Lecture 3: *Linear Programming & Approximation Algorithms*

Advanced Algorithms

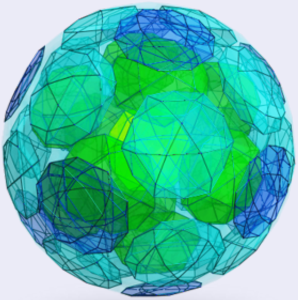
Sid Chi-Kin Chau

Australian National University

sid.chau@anu.edu.au

October 7, 2022

⋘ Polytope ⋙



What is Linear Programming (LP)

- Linear programming is a relaxation of many (integer) combinatorial optimization problems

Definition (Linear Programming)

- Given $(a_{i,j})_{i=1,...,N;j=1,...,M}$ and $(c_i)_{i=1,...,N}$ are real numbers
  - Minimize objective function $\sum_{i=1}^{N} c_i x_i$
  - Subject to $\sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ for all $j = 1, ..., M$, and $x_i \geq 0$ for all $i = 1, ..., N$
- Matrix form:
  - Minimize $(c_1, \ldots, c_N) \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix}$ (or equivalently write $\min_x c^T x$)
    - Subject to

$$\begin{pmatrix} a_{1,1} & \ldots & a_{N,1} \\ \vdots & \ddots & \vdots \\ a_{1,M} & \ldots & a_{N,M} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} \geq \begin{pmatrix} b_1 \\ \vdots \\ b_M \end{pmatrix}$$

(or equivalently write $Ax \geq b$ and $x \geq 0$)

Linear Programming: Examples

Example (Fractional Set Cover Problem)

- Given a set $\mathcal{U}$ and covers $\mathcal{S}$ with $\mathrm{Cost}(S) = c_S$ for $S \in \mathcal{S}$
- Minimize $\sum_{S \in \mathcal{S}} c_S x_S$, subject to
  - $\sum_{S \in \mathcal{S}} a_{e,S} x_S \geq 1$ for all $e \in \mathcal{U}$, where $a_{e,S} \triangleq \begin{cases} 1, & \text{if } e \in S \\ 0, & \text{otherwise} \end{cases}$
  - $x_S \geq 0$ for all $S \in \mathcal{S}$

Example (Perfect Bipartite Matching)

- Given two sets of items $\mathcal{U}$ and $\mathcal{V}$ such that $|\mathcal{U}| = |\mathcal{V}| = N$
- Minimize $\sum_{u \in \mathcal{U}, v \in \mathcal{V}} c_{u,v} x_{u,v}$, subject to
  - $\sum_{v \in \mathcal{V}} a_{u,v} x_{u,v} = 1$ for all $u \in \mathcal{U}$ and $\sum_{u \in \mathcal{U}} a_{u,v} x_{u,v} = 1$ for all $v \in \mathcal{V}$ where $a_{u,v} \triangleq \begin{cases} 1, & \text{if } u \in \mathcal{U} \text{ and } v \in \mathcal{V} \\ 0, & \text{otherwise} \end{cases}$
  - $x_{u,v} \geq 0$ for all $u \in \mathcal{U}, v \in \mathcal{V}$

Linear Programming: Examples

Example (Fractional Minimum Spanning Tree)

- Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- For each $S \subseteq \mathcal{V}$, define $\mathcal{E}(S)$ as the set of links such that both end-vertices are in S
- Minimize $\sum_{e \in \mathcal{E}} c_e x_e$
  - Subject to
    - $\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1$ for all $\varnothing \neq S \subset \mathcal{V}$
    - $\sum_{e \in \mathcal{E}(\mathcal{V})} x_e = |\mathcal{V}| - 1$
    - $x_e \geq 0$ for all $e \in \mathcal{E}$

- Although integer minimum spanning tree problem is easy to solve, other variants of minimum spanning trees are hard to solve (e.g. degree bound minimum spanning trees)
- Fractional minimum spanning tree problems are easy to solve, and can give us insight to approximate the integer version

Linear Programming: Examples

Example (Fractional Network Design Problem)

- Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Define $r(u, v)$ as the required number of disjoint paths between $u, v \in \mathcal{V}$
- Define $\delta(S)$ as the set of links with only one end-vertex in $S \subset \mathcal{V}$ (i.e. the cut set of S)
- Minimize $\sum_{e\in\mathcal{E}} c_e x_e$, subject to
  - $\sum_{e\in\delta(S)} x_e \geq \max_{u\in S, v\notin S} r(u, v)$ for all $S \subset \mathcal{V}$
  - $0 \leq x_e \leq 1$ for all $e \in \mathcal{E}$

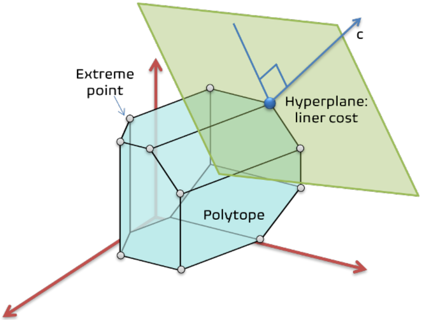
- Many networking problems are instances of network design problem
  - Source-destination connectivity problem
  - Minimum spanning tree problem
  - Minimum Steiner tree problem

Feasible Solutions of Linear Programming

Definition (Feasible Solutions)

- If x satisfies $(Ax \geq b, x \geq 0)$, then x is a *feasible* solution
- The set of feasible solutions define a *polytope* (i.e. a multi-dimensional polygon in multi-dimensional space), let $\mathbb{P} = \{x : Ax \geq b, x \geq 0\}$
- $x \in \mathbb{P}$ is called an *optimal* solution, if it minimizes $c^T x$
- $x \in \mathbb{P}$ is called an *extreme point* of a polytope $\mathbb{P}$, if there does not exist y such that $x + y \in \mathbb{P}$ and $x - y \in \mathbb{P}$ (i.e. extreme points are the end corner points)
- Extreme points are also called *vertex solutions*

Visualization of Polytopes



- 2D/3D polytopes of LP problem can be visualized easily
- A LP problem defines a class of polytopes
- The geometry of polytope tells a lot about the solutions of a LP problem

Feasible Solutions of Linear Programming

Lemma

If $\mathbb{P}$ is finite, then there exists an extreme point that is an optimal solution

Basic idea:

- Points of tangent intersecting a hyperplane of objective function and a polytope must lie on the boundary of $\mathbb{P}$
- An optimal solution must lie on the boundary of $\mathbb{P}$ (e.g. hyperplanes or extreme points)
- If $\mathbb{P}$ is finite, every edge and hyperplane contains end-points, which are extreme points
- Hence, we can restrict our attention to extreme points (or vertex solutions) only

Properties of Vertex Solutions

Definition (Linear Independence & Rank)

- A set of $V = \{v_1, v_2, ..., v_n\}$ is *linearly independent*, if none of them can be expressed as a linear combination of finitely many vectors in V
- A linear mapping can be represented as a matrix $x \mapsto Ax$
  - If column vectors in A are not linear independent, then A can be replaced by another matrix A with lower dimension. Linear independence captures irreducibility of linear mapping

Definition (Column or Row Rank)

- The *column* (or *row*) rank of a matrix A is the maximum number of linearly independent column (or row) vectors of A

Lemma (Column Rank = Row Rank)

- *The column rank of a matrix A is the same as the row rank of A*
  - *Proof using Gaussian elimination*

Optimal Solutions of Linear Programming

Lemma

- *Given a polytope* $\mathbb{P} = \{x : Ax \geq b, x \geq 0\}$. *For a particular* $x \in \mathbb{P}$, *let* $A_x^=$ *be the submatrix, such that the* j*-th column vector of* A *is in* $A_x^=$ *if* $A_j x = b_j$ *and* $x_j > 0$
- *If the column vectors of* $A_x^=$ *are linear independent, and* $\mathrm{Rank}(A) = \mathrm{Rank}(A_x^=)$, *then* x *is an extreme point of* $\mathbb{P}$

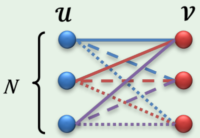
Basic idea:

- Extreme points are uniquely determined by the tight constraints (e.g. $Ax = b$)
- For $x_j > 0$, x_j must be uniquely determined by a constraint $A_x^= x = b$
- Linear independence can guarantee a unique solution of $A_x^= x = b$
- Hence, we can characterize the vertex solutions of LP problem by the linear independence of matrix A

Optimal Integer Solutions of Perfect Bipartite Matching

Example (Perfect Bipartite Matching)

- Minimize $\sum_{u\in\mathcal{U},v\in\mathcal{V}} c_{u,v}x_{u,v}$, subject to
  - $\sum_{v\in\mathcal{V}} a_{u,v}x_{u,v} = 1$ for all $u \in \mathcal{U}$ and $\sum_{u\in\mathcal{U}} a_{u,v}x_{u,v} = 1$ for all $v \in \mathcal{V}$
  - $x_{u,v} \geq 0$ for all $u \in \mathcal{U}, v \in \mathcal{V}$
  - Let $|\mathcal{U}| = |\mathcal{V}| = N$



Theorem (Integrity of LP Perfect Bipartite Matching Solution)

Optimal vertex solutions for LP bipartite matching are integers $\{0, 1\}$

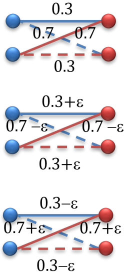
Proof:

- Use contradiction – suppose some non-zero $x_{u,v}$ is fractional (e.g. $x_{u,v} < 1$ for some u, v)
- Since $\sum\limits_{v\in\mathcal{V}} a_{u,v}x_{u,v} = 1$, there is at least one another fractional $x_{u,v}$
- There are at least $2k$ fractional $x_{u,v}$'s in $2k$ constraints. The fractional $x_{u,v}$'s form a cycle

Optimal Integer Solutions of Perfect Bipartite Matching

Proof (Cont.):

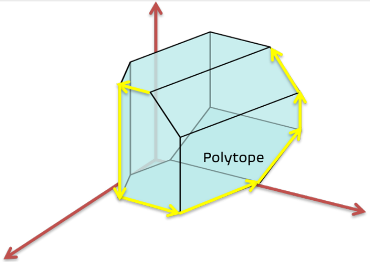
- Divide the fractional $x_{u,v}$'s in the cycle into odd and even edges
- Let x' be the same as the original x, except that the fractional value of every odd edge is increased by ϵ and the fractional value of every even edge is decreased by ϵ for some $\epsilon > 0$
- Let x'' be the same as the original x, except that the fractional value of every odd edge is decreased by ϵ and the fractional value of every even edge is increased by ϵ for the same $\epsilon > 0$
- Therefore, the fractional solution x can not be a vertex solution since x', x'' are also feasible. This proves that a vertex solution must be integers $\{0, 1\}$



How to solve Linear Programming?

Simplex Algorithm (Informal Description)

- Start from an extreme point
- Move to a better neighbor that improves the cost
- Iterate until cannot find a better neighbor



- How do we choose a neighbor when there multiple choices?
  - No rule can guarantee polynomial running time of simplex algorithm in the worst case

How to solve Linear Programming?

- Transform an optimization problem into a problem of finding a point inside $\mathbb{P}$
  - But the solution is not exact with small tolerance

Ellipsoid Algorithm (Informal Description)

- Start with a large ellipsoid which contains $\mathbb{P}$
- Test if the center of ellipsoid c is inside $\mathbb{P}$
- If not, identify the linear constraint in $Ax \geq b$ for which c is violated.
- Find a minimum ellipsoid which contains the intersection of previous ellipsoid and the violated constraint
- Iterate with the new (smaller) ellipsoid until ellipsoid is sufficiently small

- The running time of Ellipsoid Algorithm is polynomial under some assumptions
  - It needs a separation oracle: Given x, check if $x \in \mathbb{P}$ or return the violated linear constraint
  - Separation oracle must have polynomial running time (e.g. when there are a polynomial number of constraints)

Solve Integer Problems by LP-based Approximation

An Informal Recipe for LP-based Approximation Algorithm 🧑‍🍳

- Relax the integer constraints to allows fractional solutions
- Solve the fractional LP solutions by a polynomial-time LP algorithm
- Transform the fractional LP solutions to integer solutions to satisfy the feasibility of integer constraints
  - For example, rounding the fractional solutions to the nearest integers

- How to transform fractional LP solutions to integer solutions while satisfying the feasibility of integer constraints?
- How to guarantee that the rounding error is small? How to characterize the approximation ratio of rounding?
- ⚠ But rounding may be infeasible or give a large error

Is Rounding always Feasible?

- $\min_x c^T x$ subject to $Ax = b$ and $x \geq 0 \quad \Leftrightarrow \quad \min_x c^T x$ subject to $A'x \geq b$ and $x \geq 0$
- LP with equality constraints is equivalent to LP with inequality constraints:
  - $\sum_{i=1}^{N} a_{i,j} x_i = b_j \quad \Leftrightarrow \quad \sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ and $\sum_{i=1}^{N} a_{i,j} x_i \leq b_j$

Example (Infeasibility of Rounding in LP Approximation)

- Consider a simple linear programing problem:
- Maximize $2x_1 + x_2$, subject to
  - $3x_1 + 3x_2 = 2$
  - $x_1, x_2 \geq 0$ for all $u, v \in \mathcal{V}$
- The optimal fractional solution is fractional
- However, there is no integer feasible solutions

⋘ Duality ⋙

DUALITY [NOUN]: *the quality or state of having two different or opposite parts or elements* – **Merriam-Webster**

Duality of Linear Programming

Definition (Primal)

- Minimize objective function $\sum_{i=1}^{N} c_i x_i$
- subject to
  - $\sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ for all $j = 1, ..., M$
  - $x_i \geq 0$ for all $i = 1, ..., N$

Definition (Dual)

- Maximize objective function $\sum_{j=1}^{M} b_j y_j$
- subject to
  - $\sum_{j=1}^{M} a_{i,j} y_i \leq c_i$ for all $i = 1, ..., N$
  - $y_j \geq 0$ for all $j = 1, ..., M$

	Primal	Dual
Objective	Minimization	Maximization
Variables	$\{x_i : i = 1, ..., N\}$	$\{y_j : j = 1, ..., M\}$
Linear Costs	$\{c_i : i = 1, ..., N\}$	$\{b_j : j = 1, ..., M\}$
Linear Constraints	$\{b_j : j = 1, ..., M\}$	$\{c_i : i = 1, ..., N\}$

Duality of Linear Programming

Example (MaxFlow and MinCut)

- Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a set of paths $\mathcal{P}$ in $\mathcal{G}$
- Each link $e \in \mathcal{E}$ has a capacity c_e

Definition (MinCut Problem)

- Minimize $\sum_{e \in \mathcal{E}} c_e x_e$
- subject to
  - $\sum_{e \in p} x_e \geq 1$ for all $p \in \mathcal{P}$
  - $x_e \geq 0$ for all $e \in \mathcal{E}$

Definition (MaxFlow Problem)

- Maximize $\sum_{p \in \mathcal{P}} y_p$
- subject to
  - $\sum_{p \in \mathcal{P}: e \in p} y_p \leq c_e$ for all $e \in \mathcal{E}$
  - $y_p \geq 0$ for all $p \in \mathcal{P}$

Duality of Linear Programming

Example (Covering and Packing)

- $\sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ can be regarded as covering constraints
- $\sum_{i=1}^{N} a_{i,j} y_j \leq c_i$ can be regarded as packing constraints
- Many covering and packing problems are dual to each other:

Primal (Covering problems)	Dual (Packing problems)
Minimum set cover	Maximum set packing
Minimum vertex cover	Maximum matching
Minimum edge cover	Maximum independent set

Why Duality?

- A change of perspective could give an alternate (maybe easier) approach
  - Vertex Cover (Hard) $\leftrightarrow$ Maximal Matching (Easy)
- Combining primal and dual programs give a complete picture to solve a problem
  - **Primal-Dual Schema** for approximation algorithms:
    - ★ Start with initial feasible solutions to the primal and dual programs
    - ★ Iteratively, modify the primal and dual solutions integrally to satisfy *complementary slackness* conditions
    - ★ Output the solutions when all complementary slackness conditions are satisfied

Duality of Linear Programming

Theorem (Weak Duality)

For any feasible solution of primal problem $\{x_i : i = 1, ..., N\}$ *and any feasible solution of dual problem* $\{y_j : i = 1, ..., M\}$*, we have*

$$\sum_{i=1}^{N} c_i x_i \geq \sum_{j=1}^{M} b_j y_j$$

Proof:

- $\sum_{i=1}^{N} x_i c_i \geq \sum_{i=1}^{N} x_i \big(\sum_{j=1}^{M} a_{i,j} y_j \big) = \sum_{j=1}^{M} y_j \big(\sum_{i=1}^{N} a_{i,j} x_i \big) \geq \sum_{j=1}^{M} y_j b_j$

Theorem (Strong Duality)

Let an optimal solution to primal problem be $\{x_i^* : i = 1, ..., N\}$ *and an optimal solution to dual problem be* $\{y_j^* : i = 1, ..., M\}$*, we have*

$$\sum_{i=1}^{N} c_i x_i^* = \sum_{j=1}^{M} b_j y_j^*$$

Duality of Linear Programming

Theorem (Complementary Slackness)

For an optimal solution of primal problem $\{x_i^* : i = 1, ..., N\}$ *and an optimal solution of dual problem* $\{y_j^* : i = 1, ..., M\}$*, we have*

- *Either* $x_i^* = 0$ *or* $\sum_{j=1}^{M} a_{i,j} y_j^* = c_i$ *(primal complementary slackness condition)*
- *Either* $y_j^* = 0$ *or* $\sum_{i=1}^{N} a_{i,j} x_i^* = b_j$ *(dual complementary slackness condition)*

Proof:

- $\sum_{i=1}^{N} x_i^* c_i \geq \sum_{i=1}^{N} x_i^* \big(\sum_{j=1}^{M} a_{i,j} y_j^* \big) = \sum_{j=1}^{M} y_j^* \big(\sum_{i=1}^{N} a_{i,j} x_i^* \big) \geq \sum_{j=1}^{M} y_j^* b_j$
- By strong duality ($\sum_{i=1}^{N} c_i x_i^* = \sum_{j=1}^{M} b_j y_j^*$), to make the equality holds, we must require
  - $x_i^* > 0 \Rightarrow \sum_{j=1}^{M} a_{i,j} y_j^* = c_i$
  - $y_j^* > 0 \Rightarrow \sum_{i=1}^{N} a_{i,j} x_i^* = b_j$
  - $\sum_{j=1}^{M} a_{i,j} y_j^* < c_i \Rightarrow x_i^* = 0$
  - $\sum_{i=1}^{N} a_{i,j} x_i^* > b_j \Rightarrow y_j^* = 0$

Primal vs. Dual

Primal Fractional Problem

- Minimize objective function $\sum_{i=1}^{N} c_i x_i$
- subject to
  - $\sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ for all $j = 1, ..., M$
  - $x_i \geq 0$ for all $i = 1, ..., N$

Dual Fractional Problem

- Maximize objective function $\sum_{j=1}^{M} b_j y_j$
- subject to
  - $\sum_{j=1}^{M} a_{i,j} y_i \leq c_i$ for all $i = 1, ..., N$
  - $y_j \geq 0$ for all $j = 1, ..., M$

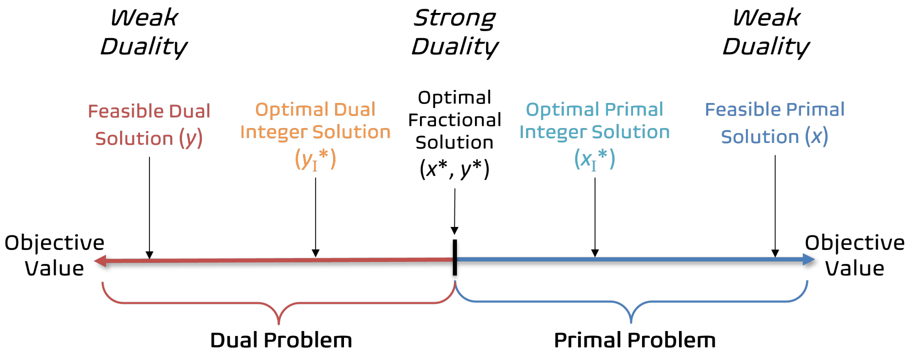
Primal Integer Problem

- Minimize objective function $\sum_{i=1}^{N} c_i x_i$
- subject to
  - $\sum_{i=1}^{N} a_{i,j} x_i \geq b_j$ for all $j = 1, ..., M$
  - x_i is a non-negative integer for all $i = 1, ..., N$

Dual Integer Problem

- Maximize objective function $\sum_{j=1}^{M} b_j y_j$
- subject to
  - $\sum_{j=1}^{M} a_{i,j} y_i \leq c_i$ for all $i = 1, ..., N$
  - y_j is a non-negative integer for all $j = 1, ..., M$

Primal vs. Dual



Approximation Algorithm by Primal-Dual Schema

- Relaxed primal and dual complementary slackness conditions:
  - Allow a violation gap of $\alpha \geq 1$, such that for each $1 \leq i \leq N$,
  $$\text{either } x_i = 0 \text{ or } \tfrac{c_i}{\alpha} \leq \sum_{j=1}^{M} a_{i,j} y_j \leq c_i$$
  - Allow a violation gap of $\beta \geq 1$, such that for each $1 \leq j \leq M$,
  $$\text{either } y_j = 0 \text{ or } b_i \leq \sum_{i=1}^{N} a_{i,j} x_i \leq \beta b_j$$
- Then, we obtain an approximation ratio as $\alpha\beta$:
$$\sum_{j=1}^{M} b_j y_j \leq \sum_{i=1}^{N} c_i x_i \leq \alpha\beta \cdot \sum_{j=1}^{M} b_j y_j$$
- No violation in primal complementary slackness condition: set $\alpha = 1$, but let $\beta > 1$
- No violation in dual complementary slackness condition: set $\beta = 1$, but let $\alpha > 1$

SetCover Approximation by Primal-Dual Schema

Definition (SetCover)

- Minimize $\sum_{S\in\mathcal{K}}^{N} c_S x_S$
- subject to
  - $\sum_{S:e\in S} x_S \geq 1$ for all $e \in \mathcal{U}$
  - $x_S \in \{0,1\}$ for all $S \in \mathcal{K}$

Definition (Primal (Fractional SetCover))

- Minimize $\sum_{S\in\mathcal{K}}^{N} c_S x_S$
- subject to
  - $\sum_{S:e\in S} x_S \geq 1$ for all $e \in \mathcal{U}$
  - $x_S \leq 1$ for all $S \in \mathcal{K}$
  - $x_S \geq 0$ for all $S \in \mathcal{K}$

Definition (Dual)

- Maximize objective function $\sum_{e\in\mathcal{U}} y_e$
- subject to
  - $\sum_{e\in\mathcal{U}:e\in S} y_e - z_S \leq c_S$ for all $S \in \mathcal{K}$
  - $y_e \geq 0$ for all $e \in \mathcal{U}$
  - $z_S \geq 0$ for all $S \in \mathcal{K}$

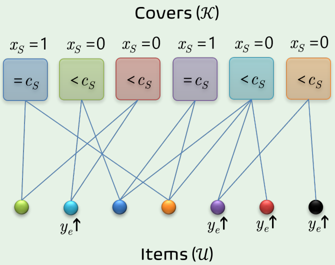
SetCover Approximation by Primal-Dual Schema

Algorithm $\mathcal{A}_{\text{PDsetcover}}$

- $y \leftarrow 0;\ x \leftarrow 0;\ \tilde{\mathcal{K}} \leftarrow \varnothing;\ \mathcal{C}_1 \leftarrow \varnothing;\ k \leftarrow 1$
- While $\mathcal{C}_k \neq \mathcal{U}$
  - For each $e \in \mathcal{U} \backslash \mathcal{C}_k$, increase the corresponding y_e at the same rate, until there is some S, where $e \in S$, such that

$$\sum_{e \in \mathcal{U}: e \in S} y_e = c_S$$

  - $x_S \leftarrow 1;\ \tilde{\mathcal{K}} \leftarrow \tilde{\mathcal{K}} \cup \{S\}$
  - $\mathcal{C}_{k+1} \leftarrow \mathcal{C}_k \cup S$
  - $k \leftarrow k + 1$
- Return $\tilde{\mathcal{K}}$



SetCover Approximation by Primal-Dual Schema

- Let f be the maximum number of covers in $\mathcal{K}$ that cover the same item

$$f \triangleq \max_{e \in \mathcal{U}} \Big| \{S \in \mathcal{K} : e \in S\} \Big|$$

Theorem

The approximation ratio of $\mathcal{A}_{\text{PDsetcover}}$ is f

Proof:

- Since each item can be covered by at most f times, the violation of dual complementary slackness condition is at most f
- Namely, $1 \leq \sum_{S:e \in S} x_S \leq f$, noting that $x_S \leq 1$ for all S
- The violation gaps are $\beta = f$ and $\alpha = 1$

Primal-Dual Schema for Online Algorithms

- The approximation ratio of $\mathcal{A}_{\text{PDsetcover}}$ is $f = \mathrm{O}(n)$, but the one of greedy algorithm $\mathcal{A}_{\text{setcover}}$ is $\mathrm{O}(\log(n))$. Why do we bother with primal-dual schema?
  - If f is small, $\mathcal{A}_{\text{PDsetcover}}$ can outperform $\mathcal{A}_{\text{setcover}}$
  - Primal-dual schema also allows online decisions - it does not depend on other unknown covers
- Example of primal-dual schema for online algorithms: Ad Auction
  - Match buyers with a stream of unknown keywords to maximize total revenue

References

Reference Materials

- Design of Approximation Algorithms (Williamson, Shmoys), Cambridge University Press
  - Chapters 1, 7, Appendix A
- Approximation Algorithms (V. Vazirani), Springer
  - Chapters 12-13