# Randomized Algorithms

Ioannis Caragiannis (this time) and Kasper Green Larsen

# Today

- Maximum Satisfiability
- Simple randomized algorithms
- Linear programming and randomized rounding

# The MAXSAT problem

# Variables, literals, and clauses

- Boolean **variables**: $x_i$ for $i = 1, 2, \dots, n$
- **Literals**: the appearance of a variable either positively, as $x_i$, or negatively, as $\bar{x}_i$
- **Clauses**: $C_j$ for $j = 1, 2, \dots, m$
- Clause $C_j$ consists of a set of literals. For example: $C_1 = (x_2, \bar{x}_3, x_5, \bar{x}_8, \bar{x}_9)$
- Boolean **assignment**: an assignment of binary/logical value (0/1 or false/true) to the variables
- **A clause is true if some of its literals is true**, i.e., consider a clause as an OR operation applied on its literals
- E.g., $C_1 = x_2 \vee \bar{x}_3 \vee x_5 \vee \bar{x}_8 \vee \bar{x}_9$

# The SAT decision problem

- SAT: Given a set of $n$ variables and a set of $m$ clauses over these variables, **is there an assignment that makes all clauses true?**
- Example: $C_1 = x_2 \vee \bar{x}_3, C_2 = x_1 \vee x_3, C_3 = \bar{x}_2 \vee x_3, C_4 = \bar{x}_1 \vee \bar{x}_3$
- Here, the assignment $x_1 = 1, x_2 = 0$, and $x_3 = 0$ satisfies all clauses (makes them all true). Also: $x_1 = 0, x_2 = 1$, and $x_3 = 1$
- Very important as it can be used to represent any non-deterministic computation (Cook, 1970)
- SAT is the most basic **NP-complete** problem
- 3SAT is NP-complete (each clause consists of exactly three literals)
- 2SAT is polynomial-time solvable

# Optimization version of satisfiability

- Input:
  - A set of $n$ variables $x_1, x_2, \ldots, x_n$ and $m$ clauses $C_1, C_2, \ldots, C_m$ over these variables
  - In addition, each clause $C_j$ has a positive weight $w_j$
- Some notation: $\boldsymbol{C_j(x)}$ is equal to 1 if the assignment $x$ makes clause $C_j$ true, and is equal to 0 otherwise
- The **MAXSAT** problem: Compute an assignment to the variables so that the total weight in satisfied clauses is maximized
- I.e., find assignment $x$ so that $\sum_{j=1}^{m} C_j(x) \cdot w_j$ is maximized
- As SAT is NP-complete, **MAXSAT is NP-hard** (i.e., we should not expect to solve MAXSAT exactly with polynomial-time algorithms)

# Approximation algorithms

- We are interested in algorithms that always return **nearly-optimal solutions**
- Optimal assignment $\hat{x}$: an assignment to the variables so that $\sum_{j=1}^{m} C_j(\hat{x}) \cdot w_j$ is as high as possible
- For a factor $\rho \in [0,1]$, a **$\rho$-approximation algorithm** for MAXSAT computes an assignment $x$ so that

$$\underbrace{\sum_{j=1}^{m} C_j(x) \cdot w_j}_{\text{quality/benefit/gain of assignment } x} \geq \rho \cdot \underbrace{\sum_{j=1}^{m} C_j(\hat{x}) \cdot w_j}_{\text{optimal quality/benefit/gain}} = \rho \cdot \mathrm{OPT}$$

# Approximation algorithms

- We would like the approximation factor $\rho$ to be **as close to 1 as possible**
- This is not always possible though: MAXSAT is **NP-hard to approximate** within a factor better than 7/8, even when all clauses have exactly three literals and weight 1 (Hastad, 2001)
- Today: How high can $\rho$ become using **randomized algorithms**?
- Our goal: compute a possibly random assignment $x$ so that

$$\underbrace{\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right]}_{\text{expected quality of } x} \geq \rho \cdot \mathrm{OPT}$$

# A simple randomized algorithm

# A simple randomized algorithm

- **Set each variable to either 0 or 1 equiprobably** and independently of the other variables
- Standard steps in the analysis:

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

# A simple randomized algorithm

- **Set each variable to either 0 or 1 equiprobably** and independently of the other variables
- Standard steps in the analysis:

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

linearity of expectation

recall that $C_j(x) \in \{0,1\}$

# A simple randomized algorithm

- **Set each variable to either 0 or 1 equiprobably** and independently of the other variables
- Standard steps in the analysis:

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

- We will show that $\mathbf{Pr}[\boldsymbol{C_j(x)} = \mathbf{1}] \geq \mathbf{1/2}$ for every clause $C_j$. Then,

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] \geq \frac{1}{2} \cdot \underbrace{\sum_{j=1}^{m} w_j}_{\text{total weight in all clauses}} \geq \frac{1}{2} \cdot \mathrm{OPT}$$

# Analysis

- We need to show that $\Pr[C_j(x) = 1] \geq 1/2$ for every clause $C_j$.
- If $C_j$ contains a positive and a negative literals of the same variable, then $C_j(x) = 1$ with certainty
- Otherwise, each literal is independendently set to either 0 or 1 equiprobably
- If $C_j$ contains $k$ literals in total, the probability that none of them is set to 1 is $2^{-k}$
- Hence, $\Pr\left[C_j(x) = 1\right] = 1 - 2^{-|C_j|} \geq 1/2$

# Observations

- The most difficult clauses seem to be **those with just a single literal**
- If all clauses have at least **two literals**, the approximation ratio we get is 3/4
- If all clauses have at least **three literals**, the approximation ratio we get is 7/8, i.e., **matching the inapproximability bound** of Hastad (2001)

# Flipping biased coins

# A better algorithm

- Without loss of generality, assume that for every variable $x_i$, the weight of the clause that consists only of literal $x_i$ is at least as high as the weight of the clause that consist only of literal $\bar{x}_i$
- If this is not the case, exchange literals $x_i$ and $\bar{x}_i$ wherever they appear
- Denote by $C$ the set of all clauses besides the ones that consist of a single negative literal
- Hence, $\mathrm{OPT} \le \sum_{j \in C} w_j$
- Algorithm: Set each variable to 1 with probability $\boldsymbol{p > 1/2}$ and to 0 with probability $1 - p$, independently on the other variables

# A better algorithm

- Without loss of generality, assume that for every variable $x_i$, the weight of the clause that consists only of literal $x_i$ is at least as high as the weight of the clause that consist only of literal $\bar{x}_i$
- If this is not the case, exchange literals $x_i$ and $\bar{x}_i$ wherever they appear
- Denote by $C$ the set of all clauses besides the ones that consist of a single negative literal
- Hence, $\boxed{\mathrm{OPT} \leq \sum_{j \in C} w_j}$ ⟵ better bound for OPT
- Algorithm: Set each variable to 1 with probability $\boldsymbol{p > 1/2}$ and to 0 with probability $1 - p$, independently on the other variables

# A better algorithm (analysis)

- In the analysis, we will account only for the contribution from clauses of $C$

$$\mathbb{E}\left[\sum_{j\in C} C_j(x)\cdot w_j\right] = \sum_{j\in C} \mathbb{E}[C_j(x)]\cdot w_j = \sum_{j\in C} \Pr[C_j(x)=1]\cdot w_j$$

- A clause with a single positive literal has $\Pr\left[C_j(x)=1\right]=p$
- A clause with at least two literals, of which $a$ are negative and $b$ are positive, has $\Pr\left[C_j(x)=0\right]=p^a(1-p)^b < p^{a+b} \leq p^2$
- Hence, $\Pr\left[C_j(x)=1\right] \geq 1-p^2$

# A better algorithm (analysis)

- In the analysis, we will account only for the contribution from clauses of $C$

$$\mathbb{E}\left[\sum_{j\in C} C_j(x)\cdot w_j\right] = \sum_{j\in C} \mathbb{E}[C_j(x)]\cdot w_j = \sum_{j\in C} \Pr[C_j(x) = 1]\cdot w_j$$

- A clause with a single positive literal has $\Pr\left[C_j(x) = 1\right] = p$
- A clause with at least two literals, of which $a$ are negative and $b$ are positive, has $\Pr\left[C_j(x) = 0\right] = p^a(1-p)^b < p^{a+b} \leq p^2$
- Hence, $\Pr\left[C_j(x) = 1\right] \geq 1 - p^2$

# A better algorithm (analysis)

- We have

$$\mathbb{E}\left[\sum_{j\in C} C_j(x)\cdot w_j\right] \geq \sum_{j\in C} \min\{p, 1-p^2\}\cdot w_j \geq \min\{p, 1-p^2\}\cdot \mathrm{OPT}$$

- Setting $p = \frac{\sqrt{5}-1}{2} \approx 0.618$, we have $\min\{p, 1-p^2\} = 0.618$, which yields

$$\mathbb{E}\left[\sum_{j\in C} C_j(x)\cdot w_j\right] \geq 0.618\cdot \mathrm{OPT}$$

# Linear programming and combinatorial optimization

# An integer linear program for MAXSAT

- Use the integer variable $y_i$ to denote whether the boolean variable $x_i$ is true ($y_i = 1$) or false ($y_i = 0$)
- Use the integer variable $z_j$ to denote whether an assignment to the boolean variables $x$ satisfies clause $C_j$ ($z_j = 1$) or not ($z_j = 0$)
- Some additional notation:
  - Let $P_j$ (respectively, $N_j$) be the list of indices $i$ so that variable $x_i$ appears as positive literal $x_i$ (respectively, negative literal $\bar{x}_i$) in $C_j$
  - For example, $C_1 = (x_2, \bar{x}_3, x_5, \bar{x}_8, \bar{x}_9)$. Here, $P_1 = \{2, 5\}$ and $N_1 = \{3, 8,9\}$

# An integer linear program for MAXSAT

$$\text{maximize} \sum_{j=1}^{m} w_j \cdot z_j$$

$$\text{subject to} \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \geq z_j \qquad \text{for } j = 1, 2, \dots, m$$

$$y_i \in \{0,1\} \qquad \text{for } i = 1, 2, \dots, n$$

$$z_j \in \{0,1\} \qquad \text{for } j = 1, 2, \dots, m$$

integrality constraints

# An integer linear program for MAXSAT

- What can we do with it?
- Not much: it is an **equivalent formulation of an NP-hard problem**
- Instead, by **relaxing the integrality constraint**, we get a **linear program** that is **solvable in polynomial time** (Khachiyan, 1979)
- Its solution can be far from what we need but can be very **helpful**

# A linear programming relaxation for MAXSAT

$$\text{maximize} \sum_{j=1}^{m} w_j \cdot z_j$$

$$\text{subject to} \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i) \geq z_j \qquad \text{for } j = 1, 2, \dots, m$$

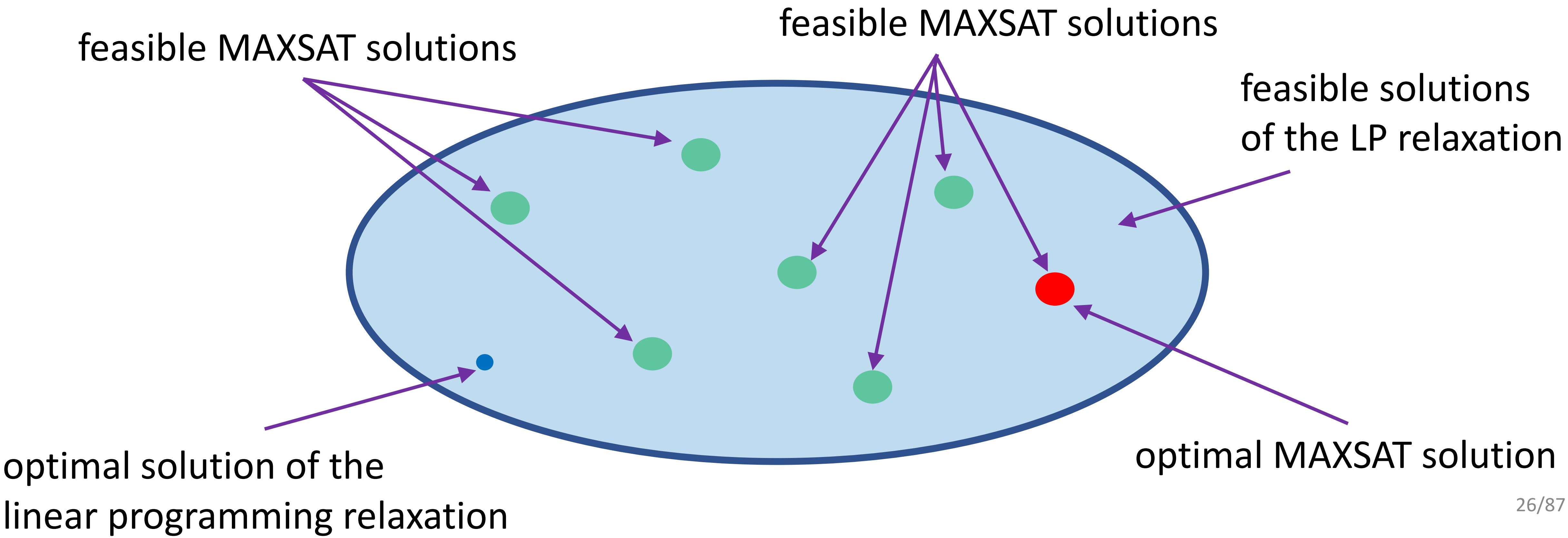
$$0 \leq y_i \leq 1 \qquad \text{for } i = 1, 2, \dots, n$$

$$0 \leq z_j \leq 1 \qquad \text{for } j = 1, 2, \dots, m$$

integrality constraints
are now relaxed

# A linear programming relaxation for MAXSAT

- The structure of feasible solutions of the linear programming relaxation



# A linear programming relaxation for MAXSAT

Why is it **useful**?

- First, it provides a hopefully better lower bound of OPT (**better analysis**)
- Let $y^*, z^*$ be the optimal feasible solution to the linear programming relaxation
- Then, $\boxed{\mathrm{OPT} \le \sum_{j=1}^{m} w_j \cdot z_j^*}$

- Second, it gives us some indication of how "good solutions" look like (**better algorithm design**)
- E.g., **round each fractional variable** to the closer integer value

# A general recipe for approximation algorithms

- Formulate the problem as an **integer linear program**
- Solve its linear programming relaxation to get a **fractional solution**
- **Round the fractional solution** to get a solution for the original problem

# Combinatorial optimization and linear programming: Some examples

# Set cover

- Input: a universe $U$ of elements, a collection $C$ of subsets of $U$, and positive weight $w(S)$ for each set $S$ of $C$
- Goal: find a subcollection of $C$ of minimum total weight, so that each element of $U$ belongs in at least one set of the subcollection

$$
\begin{aligned}
&\text{minimize} \sum_{S \in C} w(S) \cdot x_S \\
&\text{subject to} \sum_{S \in C : e \in S} x_S \geq 1 \qquad && \forall e \in U \\
&\qquad\qquad x_S \in \{0,1\} && \forall S \in C
\end{aligned}
$$

# Vertex cover

- Input: a graph $G = (V, E)$ with positive weight $w(v)$ at every node $v \in V$
- Goal: compute a set $S$ of nodes of minimum total weight, so that each edge has at least one endpoint in $S$

$$
\begin{aligned}
\text{minimize} \sum_{v \in V} w(v) \cdot x_v & \\
\text{subject to } x_u + x_v \geq 1 & \qquad \forall (u, v) \in E \\
x_v \in \{0,1\} & \qquad \forall v \in V
\end{aligned}
$$

# Maximum degree-constrained subgraph

- Input: a graph $G = (V, E)$, positive weight $w(e)$ for each edge e $\in E$ and an integer bound $\Delta(v)$ for every node $v \in V$
- Goal: compute a subgraph of $G$, consisting of edges of maximum total weight so that the degree of node $v$ in the subgraph does not exceed $\Delta(v)$

$$\text{maximize} \sum_{e \in E} w(e) \cdot x(e)$$

$$\text{subject to} \sum_{e \in E(v)} x(e) \leq \Delta(v) \qquad \forall v \in V$$

$$x(e) \in \{0,1\} \qquad \forall e \in E$$

set of edges incident at node $v$

# Randomized rounding

# A general recipe for approximation algorithms

- Formulate the problem as an **integer linear program**
- Solve its linear programming relaxation to get a **fractional solution**
- **Round the fractional solution** to get a solution for the original problem

- Today: **randomized rounding**

# Simple randomized rounding for MAXSAT

- Let $y^*, z^*$ be an optimal fractional solution of the linear programming relaxation
- **Set each variable $x_i$ to 1 with probability $y_i^*$** and to 0 with probability $1 - y_i^*$, independently of the other variables

# Simple randomized rounding for MAXSAT

- Let $y^*, z^*$ be an optimal fractional solution of the linear programming relaxation
- **Set each variable $x_i$ to 1 with probability $y_i^*$** and to 0 with probability $1 - y_i^*$, independently of the other variables
- Analysis roadmap:

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

$$\geq \cdots \geq \rho \cdot \sum_{j=1}^{m} w_j \cdot z_j^* \geq \rho \cdot \mathrm{OPT}$$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

$$\Pr\left[C_j(x) = 0\right] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

$$\Pr[C_j(x) = 0] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \prod_{i \in P_j} e^{-y_i^*} \cdot \prod_{i \in N_j} e^{y_i^* - 1}$$

$e^y \geq y + 1, \forall y \in \mathbb{R}$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

$$\Pr\left[C_j(x) = 0\right] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \prod_{i \in P_j} e^{-y_i^*} \cdot \prod_{i \in N_j} e^{y_i^* - 1} = \exp\left(-\left(\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)\right)\right)$$

$e^y \geq y + 1, \forall y \in \mathbb{R}$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

$$\Pr[C_j(x) = 0] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \prod_{i \in P_j} e^{-y_i^*} \cdot \prod_{i \in N_j} e^{y_i^* - 1} = \exp\left(-\left(\boxed{\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)}\right)\right)$$

$e^y \geq y + 1, \forall y \in \mathbb{R}$

# A linear programming relaxation for MAXSAT

LP constraint for clause $C_j$

$$\text{maximize} \sum_{j=1}^{m} w_j \cdot z_j$$

$$\text{subject to } \boxed{\sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i)} \geq z_j \qquad \text{for } j = 1, 2, \ldots, m$$

$$0 \leq y_i \leq 1 \qquad \text{for } i = 1, 2, \ldots, n$$

$$0 \leq z_j \leq 1 \qquad \text{for } j = 1, 2, \ldots, m$$

# Analysis of randomized rounding

- The clause $C_j$ is false if each of its literals are false
- The probability that a positive literal $x_i$ is false is $1 - y_i^*$
- The probability that a negative literal $\bar{x}_i$ is false is $y_i^*$

$$\Pr\left[C_j(x) = 0\right] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\le \prod_{i \in P_j} e^{-y_i^*} \cdot \prod_{i \in N_j} e^{y_i^* - 1} = \exp\left(-\left(\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)\right)\right) \le \exp(-z_j^*)$$

$e^y \ge y + 1, \forall y \in \mathbb{R}$

LP constraint for clause $C_j$

# Analysis of randomized rounding

- So, we have shown $\Pr\left[C_j(x) = 1\right] \geq 1 - \exp(-z_j^*)$

# Analysis of randomized rounding

nonlinear dependence on $z_j^*$

- So, we have shown $\Pr\left[C_j(x) = 1\right] \geq 1 - \exp(-z_j^*)$

# Simple randomized rounding for MAXSAT

- Let $y^*, z^*$ be an optimal fractional solution of the linear programming relaxation
- Set each variable $x_i$ to 1 with probability $y_i^*$ and to 0 with probability $1 - y_i^*$, independently of the other variables
- Analysis roadmap:

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

$$\geq \cdots \geq \boxed{\rho \cdot \sum_{j=1}^{m} w_j \cdot z_j^*} \geq \rho \cdot \mathrm{OPT}$$

we are still here

target

# Analysis of randomized rounding

nonlinear dependence on $z_j^*$

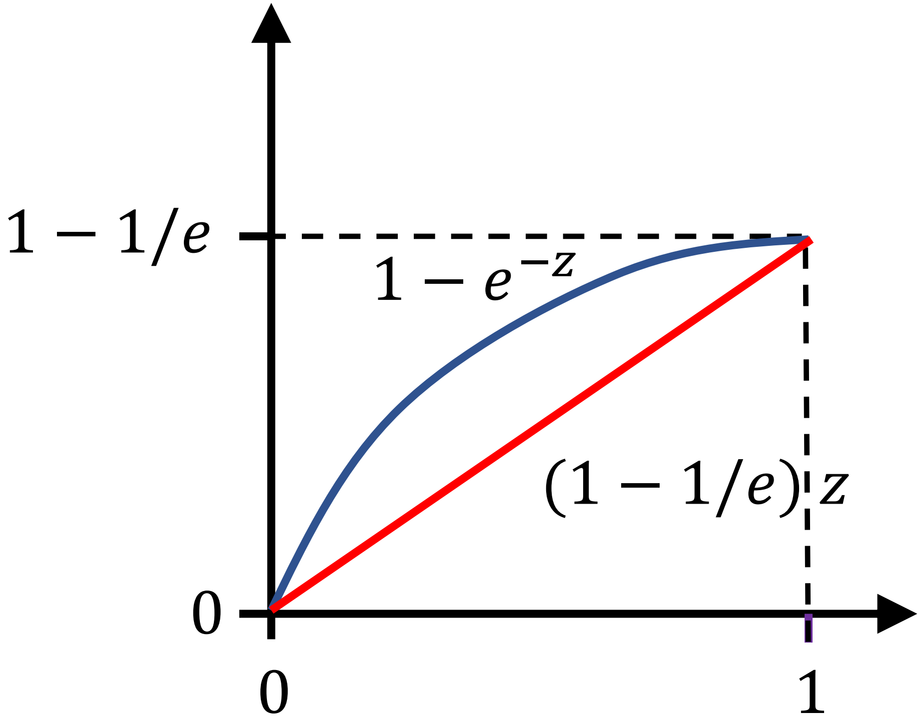
- So, we have shown $\Pr\left[C_j(x) = 1\right] \geq 1 - \exp(-z_j^*)$

# Analysis of randomized rounding

nonlinear dependence on $z_j^*$

- So, we have shown $\Pr\left[C_j(x) = 1\right] \geq 1 - \exp(-z_j^*)$
- Observe that the function $1 - e^{-z}$ is concave; thus, $1 - e^{-z} \geq (1 - 1/e)\, z$
- Hence, $\Pr\left[C_j(x) = 1\right] \geq (1 - 1/e)\, z_j^*$

linear dependence on $z_j^*$



# Analysis of randomized rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

# Analysis of randomized rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

$$\geq \sum_{j=1}^{m} \left(1 - \frac{1}{e}\right) z_j^* \cdot w_j$$

# Analysis of randomized rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x)\cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)]\cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x)=1]\cdot w_j$$

$$\geq \sum_{j=1}^{m}\left(1-\frac{1}{e}\right) z_j^* \cdot w_j = \left(1-\frac{1}{e}\right)\cdot\sum_{j=1}^{m} w_j\cdot z_j^*$$

# Analysis of randomized rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x)\cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)]\cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x)=1]\cdot w_j$$

$$\geq \sum_{j=1}^{m}\left(1-\frac{1}{e}\right) z_j^* \cdot w_j = \left(1-\frac{1}{e}\right)\cdot \sum_{j=1}^{m} w_j \cdot z_j^* \geq \left(1-\frac{1}{e}\right)\cdot \mathrm{OPT}$$

# Analysis of randomized rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x)\cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)]\cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x)=1]\cdot w_j$$

$$\geq \sum_{j=1}^{m}\left(1-\frac{1}{e}\right) z_j^* \cdot w_j = \left(1-\frac{1}{e}\right)\cdot \sum_{j=1}^{m} w_j \cdot z_j^* \geq \left(1-\frac{1}{e}\right)\cdot \mathrm{OPT}$$

- Notice that $1-\frac{1}{e} = 0.632 > 0.618$

# Choosing the better of two solutions

# Choosing the better of two solutions

- Run the algorithm that computes an assignment $x^1$ by setting each variable to 1 or 0 equiprobably
- Run the randomized rounding algorithm and denote the assignment returned by $x^2$
- Pick the best among the two solutions

# Analysis roadmap

# Analysis roadmap

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,, \sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right]$$

# Analysis roadmap

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,,\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right] \geq \mathbb{E}\left[\frac{1}{2}\sum_{j=1}^{m} C_j(x^1)\cdot w_j+\frac{1}{2}\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right]$$

# Analysis roadmap

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,,\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right] \geq \mathbb{E}\left[\frac{1}{2}\sum_{j=1}^{m} C_j(x^1)\cdot w_j + \frac{1}{2}\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right]$$

$$= \frac{1}{2}\sum_{j=1}^{m}\left(\mathbb{E}\left[C_j(x^1)\right] + \mathbb{E}[C_j(x^2)]\right)\cdot w_j$$

# Analysis roadmap

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,,\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right] \geq \mathbb{E}\left[\frac{1}{2}\sum_{j=1}^{m} C_j(x^1)\cdot w_j + \frac{1}{2}\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right]$$

$$= \frac{1}{2}\sum_{j=1}^{m}\big(\mathbb{E}\big[C_j(x^1)\big] + \mathbb{E}[C_j(x^2)]\big)\cdot w_j = \frac{1}{2}\sum_{j=1}^{m}\big(\Pr\big[C_j(x^1) = 1\big] + \Pr[C_j(x^2) = 1]\big)\cdot w_j$$

# Analysis roadmap

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,,\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right] \geq \mathbb{E}\left[\frac{1}{2}\sum_{j=1}^{m} C_j(x^1)\cdot w_j + \frac{1}{2}\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right]$$

$$= \frac{1}{2}\sum_{j=1}^{m}\left(\mathbb{E}\left[C_j(x^1)\right] + \mathbb{E}[C_j(x^2)]\right)\cdot w_j = \frac{1}{2}\sum_{j=1}^{m}\left(\Pr\left[C_j(x^1) = 1\right] + \Pr[C_j(x^2) = 1]\right)\cdot w_j$$

$$\geq \cdots \geq \rho\cdot\sum_{j=1}^{m} w_j\cdot z_j^* \geq \rho\cdot \mathrm{OPT}$$

# Analysis of random assignment

- Recall that we proved $\Pr\left[C_j(x^1) = 1\right] = 1 - 2^{-|C_j|} \geq \left(1 - 2^{-|C_j|}\right) z_j^*$
- I.e., **the coefficient of $z_j^*$ increases with the number of literals in $C_j$**

# Analysis of random assignment

- Recall that we proved $\Pr\left[C_j(x^1) = 1\right] = 1 - 2^{-|C_j|} \geq \left(1 - 2^{-|C_j|}\right) z_j^*$
- I.e., **the coefficient of $z_j^*$ increases with the number of literals in $C_j$**
- Unfortunately, the coefficient of $z_j^*$ in our analysis of randomized rounding does not depend on the number of literals in $C_j$
- Different analysis of randomized rounding is needed!

# Alternative analysis of randomized rounding

- Useful tool: **arithmetic-geometric mean inequality**
- For any non-negative $a_1, a_2, \ldots, a_k$

$$\left(\prod_{t=1}^{k} a_t\right)^{1/k} \leq \frac{1}{k}\sum_{t=1}^{k} a_t$$

# Alternative analysis of randomized rounding

$$\Pr\left[C_j(x^2) = 0\right] = \prod_{i \in P_j} \Pr[x_i^2 = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i^2 = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

# Alternative analysis of randomized rounding

$$\Pr[C_j(x^2) = 0] = \prod_{i \in P_j} \Pr[x_i^2 = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i^2 = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \left[ \frac{1}{|C_j|} \left( \sum_{i \in P_j} (1 - y_i^*) + \sum_{i \in N_j} y_i^* \right) \right]^{|C_j|}$$

arithmetic-geometric
mean inequality

# Alternative analysis of randomized rounding

$$\Pr[C_j(x^2) = 0] = \prod_{i \in P_j} \Pr[x_i^2 = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i^2 = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \left[\frac{1}{|C_j|}\left(\sum_{i \in P_j} (1 - y_i^*) + \sum_{i \in N_j} y_i^*\right)\right]^{|C_j|} = \left[1 - \frac{1}{|C_j|}\left(\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)\right)\right]^{|C_j|}$$

arithmetic-geometric
mean inequality

# Alternative analysis of randomized rounding

$$\Pr[C_j(x^2) = 0] = \prod_{i \in P_j} \Pr[x_i^2 = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i^2 = 0] = \prod_{i \in P_j} (1 - y_i^*) \cdot \prod_{i \in N_j} y_i^*$$

$$\leq \left[\frac{1}{|C_j|}\left(\sum_{i \in P_j} (1 - y_i^*) + \sum_{i \in N_j} y_i^*\right)\right]^{|C_j|} = \left[1 - \frac{1}{|C_j|}\left(\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)\right)\right]^{|C_j|}$$

$$\leq \left(1 - \frac{z_j^*}{|C_j|}\right)^{|C_j|}$$

arithmetic-geometric mean inequality

LP constraint for clause $C_j$

# Bounding $\Pr\left[C_j(x^2) = 1\right]$

nonlinear dependence on $z_j^*$

- Hence, $\Pr\left[C_j(x^2) = 1\right] \geq 1 - \left(1 - \frac{z_j^*}{|C_j|}\right)^{|C_j|}$

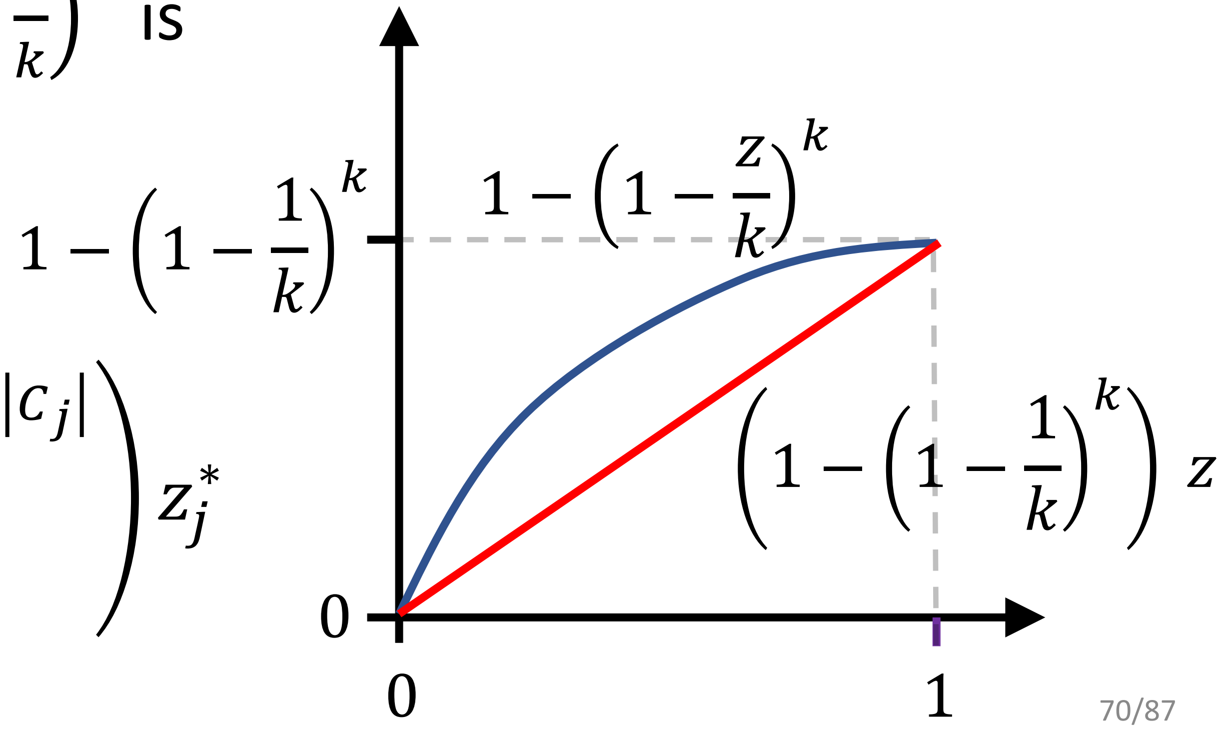
# Bounding $\Pr[C_j(x^2) = 1]$

nonlinear dependence on $z_j^*$

- Hence, $\Pr[C_j(x^2) = 1] \geq 1 - \left(1 - \frac{z_j^*}{|C_j|}\right)^{|C_j|}$
- Observe that the function $1 - \left(1 - \frac{z}{k}\right)^k$ is concave wrt $z$
- Hence,

$$\Pr[C_j(x^2) = 1] \geq \left(1 - \left(1 - \frac{1}{|C_j|}\right)^{|C_j|}\right) z_j^*$$

linear dependence on $z_j^*$



# Putting everything together

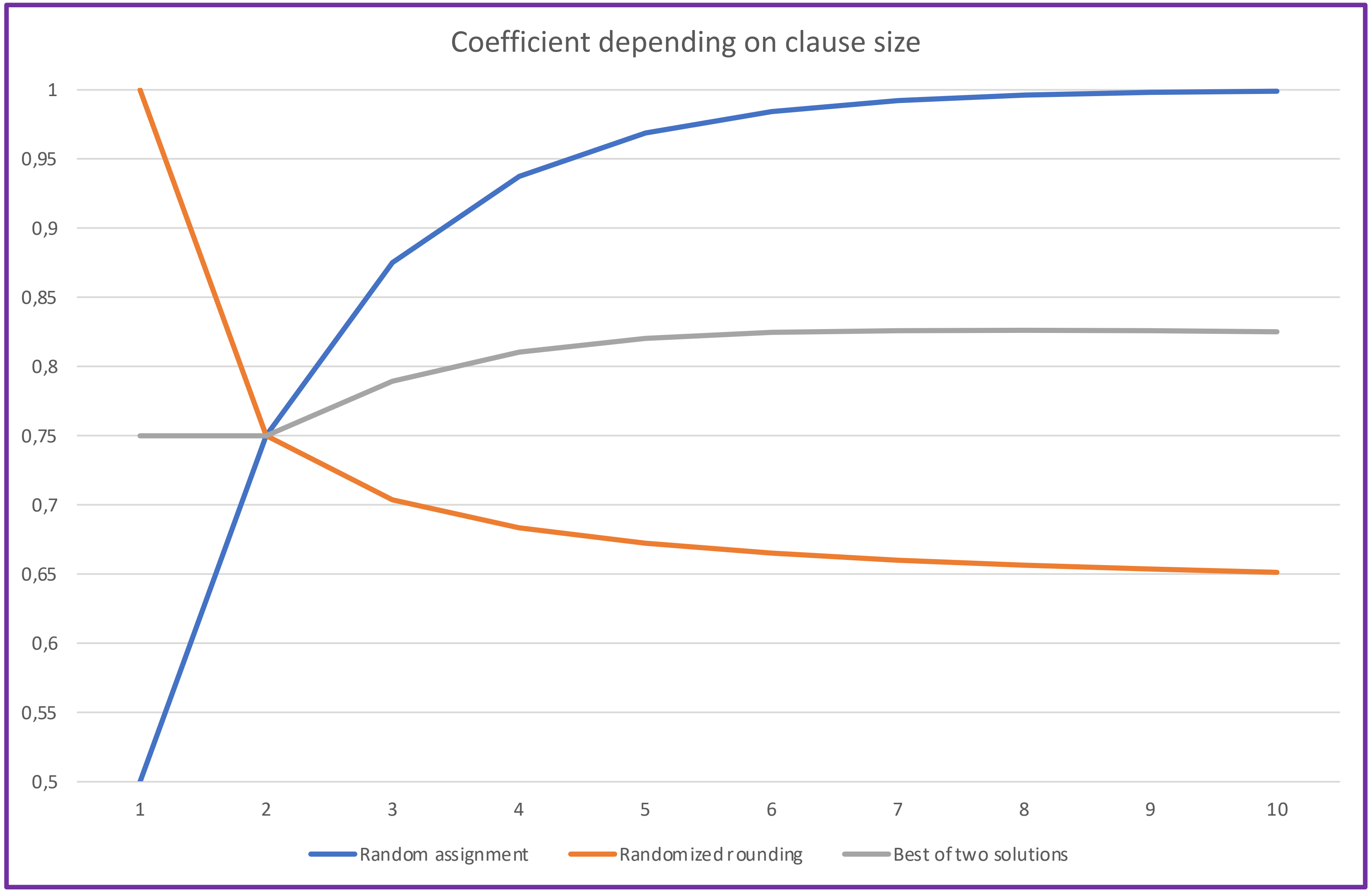
So far, we have proved:

- $\Pr\left[C_j(x^1) = 1\right] \geq \left(1 - 2^{-|C_j|}\right) z_j^*$ and $\Pr\left[C_j(x^2) = 1\right] \geq \left(1 - \left(1 - \frac{1}{|C_j|}\right)^{|C_j|}\right) z_j^*$

I.e.,

$$\frac{1}{2}\left(\Pr\left[C_j(x^1) = 1\right] + \Pr\left[C_j(x^2) = 1\right]\right) \geq \left(1 - 2^{-|C_j|-1} - \frac{1}{2}\left(1 - \frac{1}{|C_j|}\right)^{|C_j|}\right) z_j^* \geq \frac{3}{4} z_j^*$$

# Putting everything together



# Analysis roadmap (including the missing pieces)

$$\mathbb{E}\left[\max\left\{\sum_{j=1}^{m} C_j(x^1)\cdot w_j\,, \sum_{j=1}^{m} C_j(x^2)\cdot w_j\right\}\right] \geq \mathbb{E}\left[\frac{1}{2}\sum_{j=1}^{m} C_j(x^1)\cdot w_j + \frac{1}{2}\sum_{j=1}^{m} C_j(x^2)\cdot w_j\right]$$

$$= \frac{1}{2}\sum_{j=1}^{m}\Big(\mathbb{E}\big[C_j(x^1)\big] + \mathbb{E}[C_j(x^2)]\Big)\cdot w_j = \frac{1}{2}\sum_{j=1}^{m}\Big(\Pr\big[C_j(x^1) = 1\big] + \Pr[C_j(x^2) = 1]\Big)\cdot w_j$$

$$\geq \frac{3}{4}\cdot\sum_{j=1}^{m} w_j\cdot z_j^* \geq \frac{3}{4}\cdot \mathrm{OPT}$$

# Nonlinear randomized rounding

# Nonlinear randomized rounding

- Again, use the **optimal fractional solution $\boldsymbol{y}^*, \boldsymbol{z}^*$** of the linear programming relaxation
- Use a **rounding function $\boldsymbol{f}$**
- **Set variable $\boldsymbol{x_i}$ to true with probability $\boldsymbol{f(y_i^*)}$** and to false with probability $1 - f(y_i^*)$, independently on the other variables
- Our goal: a **3/4-approximation** algorithm

# The rounding function $f$

- $f$ is a function from $[0,1]$ to $[0,1]$
- Select $f$ such that $1 - 4^{-y} \leq f(y) \leq 4^{y-1}$ for every $y \in [0,1]$
- $f$ does exists (why?)

# Analysis of nonlinear rounding

# Analysis of nonlinear rounding

$$\Pr\left[C_j(x) = 0\right] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0]$$

# Analysis of nonlinear rounding

by the definition of
nonlinear rounding

$$\Pr\left[C_j(x) = 0\right] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - f(y_i^*)) \cdot \prod_{i \in N_j} f(y_i^*)$$

# Analysis of nonlinear rounding

by the definition of
nonlinear rounding

$$\Pr[C_j(x) = 0] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - f(y_i^*)) \cdot \prod_{i \in N_j} f(y_i^*)$$

$$\leq \prod_{i \in P_j} 4^{-y_i^*} \cdot \prod_{i \in N_j} 4^{y_i^* - 1}$$

by the properties of
the rounding function

# Analysis of nonlinear rounding

by the definition of nonlinear rounding

$$\Pr[C_j(x) = 0] = \prod_{i \in P_j} \Pr[x_i = 0] \cdot \prod_{i \in N_j} \Pr[\bar{x}_i = 0] = \prod_{i \in P_j} (1 - f(y_i^*)) \cdot \prod_{i \in N_j} f(y_i^*)$$

$$\leq \prod_{i \in P_j} 4^{-y_i^*} \cdot \prod_{i \in N_j} 4^{y_i^* - 1} = 4^{-\left(\sum_{i \in P_j} y_i^* + \sum_{i \in N_j} (1 - y_i^*)\right)} \leq 4^{-z_j^*}$$

by the properties of the rounding function

LP constraint for clause $C_j$

# Analysis of nonlinear rounding

nonlinear dependence on $z_j^*$

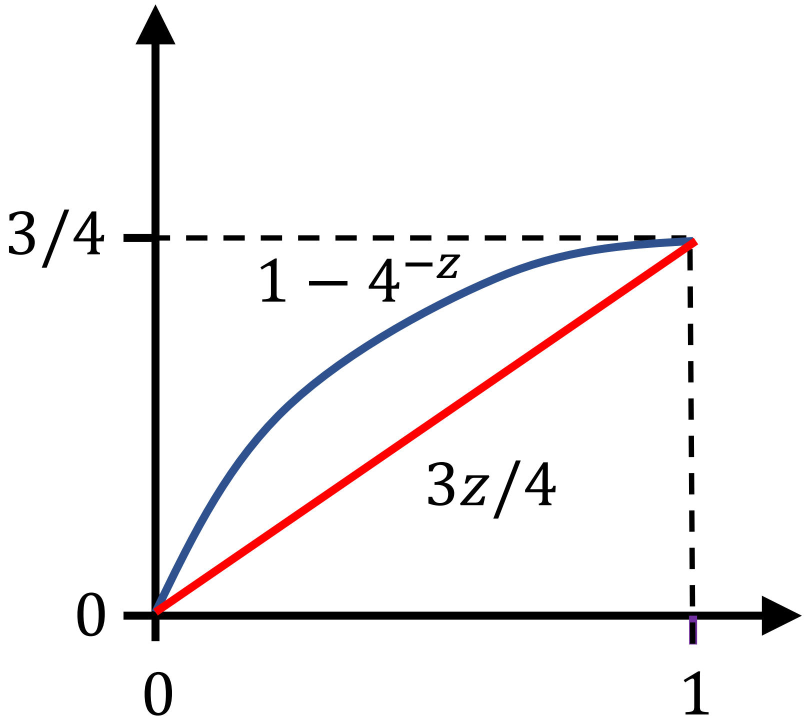
- Hence, $\Pr\left[C_j(x) = 1\right] \geq 1 - 4^{-z_j^*}$

# Analysis of nonlinear rounding

nonlinear dependence on $z_j^*$

- Hence, $\Pr\left[C_j(x) = 1\right] \geq 1 - 4^{-z_j^*}$
- Again, the function $1 - 4^{-z}$ is concave and satisfies $1 - 4^{-z} \geq 3z/4$
- I.e., $\Pr\left[C_j(x) = 1\right] \geq \frac{3}{4} z_j^*$

linear dependence on $z_j^*$



# Analysis of nonlinear rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

# Analysis of nonlinear rounding

$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] = \sum_{j=1}^{m} \mathbb{E}[C_j(x)] \cdot w_j = \sum_{j=1}^{m} \Pr[C_j(x) = 1] \cdot w_j$$

$$\geq \frac{3}{4} \cdot \sum_{j=1}^{m} w_j \cdot z_j^* \geq \frac{3}{4} \cdot \mathrm{OPT}$$

# Are improvements possible?

- MAXSAT instance: $C_1 = x_1 \vee x_2, C_2 = x_1 \vee \bar{x}_2, C_3 = \bar{x}_1 \vee x_2, C_4 = \bar{x}_1 \vee \bar{x}_2,$ unit weights
- $\mathrm{OPT} = 3$
- LP objective value = 4 (by setting $y_1 = y_2 = 1/2$ and $z_i = 1, \forall i$)
- For this instance, a (nonlinear) randomized rounding algorithm with approximation ratio $\rho > 3/4$ would imply
$$\mathbb{E}\left[\sum_{j=1}^{m} C_j(x) \cdot w_j\right] \geq \cdots \geq \rho \cdot \sum_{j=1}^{m} w_j \cdot z_j^* > \mathrm{OPT}$$
i.e., a contradiction
- The best approx. ratio we can hope for is the **integrality gap** of the LP

# Last slide

- Maximum Satisfiability
- Simple randomized algorithms
- Linear programming and randomized rounding