# SAT Solvers

Program Verification - Laborator

FMI · Denisa Diaconescu · Spring 2022

# SAT solvers

- There are plenty SAT solvers
  - Glucose
  - MiniSAT, PicoSAT
  - RelSAT
  - GRASP
  - ...
- In order to solve the problems, you can use any SAT solver you prefer
- There are also online SAT solvers:
  - Logictools
  - ...

# DIMACS Format

- the most common input format for SAT solvers
- a way to encode CNF formulas

**Example**
The input

```
c This is a comment
c This is another comment
p cnf 6 3
1 -2 3 0
2 4 5 0
4 6 0
```

represents the CNF formula $(x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee x_4 \vee x_5) \wedge (x_4 \vee x_6)$

# DIMACS Format

- At the beginning there can exist one or more comment line.
- Comment lines start with a c
- The following lines are information about the expression itself
- the Problem line starts with a p:

```
p FORMAT VARIABLES CLAUSES
```

- `FORMAT` should always be cnf
- `VARIABLES` is the number of variables in the expression
- `CLAUSES` is the number of clauses in the expression

**Example**

`p cnf 6 3` expresses that there are 6 variables and 3 clauses

# DIMACS Format

- The next `CLAUSES` lines are for the clauses themselves
- Variables are enumerated from 1 to `VARIABLES`
- A negation is represented by $-$
- Each variable information is separated by a blank space
- A 0 is added at the end to mark the end of the clause

**Example**
`1 -2 3 0` expresses the clause $(x_1 \vee \neg x_2 \vee x_3)$

# Exercise 1 - A planning problem encoded in SAT

Problem.

Scheduling a meeting considering the following constraints:

- Adam can only meet on Monday and Wednesday
- Bridget cannot meet on Wednesday
- Charles cannot meet on Friday
- Darren can only meet on Thursday or Friday

# Exercise 1 - A planning problem encoded in SAT

Solution.

- We represent week day *Monday*, *Tuesday*, ... as variables $x_1, x_2, \ldots$.
- We obtain the following formula in CNF:

$$\begin{aligned}
\varphi = {} & (x_1 \vee x_3) \wedge (\neg x_3) \wedge (\neg x_5) \wedge (x_4 \vee x_5) \wedge \\
& (\neg x_1 \vee \neg x_2) \wedge (\neg x_1 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_1 \vee \neg x_5) \wedge \\
& (\neg x_2 \vee \neg x_3) \wedge (\neg x_2 \vee \neg x_4) \wedge (\neg x_2 \vee \neg x_5) \wedge \\
& (\neg x_3 \vee \neg x_4) \wedge (\neg x_3 \vee \neg x_5) \wedge \\
& (\neg x_4 \vee \neg x_5)
\end{aligned}$$

## Exercise 1 - A planning problem encoded in SAT

Todo.

- Encode this solution into DIMACS Format
- Search for a solution using a SAT solver

# Exercise 2 - Graph Colouring encoded in SAT

Problem.

Given an undirected graph $G = (V, E)$, a graph colouring assigns a colour to each node such that all adjacent nodes have a different colour.

A graph colouring using at most $k$ colours is called a $k$-colouring.

The Graph Colouring Problem asks whether a $k$-colouring for $G$ exists.

# Exercise 2 - Graph Colouring encoded in SAT

## Problem.

Given an undirected graph $G = (V, E)$, a graph colouring assigns a colour to each node such that all adjacent nodes have a different colour.

A graph colouring using at most $k$ colours is called a $k$-colouring.

The Graph Colouring Problem asks whether a $k$-colouring for $G$ exists.

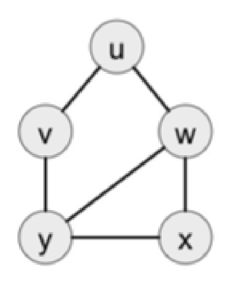
## Solution

- SAT encoding: use $k \cdot |V|$ Boolean variables
- For every $v \in V$ and $1 \leq j \leq k$, variable $v_j$ is true if node $v$ gets colour $j$
- Clauses?
    - Every node gets a colour: $(v_1 \vee \ldots \vee v_k)$ for $v \in V$
    - Adjacent nodes have diff. colours: $\neg u_j \vee \neg v_j$ for $u, v \in V$, $u \neq v$, $u, v$ adjacent, $1 \leq j \leq k$
    - What about multiple colours for a node? At-most-one constraints

# Exercise 2 - Graph Colouring encoded in SAT

Todo. Encode the following graph colouring problem into SAT and use a SAT solver to find a solution.

- $V = \{u, v, w, x, y\}$
- Colours: red (=1), green (=2), blue (=3)



# Exercise 3 - Einstein's riddle encoded in SAT

Problem.

The true source of the riddle is unknown, it is stated that there are five houses in a row with each house a different colour and each house owned by a man of a different nationality, having a different pet, prefering different kind of drink, and smoking a different brand of cigarette.

Using the following information, the question is: who owns the fish?

# Exercise 3 - Einstein's riddle encoded in SAT

Problem data.

- The Brit lives in the red house.
- The Swede keeps dogs as pets.
- The Dane drinks tea.
- The green house is next to the white house, on the left.
- The owner of the green house drinks coffee.
- The person who smokes Pall Mall rears birds.
- The owner of the yellow house smokes Dunhill.
- The man living in the centre house drinks milk.
- The Norwegian lives in the first house.
- The man who smokes Blends lives next to the one who keeps cats.
- The man who keeps horses lives next to the man who smokes Dunhill.
- The man who smokes Blue Master drinks beer.
- The German smokes Prince.
- The Norwegian lives next to the blue house.
- The man who smokes Blends has a neighbour who drinks water.

# Exercise 3 - Einstein's riddle encoded in SAT

Todo.

- Describe how would you model the problem data to be solved by a SAT solver.
- Choose 5 of the problem data items above and encode them into DIMACS Format.

# Exercise 4 - Sudoku encoded in SAT

Problem. Represent a Sudoku puzzle as a SAT problem.

Solution.

- The grid for the puzzle is $9 \times 9$.
- Encoding Sudoku puzzles into CNF requires $9 \cdot 9 \cdot 9 = 729$ propositional variables.
- For each entry in the $9 \times 9$ grid $\mathcal{S}$, we associate 9 variables.
- Let us use the denotation $s_{xyz}$ to refer to variables.
- Variable $s_{xyz}$ is assigned true iff the entry in row x and column y is assigned number $z$.
- For example, $s_{483} = 1$ means that $\mathcal{S}[4, 8] = 3$.
- Naturally, the pre-assigned entries of the Sudoku grid will be represented as unit clauses.

# Exercise 4 - Sudoku encoded in SAT

The add the following constraints:

- There is at least one number in each entry:

$$\bigwedge_{x=1}^{9} \bigwedge_{y=1}^{9} \bigvee_{z=1}^{9} s_{xyz}$$

- Each number appears at most once in each row:

$$\bigwedge_{y=1}^{9} \bigwedge_{z=1}^{9} \bigwedge_{x=1}^{8} \bigwedge_{i=x+1}^{9} (\neg s_{xyz} \vee \neg s_{iyz})$$

- Each number appears at most once in each columns:

$$\bigwedge_{x=1}^{9} \bigwedge_{z=1}^{9} \bigwedge_{y=1}^{8} \bigwedge_{i=y+1}^{9} (\neg s_{xyz} \vee \neg s_{xiz})$$

# Exercise 4 - Sudoku encoded in SAT

The add the following constraints:

- Each number appears at most once in each $3 \times 3$ sub-grid:

$$\bigwedge_{z=1}^{9}\bigwedge_{i=0}^{2}\bigwedge_{j=0}^{2}\bigwedge_{x=1}^{3}\bigwedge_{y=1}^{3}\bigwedge_{k=y+1}^{3}\left(\neg s_{(3i+x)(3j+y)z} \vee \neg s_{(3i+x)(3j+k)z}\right)$$

$$\bigwedge_{z=1}^{9}\bigwedge_{i=0}^{2}\bigwedge_{j=0}^{2}\bigwedge_{x=1}^{3}\bigwedge_{y=1}^{3}\bigwedge_{k=x+1}^{3}\bigwedge_{l=1}^{3}\left(\neg s_{(3i+x)(3j+y)z} \vee \neg s_{(3i+k)(3j+l)z}\right)$$

The encoding is from the paper

I. Lynce, J. Ouaknine, *Sudoku as a SAT Problem* (link)

# Exercise 4 - Sudoku encoded in SAT

Todo

- Write a program in your favourite language to generate all constraints into DIMACS Format
- Solve a Sudoku puzzle with a SAT solver