# Structural Iterative Rounding for Generalized $k$-Median Problems

Anupam Gupta, Ben Moseley, **Rudy Zhou**

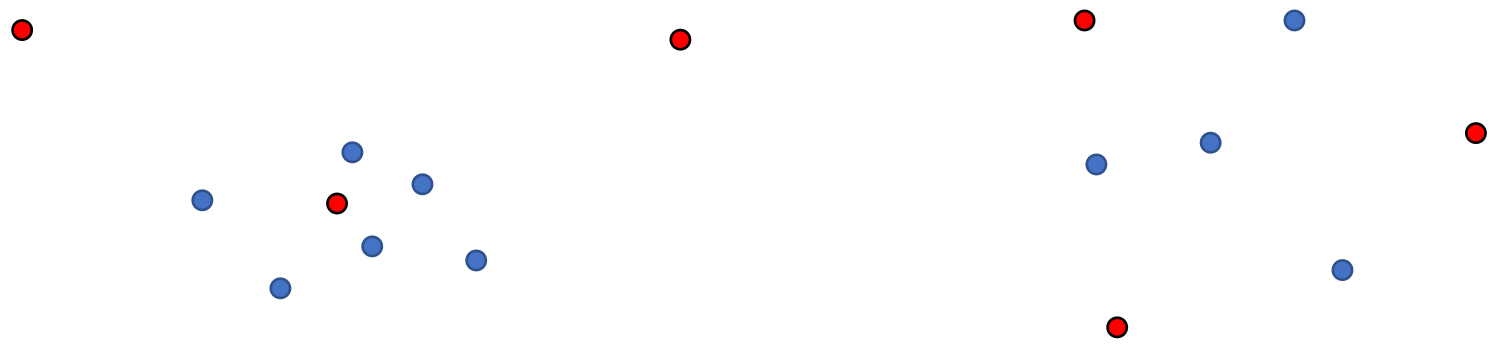
Carnegie Mellon

# $k$-Median

- $k$-Median
  - $C$ = set of Clients

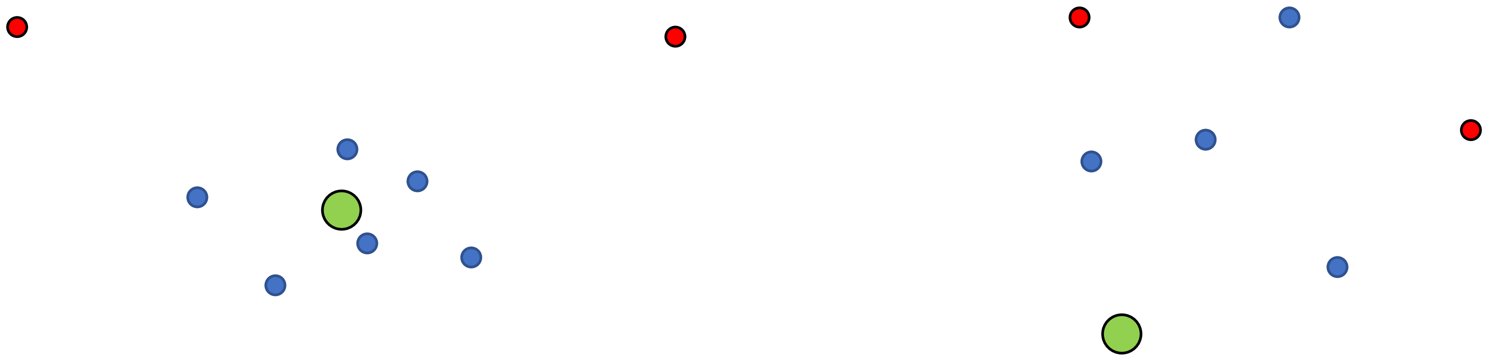
# $k$-Median

- $k$-Median
  - $C$ = set of Clients
  - $F$ = set of Facilities



# $k$-Median

- $k$-Median
  - $C$ = set of Clients
  - $F$ = set of Facilities
  - $k$ = # of Facilities to open

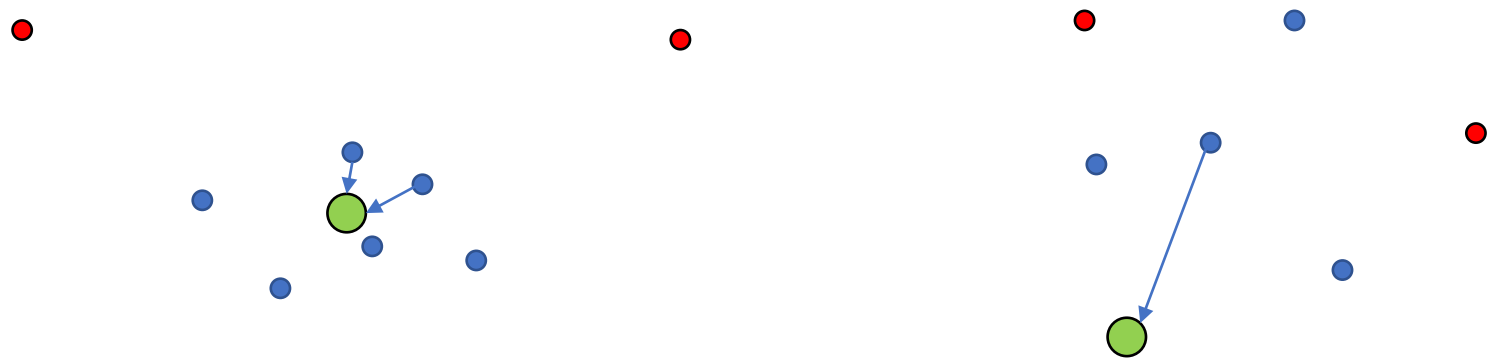


# $k$-Median

- $k$-Median
  - $C$ = set of Clients
  - $F$ = set of Facilities
  - $k$ = # of Facilities to open

**Objective:**

minimize: $\sum_{j \in C} dist(j, nearest\ open\ facil.)$



# $k$-Median Variants

- Different constraints on Clients and Facilities.

# $k$-Median Variants

- Different constraints on Clients and Facilities.

**Knapsack Median:** open $k$ facils. ⇒ open facils. satisfy knapsack constraint

# $k$-Median Variants

- Different constraints on Clients and Facilities.

**Knapsack Median:** open $k$ facils. ⇒ open facils. satisfy knapsack constraint

**$k$-Median with Outliers:** serve all clients ⇒ choose $m$ clients to serve

# $k$-Median Variants

- Different constraints on Clients and Facilities.

**Knapsack Median:** open $k$ facils. ⇒ open facils. satisfy knapsack constraint

**$k$-Median with Outliers:** serve all clients ⇒ choose $m$ clients to serve

Best approx. for both is $\sim\mathbf{7}$ via Iterative Rounding (KLS, 2018)

*Ravishankar Krishnaswamy, Shi Li, Sai Sandeep:*
*Constant approximation for k-median and k-means with outliers via iterative rounding. STOC 2018: 646-659*

# Improving on KLS: Our Results

- **Knapsack Median:** $\sim 6.3$
- $\boldsymbol{k}$**-Median with Outliers:** $\sim 6.9$
- $\boldsymbol{k}$ $-$**Median with** $\boldsymbol{O}(\mathbf{1})$ **Side Constraints:** $\sim$6.3 (pseudo-approx.)

# Improving on KLS: Our Results

- **Knapsack Median:** $\sim 6.3$
- $\boldsymbol{k}$**-Median with Outliers:** $\sim 6.9$ ← This Talk
- $\boldsymbol{k}$ **−Median with** $\boldsymbol{O(1)}$ **Side Constraints:** $\sim$6.3 (pseudo-approx.)

# Basic LP: $k$-Median with Outliers

$$\min \sum_{i \in F, j \in C} d(i,j)\, x_{ij}$$

$x_{ij} \leq y_i \qquad \forall i \in F, j \in C$

$\sum_{i \in F} x_{ij} \leq 1 \qquad \forall j \in C$

$\sum_{i \in F, j \in C} x_{ij} \geq m$

$\sum_{i \in F} y_i \leq k$

$0 \leq x, y \leq 1$

# Basic LP: $k$-Median with Outliers

$$\min \sum_{i\in F, j\in C} d(i,j)\, x_{ij}$$

$x_{ij} \le y_i \qquad \forall i \in F, j \in C$

$\sum_{i\in F} x_{ij} \le 1 \qquad \forall j \in C$

$\sum_{i\in F, j\in C} x_{ij} \ge m$

$\sum_{i\in F} y_i \le k$

$0 \le x, y \le 1$

$x_{ij}$ ~ connect client $j$ to facil. $i$

$y_i$ ~ open facil. $i$

# Basic LP: $k$-Median with Outliers

$x_{ij} \sim$ connect client $j$ to facil. $i$

$y_i \sim$ open facil. $i$

$$\min \sum_{i \in F, j \in C} d(i,j)\, x_{ij}$$

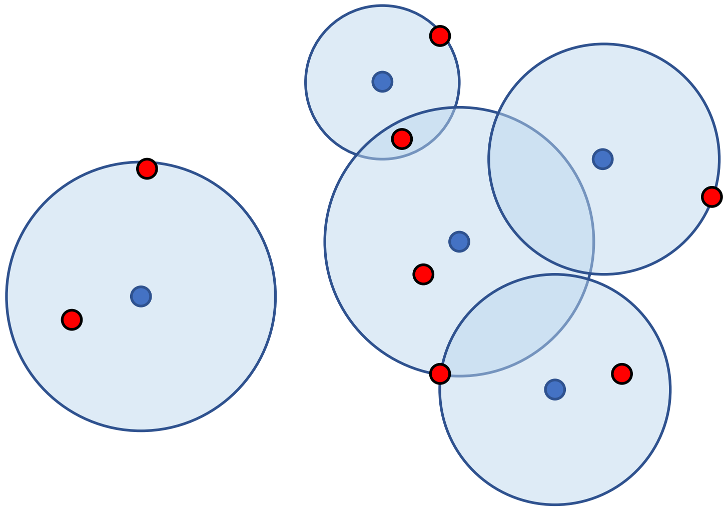
$$x_{ij} \le y_i \qquad \forall i \in F, j \in C$$

$$\sum_{i \in F} x_{ij} \le 1 \qquad \forall j \in C$$

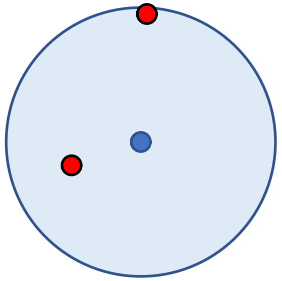
$$\sum_{i \in F, j \in C} x_{ij} \ge m$$

$$\sum_{i \in F} y_i \le k$$

$$0 \le x, y \le 1$$

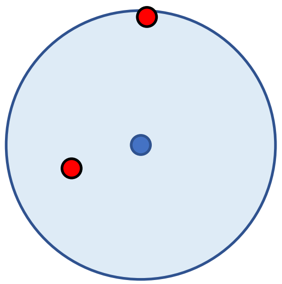


# ‘Set Cover’ LP: $k$-Median with Outliers



$$\sum_{i \in F} x_{ij} \leq 1 \qquad \forall\, j \in C$$

# ‘Set Cover’ LP: $k$-Median with Outliers

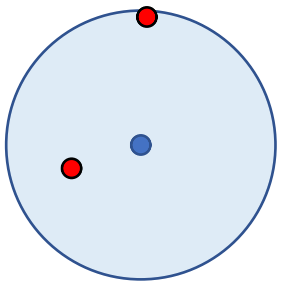


$$\sum_{i \in F} x_{ij} \leq 1 \qquad \forall\, j \in C$$

$$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall\, j \in C$$

$Ball_j = Client\ j's\ allowed\ connections$

# ‘Set Cover’ LP: $k$-Median with Outliers



$$\sum_{i \in F} x_{ij} \leq 1 \qquad \forall j \in C$$

$$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall j \in C$$

$Ball_j = Client\ j's\ allowed\ connections$

$$\min \sum_{j \in C, i \in Ball_j} d(i,j)\, y_i$$

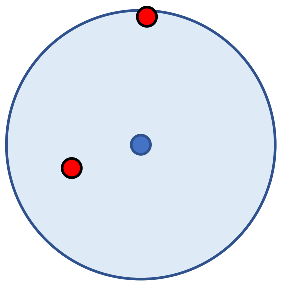
$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall j \in C$

$\sum_{j \in C, i \in Ball_j} y_i \geq m$

$\sum_{i \in F} y_i \leq k$

$0 \leq y \leq 1$

# ‘Set Cover’ LP: $k$-Median with Outliers



$$\sum_{i \in F} x_{ij} \leq 1 \qquad \forall\, j \in C$$

→

$$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall\, j \in C$$

$Ball_j = Client\ j's\ allowed\ connections$

$$\min \sum_{j \in C, i \in Ball_j} d(i,j)\, y_i$$

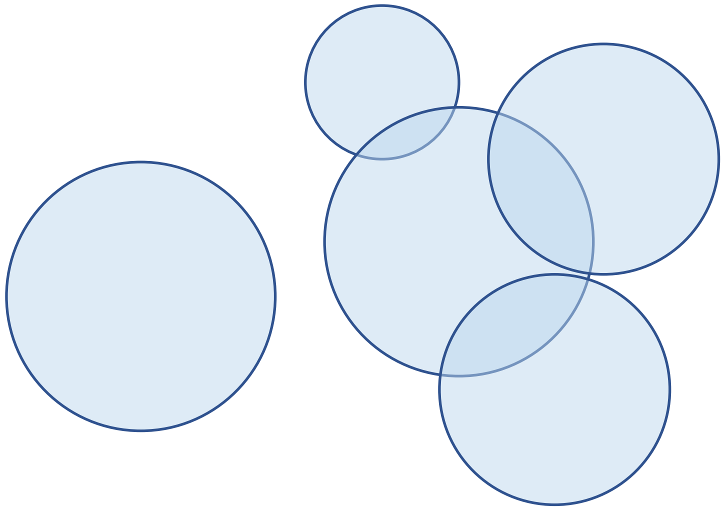
$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall\, j \in C$

$\sum_{j \in C, i \in Ball_j} y_i \geq m$

$\sum_{i \in F} y_i \leq k$

$0 \leq y \leq 1$

Cover $m$ Balls with $k$ open facils.



# Iterative Rounding Framework

**Want:**

1. Solve LP

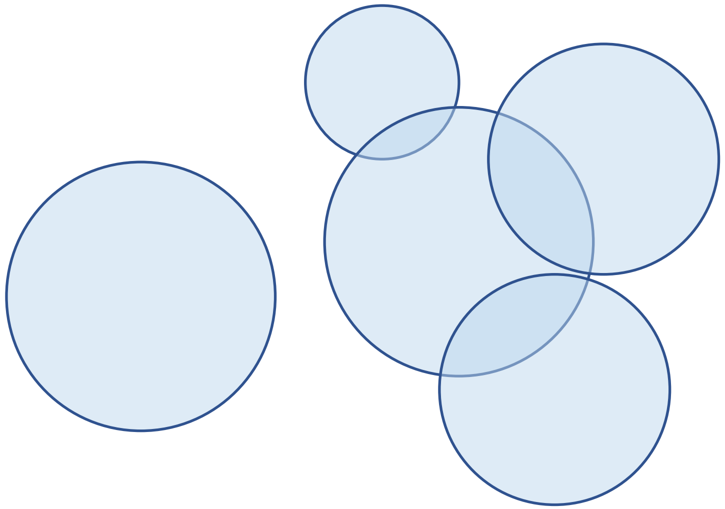
$$\min \sum_{j \in C, i \in Ball_j} d(i,j)\, y_i$$

$$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall\, j \in C$$

$$\sum_{j \in C, i \in Ball_j} y_i \geq m$$

$$\sum_{i \in F} y_i \leq k$$

$$0 \leq y \leq 1$$



# Iterative Rounding Framework

**Want:**
1. Solve LP
2. **Tight constraints are disjoint balls**

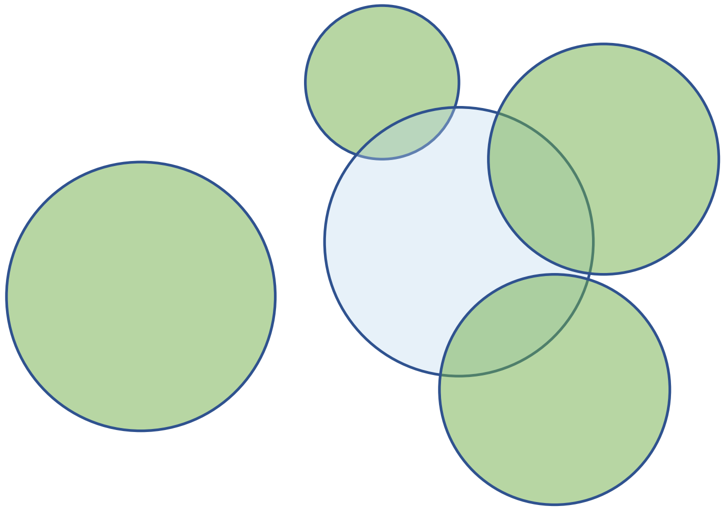
$$\min \sum_{j \in C, i \in Ball_j} d(i,j)\, y_i$$

$$\sum_{i \in Ball_j} y_i \leq 1 \qquad \forall\, j \in C$$

$$\sum_{j \in C, i \in Ball_j} y_i \geq m$$

$$\sum_{i \in F} y_i \leq k$$

$$0 \leq y \leq 1$$



# Iterative Rounding Framework

**Want:**
1. Solve LP
2. **Tight constraints are disjoint balls**

→ $\boldsymbol{O(1)}$ **frac. vars.**

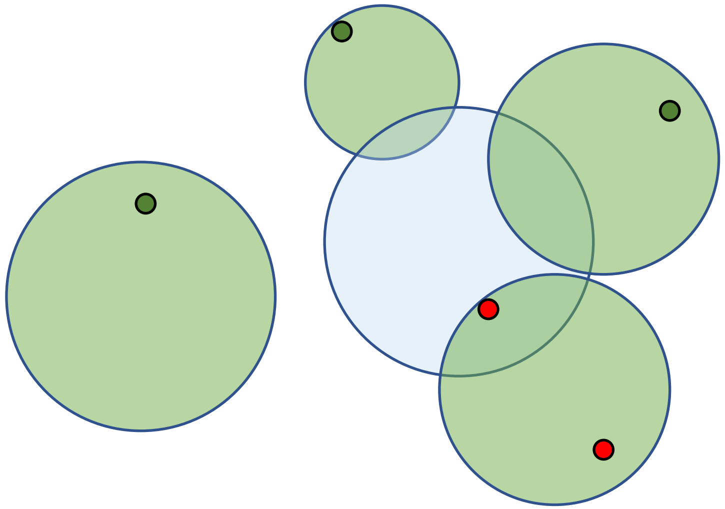
$$\min \sum_{j\in C, i\in Ball_j} d(i,j)\, y_i$$

$$\textstyle\sum_{i\in Ball_j} y_i \le 1 \qquad \forall\, j \in C$$

$$\textstyle\sum_{j\in C, i\in Ball_j} y_i \ge m$$

$$\textstyle\sum_{i\in F} y_i \le k$$

$$0 \le y \le 1$$



# Iterative Rounding Framework

**Want:**
1. Solve LP
2. **Tight constraints are disjoint balls**

→ $O(1)$ **frac. vars.**

$$\min \sum_{j \in C, i \in Ball_j} d(i,j)\, y_i$$

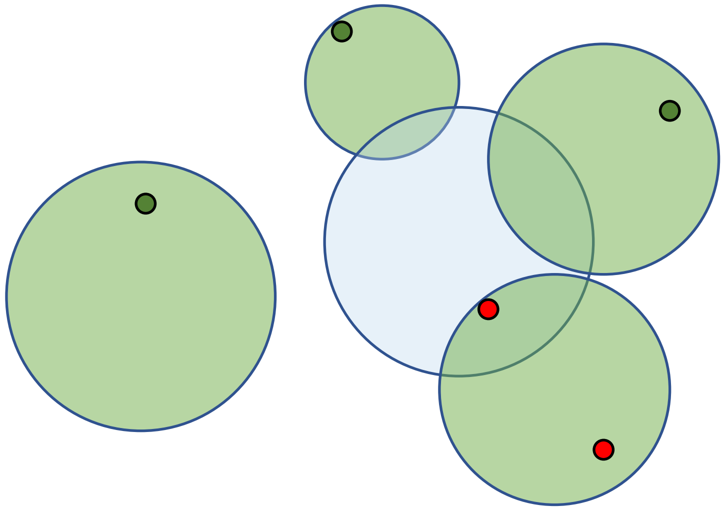
$$\sum_{i \in Ball_j} y_i \le 1 \qquad \forall j \in C$$

$$\sum_{j \in C, i \in Ball_j} y_i \ge m$$

$$\sum_{i \in F} y_i \le k$$

$$0 \le y \le 1$$

**Main Idea:** Control tight balls



# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect $\Rightarrow$ delete larger radius one

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2

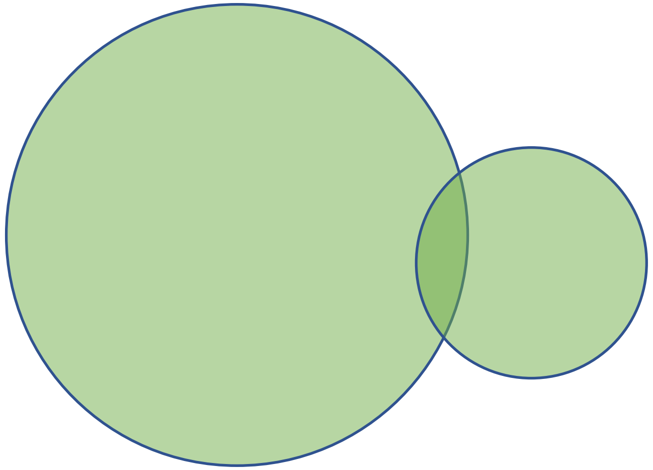
# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

**Framework:**

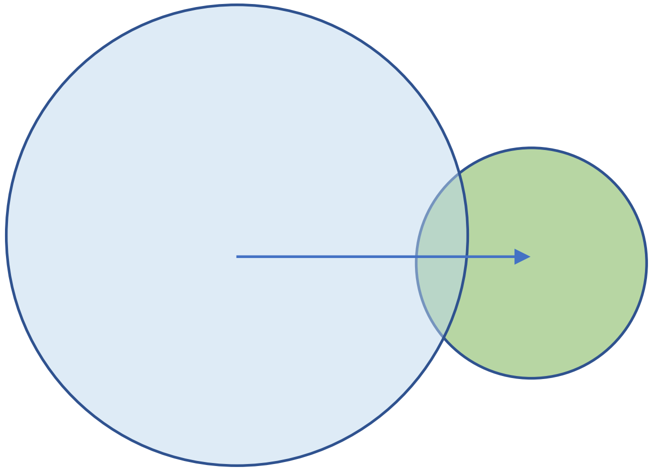
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2



# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one



**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

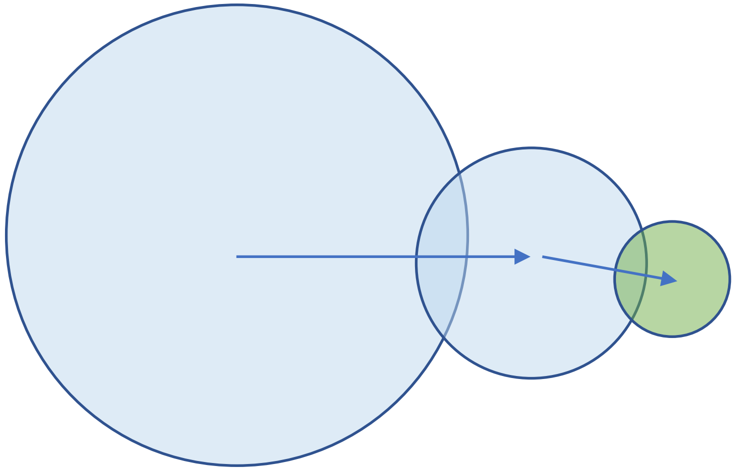
**Assume:** Radii of balls are distinct powers of 2

# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

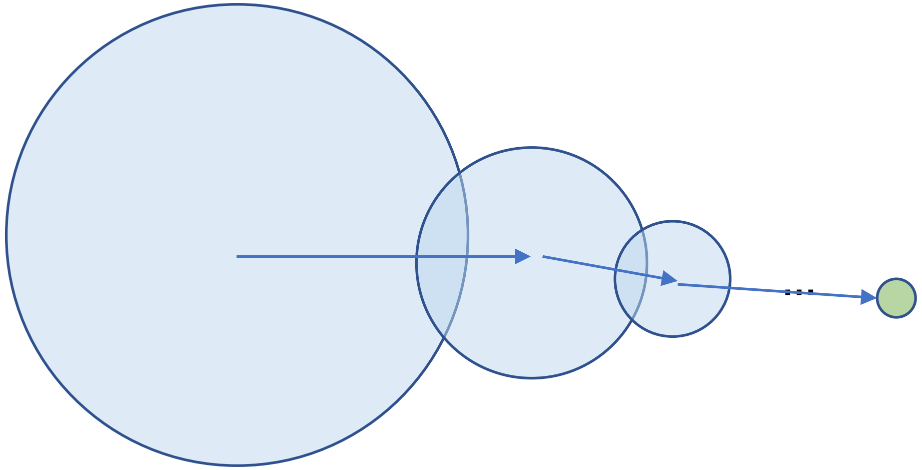
**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2



# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one




**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

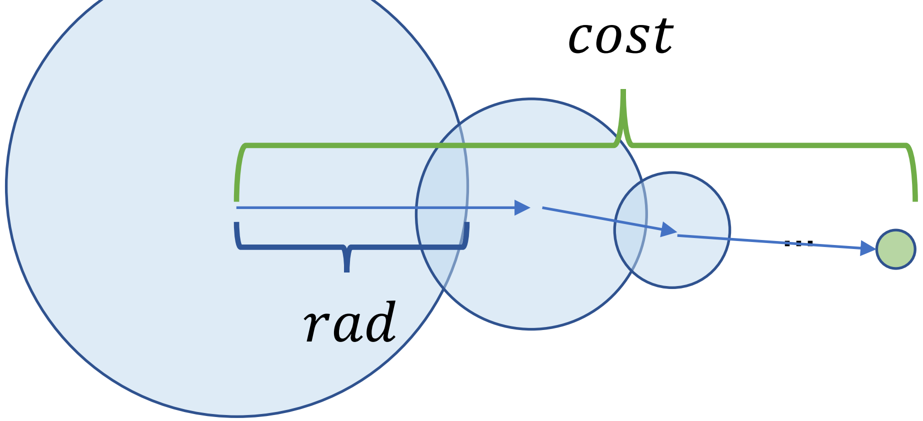
**Assume:** Radii of balls are distinct powers of 2

# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2



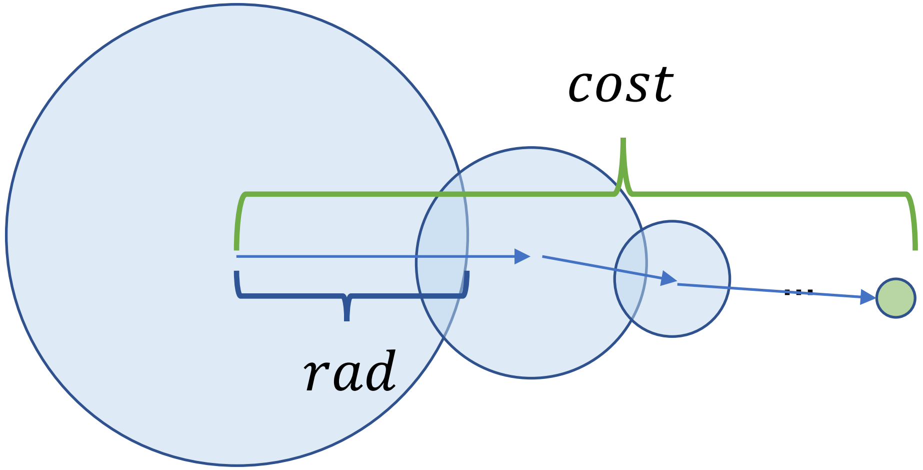
# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2




Extra dist. halves at each step

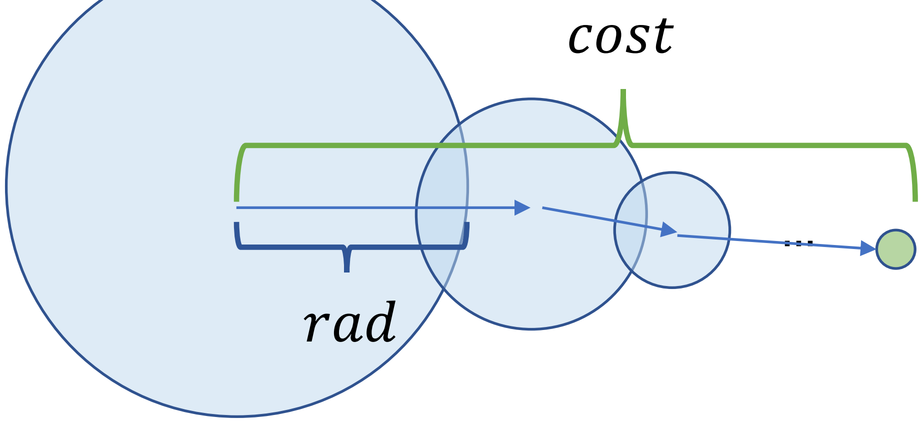
$\Rightarrow \boldsymbol{cost = O(1)\, rad}$

# Improving KLS: Our Techniques

- KLS:
  - Desired Structure = disjoint balls
  - Modify = if balls intersect ⇒ delete larger radius one

**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2




Extra dist. halves at each step
$\Rightarrow \boldsymbol{cost = O(1)\, rad}$

**Assume:** Client connection cost $= rad$

# Improving KLS: Our Techniques

- **Our Improvement:**
  - Desired Structure = **two sets of** disjoint balls
  - Modify = if **3 balls** intersect ⇒ delete **largest** radius one

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

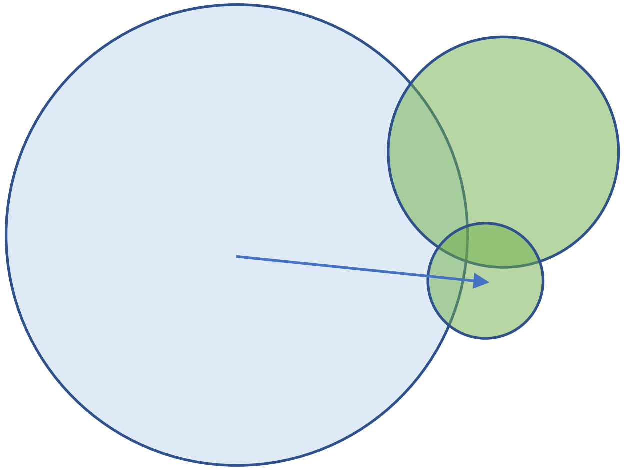
**Assume:** Radii of balls are distinct powers of 2

# Improving KLS: Our Techniques

- **Our Improvement:**
  - Desired Structure = **two sets of** disjoint balls
  - Modify = if **3 balls** intersect ⇒ delete **largest** radius one

**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

**Assume:** Radii of balls are distinct powers of 2



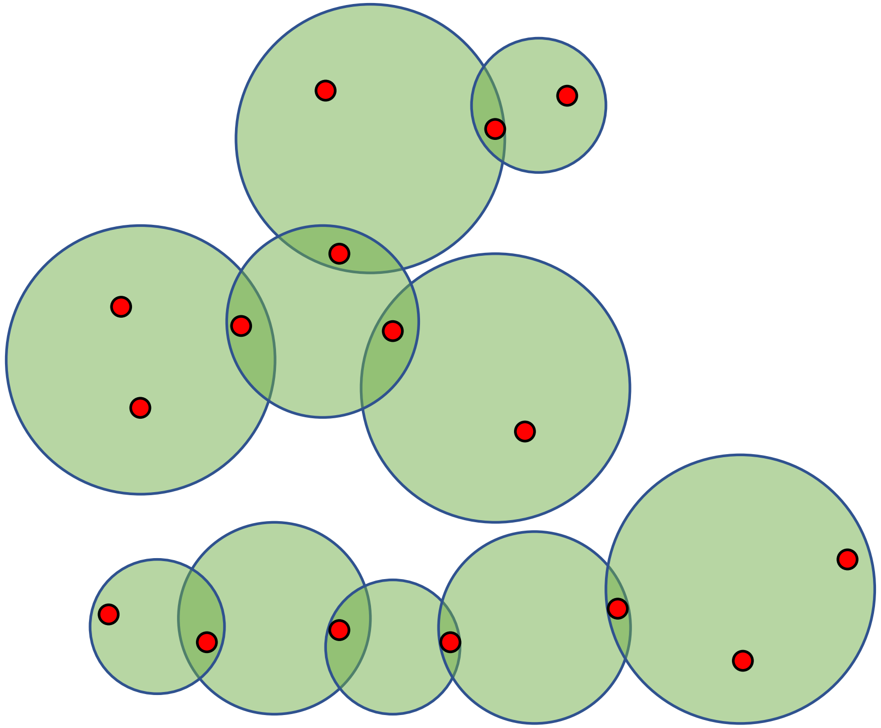
Extra dist. **quarters** at each step

# Technical Challenges

- Extra dist. halves ⇒ Extra dist. quarters (Improved approx. ratio)

# Technical Challenges

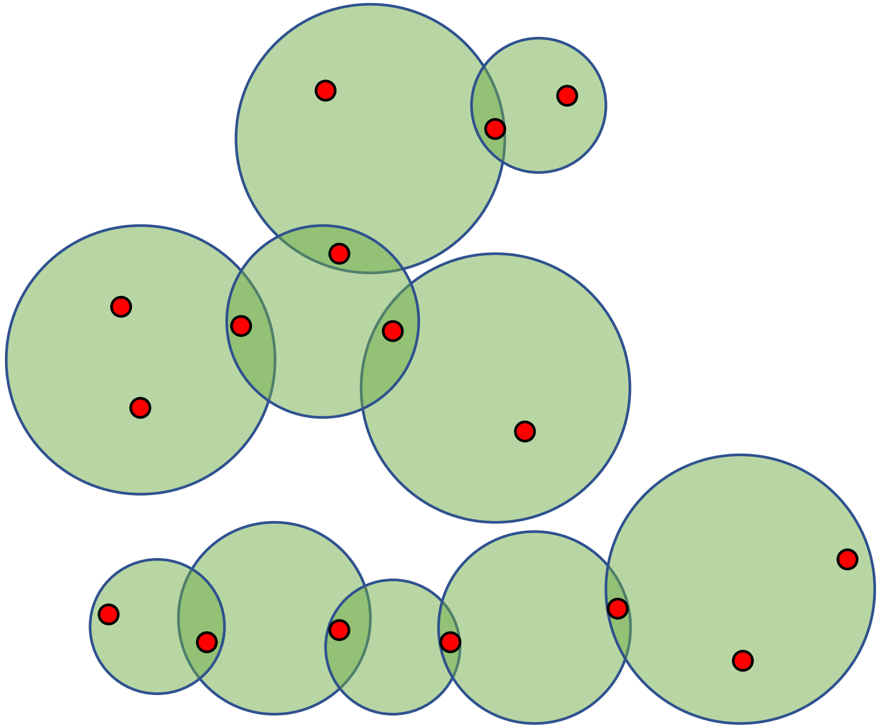
- Extra dist. halves ⇒ Extra dist. quarters (Improved approx. ratio)
- One set of disjoint balls ⇒ two sets of disjoint balls (Unbounded # of frac. vars.)



# Technical Challenges

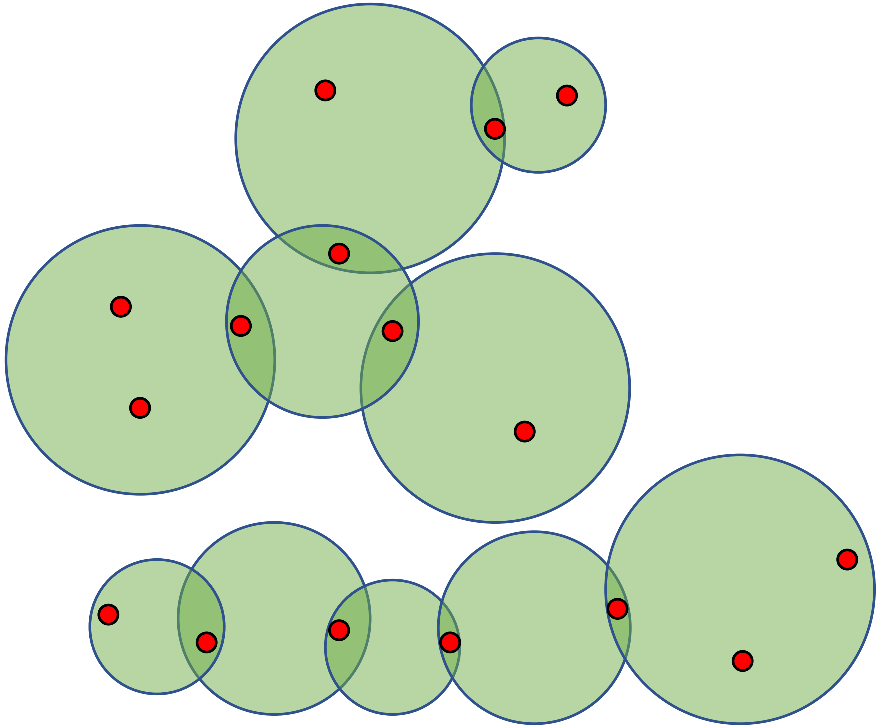
- Extra dist. halves ⇒ Extra dist. quarters (Improved approx. ratio)
- One set of disjoint balls ⇒ two sets of disjoint balls (Unbounded # of frac. vars.)

**Remains:** Reduce frac. vars. to $O(1)$



# Technical Challenges

- Extra dist. halves ⇒ Extra dist. quarters (Improved approx. ratio)
- One set of disjoint balls ⇒ two sets of disjoint balls (Unbounded # of frac. vars.)

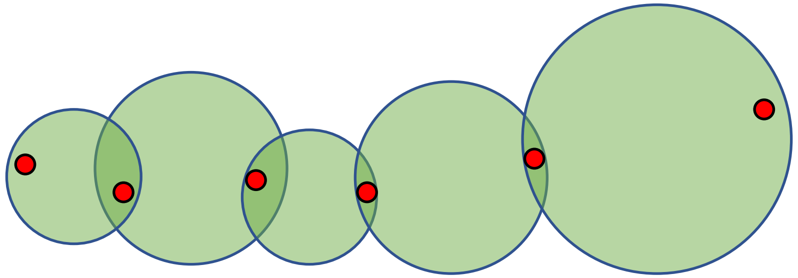


**Remains:** Reduce frac. vars. to $O(1)$

**Main Technique:** Show that extreme points are **highly-structured**; use structure to **further modify** balls

# Structure of Extreme Points

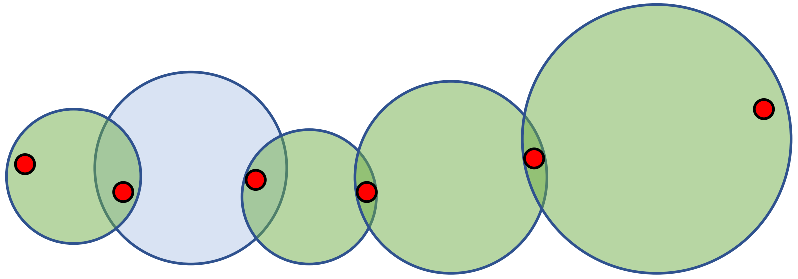
**Theorem (Informal):** The two sets of disjoint balls form **$O(1)$ disjoint 'chains' (chain decomposition)**



$\times O(1)$

# Structure of Extreme Points

**Theorem (Informal):** The two sets of disjoint balls form $\boldsymbol{O(1)}$ **disjoint 'chains' (chain decomposition)**



$\times\, O(1)$

Use chains to modify balls

# Updated Framework

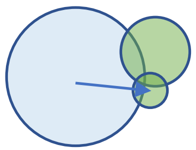
**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

# Updated Framework

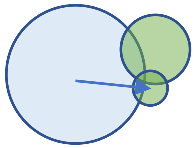
**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure



# Updated Framework

**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure
   ⇒ chain decomposition

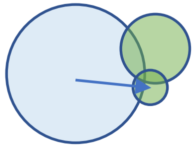


# Updated Framework

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure

   ⇒ chain decomposition



If $O(1)$ frac. vars.

Done

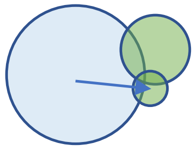
# Updated Framework

Repeat

Else $\Omega(1)$ frac. vars.

**Framework:**
1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure
   $\Rightarrow$ chain decomposition

**To do:** modify tight constraints in chain decomposition

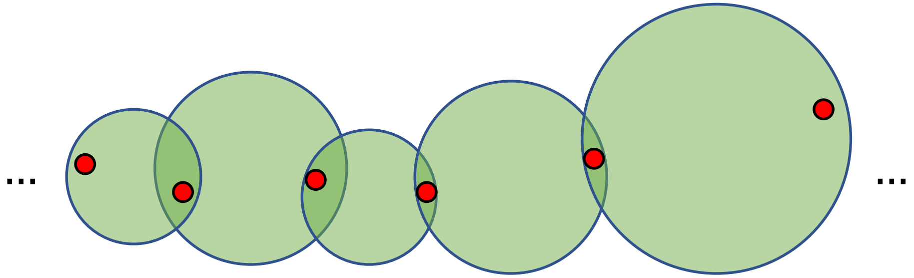


If $O(1)$ frac. vars.

Done

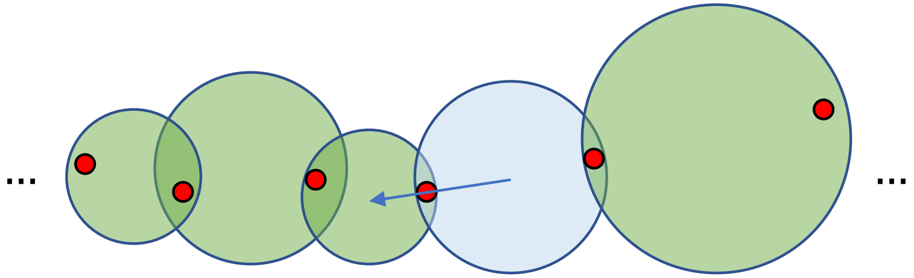
# Using Chain Decompositions

- $\Omega(1)$ frac. vars. $\Rightarrow$ some chain is long



# Using Chain Decompositions

- $\Omega(1)$ frac. vars. $\Rightarrow$ some chain is long



- $\Rightarrow$ delete ball in middle with larger radius

# Updated Framework

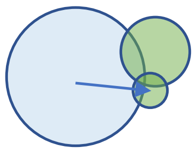
Repeat

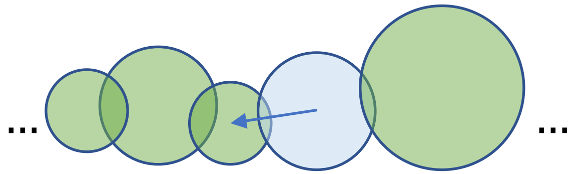
Else $\Omega(1)$ frac. vars.

**Framework:**

1. Solve LP
2. Modify tight constraints
3. Repeat until tight constraints have desired structure
   $\Rightarrow$ chain decomposition

**To do:** modify tight constraints in chain decomposition

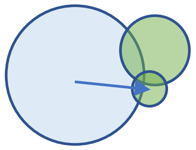
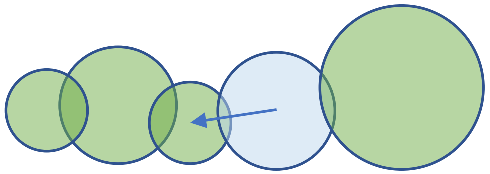




If $O(1)$ frac. vars.

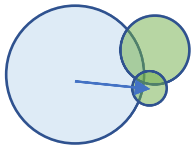
Done

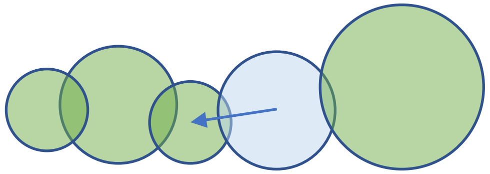
# Final Approx.

- Quarter-steps:  Half-steps: ...  ...

# Final Approx.

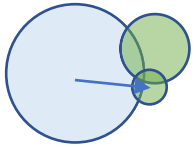
**KLS:** only half-steps

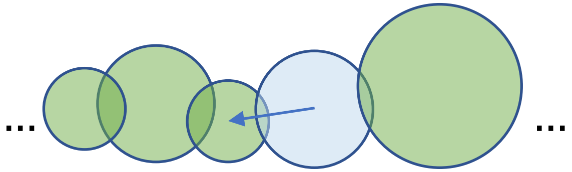
- Quarter-steps: 

Half-steps: …  …

# Final Approx.

KLS: only half-steps

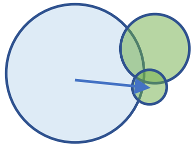
- Quarter-steps: 

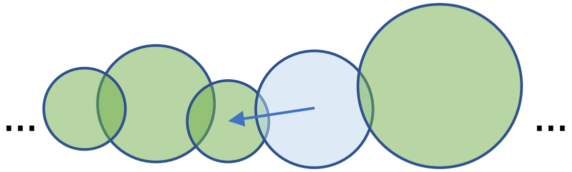
Half-steps: 

**Our Alg.:** every half-step followed by quarter-step

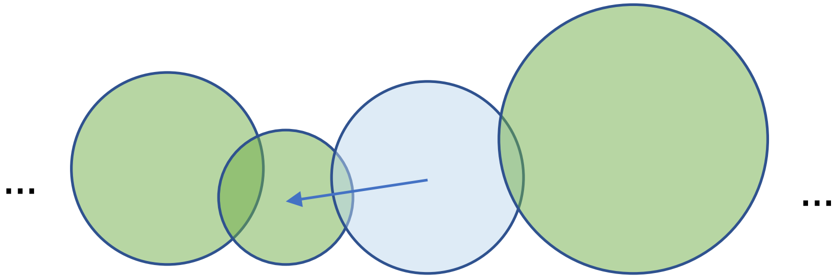
# Final Approx.

**KLS:** only half-steps

- Quarter-steps: 

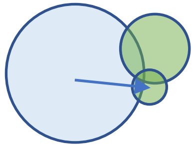
Half-steps: 

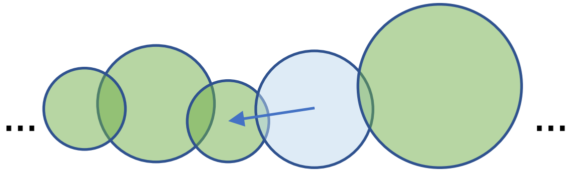
**Our Alg.:** every half-step followed by quarter-step



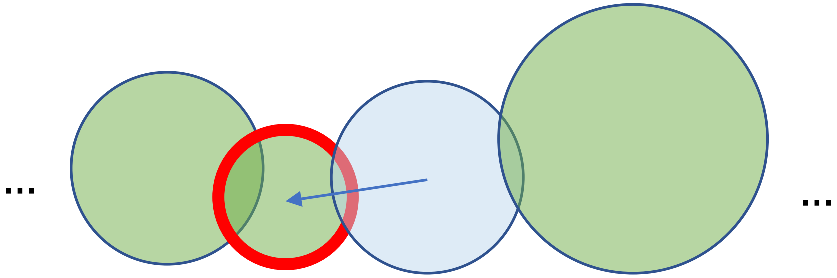
# Final Approx.

**KLS:** only half-steps

- Quarter-steps: 

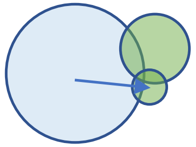
Half-steps: 

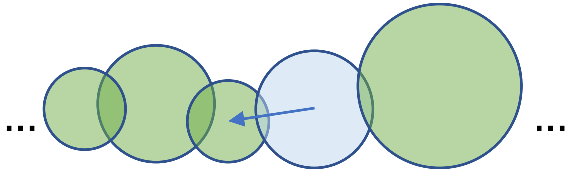
**Our Alg.:** every half-step followed by quarter-step



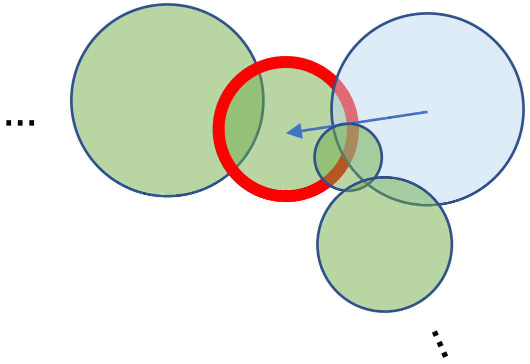
# Final Approx.

**KLS:** only half-steps

- Quarter-steps: 

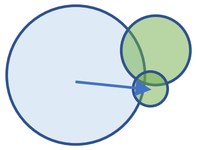
Half-steps: 

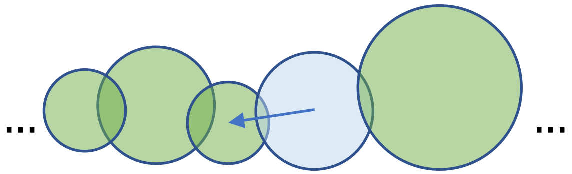
**Our Alg.:** every half-step followed by quarter-step



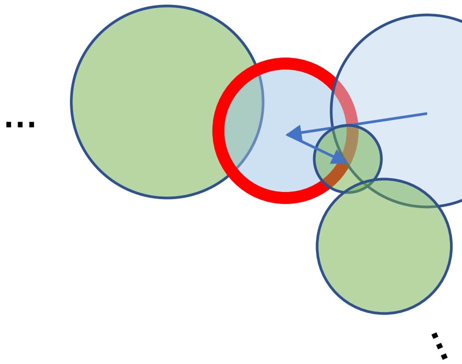
# Final Approx.

**KLS:** only half-steps

- Quarter-steps: 

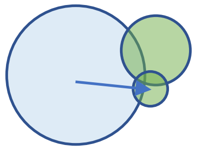
Half-steps: 

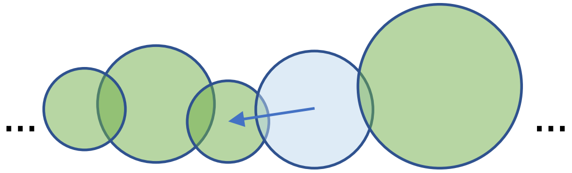
**Our Alg.:** every half-step followed by quarter-step



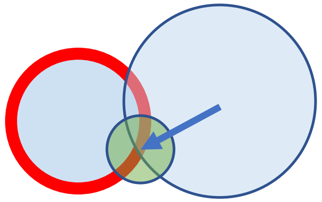
# Final Approx.

**KLS:** only half-steps

- Quarter-steps: 

Half-steps: 

**Our Alg.:** every half-step followed by quarter-step



# Summary

- Improve approx. ratio of KLS by allowing **richer sets of tight constraints**

# Summary

- Improve approx. ratio of KLS by allowing **richer sets of tight constraints**
  - **Knapsack Median:** $\sim 7 \Rightarrow \boldsymbol{\sim 6.3}$
  - $\boldsymbol{k}$**-Median with Outliers:** $\sim 7 \Rightarrow \boldsymbol{\sim 6.9}$

# Summary

- Improve approx. ratio of KLS by allowing **richer sets of tight constraints**
  - **Knapsack Median:** $\sim 7 \Rightarrow \mathbf{\sim 6.3}$
  - $\boldsymbol{k}$**-Median with Outliers:** $\sim 7 \Rightarrow \mathbf{\sim 6.9}$
- Leverage **'chain decomposition'** of extreme points to turn unbounded # of frac. vars. into $O(1)$-many.

# Summary

- Improve approx. ratio of KLS by allowing **richer sets of tight constraints**
  - **Knapsack Median:** $\sim 7 \Rightarrow \boldsymbol{\sim 6.3}$
  - $\boldsymbol{k}$**-Median with Outliers:** $\sim 7 \Rightarrow \boldsymbol{\sim 6.9}$
- Leverage **'chain decomposition'** of extreme points to turn unbounded # of frac. vars. into $O(1)$-many.

**Open Questions:**

- Can we allow even richer sets of tight constraints?
- Can we approximate $k$-Median with other side constraints? $O(1)$-many knapsack/coverage constraints?