Lecture 18: Randomized rounding

Approximation guarantees and derandomization

Alex Gittens
Department of Computer Science, Rensselaer Polytechnic Institute

Fall 2026

From discrete choices to an integer program

Many problems ask us to select items, open locations, or assign truth values. Use $x_i = 1$ for a choice we make and $x_i = 0$ otherwise.

From discrete choices to an integer program

Many problems ask us to select items, open locations, or assign truth values. Use $x_i = 1$ for a choice we make and $x_i = 0$ otherwise.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ \text{subject to} \quad & Ax \geq b, \\ & x \in \{0,1\}^m. \end{aligned}$$

The objective measures cost; the inequalities describe the requirements.

From discrete choices to an integer program

Many problems ask us to select items, open locations, or assign truth values. Use $x_i = 1$ for a choice we make and $x_i = 0$ otherwise.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ \text{subject to} \quad & Ax \geq b, \\ & x \in \{0,1\}^m. \end{aligned}$$

The objective measures cost; the inequalities describe the requirements. Replace $x \in \{0,1\}^m$ by $0 \leq x_i \leq 1$: now we have a linear program, or

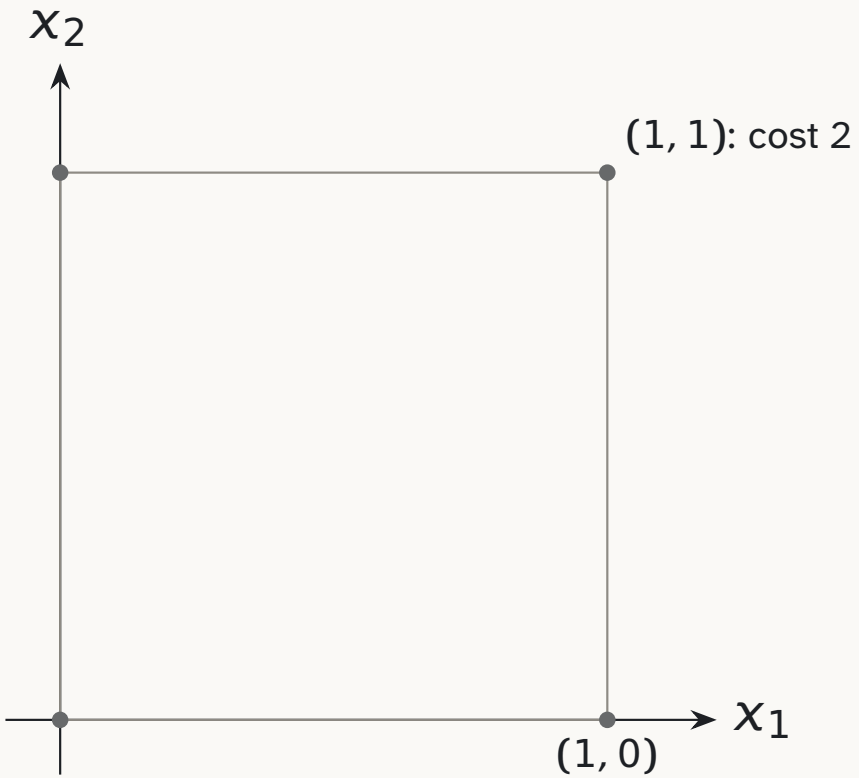
LP relaxation. Every binary feasible solution is still feasible, so

$\mathrm{OPT}_{\mathrm{LP}} \leq \mathrm{OPT}$.

Solve the relaxation, then round

$$\min x_1 + x_2 \quad \text{s.t. } 2x_1 + 2x_2 \geq 3.$$

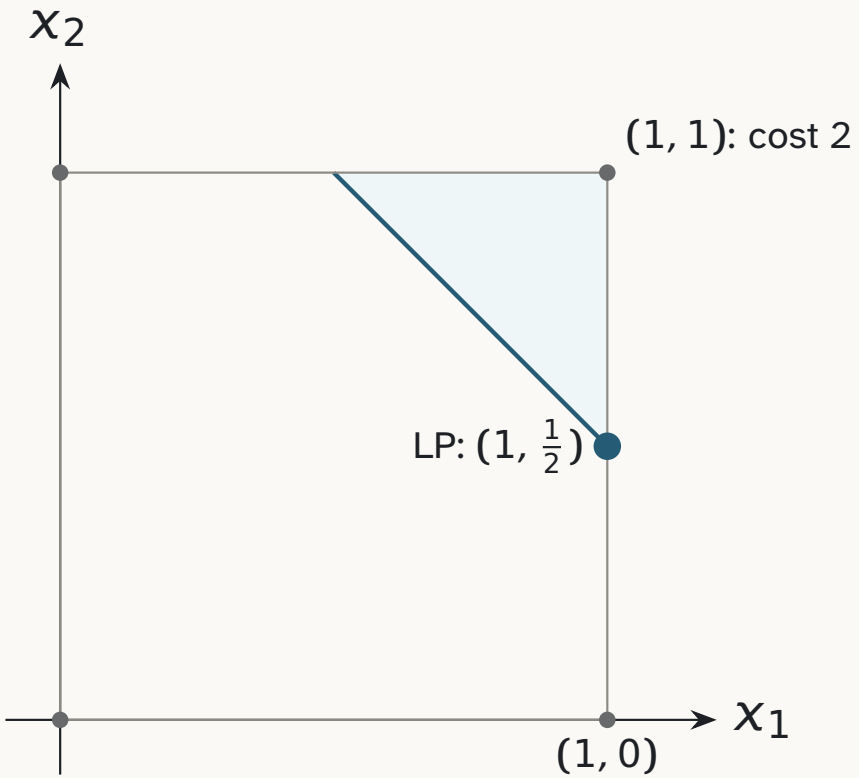
Binary choices: only $(1, 1)$ is feasible. Its cost is 2.



Solve the relaxation, then round

$$\min x_1 + x_2 \quad \text{s.t. } 2x_1 + 2x_2 \geq 3.$$

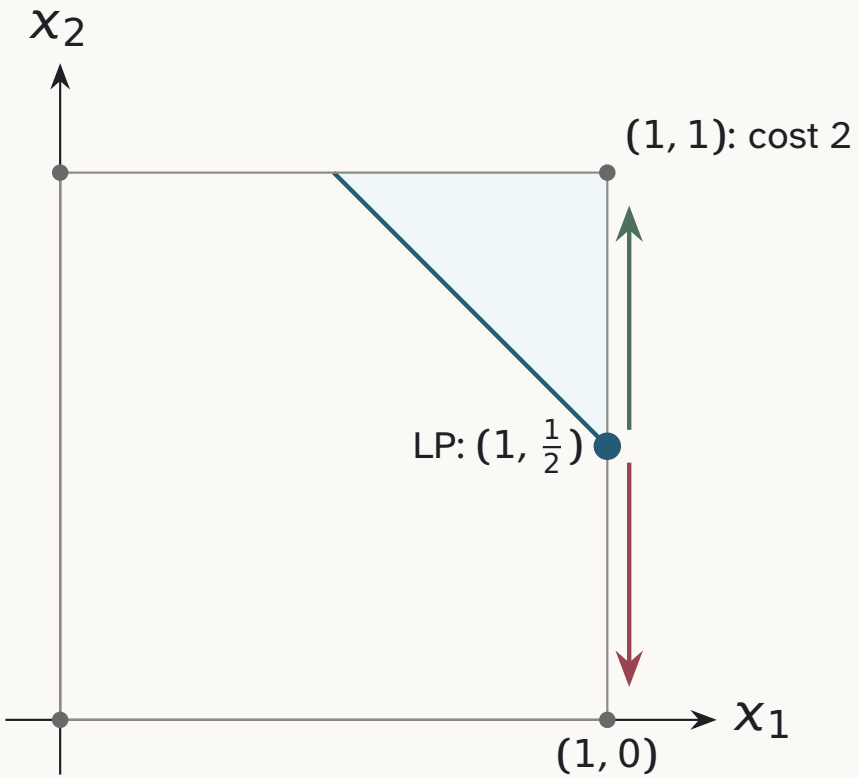
Binary choices: only $(1, 1)$ is feasible. Its cost is 2. **LP relaxation:** allow $x_i \in [0, 1]$. The point $(1, \frac{1}{2})$ is optimal, with cost $3/2$.



Solve the relaxation, then round

$$\min x_1 + x_2 \quad \text{s.t. } 2x_1 + 2x_2 \geq 3.$$

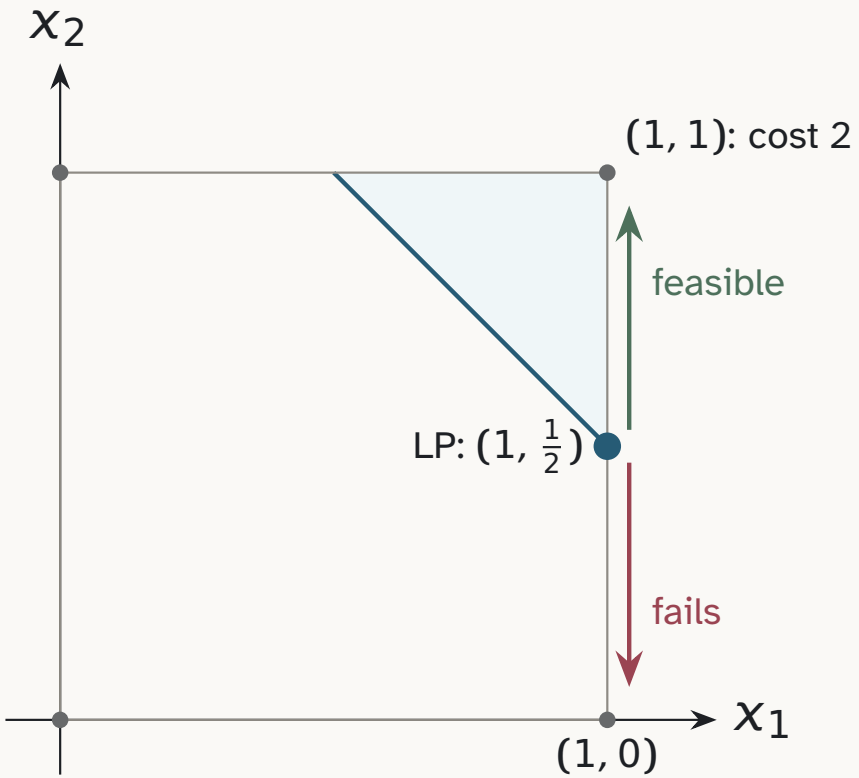
Binary choices: only $(1, 1)$ is feasible. Its cost is 2. **LP relaxation:** allow $x_i \in [0, 1]$. The point $(1, \frac{1}{2})$ is optimal, with cost $3/2$. **Round:** keep $x_1 = 1$; choose $X_2 = 1$ with probability $1/2$.



Solve the relaxation, then round

$$\min x_1 + x_2 \quad \text{s.t. } 2x_1 + 2x_2 \geq 3.$$

Binary choices: only $(1, 1)$ is feasible. Its cost is 2. **LP relaxation:** allow $x_i \in [0, 1]$. The point $(1, \frac{1}{2})$ is optimal, with cost $3/2$. **Round:** keep $x_1 = 1$; choose $X_2 = 1$ with probability $1/2$. The cost has the right expectation, but $(1, 0)$ violates the constraint. We must analyze what rounding preserves.



Two examples of LP rounding

1. **MAX-SAT:** use fractional truth values as probabilities. How many clauses will the resulting assignment satisfy?
2. **Set Cover:** use fractional set choices as probabilities. How can we cover every element without selecting too many sets?

Two examples of LP rounding

1. **MAX-SAT:** use fractional truth values as probabilities. How many clauses will the resulting assignment satisfy?
2. **Set Cover:** use fractional set choices as probabilities. How can we cover every element without selecting too many sets?

For a minimization problem, the LP gives a lower bound on the optimum. For a maximization problem, it gives an upper bound. We will analyze

both randomized algorithms and then remove their random choices using conditional expectation.

PART 1

MAX-SAT

1

MAX-SAT

We have Boolean variables and a collection of clauses. A clause is an OR of literals, such as

$$C \vee B \vee A, \qquad \neg C \vee D.$$

We want an assignment satisfying **as many clauses as possible**. Sometimes clauses conflict, so satisfying every one is impossible.

MAX-SAT

We have Boolean variables and a collection of clauses. A clause is an OR of literals, such as

$$C \vee B \vee A, \qquad \neg C \vee D.$$

We want an assignment satisfying **as many clauses as possible**. Sometimes clauses conflict, so satisfying every one is impossible.

We will use an explicit fourteen-clause example with six variables C, B, A, D, R, T. The letters are just variable names.

Our fourteen-clause instance

Six unit clauses

$$\neg C, \quad \neg B, \quad \neg A,$$
$$D, \quad R, \quad \neg T.$$

Four two-literal clauses

$$\neg A \vee R, \qquad \neg C \vee D,$$
$$\neg B \vee R, \qquad \neg T \vee D.$$

Four three-literal clauses

$$C \vee B \vee A,$$
$$C \vee B \vee \neg D,$$
$$C \vee A \vee \neg D,$$
$$B \vee A \vee \neg D.$$

Our fourteen-clause instance

Six unit clauses

$$\neg C, \quad \neg B, \quad \neg A,$$
$$D, \quad R, \quad \neg T.$$

Four two-literal clauses

$$\neg A \vee R, \qquad \neg C \vee D,$$
$$\neg B \vee R, \qquad \neg T \vee D.$$

Four three-literal clauses

$$C \vee B \vee A,$$
$$C \vee B \vee \neg D,$$
$$C \vee A \vee \neg D,$$
$$B \vee A \vee \neg D.$$

The unit clauses favor $C = B = A = 0$ and $D = 1$, while the four three-literal clauses push against those choices.

Start with uniform random assignment

Set each variable independently to 1 with probability $1/2$.
The instance has six unit clauses, four clauses of length two, and four of length three. A clause of length k fails only if all its literals are false:

$$\Pr(\text{clause satisfied}) = 1 - 2^{-k}.$$

Start with uniform random assignment

Set each variable independently to 1 with probability $1/2$.
The instance has six unit clauses, four clauses of length two, and four of length three. A clause of length k fails only if all its literals are false:

$$\Pr(\text{clause satisfied}) = 1 - 2^{-k}.$$

Thus the expected number of satisfied clauses is

$$6 \cdot \tfrac{1}{2} + 4 \cdot \tfrac{3}{4} + 4 \cdot \tfrac{7}{8} = 9.5.$$

Start with uniform random assignment

Set each variable independently to 1 with probability $1/2$.
The instance has six unit clauses, four clauses of length two, and four of length three. A clause of length k fails only if all its literals are false:

$$\Pr(\text{clause satisfied}) = 1 - 2^{-k}.$$

Thus the expected number of satisfied clauses is

$$6 \cdot \tfrac{1}{2} + 4 \cdot \tfrac{3}{4} + 4 \cdot \tfrac{7}{8} = 9.5.$$

These probabilities ignore which variables appear in the clauses. Perhaps some variables should be 1 more often than others. We will use an LP relaxation to choose the probabilities and then analyze the resulting score.

The MAX-SAT integer program and relaxation

Let $y_i \in \{0, 1\}$ be variable i's truth value. Let $z_j \in \{0, 1\}$ record whether clause j contributes to the score. P_j, N_j index positive and negative literals.

$$\begin{aligned} \text{maximize} \quad & \sum_j z_j \\ \text{subject to} \quad & z_j \leq \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i), \\ & y_i, z_j \in \{0, 1\}. \end{aligned}$$

The MAX-SAT integer program and relaxation

Let $y_i \in \{0, 1\}$ be variable i's truth value. Let $z_j \in \{0, 1\}$ record whether clause j contributes to the score. P_j, N_j index positive and negative literals.

$$\begin{aligned} \text{maximize} \quad & \sum_j z_j \\ \text{subject to} \quad & z_j \le \sum_{i \in P_j} y_i + \sum_{i \in N_j} (1 - y_i), \\ & y_i, z_j \in \{0, 1\}. \end{aligned}$$

For Boolean y, z, a clause contributes 1 only if it is satisfied, and maximization gives every satisfiable clause credit.

LP relaxation: replace the binary restrictions by $0 \le y_i, z_j \le 1$. This enlarges the feasible region, so

$$\mathrm{OPT} \le \mathrm{OPT}_{\mathrm{LP}}.$$

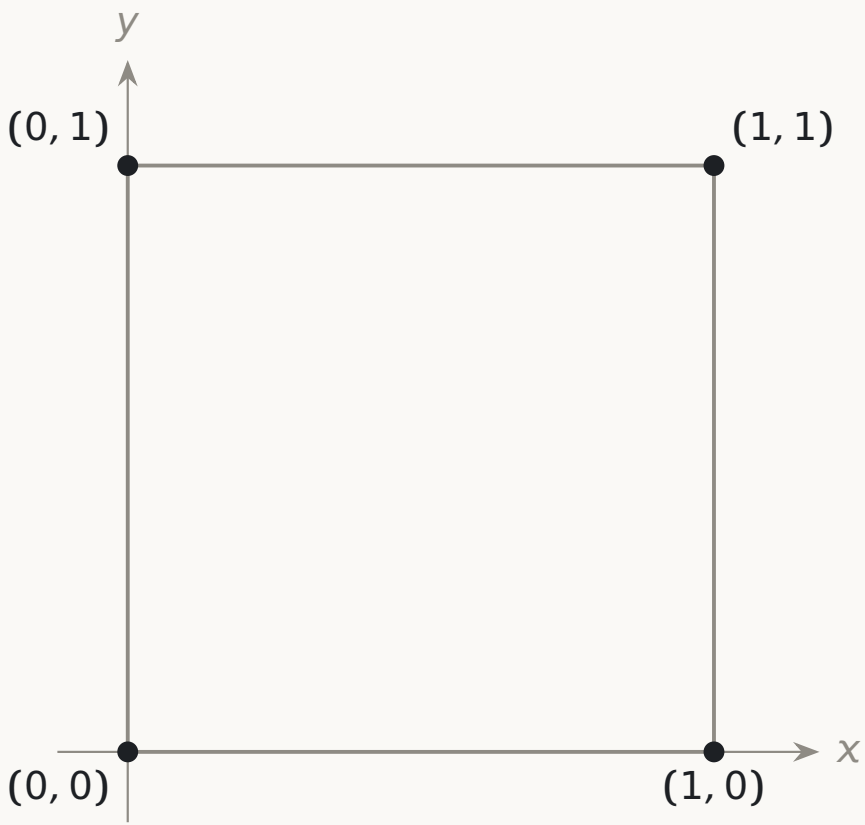
An LP value $z_j^\star$ is a fractional *credit*, not the probability that clause j will be satisfied.

What does rounding do geometrically?

Consider four conflicting clauses over just two variables:

$$(x \vee y), \quad (x \vee \neg y), \quad (\neg x \vee y), \quad (\neg x \vee \neg y).$$

Binary problem. Each corner gives a Boolean assignment, and every corner satisfies exactly 3 clauses. Thus $\mathrm{OPT} = 3$.



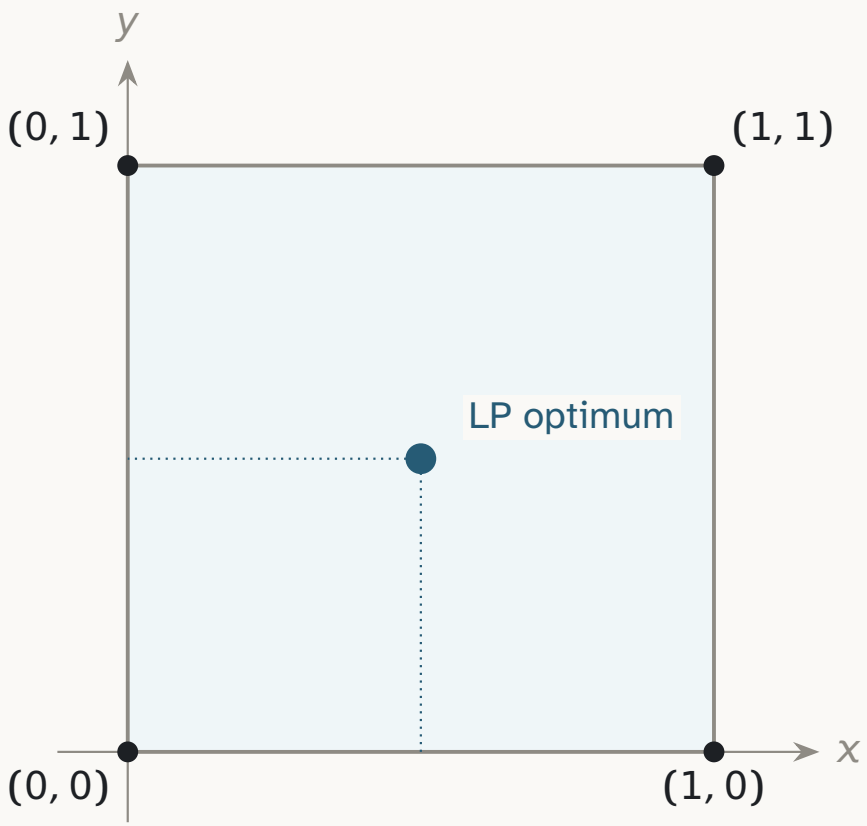
What does rounding do geometrically?

Consider four conflicting clauses over just two variables:

$$(x \vee y), \quad (x \vee \neg y), \quad (\neg x \vee y), \quad (\neg x \vee \neg y).$$

Relaxation. Allow $0 \le x, y \le 1$. At $(x^\star, y^\star) = (\frac{1}{2}, \frac{1}{2})$ all four clause credits can be $z_j^\star = 1$, so $\mathrm{OPT}_{\mathrm{LP}} = 4$.

Fractional LP optimum: $\left(\frac{1}{2}, \frac{1}{2}\right)$.



What does rounding do geometrically?

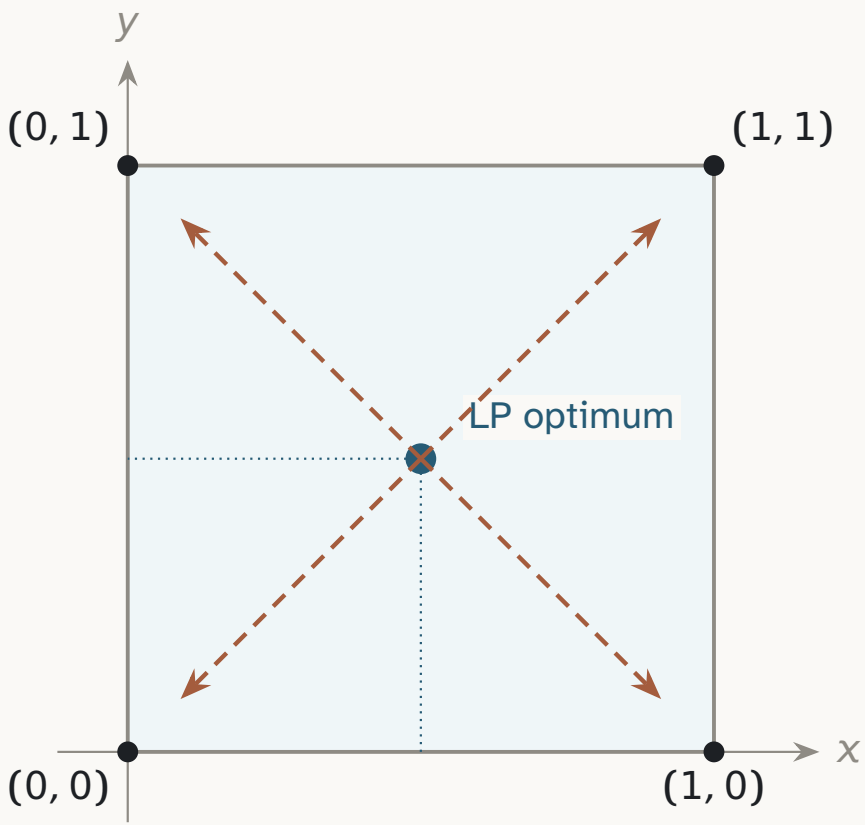
Consider four conflicting clauses over just two variables:

$$(x \vee y), \quad (x \vee \neg y), \quad (\neg x \vee y), \quad (\neg x \vee \neg y).$$

Independent rounding. Set
$X \sim \text{Bernoulli}(1/2)$ and
$Y \sim \text{Bernoulli}(1/2)$, independently.
The four corners each have probability $1/4$.

Fractional LP optimum: $\left(\frac{1}{2}, \frac{1}{2}\right)$.
Possible rounded outcomes:

$(0, 0), (1, 0), (0, 1), (1, 1)$.



What does rounding do geometrically?

Consider four conflicting clauses over just two variables:

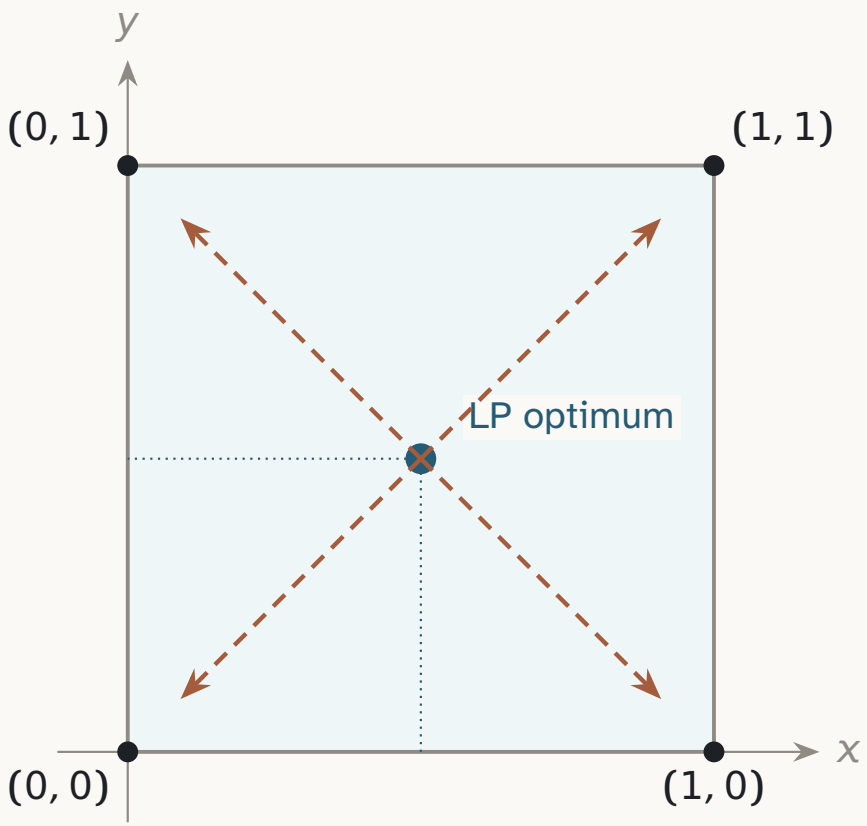
$$(x \vee y), \quad (x \vee \neg y), \quad (\neg x \vee y), \quad (\neg x \vee \neg y).$$

After rounding. Every outcome satisfies 3 clauses. Here $\mathbb{E}[V] = 3$, while the fractional LP scored 4. This is why an LP objective is an upper bound, not the rounded score.

Fractional LP optimum: $\left(\frac{1}{2}, \frac{1}{2}\right)$.

Possible rounded outcomes:

$(0, 0), (1, 0), (0, 1), (1, 1)$.



each corner: 3 clauses

LP rounding probabilities for our instance

The fourteen-clause instance has the following optimal LP solution:

variable	C	B	A	D	R	T
$y_i^\star$	1/3	1/3	1/3	2/3	1	0

Its LP objective is $38/3 = 12\frac{2}{3}$; the best Boolean assignment satisfies 12 clauses.

LP rounding probabilities for our instance

The fourteen-clause instance has the following optimal LP solution:

variable	C	B	A	D	R	T
$y_i^\star$	$1/3$	$1/3$	$1/3$	$2/3$	1	0

Its LP objective is $38/3 = 12\frac{2}{3}$; the best Boolean assignment satisfies 12 clauses. Round variables independently with $\Pr(\widehat{y}_i = 1) = y_i^\star$. Two clause probabilities are

$$\Pr(C \vee B \vee A) = 1 - (2/3)^3 = 19/27,$$

$$\Pr(\neg C \vee D) = 1 - (1/3)(1/3) = 8/9.$$

LP rounding probabilities for our instance

The fourteen-clause instance has the following optimal LP solution:

variable	C	B	A	D	R	T
$y_i^\star$	$1/3$	$1/3$	$1/3$	$2/3$	1	0

Its LP objective is $38/3 = 12\frac{2}{3}$; the best Boolean assignment satisfies 12 clauses. Round variables independently with $\Pr(\widehat{y}_i = 1) = y_i^\star$. Two clause probabilities are

$$\Pr(C \vee B \vee A) = 1 - (2/3)^3 = 19/27,$$

$$\Pr(\neg C \vee D) = 1 - (1/3)(1/3) = 8/9.$$

Both clauses have LP credit $z_j^\star = 1$, although neither has satisfaction probability 1. Summing over all clauses gives $\mathbb{E}[V] = 307/27 \approx 11.37$, compared with 9.5 for fair coins.

Appendix: why $\mathrm{OPT}_{\mathrm{LP}} = 38/3$ and $\mathrm{OPT} = 12$.

One independently rounded assignment

Draw independent $U_i \sim \text{Uniform}[0, 1]$ and set $y_i = \mathbf{1}\{U_i < y_i^\star\}$.

	C	B	A	D	R	T
$y_i^\star$	1/3	1/3	1/3	2/3	1	0
U_i	.2	.8	.6	.9	.4	.7

One independently rounded assignment

Draw independent $U_i \sim \text{Uniform}[0, 1]$ and set $y_i = \mathbf{1}\{U_i < y_i^\star\}$.

	C	B	A	D	R	T
$y_i^\star$	1/3	1/3	1/3	2/3	1	0
U_i	.2	.8	.6	.9	.4	.7
y_i	1	0	0	0	1	0

One independently rounded assignment

Draw independent $U_i \sim \text{Uniform}[0, 1]$ and set $y_i = \mathbf{1}\{U_i < y_i^\star\}$.

	C	B	A	D	R	T
$y_i^\star$	1/3	1/3	1/3	2/3	1	0
U_i	.2	.8	.6	.9	.4	.7
y_i	1	0	0	0	1	0

The false clauses are exactly $\neg C$, D, and $\neg C \vee D$. The other eleven hold, so this rounded assignment satisfies **11 of 14 clauses**.

LP rounding for MAX-SAT

Algorithm: LP rounding for MAX-SAT

Input: clauses $C_1, \ldots, C_q$ over $v_1, \ldots, v_N$.
Output: an assignment $\widehat{y} \in \{0,1\}^N$.

1. Solve the LP; let $(y^\star, z^\star)$ be an optimal solution.
2. Independently for each i, set $\widehat{y}_i = 1$ with probability $y_i^\star$.

Theorem

Let V be the number of clauses satisfied by $\widehat{y}$. Then

$$\mathbb{E}[V] \;\geq\; (1 - 1/e)\,\mathrm{OPT}_{\mathrm{LP}} \;\geq\; (1 - 1/e)\,\mathrm{OPT} \approx 0.632\;\mathrm{OPT}.$$

The bound is in expectation. An individual assignment may satisfy fewer clauses.

Probability that a clause is satisfied

We may assume each clause has distinct variables (drop repeats; a clause containing v_i and $\neg v_i$ is always satisfied).
C_j fails exactly when every positive variable is 0 and every negated one is 1:

$$\Pr(C_j \text{ fails}) = \prod_{i \in P_j} (1 - y_i^\star) \prod_{i \in N_j} y_i^\star.$$

Probability that a clause is satisfied

We may assume each clause has distinct variables (drop repeats; a clause containing v_i and $\neg v_i$ is always satisfied).
C_j fails exactly when every positive variable is 0 and every negated one is 1:

$$\Pr(C_j \text{ fails}) = \prod_{i \in P_j} (1 - y_i^\star) \prod_{i \in N_j} y_i^\star .$$

Apply $1 - a \le e^{-a}$ to every factor, writing $y_i^\star = 1 - (1 - y_i^\star)$ for the negated ones:

$$\Pr(C_j \text{ fails}) \le \exp\Big(- \underbrace{\Big[\textstyle\sum_{i \in P_j} y_i^\star + \sum_{i \in N_j} (1 - y_i^\star) \Big]}_{\ge z_j^\star \text{ by the LP constraint}} \Big) \le e^{-z_j^\star} .$$

So $\Pr(C_j \text{ satisfied}) \ge 1 - e^{-z_j^\star}$.

Expected number of satisfied clauses

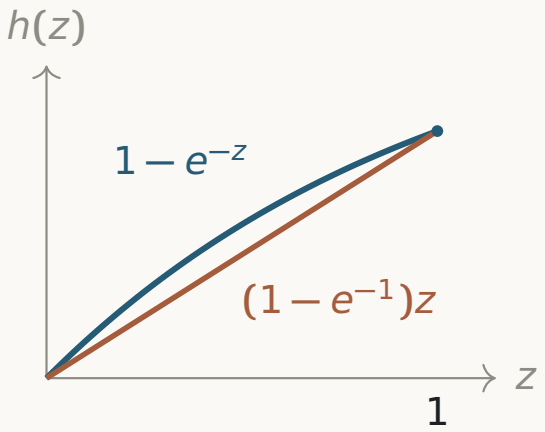
We already have

$$\mathbb{E}[V] = \sum_j \Pr(C_j \text{ satisfied}) \geq \sum_j (1 - e^{-z_j^\star}).$$

The LP objective is $\mathrm{OPT}_{\mathrm{LP}} = \sum_j z_j^\star$. We need a lower bound on $1 - e^{-z}$ proportional to z.

$h(z) = 1 - e^{-z}$ is concave. On $[0, 1]$ it lies above its chord:

$$1 - e^{-z} \geq (1 - 1/e)\, z.$$



Expected number of satisfied clauses

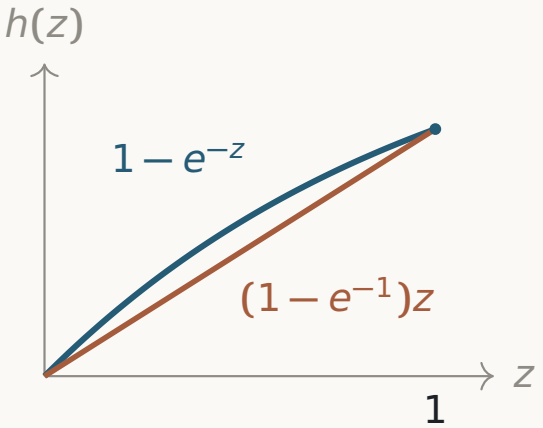
We already have

$$\mathbb{E}[V] = \sum_j \Pr(C_j \text{ satisfied}) \geq \sum_j (1 - e^{-z_j^\star}).$$

The LP objective is $\mathrm{OPT}_{\mathrm{LP}} = \sum_j z_j^\star$. We need a lower bound on $1 - e^{-z}$ proportional to z.

$h(z) = 1 - e^{-z}$ is concave. On $[0, 1]$ it lies above its chord:

$$1 - e^{-z} \geq (1 - 1/e)\, z.$$



Hence $\Pr(C_j \text{ sat}) \geq (1 - \frac{1}{e}) z_j^\star$, and by linearity of expectation

$$\mathbb{E}[V] = \sum_j \Pr(C_j \text{ sat}) \geq \left(1 - \frac{1}{e}\right) \sum_j z_j^\star = \left(1 - \frac{1}{e}\right) \mathrm{OPT}_{\mathrm{LP}}\,. \qquad \square$$

Derandomizing MAX-SAT rounding

The randomized algorithm guarantees an *expected* score. We can fix variables one at a time, keeping that expected score from decreasing.

Suppose some variables are fixed. Let F be the expected score when the rest keep their original independent probabilities.

Derandomizing MAX-SAT rounding

The randomized algorithm guarantees an *expected* score. We can fix variables one at a time, keeping that expected score from decreasing.

Suppose some variables are fixed. Let F be the expected score when the rest keep their original independent probabilities. For the next variable,

with probability p of being 1, let F_0, F_1 be the expectations after fixing it to 0 or 1. The tower property gives

$$F = (1-p)F_0 + pF_1 \leq \max\{F_0, F_1\}.$$

Choose the value with the larger expectation. If p is 0 or 1, use its forced value.

Derandomizing MAX-SAT rounding

The randomized algorithm guarantees an *expected* score. We can fix variables one at a time, keeping that expected score from decreasing.

Suppose some variables are fixed. Let F be the expected score when the rest keep their original independent probabilities. For the next variable,

with probability p of being 1, let F_0, F_1 be the expectations after fixing it to 0 or 1. The tower property gives

$$F = (1-p)F_0 + pF_1 \le \max\{F_0, F_1\}.$$

Choose the value with the larger expectation. If p is 0 or 1, use its forced value. Repeat until every variable is fixed. At the end, F is the actual

number satisfied. It never decreased, so the output satisfies at least $(1 - 1/e)\,\mathrm{OPT}_{\mathrm{LP}}$ clauses.

Computing the conditional expectation

To choose the next variable y_i, compare the expected *total* scores:

$$F_b = \sum_{j=1}^{q} \Pr(C_j \text{ satisfied} \mid a,\ y_i = b), \qquad b \in \{0, 1\},$$

where a is the partial assignment already fixed.

Computing the conditional expectation

To choose the next variable y_i, compare the expected *total* scores:

$$F_b = \sum_{j=1}^{q} \Pr(C_j \text{ satisfied} \mid a,\ y_i = b), \qquad b \in \{0, 1\},$$

where a is the partial assignment already fixed. Each clause contributes one term to this sum:

- A fixed literal is true: contribution 1.
- All literals are fixed false: contribution 0.
- Otherwise, the contribution is

$$1 - \prod_{\text{unfixed literals } \ell} \Pr(\ell \text{ false}).$$

Computing the conditional expectation

To choose the next variable y_i, compare the expected *total* scores:

$$F_b = \sum_{j=1}^{q} \Pr(C_j \text{ satisfied} \mid a,\ y_i = b), \qquad b \in \{0, 1\},$$

where a is the partial assignment already fixed. Each clause contributes one term to this sum:

- A fixed literal is true: contribution 1.
- All literals are fixed false: contribution 0.
- Otherwise, the contribution is

$$1 - \prod_{\text{unfixed literals } \ell} \Pr(\ell \text{ false}).$$

Compute the sum once with $y_i = 0$ and once with $y_i = 1$. Keep the choice with the larger sum, so the expected score cannot decrease.

A MAX-SAT decision in our instance

Initially, $\Pr(C = 1) = \Pr(B = 1) = \Pr(A = 1) = 1/3$, $\Pr(D = 1) = 2/3$, $R = 1$, and $T = 0$. To fix C, only five clauses change their expected contributions:

Clause	$C = 0$	$C = 1$
$\neg C$	1	0
$\neg C \vee D$	1	2/3
$C \vee B \vee A$	5/9	1
$C \vee B \vee \neg D$	5/9	1
$C \vee A \vee \neg D$	5/9	1

A MAX-SAT decision in our instance

Initially, $\Pr(C = 1) = \Pr(B = 1) = \Pr(A = 1) = 1/3$, $\Pr(D = 1) = 2/3$, $R = 1$, and $T = 0$. To fix C, only five clauses change their expected contributions:

Clause	$C = 0$	$C = 1$
$\neg C$	1	0
$\neg C \vee D$	1	2/3
$C \vee B \vee A$	5/9	1
$C \vee B \vee \neg D$	5/9	1
$C \vee A \vee \neg D$	5/9	1

These five contributions sum to $11/3$ under either choice. All other contributions are unchanged, so $F_0 = F_1$. Either choice preserves the expectation; choose $C = 0$ to break the tie.

Deterministic MAX-SAT rounding

Fix one variable at a time

Input: clauses and an optimal LP solution $(y^\star, z^\star)$.
Output: an assignment satisfying at least $(1 - 1/e)\,\mathrm{OPT}_{\mathrm{LP}}$ clauses.

1. Fix variables with probability 0 or 1 to those values.
2. For each remaining variable, compute F_0, F_1 and choose the larger. If equal, choose 0.

Deterministic MAX-SAT rounding

Fix one variable at a time

Input: clauses and an optimal LP solution $(y^\star, z^\star)$.
Output: an assignment satisfying at least $(1 - 1/e)\,\mathrm{OPT}_{\mathrm{LP}}$ clauses.

1. Fix variables with probability 0 or 1 to those values.
2. For each remaining variable, compute F_0, F_1 and choose the larger. If equal, choose 0.

The conditional expectation never decreases. After all variables are fixed, it equals the actual number of satisfied clauses.

How much work does derandomization take?

Let N be the number of variables, q the number of clauses, and L the total number of literal occurrences.

1. For one clause, scan its literals: check fixed values and multiply the false-literal probabilities for the remaining variables.

This is the work *after* solving the LP. LP solving is also polynomial in the encoded input size; rational arithmetic has polynomial bit cost.

How much work does derandomization take?

Let N be the number of variables, q the number of clauses, and L the total number of literal occurrences.

1. For one clause, scan its literals: check fixed values and multiply the false-literal probabilities for the remaining variables.
2. For one candidate choice, scan all clauses and add their contributions: $O(L + q)$ arithmetic operations.

This is the work *after* solving the LP. LP solving is also polynomial in the encoded input size; rational arithmetic has polynomial bit cost.

How much work does derandomization take?

Let N be the number of variables, q the number of clauses, and L the total number of literal occurrences.

1. For one clause, scan its literals: check fixed values and multiply the false-literal probabilities for the remaining variables.
2. For one candidate choice, scan all clauses and add their contributions: $O(L + q)$ arithmetic operations.
3. Compare two choices for each of at most N variables:

$$O\big(N(L + q)\big) \text{ arithmetic operations.}$$

This is the work *after* solving the LP. LP solving is also polynomial in the encoded input size; rational arithmetic has polynomial bit cost.

How much work does derandomization take?

Let N be the number of variables, q the number of clauses, and L the total number of literal occurrences.

1. For one clause, scan its literals: check fixed values and multiply the false-literal probabilities for the remaining variables.
2. For one candidate choice, scan all clauses and add their contributions: $O(L + q)$ arithmetic operations.
3. Compare two choices for each of at most N variables:

$$O\big(N(L + q)\big) \text{ arithmetic operations.}$$

4. For nonempty clauses, $q \leq L$, giving $O(NL)$. If each clause has at most k literals, this is $O(Nqk)$.

This is the work *after* solving the LP. LP solving is also polynomial in the encoded input size; rational arithmetic has polynomial bit cost.

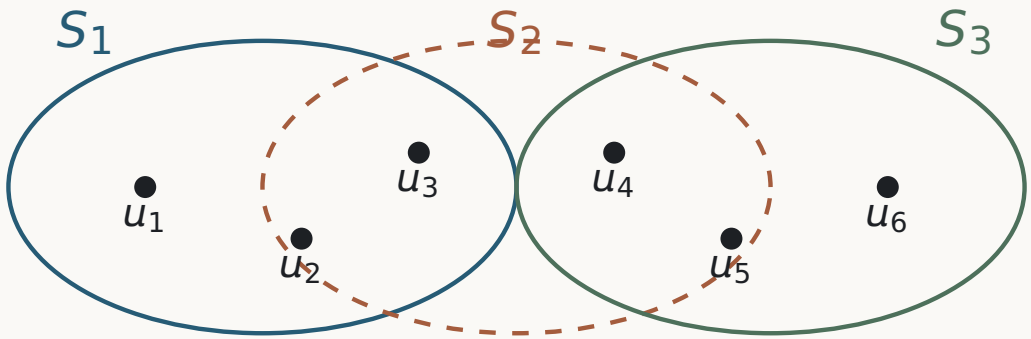
PART 2

Set Cover: coverage and cost

2

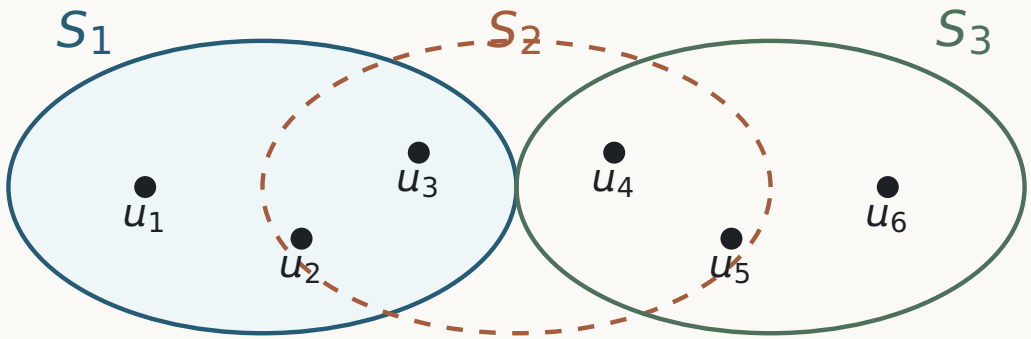
What does a set cover look like?

Each dot is an element to cover; each outlined region is an available set.



What does a set cover look like?

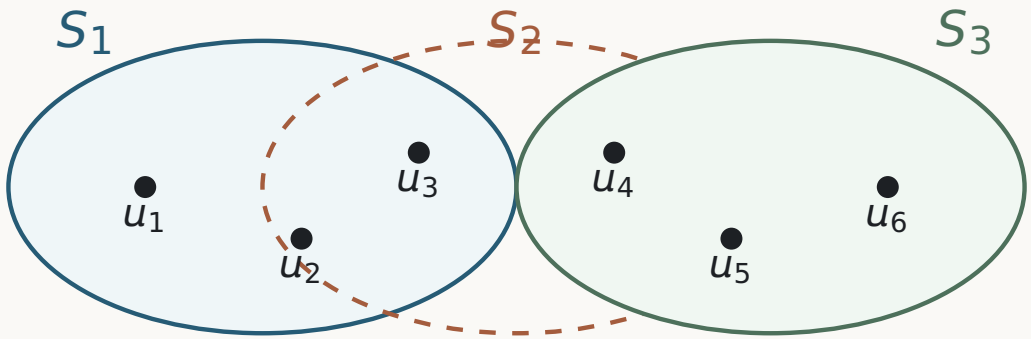
Each dot is an element to cover; each outlined region is an available set.



Select S_1: it covers u_1, u_2, u_3, but three elements remain uncovered.

What does a set cover look like?

Each dot is an element to cover; each outlined region is an available set.



Select S_1: it covers u_1, u_2, u_3, but three elements remain uncovered. Select S_3 as well: every element is covered. The selected sets may overlap; every element needs to belong to *at least one* selected set.

Set Cover: choose as few sets as possible

Input: n elements $U = \{u_1, \ldots, u_n\}$ and m sets $S_1, \ldots, S_m \subseteq U$, whose union is U.

Goal: select as few sets as possible so that every element belongs to at least one selected set.

Set Cover: choose as few sets as possible

Input: n elements $U = \{u_1, \ldots, u_n\}$ and m sets $S_1, \ldots, S_m \subseteq U$, whose union is U.

Goal: select as few sets as possible so that every element belongs to at least one selected set.

With $x_i = 1$ if we select S_i, the binary integer program is

$$\begin{aligned} \text{minimize} \quad & \sum_i x_i \\ \text{subject to} \quad & \sum_{i:u \in S_i} x_i \geq 1 \quad \text{for every } u \in U, \\ & x_i \in \{0, 1\}. \end{aligned}$$

Set Cover: choose as few sets as possible

Input: n elements $U = \{u_1, \ldots, u_n\}$ and m sets $S_1, \ldots, S_m \subseteq U$, whose union is U.

Goal: select as few sets as possible so that every element belongs to at least one selected set.

With $x_i = 1$ if we select S_i, the binary integer program is

$$\begin{aligned} \text{minimize} \quad & \sum_i x_i \\ \text{subject to} \quad & \sum_{i : u \in S_i} x_i \geq 1 \quad \text{for every } u \in U, \\ & x_i \in \{0, 1\}. \end{aligned}$$

The LP relaxation replaces $x_i \in \{0, 1\}$ by $0 \leq x_i \leq 1$. Because this is a minimization problem, $\mathrm{OPT}_{\mathrm{LP}} \leq \mathrm{OPT}$.

One-round LP rounding for Set Cover

Algorithm: one round

Input: a feasible Set Cover instance and optimal LP solution $x^\star$.
Output: a random subset of the sets.

1. Independently select set S_i with probability $x_i^\star$, for every i.
2. Return the selected sets.

One-round LP rounding for Set Cover

Algorithm: one round

Input: a feasible Set Cover instance and optimal LP solution $x^\star$.
Output: a random subset of the sets.

1. Independently select set S_i with probability $x_i^\star$, for every i.
2. Return the selected sets.

The expected number of selected sets is exactly $\sum_i x_i^\star = \mathrm{OPT}_{\mathrm{LP}}$.

But some elements may remain uncovered. Unlike MAX-SAT, **we need to control feasibility as well as cost**. A selection of very few sets does us no good if it fails to cover U.

Probability that an element is uncovered

For an element u, independence and its LP constraint give

$$\Pr(u \text{ uncovered}) = \prod_{i:u\in S_i} (1 - x_i^\star) \le \exp\left(-\sum_{i:u\in S_i} x_i^\star\right) \le e^{-1}.$$

Probability that an element is uncovered

For an element u, independence and its LP constraint give

$$\Pr(u \text{ uncovered}) = \prod_{i:u\in S_i} (1 - x_i^\star) \le \exp\left(-\sum_{i:u\in S_i} x_i^\star\right) \le e^{-1}.$$

A union bound over n elements gives n/e, which is useless for large n.

Selecting more sets cannot uncover an element. What happens if we repeat the rounding independently and keep the union of all selected sets?

Probability that an element is uncovered

For an element u, independence and its LP constraint give

$$\Pr(u \text{ uncovered}) = \prod_{i:u \in S_i} (1 - x_i^\star) \leq \exp\left(- \sum_{i:u \in S_i} x_i^\star \right) \leq e^{-1}.$$

A union bound over n elements gives n/e, which is useless for large n.

Selecting more sets cannot uncover an element. What happens if we repeat the rounding independently and keep the union of all selected sets? After t independent rounds,

$$\Pr(u \text{ uncovered}) \leq e^{-t}.$$

Each round is independent, so the probability of missing the same element in all t rounds decays exponentially.

Repeated rounding for Set Cover

Algorithm: repeated rounding for Set Cover

Input: feasible instance: $n \geq 1$ elements, m sets.
Output: $\bar{x} \in \{0,1\}^m$.

1. Solve the LP; let $x^\star$ be an optimal solution. Set $t = \lceil \ln(4n) \rceil$.
2. For $r = 1, \ldots, t$: independently draw $\widehat{x}_i^{(r)} \sim \text{Bernoulli}(x_i^\star)$ for every i.
3. Output $\bar{x}_i = \max_r \widehat{x}_i^{(r)}$ (keep every set chosen in some round).

Theorem

With probability at least $1/2$, $\bar{x}$ is a cover and

$$\mathbf{1}^\top \bar{x} < 4t \, \mathrm{OPT}_{\mathrm{LP}} \leq 4 \lceil \ln(4n) \rceil \, \mathrm{OPT}.$$

Proof: every element is covered

Set $t = \lceil \ln(4n) \rceil$. A particular element is missed only if all t rounds miss it:

$$\Pr(u \text{ uncovered}) \le e^{-t} \le \frac{1}{4n}.$$

Proof: every element is covered

Set $t = \lceil \ln(4n) \rceil$. A particular element is missed only if all t rounds miss it:

$$\Pr(u \text{ uncovered}) \le e^{-t} \le \frac{1}{4n}.$$

A union bound gives

$$\Pr(\text{some element is uncovered}) \le \sum_{u \in U} \Pr(u \text{ uncovered}) \le \frac{1}{4}.$$

No independence between different elements is needed.

Proof: the number of selected sets

Let C be the number selected after taking the union. Set i is selected with probability

$$1 - (1 - x_i^\star)^t \leq t x_i^\star.$$

Thus $\mathbb{E}[C] \leq t \, \mathrm{OPT}_{\mathrm{LP}}$. By Markov's inequality,

$$\Pr(C \geq 4t \, \mathrm{OPT}_{\mathrm{LP}}) \leq \frac{1}{4}.$$

Proof: the number of selected sets

Let C be the number selected after taking the union. Set i is selected with probability

$$1 - (1 - x_i^\star)^t \leq t x_i^\star.$$

Thus $\mathbb{E}[C] \leq t \operatorname{OPT}_{\mathsf{LP}}$. By Markov's inequality,

$$\Pr(C \geq 4t \operatorname{OPT}_{\mathsf{LP}}) \leq \frac{1}{4}.$$

Combining the two failure probabilities gives

$$\Pr(\text{a cover with } C < 4t \operatorname{OPT}_{\mathsf{LP}}) \geq 1 - \tfrac{1}{4} - \tfrac{1}{4} = \tfrac{1}{2}. \qquad \square$$

The two bad events need not be independent.

Repeat until we obtain a cover

Check and retry

Input: a feasible, nonempty Set Cover instance and its LP solution.
Output: a cover costing $< 4\lceil \ln(4n) \rceil \mathrm{OPT}_{\mathrm{LP}}$.

1. Generate a repeated-rounding selection.
2. Check coverage and the cost bound. Return it if both hold; otherwise generate a new independent selection.

Repeat until we obtain a cover

Check and retry

Input: a feasible, nonempty Set Cover instance and its LP solution.
Output: a cover costing $< 4\lceil \ln(4n) \rceil \mathrm{OPT}_{\mathrm{LP}}$.

1. Generate a repeated-rounding selection.
2. Check coverage and the cost bound. Return it if both hold; otherwise generate a new independent selection.

Each attempt succeeds with probability at least $1/2$. Therefore

$$\Pr(\text{more than } k \text{ attempts}) \le 2^{-k}, \qquad \mathbb{E}[\text{attempts}] \le 2.$$

The output always has the required properties; the running time is random.

PART 3

Facility placement in Albany

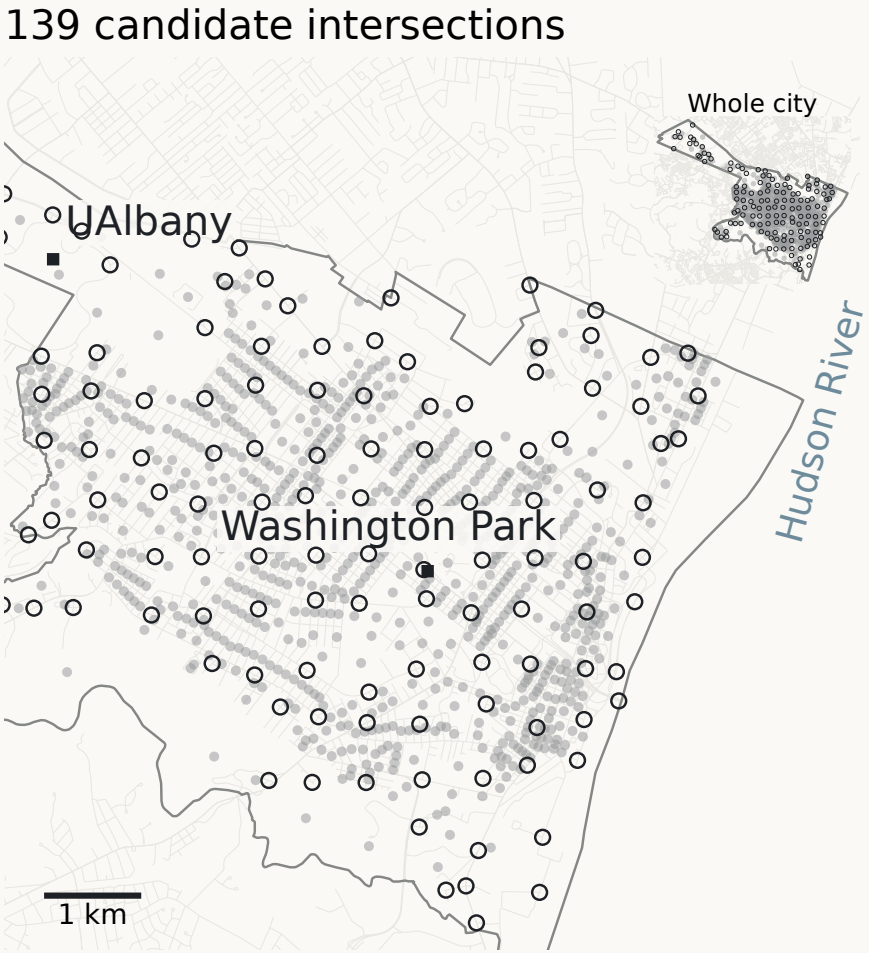
3

Where could Albany place ambulance bases?



Imagine bases where ambulances wait between calls. We want every populated block to be within a short drive of a base. Our choices are 139 hypothetical locations at real Albany intersections.

Where could Albany place ambulance bases?



Imagine bases where ambulances wait between calls. We want every populated block to be within a short drive of a base. Our choices are 139 hypothetical locations at real Albany intersections.

Coverage within four minutes

A candidate base covers a block if the estimated drive to its representative point takes at most four minutes.

- We use Census roads and assume a travel speed of 20 mph.
- We connect each block point to its nearest road junction, then add that distance to the route along the roads.
- We count each base equally and want complete coverage of all 884 block points.

Coverage within four minutes

A candidate base covers a block if the estimated drive to its representative point takes at most four minutes.

- We use Census roads and assume a travel speed of 20 mph.
- We connect each block point to its nearest road junction, then add that distance to the route along the roads.
- We count each base equally and want complete coverage of all 884 block points.

Four minutes gives us an example with a fractional LP solution. These travel estimates leave out traffic and the time needed to start a response. The model

asks where bases could be located; it does not model how many ambulances each base needs. Assumptions and data sources.

The LP for Albany

Let x_i represent opening a base at candidate intersection i. Define

$$A_{ui} = \begin{cases} 1, & \text{candidate } i \text{ can reach block point } u \text{ within four minutes,} \\ 0, & \text{otherwise.} \end{cases}$$

The LP for Albany

Let x_i represent opening a base at candidate intersection i. Define

$$A_{ui} = \begin{cases} 1, & \text{candidate } i \text{ can reach block point } u \text{ within four minutes,} \\ 0, & \text{otherwise.} \end{cases}$$

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{139} x_i \\ \text{subject to} \quad & \sum_{i=1}^{139} A_{ui} x_i \geq 1 \quad (u = 1, \ldots, 884), \\ & 0 \leq x_i \leq 1 \quad (i = 1, \ldots, 139). \end{aligned}$$

The LP for Albany

Let x_i represent opening a base at candidate intersection i. Define

$$A_{ui} = \begin{cases} 1, & \text{candidate } i \text{ can reach block point } u \text{ within four minutes,} \\ 0, & \text{otherwise.} \end{cases}$$

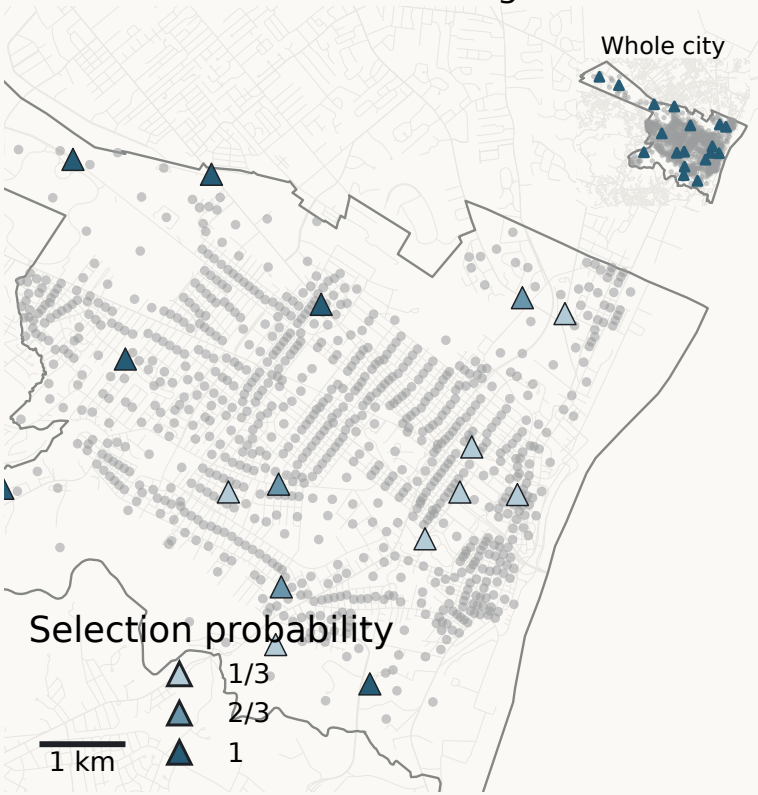
$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{139} x_i \\ \text{subject to} \quad & \sum_{i=1}^{139} A_{ui} x_i \geq 1 \quad (u = 1, \ldots, 884), \\ & 0 \leq x_i \leq 1 \quad (i = 1, \ldots, 139). \end{aligned}$$

Each row requires one block point to be covered. For block point 97 in our data, only candidates 54 and 62 can reach it in time. Its constraint is $x_{54} + x_{62} \geq 1$.

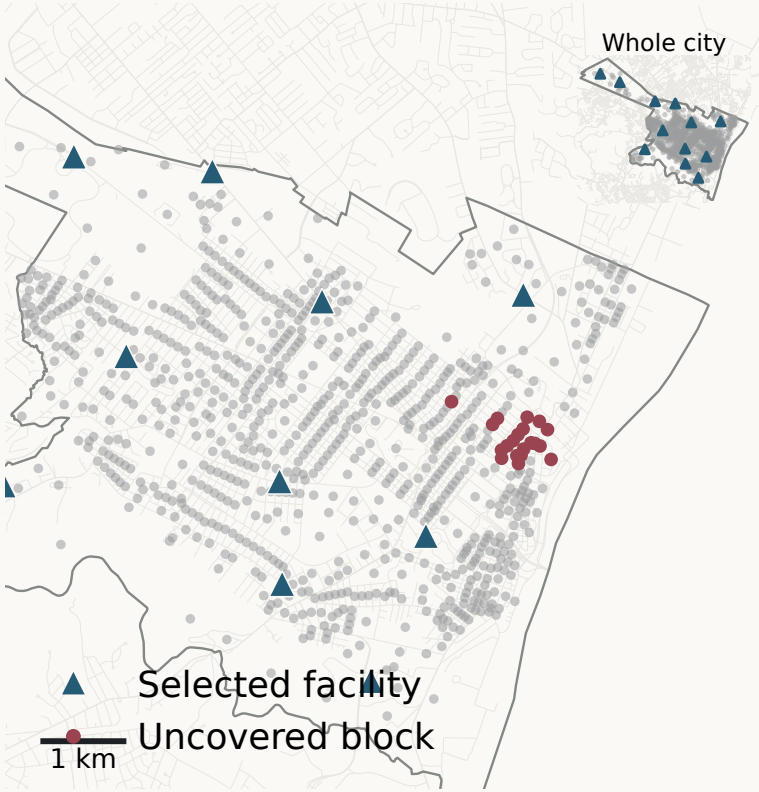
Albany: fractional solution and one round

The LP optimum is $\mathrm{OPT}_{\mathrm{LP}} = 37/3$. Eight site values are 1; ten are strictly between 0 and 1.

LP value: $12\frac{1}{3}$

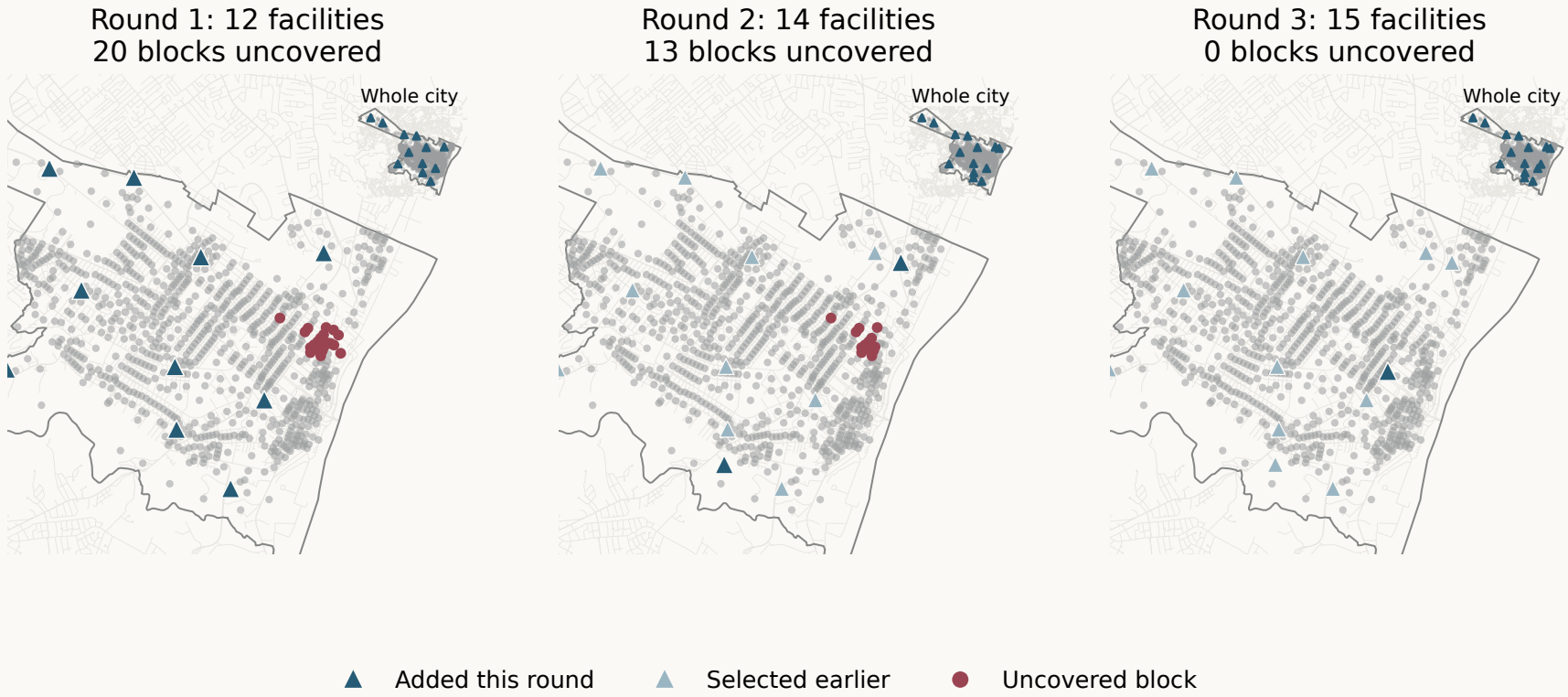


12 facilities; 20 blocks uncovered



One round selects $37/3 \approx 12.33$ facilities on average. This draw selected 12 and left 20 blocks uncovered.

Repeat the selection and keep the union



These maps continue the same run. The third round covers the remaining blocks; later rounds may add facilities that are no longer needed.

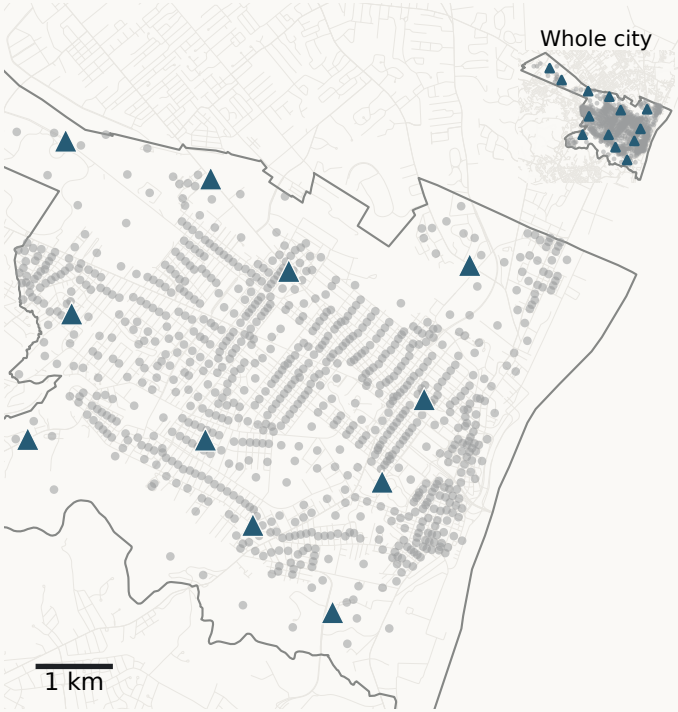
Optimal and greedy covers in Albany

Every complete cover is feasible for the LP, so it uses at least $37/3$ facilities. Since the number of facilities is an integer, it must be at least 13.

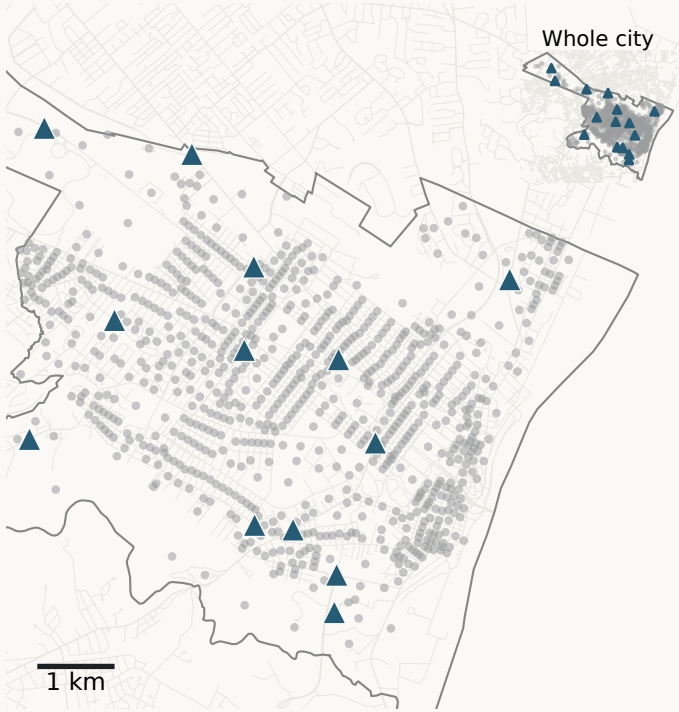
Optimal and greedy covers in Albany

Every complete cover is feasible for the LP, so it uses at least $37/3$ facilities. Since the number of facilities is an integer, it must be at least 13.

Optimal: 13 bases



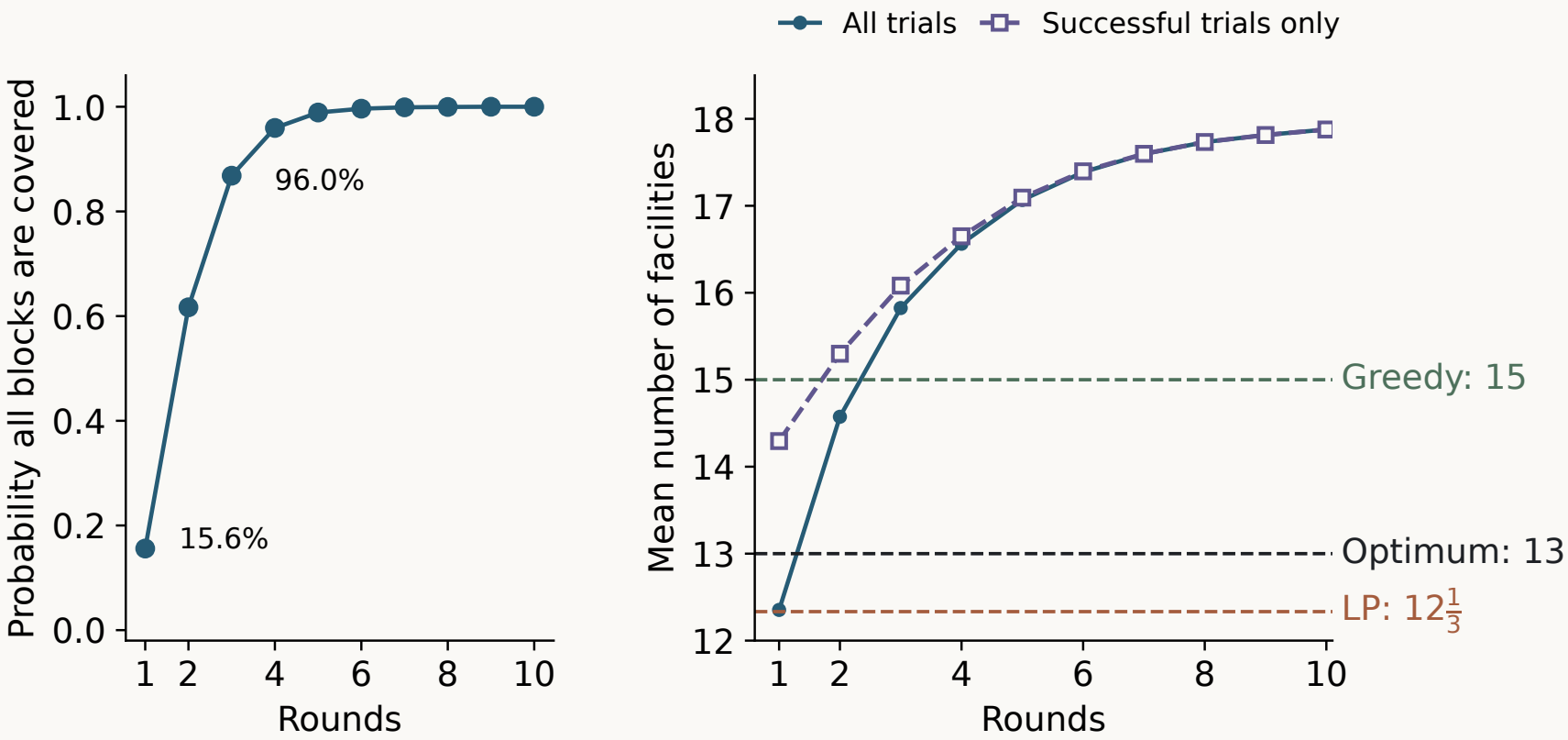
Greedy: 15 bases



A complete cover with 13 facilities proves that the optimum is 13. Greedy repeatedly covers as many remaining blocks as possible; it uses 15 facilities.

Proof of the greedy guarantee.

Coverage probability and number of facilities



After four rounds, about 96% of trials cover every block. The average selection contains 16.56 facilities. The second cost curve averages successful trials only.

10,000 trials per round count; error bars show 95% confidence intervals for the coverage probability.

PART 4

Derandomizing Set Cover

4

A penalty for coverage and cost

We need a cover that uses few sets. Let B count uncovered elements and C count selected sets.

A penalty for coverage and cost

We need a cover that uses few sets. Let B count uncovered elements and C count selected sets. Combine the two requirements in one quantity:

$$\Phi = B + \frac{C}{K},$$

where $K > 0$ is the size limit we will choose next.

A penalty for coverage and cost

We need a cover that uses few sets. Let B count uncovered elements and C count selected sets. Combine the two requirements in one quantity:

$$\Phi = B + \frac{C}{K},$$

where $K > 0$ is the size limit we will choose next.

- If any element is uncovered, $B \geq 1$, so $\Phi \geq 1$.
- If $C \geq K$, the cost term is at least 1, so $\Phi \geq 1$.

A penalty for coverage and cost

We need a cover that uses few sets. Let B count uncovered elements and C count selected sets. Combine the two requirements in one quantity:

$$\Phi = B + \frac{C}{K},$$

where $K > 0$ is the size limit we will choose next.

- If any element is uncovered, $B \geq 1$, so $\Phi \geq 1$.
- If $C \geq K$, the cost term is at least 1, so $\Phi \geq 1$.

Thus $\Phi < 1$ guarantees **complete coverage and fewer than K sets**. We will fix choices while keeping its conditional expectation below 1.

The same parameters as randomized rounding

Recall the bounds proved for repeated rounding. With $t = \lceil \ln(4n) \rceil$,

$$\mathbb{E}[B] \le n e^{-t} \le \tfrac{1}{4}.$$

With $\mathbb{E}[C] \le t\,\mathrm{OPT}_{\mathrm{LP}}$, choose $K = 4t\,\mathrm{OPT}_{\mathrm{LP}}$:

$$\Phi = B + \frac{C}{4t\,\mathrm{OPT}_{\mathrm{LP}}}, \qquad \mathbb{E}[\Phi] \le \tfrac{1}{4} + \tfrac{1}{4} = \tfrac{1}{2} < 1.$$

The same parameters as randomized rounding

Recall the bounds proved for repeated rounding. With $t = \lceil \ln(4n) \rceil$,

$$\mathbb{E}[B] \le n e^{-t} \le \tfrac{1}{4}.$$

With $\mathbb{E}[C] \le t\,\mathrm{OPT}_{\mathrm{LP}}$, choose $K = 4t\,\mathrm{OPT}_{\mathrm{LP}}$:

$$\Phi = B + \frac{C}{4t\,\mathrm{OPT}_{\mathrm{LP}}}, \qquad \mathbb{E}[\Phi] \le \tfrac{1}{4} + \tfrac{1}{4} = \tfrac{1}{2} < 1.$$

Some selection therefore has no uncovered elements and fewer than $4t\,\mathrm{OPT}_{\mathrm{LP}}$ sets.

To find one deterministically, fix the random choices without increasing the conditional expectation of Φ.

Conditional expectation of the penalty

The union of t rounds selects each facility independently with probability

$$q_i = 1 - (1 - x_i^\star)^t.$$

Selections for distinct facilities depend on disjoint coin flips, so these union indicators are independent.

Conditional expectation of the penalty

The union of t rounds selects each facility independently with probability

$$q_i = 1 - (1 - x_i^\star)^t.$$

Selections for distinct facilities depend on disjoint coin flips, so these union indicators are independent. After some decisions are fixed, let p_i be 0 for a rejected site, 1 for a selected site, and q_i for an undecided site. Then

$$\mathbb{E}[\,\Phi \mid \text{decisions}] = \underbrace{\sum_{u \in U} \prod_{i : u \in S_i} (1 - p_i)}_{\text{expected uncovered elements}} + \underbrace{\frac{\sum_i p_i}{4t\,\mathrm{OPT}_{\mathrm{LP}}}}_{\text{expected normalized cost}}.$$

Conditional expectation of the penalty

The union of t rounds selects each facility independently with probability

$$q_i = 1 - (1 - x_i^\star)^t.$$

Selections for distinct facilities depend on disjoint coin flips, so these union indicators are independent. After some decisions are fixed, let p_i be 0 for a rejected site, 1 for a selected site, and q_i for an undecided site. Then

$$\mathbb{E}[\,\Phi \mid \text{decisions}] = \underbrace{\sum_{u \in U} \prod_{i : u \in S_i} (1 - p_i)}_{\text{expected uncovered elements}} + \underbrace{\frac{\sum_i p_i}{4t\,\mathrm{OPT}_{\mathrm{LP}}}}_{\text{expected normalized cost}}.$$

Choose the next decision giving the smaller value. By the tower property, one choice is no larger than the current expectation.

Deterministic Set Cover rounding

Conditional expectation

Input: a feasible, nonempty instance with n elements.
Output: a cover costing $< 4\lceil \ln(4n) \rceil \mathrm{OPT}_{\mathrm{LP}}$.

1. Solve the LP; set $t = \lceil \ln(4n) \rceil$ and $q_i = 1 - (1 - x_i^\star)^t$.
2. Fix probability-zero/one choices to their forced values.
3. Fix each remaining choice to the value giving the smaller conditional expectation of Φ. If the values are equal, reject the set; either choice preserves the bound.

Deterministic Set Cover rounding

Conditional expectation

Input: a feasible, nonempty instance with n elements.
Output: a cover costing $< 4\lceil \ln(4n) \rceil \operatorname{OPT}_{\mathrm{LP}}$.

1. Solve the LP; set $t = \lceil \ln(4n) \rceil$ and $q_i = 1 - (1 - x_i^\star)^t$.
2. Fix probability-zero/one choices to their forced values.
3. Fix each remaining choice to the value giving the smaller conditional expectation of Φ. If the values are equal, reject the set; either choice preserves the bound.

The initial expectation is at most $1/2$. It never increases and finally equals Φ itself. Hence $B = 0$ and $C < 4t \operatorname{OPT}_{\mathrm{LP}} \le 4t \operatorname{OPT}$.

The cost of deterministic Set Cover rounding

Let $D = \sum_i |S_i|$ be the number of element–set memberships.

1. Build a list of covering sets for each element: $O(D + n + m)$ work.

Solve the LP first. Computing $q_i = 1 - (1 - x_i^\star)^t$ takes $O(mt)$ operations by direct multiplication; $t = O(\log n)$. Rational bit costs remain polynomial.

The cost of deterministic Set Cover rounding

Let $D = \sum_i |S_i|$ be the number of element–set memberships.

1. Build a list of covering sets for each element: $O(D + n + m)$ work.
2. To evaluate one candidate decision, scan those lists and compute each element's probability of being uncovered: $O(D + n)$ arithmetic operations.

Solve the LP first. Computing $q_i = 1 - (1 - x_i^\star)^t$ takes $O(mt)$ operations by direct multiplication; $t = O(\log n)$. Rational bit costs remain polynomial.

The cost of deterministic Set Cover rounding

Let $D = \sum_i |S_i|$ be the number of element–set memberships.

1. Build a list of covering sets for each element: $O(D + n + m)$ work.
2. To evaluate one candidate decision, scan those lists and compute each element's probability of being uncovered: $O(D + n)$ arithmetic operations.
3. Sum the m selection probabilities for the cost term: $O(m)$ more operations.

Solve the LP first. Computing $q_i = 1 - (1 - x_i^\star)^t$ takes $O(mt)$ operations by direct multiplication; $t = O(\log n)$. Rational bit costs remain polynomial.

The cost of deterministic Set Cover rounding

Let $D = \sum_i |S_i|$ be the number of element–set memberships.

1. Build a list of covering sets for each element: $O(D + n + m)$ work.
2. To evaluate one candidate decision, scan those lists and compute each element's probability of being uncovered: $O(D + n)$ arithmetic operations.
3. Sum the m selection probabilities for the cost term: $O(m)$ more operations.
4. Compare two choices at each of at most m sites. Total:

$$O\big(m(D + n + m)\big) \text{ arithmetic operations.}$$

Solve the LP first. Computing $q_i = 1 - (1 - x_i^\star)^t$ takes $O(mt)$ operations by direct multiplication; $t = O(\log n)$. Rational bit costs remain polynomial.

A deterministic cover

Conditional expectation: 14 facilities, every block covered



Conditional expectation selects 14 facilities and covers all 884 blocks. Compare: the optimum uses 13 facilities; greedy uses 15.

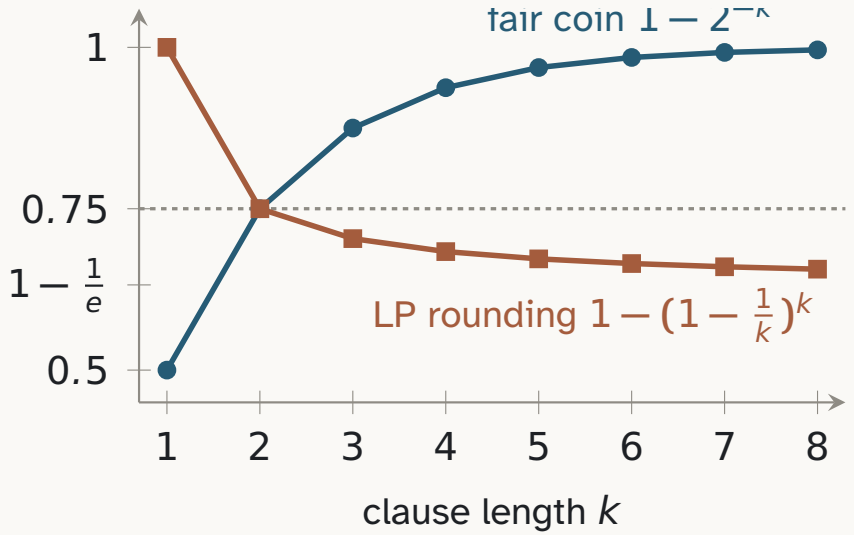
Questions?

Appendix: guide to the extra material

Section	What it adds
A. Better MAX-SAT guarantee	Combine fair coins and LP rounding for a 3/4-approximation.
B. High probability	Guarantee a prescribed failure probability in one fixed-length run.
C. Scaling probabilities	Choose higher inclusion probabilities instead of repeated rounds.
D. Greedy baseline	Prove the greedy bound used in Albany.
E. Numerical MAX-SAT example	Prove the LP and integer optimum values.
F. Albany model	Explain the threshold choice and data provenance.

Combining two MAX-SAT assignments

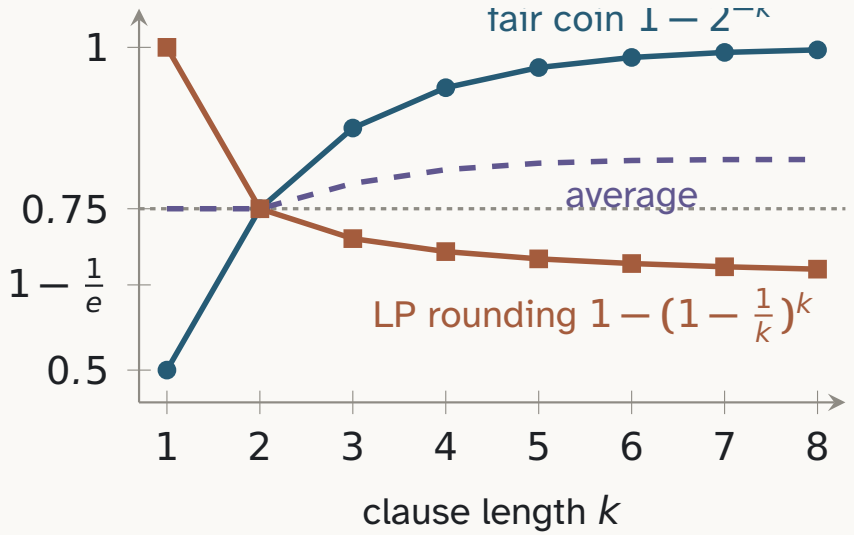
Fair coins work well for long clauses; LP rounding works well for short clauses. Run both algorithms and keep the assignment satisfying more clauses.
Guarantee per unit of $z_j^\star$ for a clause of length k:



- Uniform assignment gives $1 - 2^{-k}$.
- LP rounding gives $1 - (1 - 1/k)^k$ per unit of LP value.

Combining two MAX-SAT assignments

Fair coins work well for long clauses; LP rounding works well for short clauses. Run both algorithms and keep the assignment satisfying more clauses.
Guarantee per unit of $z_j^\star$ for a clause of length k:

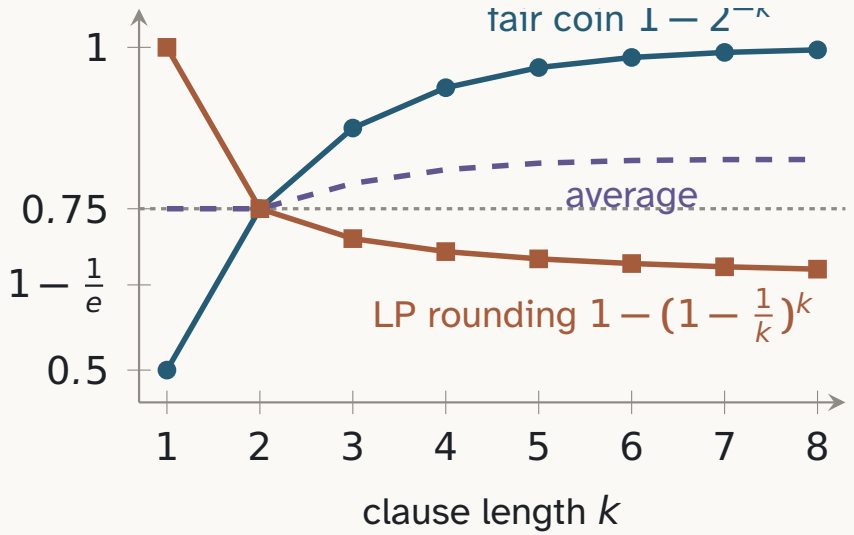


- Uniform assignment gives $1-2^{-k}$.
- LP rounding gives $1-(1-1/k)^k$ per unit of LP value.

At every k the average is $\geq 3/4$, with equality at $k=1,2$.

Combining two MAX-SAT assignments

Fair coins work well for long clauses; LP rounding works well for short clauses. Run both algorithms and keep the assignment satisfying more clauses.
Guarantee per unit of $z_j^\star$ for a clause of length k:



- Uniform assignment gives $1 - 2^{-k}$.
- LP rounding gives $1 - (1 - 1/k)^k$ per unit of LP value.

At every k the average is $\geq 3/4$, with equality at $k = 1, 2$.

Theorem (Goemans–Williamson 1994)

Input: clauses. **Steps:** run fair-coin assignment and LP rounding; score both. **Output:** the assignment satisfying more clauses. Its expected score is at least $\frac{3}{4}\,\mathrm{OPT}_{\mathrm{LP}} \geq \frac{3}{4}\,\mathrm{OPT}$ clauses.

A sharper bound for a clause

Lemma

If C_j has $k \geq 1$ distinct variables, LP rounding satisfies it with probability at least $\beta_k z_j^\star$, where $\beta_k = 1 - (1 - 1/k)^k$.

Proof. Write the k literal values as $a_1, \ldots, a_k$: $a_i = y_i^\star$ for $i \in P_j$ and $a_i = 1 - y_i^\star$ for $i \in N_j$. The LP constraint says $\sum_i a_i \geq z_j^\star$, and C_j fails with probability $\prod_i (1 - a_i)$. By AM–GM,

$$\prod_{i=1}^{k} (1 - a_i) \leq \Big(1 - \frac{1}{k} \sum_i a_i\Big)^k \leq \Big(1 - \frac{z_j^\star}{k}\Big)^k.$$

So $\Pr(C_j \text{ sat}) \geq g(z_j^\star)$ with $g(z) = 1 - (1 - z/k)^k$. Since g is concave with $g(0) = 0$, on $[0, 1]$ it lies above its chord: $g(z) \geq g(1)\, z = \beta_k z$. □

$\beta_1 = 1$, $\beta_2 = 3/4$, $\beta_3 \approx 0.704$, decreasing to $1 - 1/e$. Since $\beta_k \geq 1 - 1/e$ for every k, this implies the main-body theorem.

Proof of the $3/4$-approximation

Let V_1 be the number of clauses the fair coin satisfies and V_2 the number LP rounding satisfies. For a clause of length k:

$$\Pr(\text{fair coin satisfies } C_j) = 1 - 2^{-k} \geq (1 - 2^{-k})\, z_j^\star \qquad (\text{since } z_j^\star \leq 1),$$

and by the lemma, LP rounding satisfies it with probability $\geq \beta_k z_j^\star$.

Therefore

$$\mathbb{E}[\max(V_1, V_2)] \geq \tfrac{1}{2}\mathbb{E}[V_1] + \tfrac{1}{2}\mathbb{E}[V_2] \geq \sum_j \tfrac{1}{2}\big[(1 - 2^{-k_j}) + \beta_{k_j}\big] z_j^\star.$$

Proof of the $3/4$-approximation

Let V_1 be the number of clauses the fair coin satisfies and V_2 the number LP rounding satisfies. For a clause of length k:

$$\Pr(\text{fair coin satisfies } C_j) = 1 - 2^{-k} \geq (1 - 2^{-k})\, z_j^\star \qquad (\text{since } z_j^\star \leq 1),$$

and by the lemma, LP rounding satisfies it with probability $\geq \beta_k z_j^\star$.

Therefore

$$\mathbb{E}[\max(V_1, V_2)] \geq \tfrac{1}{2}\mathbb{E}[V_1] + \tfrac{1}{2}\mathbb{E}[V_2] \geq \sum_j \tfrac{1}{2}\big[(1 - 2^{-k_j}) + \beta_{k_j}\big] z_j^\star.$$

The bracket is at least $3/2$ for every k:

k	1	2	≥ 3
$(1-2^{-k}) + \beta_k$	$\frac{1}{2} + 1$	$\frac{3}{4} + \frac{3}{4}$	$\geq \frac{7}{8} + (1 - \frac{1}{e}) > 1.50$

Hence $\mathbb{E}[\max(V_1, V_2)] \geq \frac{3}{4} \sum_j z_j^\star = \frac{3}{4}\,\mathrm{OPT}_{\mathrm{LP}} \geq \frac{3}{4}\,\mathrm{OPT}$. □

Derandomizing each algorithm (main body) and keeping the better assignment gives a deterministic $3/4$-approximation.

A high-probability Set Cover bound

How many rounds suffice if we want one attempt to succeed with probability at least $1 - \delta$?
Let $0 < \delta < 1$, $t = \lceil \ln(2n/\delta) \rceil$, and round t times. Coverage fails with probability at most $ne^{-t} \le \delta/2$.

The union indicators are independent Bernoulli variables. Put $C = \sum_i X_i$ and $M = t\,\mathrm{OPT}_{\mathrm{LP}} \ge \mathbb{E}[C]$. For $s > 0$,

$$\mathbb{E}[e^{sC}] = \prod_i (1 + p_i(e^s - 1)) \le \exp((e^s - 1)\mathbb{E}[C]) \le \exp((e^s - 1)M).$$

A high-probability Set Cover bound

How many rounds suffice if we want one attempt to succeed with probability at least $1 - \delta$?
Let $0 < \delta < 1$, $t = \lceil \ln(2n/\delta) \rceil$, and round t times. Coverage fails with probability at most $ne^{-t} \le \delta/2$.

The union indicators are independent Bernoulli variables. Put $C = \sum_i X_i$ and $M = t\,\mathrm{OPT}_{\mathrm{LP}} \ge \mathbb{E}[C]$. For $s > 0$,

$$\mathbb{E}[e^{sC}] = \prod_i (1 + p_i(e^s - 1)) \le \exp((e^s - 1)\mathbb{E}[C]) \le \exp((e^s - 1)M).$$

Markov with $s = 1$ gives

$$\Pr(C \ge eM) \le e^{-eM}\mathbb{E}[e^C] \le e^{-M}.$$

A nonempty feasible instance has $\mathrm{OPT}_{\mathrm{LP}} \ge 1$, so $e^{-M} \le e^{-t} \le \delta/(2n) \le \delta/2$.

A union bound proves: with probability at least $1 - \delta$, the output is a cover costing $< e\lceil \ln(2n/\delta) \rceil\,\mathrm{OPT}_{\mathrm{LP}} \le e\lceil \ln(2n/\delta) \rceil\,\mathrm{OPT}$.

Scaling the selection probabilities

Input: a feasible, nonempty Set Cover instance and its LP solution. **Output:** a sampled facility subset.
Instead of t rounds, sample once with raised probabilities:

$$p_i = \min\{1,\ \lambda x_i^\star\}, \qquad \lambda = \ln(4n), \qquad X_i \sim \text{Bernoulli}(p_i) \text{ independently.}$$

Cost: $\mathbb{E}[\sum_i X_i] \le \lambda\ \text{OPT}_{\text{LP}}$.
Coverage: if some set containing u has $p_i = 1$, u is covered surely; otherwise

$$\Pr(u \text{ missed}) = \prod_{i:\, u \in S_i} (1 - \lambda x_i^\star) \le e^{-\lambda \sum_{i:u\in S_i} x_i^\star} \le e^{-\lambda} = \frac{1}{4n}.$$

The same Markov and union-bound argument gives a cover of size $< 4\lambda\ \text{OPT}_{\text{LP}}$ with probability $\ge 1/2$.

At the same integer scale $\lambda = t$, repetition selects site i with probability $1 - (1 - x_i^\star)^t \le \min\{1, tx_i^\star\}$, so it is no more expensive in expectation than scaling. Scaling also permits non-integer λ.

Greedy Set Cover

In Albany, greedy found a cover with 15 facilities. Here is the algorithm and its guarantee for any feasible instance.

Greedy Set Cover

Input: a feasible instance $(U, S_1, \ldots, S_m)$.
Output: a cover G with $|G| \leq H_n \, \mathrm{OPT}$, where $H_n = \sum_{j=1}^{n} 1/j$.

1. Start with no selected facilities and every point uncovered.
2. Select the candidate covering the largest number of still-uncovered points (if tied, choose the lowest ID).
3. Mark those points covered; repeat until none remain.

Feasibility ensures a positive gain whenever a point remains uncovered. Hence greedy terminates with a cover after at most $\min(m, n)$ selections.

Proof of the greedy approximation bound

When a chosen set covers k new points, charge each new point $1/k$. The charges sum to exactly the number of chosen sets.

Proof of the greedy approximation bound

When a chosen set covers k new points, charge each new point $1/k$. The charges sum to exactly the number of chosen sets. Fix an optimal set S

of size s, and order its points by the time greedy covers them (break simultaneous ties arbitrarily). Just before its jth point is covered, at least $s - j + 1$ of its points remain uncovered.

Greedy could select S, so its selected set covers at least $s - j + 1$ new points. That point's charge is at most $1/(s - j + 1)$.

Proof of the greedy approximation bound

When a chosen set covers k new points, charge each new point $1/k$. The charges sum to exactly the number of chosen sets. Fix an optimal set S of size s, and order its points by the time greedy covers them (break simultaneous ties arbitrarily). Just before its jth point is covered, at least $s - j + 1$ of its points remain uncovered.

Greedy could select S, so its selected set covers at least $s - j + 1$ new points. That point's charge is at most $1/(s - j + 1)$. Thus charges on S sum to at most $H_s \le H_n$. Every point belongs to an optimal set; summing these bounds over an optimal cover (possibly counting points repeatedly) gives

$$|G| = \sum_{u \in U} \mathrm{charge}(u) \le H_n \,\mathrm{OPT}. \qquad \square$$

The fourteen-clause LP optimum is $38/3$

We will prove an upper bound on the LP value, then give a solution attaining it. Write $a = C$, $b = B$, $c = A$, $d = 1 - D$, and $s = a + b + c + d$.

The four unit clauses $\neg C$, $\neg B$, $\neg A$, D contribute at most $4 - s$. The four length-three clauses contain every triple from a, b, c, d.

The fourteen-clause LP optimum is $38/3$

We will prove an upper bound on the LP value, then give a solution attaining it. Write $a = C$, $b = B$, $c = A$, $d = 1 - D$, and $s = a + b + c + d$.

The four unit clauses $\neg C$, $\neg B$, $\neg A$, D contribute at most $4 - s$. The four length-three clauses contain every triple from a, b, c, d. Each variable appears in three of these triples. Their LP values therefore sum to at most $3s$, and also to at most 4. The eight clauses contribute at most

$$4 - s + \min\{4, 3s\} \le 20/3.$$

The remaining six clauses contribute at most 6, giving $\mathrm{OPT}_{\mathrm{LP}} \le 38/3$.

The fourteen-clause LP optimum is $38/3$

We will prove an upper bound on the LP value, then give a solution attaining it. Write $a = C$, $b = B$, $c = A$, $d = 1 - D$, and $s = a + b + c + d$.

The four unit clauses $\neg C, \neg B, \neg A, D$ contribute at most $4 - s$. The four length-three clauses contain every triple from a, b, c, d. Each variable appears in three of these triples. Their LP values therefore sum to at most $3s$, and also to at most 4. The eight clauses contribute at most

$$4 - s + \min\{4, 3s\} \le 20/3.$$

The remaining six clauses contribute at most 6, giving $\mathrm{OPT}_{\mathrm{LP}} \le 38/3$. The solution $(C, B, A, D, R, T) = (1/3, 1/3, 1/3, 2/3, 1, 0)$ attains this bound: its four triple-clause values and six remaining clause values are all 1.

Boolean optimum: twelve clauses

Using the same four transformed bits a, b, c, d, let k count the bits equal to 1. The four unit clauses contribute $4 - k$.

k	0	1	2	3	4
Satisfied triple clauses	0	3	4	4	4
Total of these eight clauses	4	6	6	5	4

Boolean optimum: twelve clauses

Using the same four transformed bits a, b, c, d, let k count the bits equal to 1. The four unit clauses contribute $4 - k$.

k	0	1	2	3	4
Satisfied triple clauses	0	3	4	4	4
Total of these eight clauses	4	6	6	5	4

Thus these eight clauses contribute at most 6; the remaining six contribute at most 6. Hence $\mathrm{OPT} \leq 12$.

The assignment $(C, B, A, D, R, T) = (1, 0, 0, 1, 1, 0)$ satisfies 12 clauses, proving $\mathrm{OPT} = 12$.

Other coverage thresholds

At four minutes, the LP optimum is $37/3$ but a smallest cover uses 13 facilities. The site grid and assumed speed are fixed while the coverage threshold varies:

Minutes	LP	Optimum	Fractional variables	Greedy
2–3		Some blocks have no covering site		
3.5	17	17	0	21
4	37/3	13	10	15
4.5	10.5	11	7	14
5	9	9	0	10
5.5	6.5	7	7	8
6	6	6	10	9

Fractional variables do not necessarily imply a gap in objective values: at six minutes, this LP solution is fractional but both optimum values are 6.

Albany data sources

Blocks and population: 2020 U.S. Census data, distributed as block points by Esri. Retain positive-population points inside the city boundary: 884 blocks, 99,224 people.

Boundary and roads: Census TIGERweb Albany city boundary; 2025 TIGER/Line Albany County road geometry.

Candidates: named secondary/local-road intersections. Choose the intersection nearest the center of each occupied 600 m grid cell. Sites have not been assessed for construction feasibility.

Travel model: shortest paths on an undirected road graph, at 20 mph. We add the distance from each block point to its nearest road junction (median 52 m; maximum 464 m). Roads can be traveled in either direction. The estimate leaves out traffic, turns, and time before an ambulance starts moving.

The package includes source links, the downloaded data, and scripts to check the results. Each populated block must be covered, regardless of its population; each selected base counts once.