Lecture 24

Plan:
1) Ellipsoid for LP
2) If time, examples.

Ellipsoid for LP

- Even for feasibility of $P = \{x : Ax \leq b\}$, are issues!
- Finding starting ellipse, $E$. 
- Bounding volume of $P$ can be handled in general, but
- To avoid numerical details, study important special case: (important for combo. opt.)

Assume $P = \text{conv}(X)$ for $X \in \mathbb{F}_2^m$, & $\dim P = n$.

E.g.

\[ P = \text{conv} \{ 1 \mathbf{m} : \mathbf{m} \text{ matching in } \mathbb{F}_2^3 \} \subseteq \mathbb{R}^E \]

- Can handle $\dim P < n$ by eliminating variables if $\text{aff}(P)$ known, tricky if not!
Given \( c \in \mathbb{R}^n \), want to compute
\[ \text{OPT} = \max \{ c^T x : x \in P \} \]
in polynomial time given
separation oracle for \( P \).

CSP oracle tells us \( x \in P \) or gives
separating hyperplane \( a^T x \leq b \).

What's polynomial time
here?

- **Input-size:**
  - Assume \( c \in \mathbb{Q}^n \)
  - \( \mathbb{Q}^n \) \( bc \) must store on machine
  - \( \mathbb{Z}^n \) by clearing denominators.
Assume each entry satisfies \( 10^{c_i} \leq M \in \mathbb{N} \);
need \( \lceil \log_{10} M \rceil \) digits to write each \( c_i \)
(i.e. \( \lceil \log_2 M \rceil \) bits)

\( \Rightarrow \) total input size \( \leq n \lceil \log_{2} M \rceil \).

- Thus will take polynomial time to mean \( \text{poly}(n, \ln M) \)
  \#steps / calls to SEP oracle.
Implement the binary search.

- How long do we need to do binary search?
  - Know $OPT \in \mathbb{R}$
  - Know $|OPT| \leq n M$

\[ 10n^3 (\log M)^2 \checkmark \]

\[ 2^n \log M \times \]

\[ nM^2 \times \]
Thus can exactly solve for \( \text{OPT} \) using binary search; need only check \( L = k + \frac{1}{2}, k \in \mathbb{Z}, 1 \leq k < n \).

Each time we check "is \( \text{OPT} \geq L \)?" we check

**Example:** Suppose \( Mn = 4 \), \( \text{OPT} = 2 \), just need to check the blue points.

Know \( L_2 \leq \text{OPT} \leq L_3 \), \( \text{OPT} \in \mathbb{Z} \) \( \Rightarrow \) \( \text{OPT} = 2 \).
If only one query were of the form $y = f(x)$ in step $k$, only one dot...
• How many steps?

\[ \leq \log_2 |2Mn| \]

(\# points to check halved at each step.)

• At each step, need to test if \( L < \text{opt} \), i.e. if \( P_L := \{ x \in P : c^T x \geq L \} \) is nonempty.

• For this, use ellipsoid.
Runtime of ellipsoid calls

- Recall: to test feasibility with ellipsoid, must know
  - starting ellipsoid $E_0 \supseteq P_c$
  - volume lower bound $\text{vol} P_c \geq \delta$
    for all $P_c \neq \emptyset$.

- To test:
  - run ellipsoid for
    
    $2(n+1) \ln \frac{\text{vol} E_0}{\delta}$

    steps. (or until find $x \in P_c$).
\[ \text{if haven't found } \mathbf{x} \in \mathcal{P}_L, \text{ output that } \mathcal{P}_L = \emptyset. \]

- Thus we just need lower bound \( \delta \) on \( \text{vol} \mathcal{P}_L \)
  & upper bound on \( \text{vol} \mathcal{E}_0 \)
  \[ \log(\frac{1}{\delta}), \log \text{vol} \mathcal{E}_0 \leq \text{poly}(n, \log M). \]

**Bounding starting ellipsoid**

- Simple: \( \mathcal{P} \subseteq [0, 1]^n \)
  \[ \Rightarrow \mathcal{P}_L \subseteq [0, 1]^n \]
  \[ \Rightarrow \text{any } \mathcal{E}_0 \supseteq [0, 1]^n \text{ is ok.} \]
• Can use $E_0 = \text{ball centered at } \left(\frac{1}{2}, \ldots, \frac{1}{2}\right) \text{ radius } \frac{1}{2} \sqrt{n}$.

$E_0$ goes through all points of $\frac{1}{2} \sqrt{n}$.

e.g. for $n = 2$,

\[
\text{Vol } E_0 = \left(\frac{\sqrt{n}}{2}\right)^n \cdot \text{vol (unit ball)}
\]

because $\text{unit ball } \subseteq [-1, 1]^n \subseteq (\frac{\sqrt{n}}{2})^n 2^n = n^{\frac{n}{2}}$.

$\Rightarrow \ln \text{ vol } E_0 \leq \frac{n}{2} \ln n$. 
Bounding Vol PL

- Need to show $PL = \mathcal{P} \Rightarrow \text{vol } PL \geq \delta$

  where $\log(\frac{1}{\delta}) = \text{poly}(n,M)$.

- Since $PL \neq \mathcal{P}$, contains some optimal w.r.t. $u_0 \in \mathcal{F}_0,1,\mathcal{U}$ of $\mathcal{P}$

$$c^Tu_0 = \text{OPT}$$

* e.g. $n = 3$
• One way (there are many): Find a simplex in "corner" of \( P_n \).

• Simplex in \( \mathbb{R}^n \) is convex hull of \( n+1 \) affinely independent points.

  e.g.

  • triangle in \( \mathbb{R}^2 \)
  • tetrahedron in \( \mathbb{R}^3 \)

  easy to compute volumes of simplices.
- We've assumed \( P \) is full-dimensional.

\[ \Rightarrow \exists v_1, \ldots, v_n \in \{0, 1\}^n \text{ vertices of } P \text{ s.t. } \text{conv}\{v_0, \ldots, v_n\} \text{ is full-dimensional.} \]

**e.g.** \( n = 3 \)
\( \cdot v_1, \ldots, v_n \) might not be in \( P_L \), but we can “truncating” \( \text{conv}[v_0, \ldots, v_n] \):

\[
\omega_i = \begin{cases} 
v_i & \text{if } c^Tv_i \geq L \\
v_0 + \alpha(v_i - v_0) & \text{else}
\end{cases}
\]

for some small \( \alpha > 0 \).

\[\text{e.g. } n = 3\]

For some \( \alpha \),

\[
C := \text{conv}(v_0, \omega_1, \ldots, \omega_n) \subseteq P_L
\]
• Can take $\alpha = \frac{1}{2 Mn}$, because then

$$c^T w_i = c^T v_o + \alpha c^T (v_i - v_o)$$

$$= \text{OPT} + \alpha c^T (v_i - v_o)$$

$$\geq \text{OPT} - \alpha Mn$$

$$\leq v_i - v_o \in \mathbb{E} \cup \mathbb{P} \cup \mathbb{S},$$

where $L \leq |v_i - v_o| \leq Mn$.

Thus,

$$L \geq (c^T \frac{1}{2}) - \frac{1}{2} \geq L.$$

$$\Rightarrow w_i \in P_L.$$

• Now $\text{vol } P_L \geq \text{vol } C,$
\[ \mathcal{C} := \text{conv}\{v_0, w_1, \ldots, w_n\}. \]

- Simplex \( \mathcal{C} \) = "corner" of parallelepiped \( Q \) with sides \( \alpha(v_1 - v_0), \ldots, \alpha(v_n - v_0) \).
- $\text{vol } C = \frac{1}{n!} \cdot \text{vol } (Q)$. 

**Exercise:** wlog $Q = [0,1]^n$ 

$C = \{x \in Q : \exists x_i \leq 1\}$. 

- $\text{vol } Q = \alpha^n \cdot \text{vol } Q'$,
$Q' := \text{parallelepiped w/ sides}$

$(v_1 - v_0), \ldots, (v_n - v_0)$

- $\det Q' \geq 1$, because sides are lin. indep \& they are in $\mathbb{R}^n$

$\det Q' = \left| \det \begin{pmatrix} v_1 - v_0 & \cdots & v_n - v_0 \end{pmatrix} \right| \\
\geq 1$

- So together:
\[ \forall \theta \ P_L \geq \frac{1}{n!} \alpha^n \cdot 1 = \frac{1}{n!} \left( \frac{1}{2nM} \right)^n \]

\[ \geq \frac{1}{n^n (2nM)^n} = \frac{1}{(2n^2M)^n} \]

Thus we may take

\[ \delta = \frac{1}{(2nM)^n} \]

\[ \log \frac{1}{8} = n \log (2nM) \checkmark \]

**Overall Runtime**
• \# steps of ellipsoid
\[ \leq 2(n+1) \ln \frac{\text{Vol } E_0}{\delta} \]
\[ \leq 2(n+1) \left[ \ln \left( n^{\frac{M}{2}} \right) + \ln \left( 2n^2 M \right) \right] \]
\[ = 2(n+1) \left[ \frac{M}{2} \ln(n) + n \ln(2n^2 M) \right] \]
\[ = O(n^2 \left( \ln n + \ln M \right)) \].

• \# steps of binary search
\[ \leq \log_2 (2nM) \]
\[ = O(\ln(n) + \log(M)) \]

• Overall:
$O(n^2(\ln n + \ln M)^2)$

= \text{poly}(n, \ln M) \text{ SEP calls.}

To summarize...

\textbf{Theorem}: Given a separation oracle for $P = \text{conw}(X), X \subseteq \{0,1\}^n$, s.t. $\dim P = n$, can max $cx$ over $P$ (& hence $X$) in polynomial time

$\text{in } O(n^2(\ln n + \ln M)^2) \text{ calls}$
to SEP oracle. 

- **Side Remark:** is not strongly polynomial; # iterations depends on c (albeit polynomially).

  we could have covered P = \{x : Ax \leq b\} ellipsoid can opt in poly time.

- **Éva Tardos '86:** can solve LP's \( \max \{c^T x : Ax \leq b\} \) in time \( \text{poly}(\text{input size of } A) \) arithmetic \((+,-,/)\) cost 1 unit.

  (or just poly calls to SEPoracle).

  i.e. indep of c, b!

  still uses ellipsoid.
Thus if \( A \in \{-1, 0, 1\}^{m \times n} \),
can solve LP in strongly polynomial time.
but not known for general \( A \)!

**Example: Matroid Intersection**

- Given \( M_1 = (E_1, I_1) \), \( M_2 = (E_1, I_2) \),
cost \( c: E \to \mathbb{R} \). How to find costliest common independent set?

  \[
  \text{i.e. max } c(S) = \sum_{e \in S} c(e) \quad \forall S \subseteq I_1 \cap I_2
  \]

- Equivalently, maximize \( c^T x \) over
  matroid intersection polytope

  \[
  P_{M_1 \cap M_2} = \text{conv} \{ 1_S : S \subseteq \emptyset \}\]

\}
But how to get a separation oracle?? Exponential # constraints!

- Recall: $P_{m_1 \cap m_2} = P_{m_1} \cap P_{m_2}$ matroid polytope
- SEP oracle for $P_{m_1}, P_{m_2}$ $\Rightarrow$ SEP oracle for $P_{m_1 \cap m_2}$ (check both $P_{m_1}, P_{m_2}$.)

- But we only have efficient OPT algorithms for $P_{m_1}, P_{m_2}$, not SEP!

- From GLS '81: efficient OPT algo. $\Rightarrow$ efficient SEP algorithm!
• Thus \exists \text{ efficient SEP algos. for } P_{M_1}, P_{M_2}, \Rightarrow

\exists \text{ efficient SEP } P_{M_1}, P_{M_2}, \\
\Rightarrow \text{ ellipsoid can optimize in polytime.}

\textbf{Example: nonbip. matching}

• Given \( G, \) cost \( c: E \rightarrow \mathbb{R}, \)
find max cost matching \( M. \)

• Equivalently, optimize \( \max x \)
over matching polytope

\[ P = \text{conv}\{1_M : M \text{ matching in } G^3\}. \]

• \textbf{Recall:} Matching polytope given by
\[ P = \{ x \in \mathbb{R}^E : \sum_{e \in E(N)} x_e \leq 1 \quad \forall v \in V. \}
\]

- P is full-dimensional
  - Exercise.
- However, separation oracle nontrivial! (Exp. constraints again!)
- Can implement using min-T-odd cut algorithm (Padberg-Rao).
Matching polytope SEP oracle:

- Check degree constraints; if violated, return corresponding inequality.

- Next check odd set constraints.

How?

- For $X$ satisfying degree constraints, need to decide if
  $$\exists x_e \leq |S| - 1$$
  $$e \in E(S)$$

As $S \subseteq V$, $|S|$ odd, & if not produce violated $S$. .
Assume WLOG \(|V|\) even (else add isolated vertex).

For \(v \in V\), define

\[
S_v = 1 - \sum_{e \in \delta(v)} x_e
\]

Observe: Given \(S \subseteq V\),

\[
\sum_{v \in S} x_v + \sum_{e \in E(S)} x_e = |S| - 2 \sum_{e \in E(S)} x_e - 2 \sum_{e \in E(S)} x_e
\]

Thus,

\[
|S| - 2 \sum_{e \in E(S)} x_e - 2 \sum_{e \in E(S)} x_e = \sum_{v \in S} x_v + \sum_{e \in E(S)} x_e
\]

In a bipartite graph, \(x_e = 0\) for all \(e \in E(S)\) double counted.
Thus odd set constr. holds for $S$

\[ \sum_{v \in S} x_e \geq 1 \]

Define new graph $H$ with vertex set $V + \text{new vert. } \omega$
edge set $E + \text{all edges } \omega \leftrightarrow V$

Define edge weights

\[ u_e = \begin{cases} 
  x_e & \text{if } e \in E \\
  s_v & \text{if } e = (v, \omega). 
\end{cases} \]
For a cut \( S \) in \( H \), may assume \( w \notin S \) by taking complements.

A cut \( S \subseteq V \) in \( H \) has value

\[
\sum_{u \in u \in S} x_e + \sum_{v \in S'} x_e \geq 1
\]

A odd \( S \subseteq V \) \( \iff \)

\[
\text{min } V-\text{odd cut in } H \text{ has value } \geq 1.
\]

Recall: min T-odd cut is to find min cut \( S \) subject to \( |S \cap T| \) odd.
\[\text{We have seen how to compute min T-odd cut; do so for } T = \mathcal{V} \text{ in } H.\]

\[\text{if } \exists \text{ v-odd cut } S \text{ for } H\]

\[\text{w/ value } < 1, \]

\[\text{S is violated; return } S\]

\[\text{if not, } x \in P. \quad \square\]