A primal-dual smooth perceptron-von Neumann algorithm

Javier Peña
Carnegie Mellon University
(joint work with Negar Soheili)

Shubfest, Fields Institute
May 2012

Polyhedral feasibility problems

Given $A := \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} \in \mathbb{R}^{m\times n}$, consider the alternative feasibility problems

$$A^{\mathsf{T}} y > 0, \tag{D}$$

and

$$Ax = 0,\ x \geq 0,\ x \neq 0. \tag{P}$$

Theme

Condition-based analysis of *elementary* algorithms for solving (P) and (D).

Perceptron Algorithm

Algorithm to solve

$$A^{\mathsf{T}} y > 0. \tag{D}$$

Perceptron Algorithm (Rosenblatt, 1958)

- $y := 0$
- while $A^{\mathsf{T}} y \not> 0$

  $y := y + \frac{a_j}{\|a_j\|}$, where $a_j^{\mathsf{T}} y \leq 0$

  end while

Throughout this talk: $\|\cdot\| = \|\cdot\|_2$.

Von Neumann's Algorithm

Algorithm to solve

$$Ax = 0, \; x \geq 0, \; x \neq 0. \tag{P}$$

Von Neumann's Algorithm (von Neumann, 1948)

- $x_0 := \frac{1}{n}\mathbf{1}$; $y_0 := Ax_0$
- for $k = 0, 1, \ldots$

  if $a_j^\mathsf{T} y_k := \min_i a_i^\mathsf{T} y_k > 0$ then halt: (P) is infeasible

  $\lambda_k := \operatorname{argmin}_{\lambda \in [0,1]} \|(1-\lambda)y_k - \lambda a_j\| = \frac{1 - a_j^\mathsf{T} y_k}{\|y_k\|^2 - 2a_j^\mathsf{T} y_k + 1}$

  $x_{k+1} := \lambda_k x_k + (1-\lambda_k)e_j$, where $j = \operatorname{argmin}_i a_i^\mathsf{T} y_k$

  end for

Elementary algorithms

- The perceptron and von Neumann's algorithms are "elementary" algorithms.
- "Elementary" means that each iteration involves only simple computations.

Why should we care about elementary algorithms?

- Some large-scale optimization problems (e.g., in compressive sensing) are not solvable via conventional Newton-based algorithms.
- In some cases, the entire matrix A may not be explicitly available at once.
- Elementary algorithms have been effective in these cases.

Conditioning

Throughout the sequel assume

$A = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}$, where $\|a_j\| = 1,\ j = 1, \ldots, n$.

Key parameter

$$\rho(A) := \max_{\|y\|=1} \min_{j=1,\ldots,n} a_j^{\mathsf{T}} y.$$

Goffin-Cheung-Cucker condition number

$$\mathscr{C}(A) := \frac{1}{|\rho(A)|}.$$

(This is closely related to Renegar's condition number.)

Conditioning

Notice

- $A^\mathsf{T} y > 0$ feasible $\Leftrightarrow \rho(A) > 0$.
- $Ax = 0,\ x \geq 0,\ x \neq 0$ feasible $\Leftrightarrow \rho(A) \leq 0$.

Ill-posedness

A is **ill-posed** when $\rho(A) = 0$. In this case both $A^\mathsf{T} y > 0$ and $Ax = 0, x > 0$ are on the verge of feasibility.

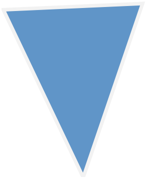
Theorem (Cheung & Cucker, 2001)

$$|\rho(A)| = \min_{\tilde{A}} \{\max_i \|\tilde{a}_i - a_i\| : \tilde{A} \textit{ is ill-posed}\}.$$

Some geometry

When $\rho(A) > 0$, it is a measure of thickness of the feasible cone:

$$\rho(A) = \max_{\|y\|=1} \left\{ r : \mathbb{B}(y, r) \subseteq \{z : A^{\mathsf{T}} z \geq 0\} \right\}.$$



large $\rho(A)$

small $\rho(A)$

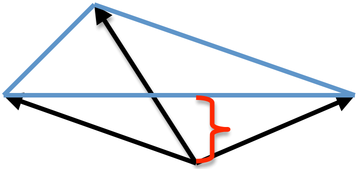
More geometry

Let

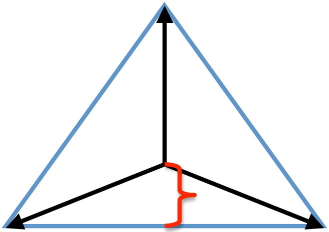
$$\Delta_n := \{x \geq 0 : \|x\|_1 = 1\}.$$

Proposition (From Renegar 1995 and Cheung-Cucker 2001)

$$|\rho(A)| = \operatorname{dist}\left(0, \partial\{Ax : x \geq 0,\ x \in \Delta_n\}\right).$$



$\rho(A) > 0$



$\rho(A) < 0$

Condition-based complexity

Recall our problems of interest

$$A^{\mathsf{T}} y > 0, \tag{D}$$

and

$$Ax = 0, \; x \in \Delta_n. \tag{P}$$

Theorem (Block-Novikoff 1962)

If $\rho(A) > 0$, then the perceptron algorithm terminates after at most

$$\frac{1}{\rho(A)^2} = \mathscr{C}(A)^2$$

iterations.

Condition-based complexity

Theorem (Dantzig, 1992)

If $\rho(A) < 0$, then von Neumann's algorithm finds an ϵ-solution to (P), *i.e, $x \in \Delta_n$ with $\|Ax\| < \epsilon$ in at most*

$$\frac{1}{\epsilon^2}$$

iterations.

Theorem (Epelman & Freund, 2000)

If $\rho(A) < 0$, then von Neumann's algorithm finds an ϵ-solution to (P) *in at most*

$$\frac{1}{\rho(A)^2} \cdot \log\left(\frac{1}{\epsilon}\right)$$

iterations.

Main Theorem

Theorem (Soheili & P, 2012)

A <u>smooth</u> version of perceptron/von Neumann's algorithm such that:

(a) *If $\rho(A) > 0$, then it finds a solution to $A^T y > 0$ in at most*

$$\mathcal{O}\left(\frac{\sqrt{n}}{\rho(A)} \cdot \log\left(\frac{1}{\rho(A)}\right)\right)$$

iterations.

(b) *If $\rho(A) < 0$, then it finds an ϵ-solution to $Ax = 0,\ x \in \Delta_n$ in at most*

$$\mathcal{O}\left(\frac{\sqrt{n}}{|\rho(A)|} \cdot \log\left(\frac{1}{\epsilon}\right)\right)$$

iterations.

(c) *Iterations are elementary (not much more complicated than those of the perceptron or von Neumann's algorithms).*

Perceptron algorithm again

Perceptron Algorithm

- $y_0 := 0$
- for $k = 0, 1, \ldots$
  $a_j^{\mathsf{T}} y_k := \min_i a_i^{\mathsf{T}} y_k$
  $y_{k+1} := y_k + a_j$

  end for

Observe

$$a_j^{\mathsf{T}} y := \min_i a_i^{\mathsf{T}} y \Leftrightarrow a_j = Ax(y), \; x(y) = \operatorname*{argmin}_{x \in \Delta_n} \langle A^{\mathsf{T}} y, x \rangle.$$

Hence in the above algorithm $y_k = Ax_k$ where $x_k \geq 0, \; \|x_k\|_1 = k$.

Normalized Perceptron Algorithm

Recall $x(y) := \underset{x \in \Delta_n}{\operatorname{argmin}} \langle A^{\mathsf{T}} y, x \rangle$.

Normalized Perceptron Algorithm

- $y_0 := 0$
- for $k = 0, 1, \ldots$
  $\theta_k := \frac{1}{k+1}$
  $y_{k+1} := (1 - \theta_k) y_k + \theta_k A x(y_k)$

  end for

In this algorithm $y_k = A x_k$ for $x_k \in \Delta_n$.

Perceptron-Von Neumann's Template

Both the perceptron and von Neumann's algorithms perform similar iterations.

PVN Template

- $x_0 \in \Delta_n; \; y_0 := Ax_0$
- for $k = 0, 1, \ldots$

  $x_{k+1} := (1 - \theta_k) x_k + \theta_k x(y_k)$

  $y_{k+1} := (1 - \theta_k) y_k + \theta_k Ax(y_k)$

  end for

Observe

- Recover (normalized) perceptron if $\theta_k = \frac{1}{k+1}$
- Recover von Neumann's if $\theta_k = \underset{\lambda \in [0,1]}{\operatorname{argmin}} \|(1-\lambda) y_k - \lambda Ax(y_k)\|.$

Smooth Perceptron-Von Neumann Algorithm

Apply Nesterov's smoothing technique (Nesterov, 2005).

Key step: Use a smooth version of

$$x(y) = \operatorname*{argmin}_{x \in \Delta_n} \langle A^\mathsf{T} y, x \rangle,$$

namely,

$$x_\mu(y) := \operatorname*{argmin}_{x \in \Delta_n} \left\{ \langle A^\mathsf{T} y, x \rangle + \frac{\mu}{2} \|x - \bar{x}\|^2 \right\},$$

for some $\mu > 0$ and $\bar{x} \in \Delta_n$.

Smooth Perceptron-Von Neumann Algorithm

Assume $\bar{x} \in \Delta_n$ and $\delta > 0$ are given inputs.

Algorithm SPVN$(\bar{x}, \delta)$

- $y_0 := A\bar{x}$; $\mu_0 := n$; $x_0 := x_{\mu_0}(y_0)$
- for $k = 0, 1, \ldots$

  $\theta_k := \frac{2}{k+3}$

  $y_{k+1} := (1-\theta_k)(y_k + \theta_k A x_k) + \theta_k^2 A x_{\mu_k}(y_k)$
  $\mu_{k+1} := (1-\theta_k)\mu_k$
  $x_{k+1} := (1-\theta_k)x_k + \theta_k x_{\mu_{k+1}}(y_{k+1})$

  if $A^\mathsf{T} y_{k+1} > 0$ then halt: y_{k+1} is a solution to (D)
  if $\|Ax_{k+1}\| \leq \delta$ then halt: x_{k+1} is δ-solution to (P)

  end for

PVN update versus SPVN update

Update in PVN template

$$y_{k+1} := (1-\theta_k)y_k + \theta_k A x(y_k)$$
$$x_{k+1} := (1-\theta_k)x_k + \theta_k x(y_k)$$

Update in Algorithm SPVN

$$y_{k+1} := (1-\theta_k)(y_k + \theta_k A x_k) + \theta_k^2 A x_{\mu_k}(y_k)$$
$$\mu_{k+1} := (1-\theta_k)\mu_k$$
$$x_{k+1} := (1-\theta_k)x_k + \theta_k x_{\mu_{k+1}}(y_{k+1})$$

Theorem (Soheili and P, 2011)

Assume $\bar{x} \in \Delta_n$ *and* $\delta > 0$ *are given.*

(a) *If* $\delta < \rho(A)$, *then Algorithm SPVN finds a solution to* (D) *in at most*

$$\frac{2\sqrt{2n}}{\rho(A)} - 1.$$

iterations.

(b) *If* $\rho(A) < 0$, *then Algorithm SPVN finds a* δ*-solution to* (P) *in at most*

$$\frac{2\sqrt{2n}}{\delta} - 1$$

iterations.

Iterated Smooth Perceptron-Von Neumann Algorithm

Assume $\gamma > 1$ is a given constant.

Algorithm ISPVN(γ)

- $\tilde{x}_0 := \frac{1}{n}\mathbf{1}$
- for $i = 0, 1, \ldots$

  $\delta_i := \frac{\|A\tilde{x}_i\|}{\gamma}$

  $\tilde{x}_{i+1} = \mathrm{SPVN}(\tilde{x}_i, \delta_i)$

  end for

Main Theorem Again

Theorem (Soheili & P, 2012)

(a) *If* $\rho(A) > 0$, *then each call to SPVN in Algorithm ISPVN halts in at most* $\frac{2\sqrt{2n}}{\rho(A)} - 1$ *iterations. Consequently, Algorithm ISPVN finds a solution to* (D) *in at most*

$$\left(\frac{2\sqrt{2n}}{\rho(A)} - 1\right) \cdot \frac{\log(1/\rho(A))}{\log(\gamma)}$$

SPVN iterations.

(b) *If* $\rho(A) < 0$, *then each call to SPVN in Algorithm ISPVN halts in at most* $\frac{2\gamma\sqrt{2n}}{|\rho(A)|} - 1$ *iterations. Hence for* $\epsilon > 0$ *Algorithm ISPVN finds an* ϵ*-solution to* (P) *in at most*

$$\left(\frac{2\gamma\sqrt{2n}}{|\rho(A)|} - 1\right) \cdot \frac{\log(1/\epsilon)}{\log(\gamma)}$$

SPVN iterations.

Observe

- A "pure" SPVN ($\delta = 0$):
  - When $\rho(A) > 0$, it solves (D) in $\mathcal{O}\left(\frac{\sqrt{n}}{\rho(A)}\right)$ iterations.
  - When $\rho(A) < 0$, it finds ϵ-solution to (P) in $\mathcal{O}\left(\frac{\sqrt{n}}{\epsilon}\right)$ iterations.
- ISPVN (iterated SPVN with gradual reduction on δ):
  - When $\rho(A) > 0$, it solves (D) in $\mathcal{O}\left(\frac{\sqrt{n}}{\rho(A)} \log\left(\frac{1}{\rho(A)}\right)\right)$ iterations.
  - When $\rho(A) < 0$, it finds ϵ-solution to (P) in $\mathcal{O}\left(\frac{\sqrt{n}}{|\rho(A)|} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations.

Perceptron and von Neumann's as subgradient algorithms

Let

$$\phi(y) := -\frac{\|y\|^2}{2} + \min_{x \in \Delta_n} \langle A^\mathsf{T} y, x\rangle.$$

Observe

$$\max_y \phi(y) = \min_{x \in \Delta_n} \frac{1}{2}\|Ax\|^2 = \begin{cases} \frac{1}{2}\rho(A)^2 & \text{if } \rho(A) > 0 \\ 0 & \text{if } \rho(A) \le 0. \end{cases}$$

PVN Template: $y_{k+1} = y_k + \theta_k(-y_k + Ax(y_k))$ is a subgradient algorithm for

$$\max_y \phi(y).$$

For $\mu > 0$ and $\bar{x} \in \Delta_n$ let

$$\begin{aligned} \phi_\mu(y) &:= -\tfrac{\|y\|^2}{2} + \min_{x \in \Delta_n} \left\{ \langle A^\mathsf{T} y, x\rangle + \frac{\mu}{2}\|x - \bar{x}\|^2 \right\} \\ &= -\tfrac{\|y\|^2}{2} + \langle A^\mathsf{T} y, x_\mu(y)\rangle + \tfrac{\mu}{2}\|x_\mu(y) - \bar{x}\|^2. \end{aligned}$$

Proof of Main Theorem

Apply Nesterov's excessive gap technique (Nesterov, 2005).

Claim

For all $x \in \Delta_n$ and $y \in \mathbb{R}^m$ we have $\phi(y) \leq \frac{1}{2}\|Ax\|^2$.

Claim

For all $y \in \mathbb{R}^m$ we have $\phi(y) \leq \phi_\mu(y) \leq \phi(y) + 2\mu$.

Lemma

The iterates $x_k \in \Delta_n$, $y_k \in \mathbb{R}^m$, $k = 0, 1, \ldots$ generated by the SPVN Algorithm satisfy the Excessive Gap Condition

$$\frac{1}{2}\|Ax_k\|^2 \leq \phi_{\mu_k}(y_k).$$

Proof of Main Theorem (a): $\rho(A) > 0$

Putting together the two claims and lemma we get

$$\frac{1}{2}\rho(A)^2 \le \frac{1}{2}\|Ax_k\|^2 \le \phi_{\mu_k}(y_k) \le \phi(y_k) + 2\mu_k.$$

So

$$\phi(y_k) \ge \frac{1}{2}\rho(A)^2 - 2\mu_k.$$

In the algorithm $\mu_k = n \cdot \frac{1}{3} \cdot \frac{2}{4} \cdots \frac{k}{k+2} = \frac{2n}{(k+1)(k+2)} < \frac{2n}{(k+1)^2}$.

Thus $\phi(y_k) > 0$, and consequently $A^\mathsf{T} y_k > 0$, as soon as

$$k \ge \frac{2\sqrt{2n}}{\rho(A)} - 1.$$

$\square$

Proof of Main Theorem (continued)

Suppose now $\rho(A) < 0$, i.e., (P) is feasible.

Let

$$S := \{x \in \Delta_n : Ax = 0\},$$

and for $v \in \mathbb{R}^n$ let

$$\mathrm{dist}(v, S) := \min\{\|v - x\| : x \in S\}.$$

Lemma

If $\rho(A) < 0$ then for all $v \in \Delta_n$

$$\mathit{dist}(v, S) \le \frac{2\|Av\|}{|\rho(A)|}.$$

Proof of Main Theorem (b): $\rho(A) < 0$

As in part (a), at iteration k of Algorithm SPVN we have

$$
\begin{aligned}
\tfrac{1}{2}\|Ax_k\|^2 &\leq \varphi_{\mu_k}(y_k) \\
&\leq \min_{x \in S}\left\{-\tfrac{\|y_k\|^2}{2} + \langle A^\mathsf{T} y_k, x\rangle + \tfrac{\mu_k}{2}\|x - \bar{x}\|^2\right\} \\
&\leq \frac{\mu_k}{2}\min_{x \in S}\|x - \bar{x}\|^2 \\
&= \frac{\mu_k}{2}\mathrm{dist}(\bar{x}, S)^2.
\end{aligned}
$$

Thus by previous lemma and the fact that $\mu_k < \frac{2n}{(k+1)^2}$ we get

$$\|Ax_k\|^2 \leq \mu_k \cdot \mathrm{dist}(\bar{x}, S)^2 \leq \frac{4\mu_k\|A\bar{x}\|^2}{\rho(A)^2} \leq \frac{8n\|A\bar{x}\|^2}{(k+1)^2\rho(A)^2}.$$

So when $k \geq \frac{2\gamma\sqrt{2n}}{|\rho(A)|} - 1$ we have $\|Ax_k\| \leq \frac{\|A\bar{x}\|}{\gamma}$ and Algorithm SPVN halts.

About the key smoothing step

We could instead use the entropy function

$$d(x) = \sum_{j=1}^{n} x_j \log(x_j).$$

Bregman distance:

$$h(x, \bar{x}) := d(x) - d(\bar{x}) - \langle \nabla d(\bar{x}), x - \bar{x} \rangle.$$

Given $\mu > 0$ and $\bar{x} \in \Delta_n$, smooth

$$x(y) = \operatorname*{argmin}_{x \in \Delta_n} \langle A^\mathsf{T} y, x \rangle,$$

to

$$x_\mu(y) := \operatorname*{argmin}_{x \in \Delta_n} \left\{ \langle A^\mathsf{T} y, x \rangle + \mu h(x, \bar{x}) \right\}.$$

Replace $\frac{1}{2}\|x - \bar{x}\|^2$ with $h(x, \bar{x})$.

About the key smoothing step

With the entropy we get stronger result for SPVN:

Theorem (Soheili and P, 2011)

Assume $\bar{x} \in \Delta_n$ and $\delta > 0$ are given.

(a) *If $\delta < \rho(A)$, then Algorithm SPVN finds a solution to* (D) *in at most*
$$\frac{2\sqrt{\log(n)}}{\rho(A)} - 1.$$
iterations.

(b) *If $\rho(A) < 0$, then Algorithm SPVN finds a δ-solution to* (P) *in at most*
$$\frac{2\sqrt{\log(n)}}{\delta} - 1$$
iterations.

However, the proof of Main Theorem (b) for ISPVN breaks down.

More general feasibility problems

Given $A \in \mathbb{R}^{m \times n}$ and a regular closed convex cone $K \subseteq \mathbb{R}^n$, consider the alternative feasibility problems

$$A^{\mathsf{T}} y \in \operatorname{int}(K^*), \tag{D}$$

and

$$Ax = 0,\ x \in K,\ x \neq 0. \tag{P}$$

Assume

For some $\mathbf{1} \in \operatorname{int}(K^*)$, we have an oracle that solves

$$x(y) := \operatorname*{argmin}_x \left\{ \langle A^{\mathsf{T}} y, x \rangle : x \in K,\ \langle \mathbf{1}, x \rangle = 1 \right\}.$$

More general feasibility problems

Recall Renegar's condition number

$$C(A) = \frac{\|A\|}{\inf_{A}\{\|A - \tilde{A}\| : \tilde{A} \text{ ill-posed}\}}.$$

Theorem (Epelman & Freund, 2000)

A generalized von Neumann's algorithm solves (D) *in*

$$\mathcal{O}(\beta \cdot C(A)^2)$$

iterations, or finds an ϵ-solution to (P) *in*

$$\mathcal{O}(\beta \cdot C(A)^2 \cdot \log(\|A\|/\epsilon))$$

iterations.

β: constant depending on specific choice of norms and $\mathbf{1} \in \operatorname{int}(K)$.

Smooth version

Assume

For some fixed $\mathbf{1} \in \text{int}(K)$, we have an oracle that solves

$$\operatorname*{argmin}_{x} \left\{ \langle A^\mathsf{T} y, x \rangle + \frac{1}{2}\|x\|^2 : x \in K,\ \langle \mathbf{1}, x \rangle = 1 \right\}.$$

Theorem (Soheili & P, 2012)

A smooth generalized von Neumann's algorithm solves (D) *in*

$$\mathcal{O}(\beta\sqrt{n} \cdot C(A) \cdot \log(C(A)))$$

iterations, or finds an ϵ-solution to (P) *in*

$$\mathcal{O}(\beta\sqrt{n} \cdot C(A) \cdot \log(\|A\|/\epsilon))$$

iterations.

Summary

- Smooth perceptron-von Neumann algorithm improves condition-based complexity roughly from $C(A)^2$ to $C(A)$.
- Smooth version preserves most of the algorithms' original simplicity.
- There seems to be room for sharper complexity results.

Happy Birthday to Mike Shub!