Cutting Plane Methods II

Gomory-Chvátal cuts

Reminder

- $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ with $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$.
- For $\lambda \in [0,1)^m$ such that $\lambda^\intercal A \in \mathbb{Z}^n$,

$$(\lambda^\intercal A)x \leq \lfloor \lambda^\intercal b \rfloor$$

  is valid for all integral points in P.

Stable Sets

Definitions

- Let $G = (V, E)$ be an undirected graph.
- $S \subseteq V$ is *stable* if $\big\{\{u,v\} \in E : u, v \in S\big\} = \emptyset$.
- Stable sets are the integer solutions to:

$$\begin{aligned} x_u + x_v &\leq 1 && \text{for all } \{u,v\} \in E \\ x_v &\geq 0 && \text{for all } v \in V \end{aligned}$$

- The stable set polytope is

$$P_{\text{stab}}(G) = \text{conv}\big\{x \in \{0,1\}^V : x_u + x_v \leq 1 \text{ for all } u, v \in E\big\}$$

Odd Cycle Inequalities

- An *odd cycle* C in G consists of an odd number of vertices $0, 1, \ldots, 2k$ and edges $\{i, i+1\}$.
- The odd cycle inequality

$$\sum_{v \in C} x_v \leq \frac{|C| - 1}{2}$$

  is valid for $P_{\text{stab}}(G)$.
- It has a cutting-plane proof that only needs one step of rounding.
- The separation problem for the class of odd cycle inequalities can be solved in polynomial time:
- Let $y \in \mathbb{Q}^V$.

- We may assume that $y \geq 0$ and $y_u + y_v \leq 1$ for all $\{u, v\} \in E$.
- Define, for each edge $e = \{u, v\} \in E$, $z_e := 1 - y_u - y_v$.
- So $z_e \geq 0$ for all $e \in E$.
- y satisfies all odd cycle constraints iff z satisfies

$$\sum_{e \in C} z_e \geq 1 \text{ for all odd cycles } C.$$

- If we view z_e as the "length" of edge e, then y satisfies all odd cycle inequalities iff the length of a shortest odd cycle is at least 1.

Shortest Odd Cycles

- A shortest odd cycle can be found in polynomial time:
- Split each node $v \in V$ into two nodes v_1 and v_2.
- For each arc (u, v) create new arcs (u_1, v_2) and (u_2, v_1), both of the same length as (u, v).
- Let D' be the digraph constructed this way.
- For each $v \in V$ find the shortest (v_1, v_2)-path in D'.
- The shortest among these paths gives us the shortest odd cycle.

Perfect Matchings

Definitions

- Let $G = (V, E)$ be an undirected graph.
- A matching $M \subseteq E$ is *perfect* if $|M| = |V|/2$.
- Perfect matchings are the integer solutions to:

$$\begin{aligned} \sum_{e \in \delta(v)} x_e &= 1 && \text{for all } v \in V \\ x_e &\geq 0 && \text{for all } e \in E \end{aligned}$$

- The perfect matching polytope is

$$P_{\mathrm{PM}}(G) = \mathrm{conv}\Big\{x \in \{0,1\}^E : \sum_{e \in \delta(v)} x_e = 1 \text{ for all } v \in V\Big\}$$

Odd Cut Inequalities

- The following inequalities are valid for $P_{\mathrm{PM}}(G)$:

$$\sum_{e \in \delta(U)} x_e \geq 1 \text{ for all } U \subset V, |U| \text{ odd}$$

- Each has a cutting-plane proof that requires rounding only once.
- The separation problem for this class of inequalities can be solved in polynomial time.

{0,1/2}-cuts

Definition

- Let
$$\mathcal{F}_{1/2}(A,b) := \left\{ (\lambda^\intercal A)x \leq \lfloor \lambda^\intercal b \rfloor : \lambda \in \{0,1/2\}^m, \lambda^\intercal A \in \mathbb{Z}^n \right\}$$
be the family of all {0,1/2}-cuts.

Question: *Can one separate efficiently over $\mathcal{F}_{1/2}(A,b)$?*

NP-Hardness

Theorem 1. *Let $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$, and $y \in \mathbb{Q}^n$ with $Ay \leq b$. Checking whether y violates some inequality in $\mathcal{F}_{1/2}(A,b)$ is NP-complete.*

Preliminaries

- Let $P = \{x : Ax \leq b\}$ and $y \in P$.

y violates a {0,1/2}-cut iff there exists $\mu \in \{0,1\}^m$ such that

- $\mu^\intercal A \equiv 0 \pmod 2$,
- $\mu^\intercal b \equiv 1 \pmod 2$, and
- $\mu^\intercal (b - A\hat{x}) < 1$.

(Because $\mu^\intercal b = 2k+1$ for some $k \in \mathbb{Z}$, and $\mu^\intercal Ax \leq 2k$ can be written as $\mu^\intercal (b - Ax) \geq 1$.)

An NP-complete Problem

- Given $Q \in \{0,1\}^{r \times t}$, $d \in \{0,1\}^r$, and a positive integer K, decide whether there exists $z \in \{0,1\}^t$ with at most K 1's such that $Qz \equiv d \pmod 2$.

Reduction

- Let $w := \frac{1}{K+1}\mathbf{1}$ and consider $P = \{x \in \mathbb{R}^n : Ax \leq b\}$ with:

$$A := \left(\begin{array}{c|c} Q^\intercal & \\ d^\intercal & 2I_{t+1} \end{array} \right), \; b := \begin{pmatrix} 2 \cdot \mathbf{1}^t \\ 1 \end{pmatrix}, \; y := \begin{pmatrix} \mathbf{0}^r \\ \mathbf{1}^t - \frac{1}{2}w^\intercal \\ \frac{1}{2} \end{pmatrix}$$

Proof Sketch
Step 1: Show (A, b, y) is a valid instance.

- $y \in P$: Observe that $$b - Ay = (w_1, \ldots, w_t, 0)^\intercal \geq 0.$$

Proof sketch
Step 2: Equivalence of "Yes"-instances.

- $\exists \mu \in \{0,1\}^m$ with $\mu^\intercal A \equiv 0 \pmod 2$, $\mu^\intercal b \equiv 1 \pmod 2$ iff $\exists z \in \{0,1\}^t$ such that $Qz \equiv d \pmod 2$:

$$A := \left(\begin{array}{c|c} Q^\intercal & \\ d^\intercal & 2I_{t+1} \end{array} \right), \; b := \begin{pmatrix} 2 \cdot \mathbf{1}^t \\ 1 \end{pmatrix}$$

Proof sketch
Step 2: Equivalence of "Yes"-instances.

- $\exists \mu$ s.th. $\mu^\intercal(b - Ay) < 1$ iff $\exists z$ s.th. $w^\intercal z < 1$ ($\Leftrightarrow \mathbf{1}^\intercal z \leq K$):

$$\mu^\intercal(b - Ay) = \mu^\intercal(w_1, \ldots, w_t, 0)^\intercal$$

□

Primal Separation

The Primal Separation Problem

- Let P be a 0/1-polytope.

- Given a point $y \in \mathbb{Q}^n$ and a vertex $\hat{x} \in P$, find $c \in \mathbb{Z}^n$ and $d \in \mathbb{Z}$ such that $cx \leq d$ for all $x \in P$, $c\hat{x} = d$, and $cy > d$, if they exist.

Theorem 2. *For 0/1-polytopes, optimization and primal separation are polynomial-time equivalent.*

Perfect Matchings

- Let $\hat{x}$ be the incidence vector of a perfect matching M.
- Let $y \in \mathbb{Q}_+^E$ be a point satisfying the node-degree equations.
- We have to find a min-weight odd cut (w.r.t. the edge weights given by y) among those that intersect M in exactly one edge.
- Let $\{s,t\} \in M$ be an arbitrary edge of M.
- Let $G_{\{s,t\}}$ be the graph obtained from G by contracting the end nodes of all edges $e \in M \setminus \{\{s,t\}\}$.
- The minimum weight odd cut among those that contain exactly the edge $\{s,t\}$ of M can be computed by finding a min-weight $\{s,t\}$-cut in $G_{\{s,t\}}$.

Theorem 3. *The primal separation problem for the perfect matching polytope of a graph $G = (V,E)$ can be solved with $|V|/2$ max-flow computations.*

Corollary 4. *A minimum weight perfect matching can be computed in polynomial time.*

Proof Sketch

$$\begin{array}{c}\text{Primal Separation}\\ \Downarrow \\ \text{Verification} \\ \Downarrow \\ \text{Augmentation} \\ \Downarrow \\ \text{Optimization}\end{array}$$

The Verification Problem

- Let $P \subseteq \mathbb{R}^n$ be a 0/1-polytope.
- Given an objective function $c \in \mathbb{Z}^n$ and a vertex $\hat{x} \in P$, decide whether $\hat{x}$ minimizes cx over P.

Primal Separation ⇒ Verification

- Let C be the cone defined by the linear inequalities of P that are tight at $\hat{x}$.
- By LP duality, $\hat{x}$ minimizes cx over P iff $\hat{x}$ minimizes cx over C.
- By the equivalence of optimization and separation, minimizing cx over C is equivalent to solving the separation problem for C.
- One can solve the separation problem for C by solving the primal separation problem for P and $\hat{x}$. □

The Augmentation Problem

- Let $P \subseteq \mathbb{R}^n$ be a 0/1-polytope.
- Given an objective function $c \in \mathbb{Z}^n$ and a vertex $x \in P$, find a vertex $x' \in P$ such that $cx' < cx$, if one exists.

Verification ⇒ Augmentation

- We may assume that $x = \mathbf{1}$.
- Use "Verification" to check whether x is optimal. If not:

$M := \sum_{i=1}^{n} |c_i| + 1$;
for $i = 1$ to n **do**
  $c_i := c_i - M$;
  call the verification oracle with input x and c;
  if x is optimal **then**
    $y_i := 0$;
    $c_i := c_i + M$
  else
    $y_i := 1$
return y.

□

Augmentation ⇒ Optimization

- We may assume that $c \geq 0$.
- Let $C := \max\{c_i : i = 1, \ldots, n\}$, and $K := \lfloor \log C \rfloor$.
- For $k = 0, \ldots, K$, define c^k by $c_i^k := \lfloor c_i / 2^{K-k} \rfloor$, $i = 1, \ldots, n$.

for $k = 0, 1, \ldots, K$ **do**
  while x^k is not optimal for c^k **do**
    $x^k := \text{AUG}(x^k, c^k)$
  $x^{k+1} := x^k$
return x^{K+1}.

Running Time

- $O(\log C)$ many phases.
- At the end of phase $k-1$, x^k is optimal with respect to c^{k-1}, and hence for $2c^{k-1}$.
- Moreover, $c^k = 2c^{k-1} + c(k)$, for some 0/1-vector $c(k)$.
- If x^{k+1} denotes the optimal solution for c^k at the end of phase k, we obtain

$$c^k(x^k - x^{k+1}) = 2c^{k-1}(x^k - x^{k+1}) + c(k)(x^k - x^{k+1}) \leq n.$$

- Thus, the algorithm determines an optimal solution by solving at most $O(n \log C)$ augmentation problems. $\square$