

Improved Deterministic Algorithms for Linear Programming in Low Dimensions

Timothy Chan

School of CS

U of Waterloo

The Problem: LP

maximize $c_1x_1 + \cdots + c_dx_d$

subject to

$$a_{11}x_1 + \cdots + a_{1d}x_d \leq b_1$$

⋮

$$a_{n1}x_1 + \cdots + a_{nd}x_d \leq b_n$$

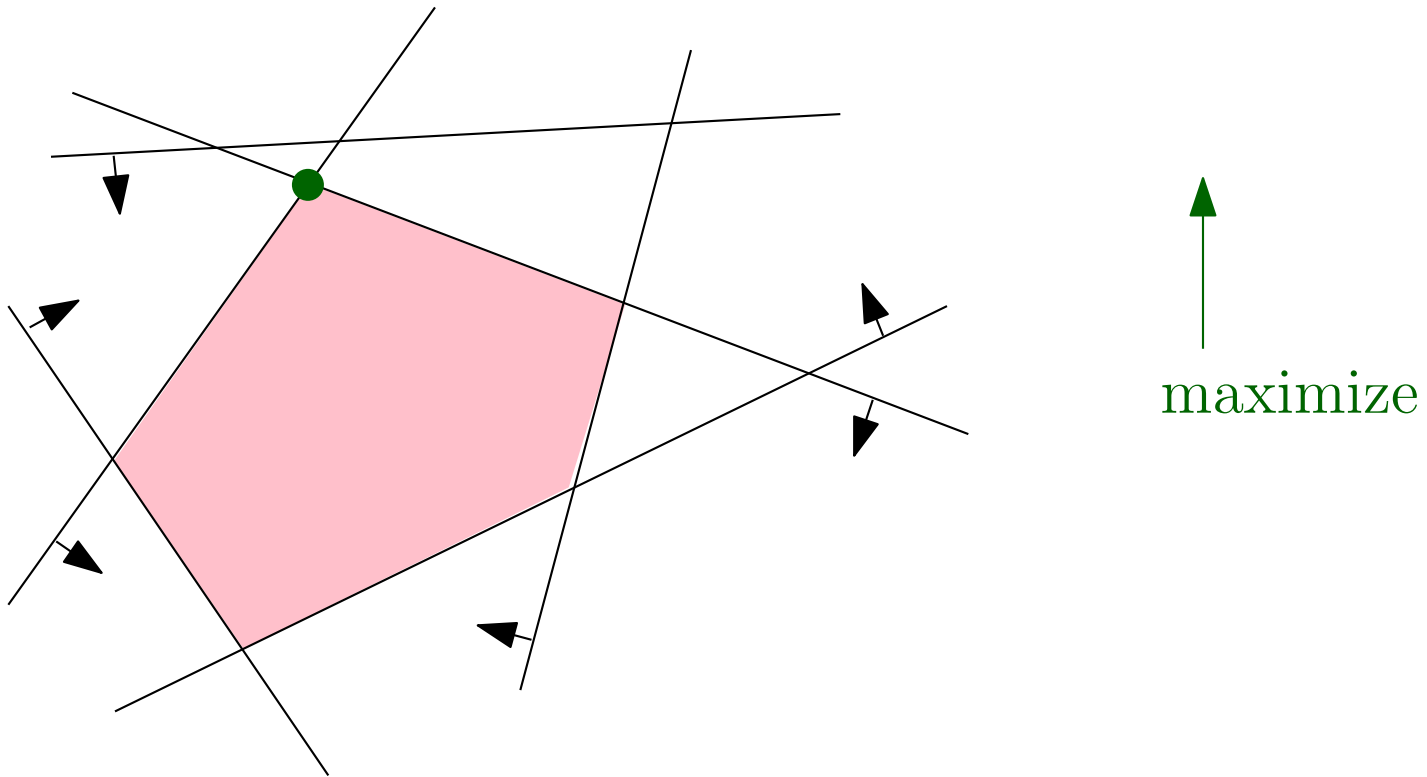
over real variables x_1, \dots, x_d

The Problem: LP

- interested in runtime of alg'ms as a function of n & d (not bit complexity)
- **Big Open Q:** \exists strongly polynomial alg'm??
- **Ex: simplex method** is exponential, with upper bound about $\binom{n}{\lfloor d/2 \rfloor} = O(n/d)^{d/2}$
whereas **ellipsoid method & interior-point methods** (from Karmarkar'84 to ... Lee–Sidford'15) aren't applicable

The Problem: LP

Our Focus: when d is small



History: Prune&Search Alg'ms

- $d = 2$ or 3 : Dyer'82 / Megiddo [FOCS'82]: $O(n)$ time

$$T_2(n) = T_2(3n/4) + O(n)$$

$$T_3(n) = T_3(15n/16) + O(n)$$

- any constant d : Megiddo'84: $O(n)$ time

$$T_d(n) = T_d((1 - 1/2^{2^{d-1}})n) + O(2^d T_{d-1}(n))$$

$$\Rightarrow 2^{O(2^d)} n$$

History: Prune&Search Alg'ms

- Clarkson'86 / Dyer'86: $O(3^{d^2})_n$ time
- Dyer–Frieze'89: $\tilde{O}(d)^{3d}_n$ time (but randomized)
- Agarwal–Sharir–Toledo'93: $\tilde{O}(d)^{10d}_n$ (deterministic)

History: Rand. Sampling Alg'ms

- Clarkson [FOCS'88]: (**recursive** version)

$$T_d(n) \approx (d + 1) \cdot (T_d(d\sqrt{n}) + O(dn))$$

(with base case $T_d(d^2) = O(d)^{d/2}$ by simplex method)

$$\Rightarrow \boxed{O(d^2n) + O(d)^{d/2}} \text{ (rand.)}$$

- Clarkson [later]: (**iterative reweighting** version)

$$T_d(n) \approx d \log n \cdot O(dn + T_d(d^2))$$

\Rightarrow roughly the same

both simple!

History: Rand. Incremental Alg'ms

- Seidel [SoCG'90]:

$$T_d(n) = T_d(n-1) + O((d/n) \cdot T_{d-1}(n))$$

$$\Rightarrow \boxed{O(d!n)} \text{ (rand.) } \text{very simple!}$$

- Kalai [STOC'92] / Matoušek–Sharir–Welzl [SoCG'92]:

$$\boxed{2^{O(\sqrt{d \log n})} n} \text{ (rand.)}$$

- combined with rand. sampling alg'ms

$$\Rightarrow \boxed{O(d^2 n) + 2^{O(\sqrt{d \log d})}} \text{ (rand.) } \text{current record}$$

(Hansen–Zwick [STOC'15]: $2^{O(\sqrt{d})}$ rand. for $n = O(d)$)

History: Back to Deterministic Alg'ms

- Chazelle–Matoušek [SODA'93]: derandomize Clarkson's recursive rand. sampling alg'm by designing an ε -net alg'm (via “ ε -approximations”, method of conditional probabilities, & a clever merge&reduce technique)

$$\Rightarrow \boxed{\tilde{O}(d)^{7d_n}} \text{ (det.)}$$

- Brönnimann–Chazelle–Matoušek [FOCS'93]: throw in “sensitive ε -approximations”

$$\Rightarrow \boxed{\tilde{O}(d)^{5d_n}} \text{ (det.)} \quad \text{current record... till now}$$

Today: New Deterministic Alg'ms

- much **simpler** derandomization of Clarkson's recursive rand. sampling alg'm (without ε -approximations, method of conditional probabilities, merge&reduce, ...)
 $\Rightarrow \boxed{\tilde{O}(d)^{3d_n}}$ (det.) \leftarrow
- combined with a new variant of Clarkson's iterative reweighting alg'm $\Rightarrow \boxed{\tilde{O}(d)^{2d_n}}$ (det.)
- throw in combinatorial bounds on $(\leq k)$ -levels
 $\Rightarrow \boxed{\tilde{O}(d)^{d_n}}$ (det.)
- new ε -net alg'm (by throwing back in sensitive ε -approximation, method of conditional probabilities, ... & a new merge&reduce)
 $\Rightarrow \boxed{\tilde{O}(d)^{d/2_n}}$ (det.) **(new current record)**

Review of Clarkson's Random Sampling Alg'm (Recursive Version)

LP(H), given set H of n halfspaces in \mathbb{R}^d :

1. choose a subset R of H
2. repeat:
3. recursively compute $p = \mathbf{LP}(R)$
4. add {all halfspaces of H violated by p } to R

Claim: # repeats $\leq d + 1$

Proof: let B^* = the d halfspaces defining optimal sol'n p^*
each iteration adds ≥ 1 halfspace of B^* to R
(if not, p inside $\cap B^* \Rightarrow p$ worse than p^* : contradiction!) \square

Review of Clarkson's Random Sampling Alg'm (Recursive Version)

$LP(H)$, given set H of n halfspaces in \mathbb{R}^d :

1. choose a subset R of H ← how?
2. repeat $d + 1$ times:
3. recursively compute $p = LP(R)$
4. add {all halfspaces of H violated by p } to R

ε -Nets

Def: $R \subset H$ is an ε -net iff $\forall p \in \mathbb{R}^d$,

p violates $> \varepsilon n$ of $H \Rightarrow p$ violates ≥ 1 of R

Fact: $\exists \varepsilon$ -net R of size $\tilde{O}(d/\varepsilon)$

Proof:

- call {all halfspaces violated by p } a “violation set”
- want R to hit all violation sets of size $> \varepsilon n$
- # diff. violation sets = $m = O\left(\binom{n}{d}\right) = O(n/d)^d$
- just take **random sample** of size $O\left((1/\varepsilon) \log m\right)$ \square

Alternate Proof: run **greedy hitting set alg'm** \square

Review of Clarkson's Random Sampling Alg'm (Recursive Version)

LP(H), given set H of n halfspaces in \mathbb{R}^d :

1. choose ε -net R of size $\tilde{O}(d/\varepsilon)$ by sampling
2. repeat $d + 1$ times:
3. recursively compute $p = \mathbf{LP}(R)$
4. add {all halfspaces of H violated by p } to R

By Def: p violates none of $R \Rightarrow p$ violates $\leq \varepsilon n$ of H

$$\Rightarrow T_d(n) \approx (d + 1) \cdot \left(T_d\left(\frac{d}{\varepsilon} + \frac{d\varepsilon n}{d\sqrt{n}}\right) + O(dn) \right)$$

Chazelle–Matoušek's Derandomization

- gave complicated alg'm to compute ε -net R of size $\tilde{O}(d/\varepsilon)$ in $\tilde{O}(d^3/\varepsilon^2)^{d_n}$ time (det.)
- set $\varepsilon \approx 1/(Cd^2)$
 - $\Rightarrow T_d(n) \approx (d+1) \cdot (T_d(\frac{d}{\varepsilon} + \frac{d\varepsilon n}{n/(Cd)}) + \tilde{O}(d)^{7d_n})$
 - $\Rightarrow \boxed{\tilde{O}(d)^{7d_n}}$ (det.)
- **New Obs:** can afford ε -net of much larger size...

New Simple Derandomization

$\text{LP}(H)$, given set H of n halfspaces in \mathbb{R}^d :

1. divide H into groups of size b ;
compute ε -net of each group by greedy hitting set alg'm;
 $R =$ union of these ε -nets
2. repeat $d + 1$ times:
3. recursively compute $p = \text{LP}(R)$
4. add {all halfspaces of H violated by p } to R

New Simple Derandomization

$\text{LP}(H)$, given set H of n halfspaces in \mathbb{R}^d :

1. divide H into groups of size b ;
compute ε -net of each group by greedy hitting set alg'm;
 $R =$ union of these ε -nets
2. repeat $d + 1$ times:
3. recursively compute $p = \text{LP}(R)$
4. add {all halfspaces of H violated by p } to R

set $\varepsilon = 1/(Cd^2)$, $b = \tilde{\Theta}(d^4)$

size of ε -net = $s = (n/b) \cdot \tilde{O}(d/\varepsilon) \leq n/(C'd)$

time to compute ε -net = $(n/b) \cdot O(b/d)^d = \tilde{O}(d)^{3d}n$

$\Rightarrow T_d(n) \approx (d + 1) \cdot (T_d(\frac{s + dcn}{n/(C''d)}) + \tilde{O}(d)^{3d}n)$

$\Rightarrow \boxed{\tilde{O}(d)^{3d}n}$ (det.)

Conclusions

- simpler, even compared to Megiddo's det. alg'm
- throw in a few more ideas $\Rightarrow \boxed{\tilde{O}(d)^{d/2}n}$ (det.)
- one barrier: for the base case $n \approx d^2$, can we beat $O\left(\binom{n}{\lfloor d/2 \rfloor}\right) = O(d)^{d/2}$ det. time?
- generalize to many **LP-type** problems (with $\tilde{O}(d)^{d}n$ det. time)
- $2^{O(d)}n$ det. alg'm??