§1. The MAXSAT Problem

- The well-known **SAT** problem is to determine if a boolean formula in Conjunctive Normal Form (CNF) has a satisfying truth assignment
  - A CNF formula is a conjunction of clauses
  - A clause is a disjunction of literals (variables or their negations)
  - $\square$ denotes the empty clause (falsified by every truth assignment)
  - A satisfying assignment assigns *true* to at least one literal of every clause.
- MAXSAT is an **optimization** extension of SAT that asks what is the maximum number of clauses that can be simultaneously satisfied

§1. The MAXSAT Problem

Example

$\mathcal{F} = (\neg x) \wedge (x \vee y) \wedge (\neg y) \wedge (z \vee w)$

§1. The MAXSAT Problem

Example

$\mathcal{F} = (\neg x) \wedge (x \vee y) \wedge (\neg y) \wedge (z \vee w)$
$\mathcal{F}$ is unsatisfiable, so no truth assignment can satisfy all 4 clauses.

§1. The MAXSAT Problem

Example

$\mathcal{F} = (\neg x) \wedge (x \vee y) \wedge (\neg y) \wedge (z \vee w)$
$\mathcal{F}$ is unsatisfiable, so no truth assignment can satisfy all 4 clauses.
Truth assignment $\pi = \{x, \neg y, z, \neg w\}$ satisfies all clauses except $(\neg x)$.

Example

$\mathcal{F} = (\neg x) \wedge (x \vee y) \wedge (\neg y) \wedge (z \vee w)$
$\mathcal{F}$ is unsatisfiable, so no truth assignment can satisfy all 4 clauses.
Truth assignment $\pi = \{x, \neg y, z, \neg w\}$ satisfies all clauses except $(\neg x)$.
$\Rightarrow$ π is a solution to the MAXSAT problem $\mathcal{F}$.

§1. The MAXSAT Problem

- Some clauses may be more important to satisfy than others
- This can be modeled by associating a positive cost with each clause C that will be incurred if C is *falsified*

Example

$\mathcal{F} = (\neg x, \infty) \wedge (x \vee y, 4) \wedge (\neg y, 1) \wedge (z \vee w, \infty)$

§1. The MAXSAT Problem

- Some clauses may be more important to satisfy than others
- This can be modeled by associating a positive cost with each clause C that will be incurred if C is *falsified*
- If it is mandatory to satisfy C, its cost is ∞ and C is called **hard**
- Otherwise, C is called **soft**

Example

$\mathcal{F} = (\neg x, \infty) \wedge (x \vee y, 4) \wedge (\neg y, 1) \wedge (z \vee w, \infty)$
In $\mathcal{F}$, $(\neg x, \infty)$ is a **hard** clause, and $(x \vee y, 4)$ is **soft** with cost 4.

§1. The MAXSAT Problem

- A truth assignment π has cost equal to the sum of the costs of the clauses it falsifies
- Goal: find an optimal feasible truth assignment, i.e., a truth assignment of minimum **finite** cost *mincost*$(\mathcal{F})$

Example

$\mathcal{F} = (\neg x, \infty) \wedge (x \vee y, 4) \wedge (\neg y, 1) \wedge (z \vee w, \infty)$
$\pi = \{\neg x, y, z, \neg w\}$ satisfies all clauses except $(\neg y, 1)$
π is optimal: *mincost*$(\mathcal{F})$=*cost*(π)=1.

§1. Notes

- We use $cost(C)$ to denote the cost of clause C.
- A solution must satisfy all hard clauses (else its cost will be infinite).
- A solution also satisfies a **maximum** total cost of soft clauses.
    - Casting as minimization problem more closely corresponds to how most MAXSAT solvers work.
- Many solutions might exist—typically we are only interested in finding one, sometimes only interested in finding out the cost of a solution.

§1.1. Categories of MAXSAT

- MAXSAT (ms) (standard MAXSAT): no hard clauses and all clause have weight 1.
  - Solution maximizes the number of satisfied clauses.
- Weighted MAXSAT (wms): no hard clauses.
- Partial MAXSAT (pms): have hard clauses but all soft clauses have weight 1.
- Weighted Partial MAXSAT (wpms): the version we have defined here (subsumes all other versions).

- Standard MAXSAT is most interesting for theory: it already has sufficient structure for theoretical insights.
- Other versions mostly an artifact of the limitations of earlier MAXSAT solvers.