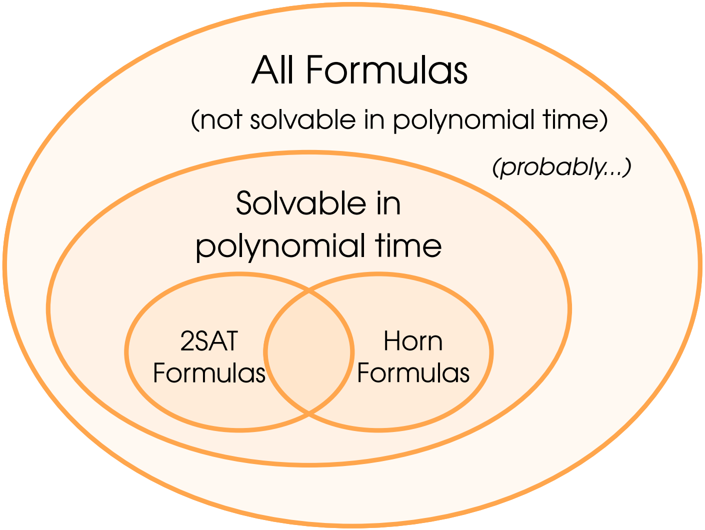
Other easy variants of SAT

- **(Definite) Propositional Horn Clauses**

$$\boxed{P \quad , \quad (P \to Q) \quad , \quad (P \wedge Q \to R) \quad , \quad (Q \to S) \quad , \quad (S \wedge R \to T)}$$

- Every clause contains **exactly one** positive literal
- Every clause can be written in **implicational / rule form** with a single **(positive) head**

Other easy variants of SAT



End of Slides!