Randomized MWU for Positive LPs

Chandra Chekuri

Kent Quanrud

Positive LPs

find x s.t.

$$Ax \le b, \quad Cx \ge d, \quad x \ge 0$$

where $A \in \mathbb{R}^{m \times n}_{\ge 0}$, $b \in \mathbb{R}^{m}_{\ge 0}$, $C \in \mathbb{R}^{m \times n}_{\ge 0}$, $d \in \mathbb{R}^{m}_{\ge 0}$

have non-negative coefficients

Packing LP	Covering LP
$\max \langle d, x\rangle$ s.t. $Ax \le b$, $x \ge 0$	$\min \langle b, x\rangle$ s.t. $Cx \ge d$, $x \ge 0$

Packing/Covering $\Rightarrow$ Positive LP

find $x \ge 0$ st $\langle d, x\rangle \ge OPT$, $Ax \le b$	find $x \ge 0$ st $\langle b, x\rangle \le OPT$, $Cx \ge d$

(A, b, c, d nonnegative)

Explicit vs Implicit

<u>Explicit</u>: coefficients given by list of N nonzeroes

<u>Implicit</u>: coefficients inferred from context

- explicit form may be much larger than input
- eg. packing spanning trees, cuts in graphs

$(1+\varepsilon)$-Relative Approximations

for fixed $\varepsilon > 0$, output

- **Packing:** $\langle d, x \rangle \geq (1-\varepsilon)\mathrm{OPT}$, $Ax \leq b$
- **Covering:** $\langle b, x \rangle \leq (1+\varepsilon)\mathrm{OPT}$, $Cx \geq d$
- **Positive LPs:** $Ax \leq (1+\varepsilon)b$, $Cx \geq (1-\varepsilon)d$,

  or certificate showing $\mathcal{LP}$ infeasible

Overview of Results

① faster $(1+\varepsilon)$-APX for <u>explicit</u> positive LPs

② fast + general MWU framework for <u>implicit</u> positive LPs

[① ⇐ ② + fast implementation of "greedy oracle" for explicit setting]

State of the art: $(1+\varepsilon)$-APX, explicit

<table>
<tr><th>Packing</th><th>Covering</th><th>Positive LPs</th></tr>
<tr><td colspan="2">$O(N) + \tilde{O}\left(\frac{m+n}{\varepsilon^2}\right)$
randomized
[Koufogiannakis & Young, 2007]</td><td rowspan="2">$\tilde{O}\left(\frac{N}{\varepsilon^2}\right)$
deterministic
[Young 2014]</td></tr>
<tr><td colspan="2">$\tilde{O}\left(\frac{N}{\varepsilon}\right)$ randomized
[Allen-Zhu, Orecchia 2015] | [Wang, Rao, Mahoney 2016]</td></tr>
</table>

Q: Can general positive LPs be approximated as fast as pure packing or covering; i.e, $\tilde{O}(N + \frac{m+n}{\varepsilon^2})$ or $\tilde{O}(N/\varepsilon)$?

(Best previous bound $\tilde{O}(N/\varepsilon^2)$ time)

Q: Can general positive LPs be approximated as fast as pure packing or covering;

i.e, $\tilde{O}(N + \frac{m+n}{\varepsilon^2})$ or $\tilde{O}(N/\varepsilon)$?

<u>Almost</u>: [this paper]

$\hat{O}\left(\frac{N}{\varepsilon} + \frac{m}{\varepsilon^2} + \frac{n}{\varepsilon^3}\right)$ randomized

Now?

(a) Randomized variant of MWU framework

($\downarrow$ reduces problem to)

(b) Randomized data structure approximating matrices coordinatewise

Now?

goal of this talk

(a) Randomized variant of MWU framework

($\downarrow$ reduces problem to)

(b) Randomized data structure approximating matrices coordinatewise

not enough time

Basic idea: simpler problems via Lagr. relax.

$$Ax \le b \quad\Longrightarrow\quad \langle v, Ax\rangle \le \langle v, b\rangle \quad\Longrightarrow\quad \max_i \frac{\langle w, Ce_i\rangle}{\langle v, Ae_i\rangle}$$
$$Cx \ge d \qquad\qquad\quad \langle w, Cx\rangle \ge \langle w, d\rangle$$

many constraints — 1 packing, 1 covering — greedy solution

- nonnegative weights $v, w \in \mathbb{R}^m_{\ge 0}$ collapse original problem into a trivial one
- each v_i or w_i controls "importance" of corresponding constraint in relaxation

multiplicative weights

find x s.t.
$Ax \le b$
$Cx \ge d$

$(\eta \approx \tilde{O}(\frac{1}{\varepsilon}))$

① start with $x = 0$

② repeatedly

(a) choose $v_i = \exp\left(\eta \frac{(Ax)_i}{b_i}\right)$

$$w_i = \begin{cases} \exp\left(-\eta \frac{(Cx)_i}{d_i}\right) & \text{if } (Cx)_i \le d_i \\ 0 & \text{otherwise} \end{cases}$$

(c) find y s.t. $\langle v, Ay \rangle \le \langle v, b \rangle$, $\langle w, By \rangle \ge \langle w, B \rangle$

(d) $x \leftarrow x + \delta y$ for small $\delta > 0$

$\boxed{\text{(d) } x \leftarrow x + \delta y \text{ for small } \delta > 0}$

how to choose δ? greedily

(as large as possible such that no v_i, w_i grows more than $(1+\varepsilon)$-mult. factor)

"width independent"

MWU in 3 pieces

① increase x along sol'n of relaxations

② relax w/ weights exponential in "load" of each constraint

③ maximize step size s.t. keeping each weight within $(1+\varepsilon)$-mult. factor

Invariants

(a) either finds infeasible relaxation, or outputs x s.t. $Ax \le (1+\varepsilon)b$, $Cx \ge (1-\varepsilon)d$

(b) packing weights v_i go from $1 \uparrow m^{O(1/\varepsilon)}$

cov. weights w_i go from $1 \downarrow m^{-O(1/\varepsilon)}$

(c) total of $\tilde{O}\left(\frac{m}{\varepsilon^2}\right)$ iterations

((b), greedy $\delta \Rightarrow$ (c) since some constraint changes by $(1+\varepsilon)$-mult factor each iter.)

Running time

each iteration we: "greedy oracle"

(a) solve relaxation

(b) update each weight v_i, w_i

m weights $\times$ $\tilde{O}\left(\frac{m}{\varepsilon^2}\right)$ iterations

$\Rightarrow$ $\tilde{O}\left(\frac{m^2}{\varepsilon^2}\right)$ running time

Reducing time spent on weight updates

[Young '01] $\tilde{O}\left(\frac{m^2}{\varepsilon^2}\right)$

[Koufogiannakis & Young '07] $\tilde{O}\left(N+\frac{m+n}{\varepsilon^2}\right)$ ← only for pure packing or covering

randomized

[Young '14] $\tilde{O}\left(\frac{N}{\varepsilon^2}\right)$

via amortized data structures

updating weights

$\tilde{O}\left(N + \frac{m+n}{\varepsilon^2}\right)$ vs $\tilde{O}\left(\frac{N}{\varepsilon^2}\right)$ time

packing or covering LPs

general positive LPs

Q: can the weight updates for positive LPs be made as fast as for pure packing / covering?

updating weights

$O(N) + \hat{\tilde{O}}\left(\frac{m+n}{\varepsilon^2}\right)$ vs $\hat{\tilde{O}}\left(\frac{N}{\varepsilon^2}\right)$

packing or covering LPs

general positive LPs

Yes

[this paper]

$\hat{O}\left(\frac{m}{\varepsilon^2}\right)$ time on updates after $O(N)$ preprocessing

Full theorem

$\tilde{O}(N + \frac{m}{\varepsilon^2})$ randomized time $+ \hat{O}(\frac{m}{\varepsilon^2})$ oracle calls, "randomized MWU" computes $(1+\varepsilon)$-APX w/h/p.

Moreover:

packing weights increase from 1 to $m^{O(1/\varepsilon)}$ and covering weights decrease from 1 to $m^{-O(1/\varepsilon)}$ along integer powers of $(1+\varepsilon)$.

Randomized MWU

find $x \geq 0$ st
$Ax \leq b$
$Cx \geq d$

1. start w/ $x=0$, $v_i = 1 \ \forall i$, $w_i = 1 \ \forall i$

2. repeatedly
   (a) find y s.t. $\langle v, Ay \rangle \leq \langle v, b \rangle$, $\langle w, By \rangle \geq \langle w, d \rangle$

   (b) $x \leftarrow x + \delta y$ for small $\delta > 0$

   (c) for each packing weight i,

   w/ probability $\frac{\delta \eta (Ay)_i}{\varepsilon b_i}$,

   $v_i \leftarrow (1+\varepsilon) v_i$

   * correct update in expectation

   * only touch v_i when it increases by $(1+\varepsilon)$-factor

   (d) (similarly for covering weights)

(c) for each packing weight i,

w/ probability $\frac{\delta\eta(A_y)_i}{\varepsilon b_i}$, * correct update in expectation

$v_i \leftarrow (1+\varepsilon) v_i$ *only touch v_i when it increases by $(1+\varepsilon)$-factor

---

Some comments:

- simple algorithm, longer analysis
- since greedy oracle seeks out mistakes, unclear we can randomize updates
- requires "online" version of Chernoff

[Koufogiannakis-Young, '07]

where we stand

(a) randomized weight updates lifts bottleneck from updating weights

(b) simplicity => good for <u>implicit</u> problems

(c) for <u>explicit</u> positive LPs, need to solve relaxations very fast subproblem:

$$\max_j \frac{\langle w, C e_j \rangle}{\langle v, A e_j \rangle}$$

where v monotonically increasing
w monotonically decreasing

- 2nd part of paper maintains uniform APX for $A^T v$ and $C^T w$ efficiently
- uses sampling techniques as well

Conclusion

- general MWU framework removes bottleneck from weight updates

- $(1+\varepsilon)$-APX for explicit positive LPs in $\tilde{O}\left(\frac{N}{\varepsilon} + \frac{m}{\varepsilon^2} + \frac{n}{\varepsilon^3}\right)$ (from $\tilde{O}(N/\varepsilon^2)$)

- Open question:

$$\tilde{O}\left(\frac{N}{\varepsilon} + \frac{m+n}{\varepsilon^2}\right)?$$

Thanks