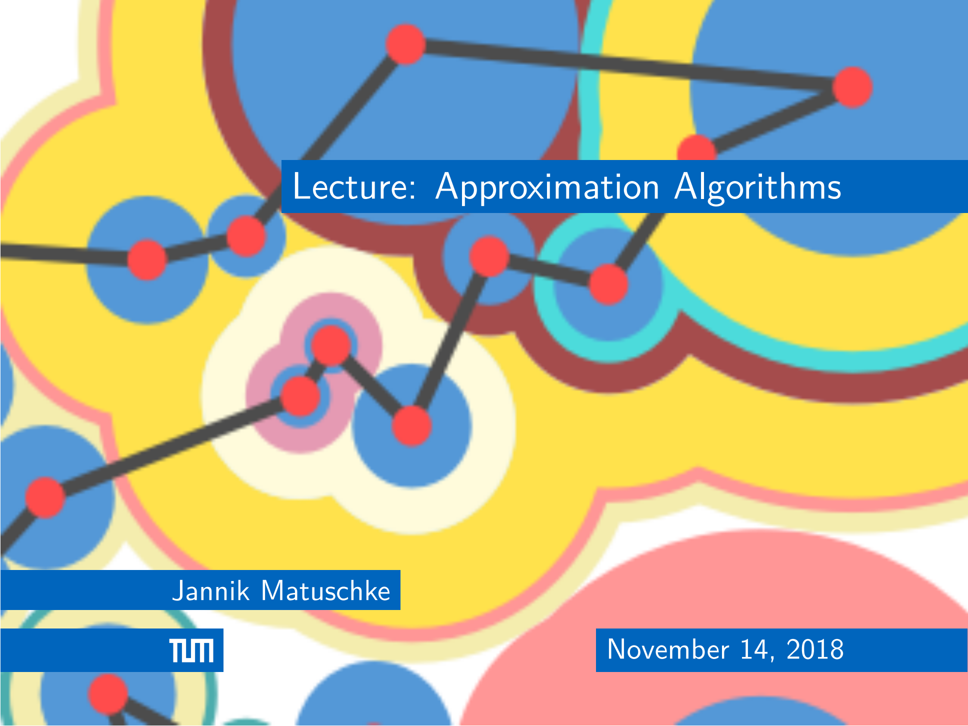


Lecture: Approximation Algorithms

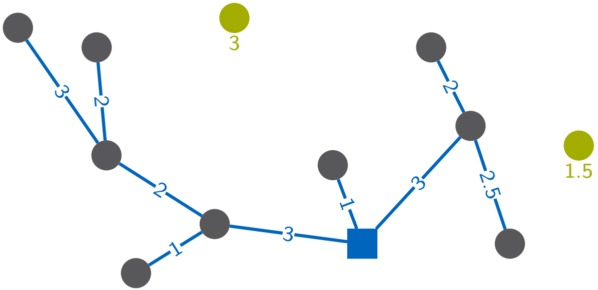
Jannik Matuschke

TUM

November 14, 2018

Previously:
LP Rounding for
Prize-collecting Steiner Tree

Approximation for PC Steiner Tree



Algorithm D (Deterministic Rounding)

1. Compute optimal solution (x^*, y^*) to LP.
2. Let $U := \{v \in V : y^*(v) \geq \alpha\}$.
3. Compute min spanning tree T on $U \cup \{r\}$.
4. Return T and U.

Algorithm B (Best of Many): select best $\alpha \in \{y^*(v) : v \in V\}$

Algorithm R (Randomized): $\alpha \in [\gamma, 1]$ uniformly at random

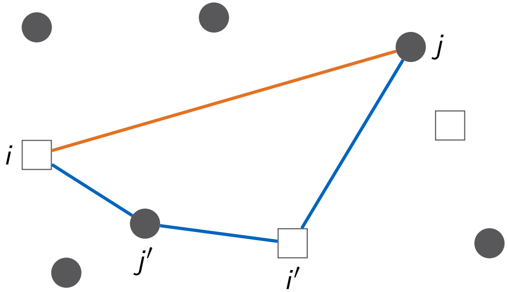
$\rightsquigarrow$ Algorithms B & R are 2.54-approximations

Now:
LP Rounding for Uncapacitated Facility Location

(Metric) Uncapacitated Facility Location

Input: facilities F, clients C, opening cost f_i for $i \in F$,
metric distances d_{ij} for $i \in F$ and $j \in C$

Task: find $S \subseteq F$, minimizing $\sum_{i \in S} f_i + \sum_{j \in C} \min_{i \in S} d_{ij}$

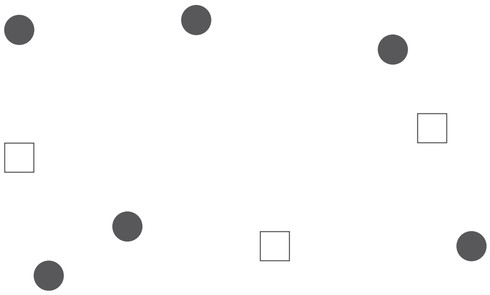


metric: $d_{ij} \leq d_{ij'} + d_{i'j'} + d_{i'j}$

(Metric) Uncapacitated Facility Location

Input: facilities F, clients C, opening cost f_i for $i \in F$, metric distances d_{ij} for $i \in F$ and $j \in C$

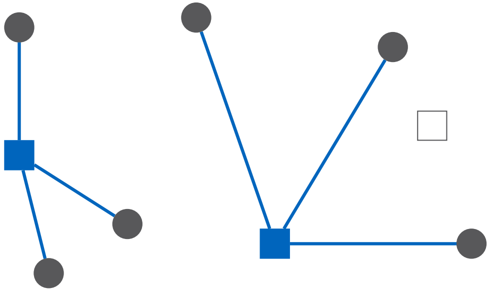
Task: find $S \subseteq F$, minimizing $\sum_{i \in S} f_i + \sum_{j \in C} \min_{i \in S} d_{ij}$



(Metric) Uncapacitated Facility Location

Input: facilities F, clients C, opening cost f_i for $i \in F$, metric distances d_{ij} for $i \in F$ and $j \in C$

Task: find $S \subseteq F$, minimizing $\sum_{i \in S} f_i + \sum_{j \in C} \min_{i \in S} d_{ij}$



LP relaxation

$$\begin{aligned} \min \quad & \sum_{i\in F}\sum_{j\in C} d_{ij}x_{ij} + \sum_{i\in F} f_i y_i & \\ \text{s.t.} \quad & \sum_{i\in F} x_{ij} = 1 & \forall\, j \in C \\ & y_i - x_{ij} \geq 0 & \forall\, i \in F, j \in C \\ & x_{ij} \in \{0,1\} & \forall\, i \in F, j \in C \\ & y_i \in \{0,1\} & \forall\, i \in F \end{aligned}$$

variables:

$x_{ij} = 1 \;\Leftrightarrow\; i$ serves j

$y_i = 1 \;\Leftrightarrow\; i \in S$

LP relaxation

$$\begin{aligned} \min \quad & \sum_{i\in F}\sum_{j\in C} d_{ij}x_{ij} + \sum_{i\in F} f_i y_i & \\ \text{s.t.} \quad & \sum_{i\in F} x_{ij} = 1 & \forall\, j \in C \\ & y_i - x_{ij} \geq 0 & \forall\, i \in F, j \in C \\ & x_{ij} \in \{0,1\} & \forall\, i \in F, j \in C \\ & y_i \in \{0,1\} & \forall\, i \in F \end{aligned}$$

variables:

$x_{ij} = 1 \;\Leftrightarrow\; i$ serves j

$y_i = 1 \;\Leftrightarrow\; i \in S$

LP relaxation

$$
\begin{aligned}
\min \quad & \sum_{i \in F} \sum_{j \in C} d_{ij} x_{ij} + \sum_{i \in F} f_i y_i \\
\text{s.t.} \quad & \sum_{i \in F} x_{ij} = 1 && \forall\, j \in C \\
& y_i - x_{ij} \geq 0 && \forall\, i \in F, j \in C \\
& x_{ij} \geq 0 && \forall\, i \in F, j \in C \\
& y_i \geq 0 && \forall\, i \in F
\end{aligned}
$$

variables:

$x_{ij} = 1 \iff i$ serves j

$y_i = 1 \iff i \in S$

LP relaxation

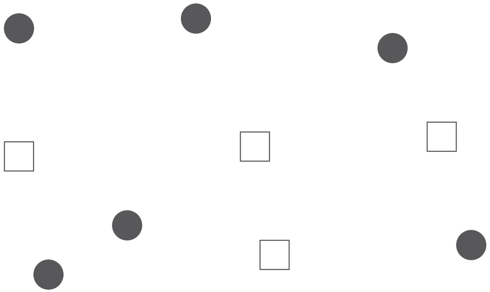
$$\min \sum_{i \in F} \sum_{j \in C} d_{ij} x_{ij} + \sum_{i \in F} f_i y_i$$

$$\begin{aligned} \text{s.t.} \quad \sum_{i \in F} x_{ij} &= 1 & \forall\, j \in C \\ y_i - x_{ij} &\geq 0 & \forall\, i \in F, j \in C \\ x_{ij} &\geq 0 & \forall\, i \in F, j \in C \\ y_i &\geq 0 & \forall\, i \in F \end{aligned}$$

variables:

$x_{ij} = 1 \Leftrightarrow i$ serves j

$y_i = 1 \Leftrightarrow i \in S$



LP relaxation

$$\min \sum_{i\in F}\sum_{j\in C} d_{ij}x_{ij} + \sum_{i\in F} f_i y_i$$

$$\text{s.t.} \quad \sum_{i\in F} x_{ij} = 1 \qquad \forall\, j\in C$$

$$y_i - x_{ij} \geq 0 \quad \forall\, i\in F, j\in C$$

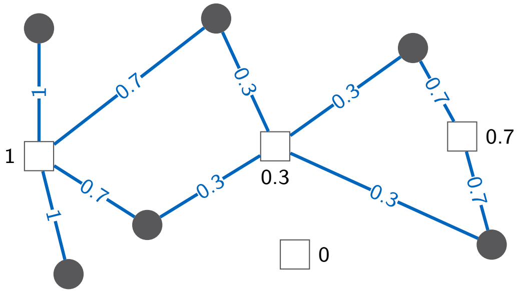
$$x_{ij} \geq 0 \quad \forall\, i\in F, j\in C$$

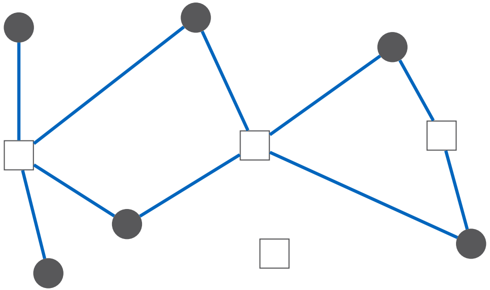
$$y_i \geq 0 \qquad \forall\, i\in F$$

variables:

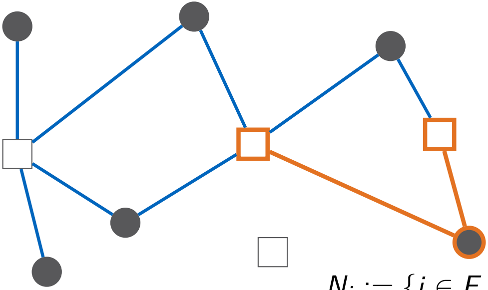
$x_{ij} = 1 \;\Leftrightarrow\; i$ serves j

$y_i = 1 \;\Leftrightarrow\; i \in S$



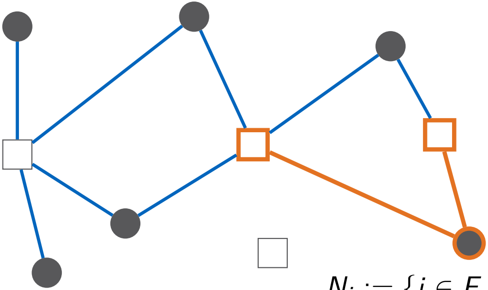


Ideas



$N_j := \{i \in F \,:\, x^*_{ij} > 0\}$

Ideas



$N_j := \{i \in F \ : \ x_{ij}^* > 0\}$

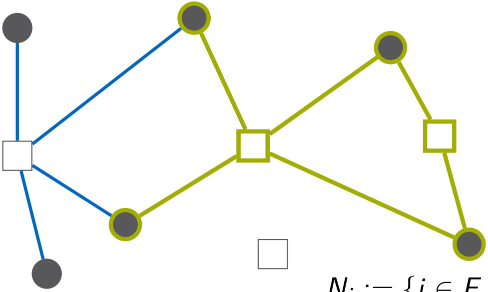
Lemma 6.4

Let $i \in N_j$, then $d_{ij} \leq v_i^*$.

Lemma 6.5

Let $i' \in N_j$ with $f_{i'} = \min_{i \in N_j} f_i$, then $f_{i'} \leq \sum_{i \in N_j} f_i y_i^*$.

Ideas



$$N_j := \{i \in F \ : \ x^*_{ij} > 0\}$$
$$N_j^2 := \{j' \in C \ : \ N_j \cap N_{j'} \neq \emptyset\}$$

Lemma 6.4

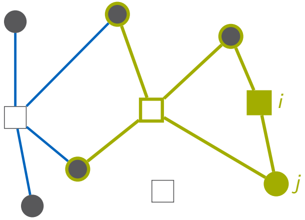
Let $i \in N_j$, then $d_{ij} \leq v^*_i$.

Lemma 6.5

Let $i' \in N_j$ with $f_{i'} = \min_{i \in N_j} f_i$, then $f_{i'} \leq \sum_{i \in N_j} f_i y^*_i$.

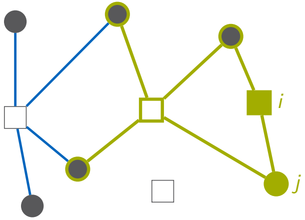
Algorithm D (Deterministic Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$
   Choose $j \in C'$ minimizing v_j^*.
   Choose $i \in N_j$ minimizing f_i.
   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.



Algorithm D (Deterministic Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$
   Choose $j \in C'$ minimizing v_j^*.
   Choose $i \in N_j$ minimizing f_i.
   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.

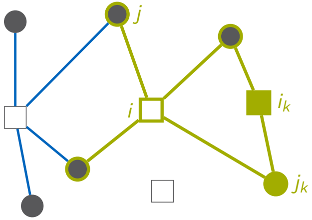


Theorem 6.6

Algorithm D is a 4-approximation algorithm for Uncapacitated Facility Location.

Algorithm D (Deterministic Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$
   Choose $j \in C'$ minimizing v_j^*.
   Choose $i \in N_j$ minimizing f_i.
   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.



Theorem 6.6

Algorithm D is a 4-approximation algorithm for Uncapacitated Facility Location.

Algorithm D* (Improved Deterministic Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$

   Choose $j \in C'$ minimizing v_j^*.

   Choose $i \in N_j$ minimizing $f_i + \sum_{j' \in N_j^2} d_{ij'}$.

   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.

Improved algorithm

Algorithm D* (Improved Deterministic Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$ $\qquad \Delta_j := \sum_{i \in F} d_{ij} x_{ij}$

   Choose $j \in C'$ minimizing $v_j^* + \Delta_j$.

   Choose $i \in N_j$ minimizing $f_i + \sum_{j' \in N_j^2} d_{ij'}$.

   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.

Improved algorithm

Algorithm R (Randomized Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$ $\Delta_j := \sum_{i \in F} d_{ij} x_{ij}$
   Choose $j \in C'$ minimizing $v_j^* + \Delta_j$.
   Choose $i \in N_j$ randomly according to probabilities x_{ij}^*.
   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.

Improved algorithm

Algorithm R (Randomized Rounding)

1. Compute optimal solutions (x^*, y^*) and (v^*, w^*) to LP relaxation and its dual.
2. Initialize $S := \emptyset$, $C' := C$.
3. while $(C' \neq \emptyset)$ $\quad \Delta_j := \sum_{i \in F} d_{ij} x_{ij}$

   Choose $j \in C'$ minimizing $v_j^* + \Delta_j$.

   Choose $i \in N_j$ randomly according to probabilities x_{ij}^*.

   $C' := C' \setminus N_j^2$ and $S := S \cup \{i\}$.
4. Return S.

Theorem 6.9

Algorithm R is a randomized 3-approximation algorithm for UFL.

Corollary 6.10

Algorithm D* is a 3-approximation algorithm for UFL.

Random Sampling: The Maximum Satisfiability Problem

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j \,:\, C_j \text{ is satisfied}} w_j$

$$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1} \qquad \underbrace{\neg x_1 \vee x_3}_{C_2} \qquad \underbrace{\neg x_3}_{C_3} \qquad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \qquad \underbrace{x_2 \vee \neg x_4}_{C_5}$$

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j : C_j \text{ is satisfied}} w_j$

$$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1} \qquad \underbrace{\neg x_1 \vee x_3}_{C_2} \qquad \underbrace{\neg x_3}_{C_3} \qquad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \qquad \underbrace{x_2 \vee \neg x_4}_{C_5}$$

$$w_1 = 2 \qquad w_2 = 3 \qquad w_3 = 1 \qquad w_4 = 2 \qquad w_5 = 1$$

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j\,:\,C_j \text{ is satisfied}} w_j$

$$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1} \checkmark \qquad \underbrace{\neg x_1 \vee x_3}_{C_2} \qquad \underbrace{\neg x_3}_{C_3} \qquad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \qquad \underbrace{x_2 \vee \neg x_4}_{C_5}$$

$$w_1 = 2 \qquad w_2 = 3 \qquad w_3 = 1 \qquad w_4 = 2 \qquad w_5 = 1$$

assignment:

$$x_1 = \texttt{true} \qquad x_2 = \texttt{true} \qquad x_3 = \texttt{false} \qquad x_4 = \texttt{true}$$

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j\,:\,C_j \text{ is satisfied}} w_j$

$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1}$ ✓ $\quad \underbrace{\neg x_1 \vee x_3}_{C_2}$ ✗ $\quad \underbrace{\neg x_3}_{C_3} \quad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \quad \underbrace{x_2 \vee \neg x_4}_{C_5}$

$w_1 = 2 \quad w_2 = 3 \quad w_3 = 1 \quad w_4 = 2 \quad w_5 = 1$

assignment:

$x_1 = \texttt{true} \quad x_2 = \texttt{true} \quad x_3 = \texttt{false} \quad x_4 = \texttt{true}$

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j\,:\,C_j \text{ is satisfied}} w_j$

$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1}$ ✓ $\quad \underbrace{\neg x_1 \vee x_3}_{C_2}$ ✗ $\quad \underbrace{\neg x_3}_{C_3}$ ✓ $\quad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4} \quad \underbrace{x_2 \vee \neg x_4}_{C_5}$

$w_1 = 2 \quad w_2 = 3 \quad w_3 = 1 \quad w_4 = 2 \quad w_5 = 1$

assignment:

$x_1 = \texttt{true} \quad x_2 = \texttt{true} \quad x_3 = \texttt{false} \quad x_4 = \texttt{true}$

Max Sat

Input: variables $x_1, \dots, x_n$, disjunctive clauses $C_1, \dots, C_m$, weights $w_1, \dots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j\,:\,C_j \text{ is satisfied}} w_j$

$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1}$ ✓ $\quad \underbrace{\neg x_1 \vee x_3}_{C_2}$ ✗ $\quad \underbrace{\neg x_3}_{C_3}$ ✓ $\quad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4}$ ✓ $\quad \underbrace{x_2 \vee \neg x_4}_{C_5}$

$w_1 = 2 \quad w_2 = 3 \quad w_3 = 1 \quad w_4 = 2 \quad w_5 = 1$

assignment:

$x_1 = \texttt{true} \quad x_2 = \texttt{true} \quad x_3 = \texttt{false} \quad x_4 = \texttt{true}$

Max Sat

Input: variables $x_1, \dots, x_n$, disjunctive clauses $C_1, \dots, C_m$, weights $w_1, \dots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j : C_j \text{ is satisfied}} w_j$

$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1}$ ✓ $\quad \underbrace{\neg x_1 \vee x_3}_{C_2}$ ✗ $\quad \underbrace{\neg x_3}_{C_3}$ ✓ $\quad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4}$ ✓ $\quad \underbrace{x_2 \vee \neg x_4}_{C_5}$ ✓

$w_1 = 2 \quad w_2 = 3 \quad w_3 = 1 \quad w_4 = 2 \quad w_5 = 1$

assignment:

$x_1 = \texttt{true} \quad x_2 = \texttt{true} \quad x_3 = \texttt{false} \quad x_4 = \texttt{true}$

Max Sat

Input: variables $x_1, \ldots, x_n$, disjunctive clauses $C_1, \ldots, C_m$, weights $w_1, \ldots, w_m \in \mathbb{R}_+$

Task: find a truth assignment maximizing $\sum_{j : C_j \text{ is satisfied}} w_j$

$\underbrace{x_1 \vee \neg x_2 \vee x_3}_{C_1}$ ✓ $\quad \underbrace{\neg x_1 \vee x_3}_{C_2}$ ✗ $\quad \underbrace{\neg x_3}_{C_3}$ ✓ $\quad \underbrace{x_2 \vee x_3 \vee x_4}_{C_4}$ ✓ $\quad \underbrace{x_2 \vee \neg x_4}_{C_5}$ ✓

$w_1 = 2 \quad w_2 = 3 \quad w_3 = 1 \quad w_4 = 2 \quad w_5 = 1$

assignment: weight: 6

$x_1 = \texttt{true} \quad x_2 = \texttt{true} \quad x_3 = \texttt{false} \quad x_4 = \texttt{true}$

Random Sampling

Algorithm 1: set each x_i to `true` with probability $1/2$ (independently)

Random Sampling

Algorithm 1: set each x_i to `true` with probability $1/2$ (independently)

Theorem 7.1

Algorithm 1 is a randomized $1/2$-approximation for Max Sat.

Random Sampling

Algorithm 1: set each x_i to `true` with probability $1/2$ (independently)

Theorem 7.1

Algorithm 1 is a randomized $1/2$-approximation for MAX SAT.

analysis shows something even better:

Algorithm 1 is a $1 - (1/2)^\ell$-approximation, where $\ell := \min_j |C_j|$.

Random Sampling

Algorithm 1: set each x_i to `true` with probability $1/2$ (independently)

Theorem 7.1

Algorithm 1 is a randomized $1/2$-approximation for Max Sat.

analysis shows something even better:
Algorithm 1 is a $1-(1/2)^\ell$-approximation, where $\ell := \min_j |C_j|$.

Theorem 7.2

There is no $(7/8+\varepsilon)$-approximation for Max E3-Sat
for any $\varepsilon > 0$, unless $P = NP$.

($|C_j| = 3$ for all j)

Derandomization

Can we make Algorithm 1 deterministic?

Derandomization

Can we make Algorithm 1 deterministic?

$$X_i := \begin{cases} 1 & \text{if Alg. 1 sets } x_i = \texttt{true} \\ 0 & \text{otherwise} \end{cases} \qquad W := \sum_{j\,:\,C_j \text{ is satisfied}} w_j$$

Derandomization

Can we make Algorithm 1 deterministic?

$$X_i := \begin{cases} 1 & \text{if Alg. 1 sets } x_i = \texttt{true} \\ 0 & \text{otherwise} \end{cases} \qquad W := \sum_{j\,:\,C_j \text{ is satisfied}} w_j$$

Idea: If $\mathbb{E}[W \mid X_1 = 1] > \mathbb{E}[W \mid X_1 = 0]$, we should set x_1 to `true`.

$\rightsquigarrow$ method of conditional expectations

Can we make Algorithm 1 deterministic?

$$X_i := \begin{cases} 1 & \text{if Alg. 1 sets } x_i = \texttt{true} \\ 0 & \text{otherwise} \end{cases} \qquad W := \sum_{j\,:\,C_j \text{ is satisfied}} w_j$$

Algorithm 2
For $i = 1$ to n:

1. $W_0 := \mathbb{E}[W \mid X_1 = b_1 \wedge \ldots \wedge X_{i-1} = b_{i-1} \wedge X_i = 0]$
2. $W_1 := \mathbb{E}[W \mid X_1 = b_1 \wedge \ldots \wedge X_{i-1} = b_{i-1} \wedge X_i = 1]$
3. If $W_1 \geq W_0$ then set $b_i := 1$ and $x_i := \texttt{true}$,
   otherwise set $b_i := 0$ and $x_i := \texttt{false}$.

Derandomization

Can we make Algorithm 1 deterministic?

$$X_i := \begin{cases} 1 & \text{if Alg. 1 sets } x_i = \texttt{true} \\ 0 & \text{otherwise} \end{cases} \qquad W := \sum_{j\,:\,C_j \text{ is satisfied}} w_j$$

Algorithm 2

For $i = 1$ to n:

1. $W_0 := \mathbb{E}[W \mid X_1 = b_1 \wedge \ldots \wedge X_{i-1} = b_{i-1} \wedge X_i = 0]$
2. $W_1 := \mathbb{E}[W \mid X_1 = b_1 \wedge \ldots \wedge X_{i-1} = b_{i-1} \wedge X_i = 1]$
3. If $W_1 \geq W_0$ then set $b_i := 1$ and $x_i := \texttt{true}$, otherwise set $b_i := 0$ and $x_i := \texttt{false}$.

Theorem 7.3

Algorithm 2 is a $1/2$-approximation algorithm for MAX SAT.

Today you learnt ...

You need good bounds on
what you can lose in a coin toss.