An Efficient Affine-Scaling Algorithm for Hyperbolic Programming

Jim Renegar

– joint work with Mutiara Sondjaja
A homogeneous polynomial \( p : \mathcal{E} \to \mathbb{R} \) is **hyperbolic** if there is a vector \( e \in \mathcal{E} \) such that for all \( x \in \mathcal{E} \), the univariate polynomial \( t \mapsto p(x + te) \) has only real roots.

" \( p \) is hyperbolic in direction \( e \) "

Example: \( \mathcal{E} = \mathbb{S}^{n \times n}, \ p(X) = \det(X), \ E = I \) (identity matrix)

- then \( p(X + tE) \) is the characteristic polynomial of \( -X \)

All roots are real because symmetric matrices have only real eigenvalues.

The **hyperbolicity cone** \( \Lambda_{++} \) is

the connected component of \( \{ x : p(x) \neq 0 \} \) containing \( e \).

For the example, \( \Lambda_{++} = \mathbb{S}_+^{n \times n} \) (cone of positive-definite matrices)

- the convexity of this particular cone
  is true of hyperbolicity cones in general...
A hyperbolic program is an optimization problem of the form

\[ \begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b \\
& \quad x \in \Lambda_+ \\
& \quad x \in \Lambda_{++} \text{ closure of } \Lambda_{++}
\end{align*} \]

Thm (Gårding, 1959): \( \Lambda_{++} \) is a convex cone

Güler (1997) introduced hyperbolic programming, motivated largely by the realization that
\[ f(x) = -\ln p(x) \]
is a self-concordant barrier function

\[ "O(\sqrt{n}) \text{ iterations to halve the duality gap }" \]
where \( n \) is the degree of \( p \)

Güler showed the barrier functions \( f(x) = -\ln p(x) \)
possess many of the nice properties of \( X \mapsto -\ln \det(X) \)

although hyperbolicity cones in general are not symmetric (i.e., self-scaled)
\[ \min \left\{ \langle c, x \rangle \right\} \]
\[ \text{s.t. } Ax = b \]
\[ x \in \Lambda_+ \]
\[ \left\} \text{HP} \right. \]

There are natural ways in which to “relax” HP to hyperbolic programs for lower degree polynomials.

For example, to obtain a relaxation of SDP \( \ldots \)

Fix \( n \), and for \( 1 \leq k \leq n \) let \( \sigma_k(\lambda_1, \ldots, \lambda_n) := \sum_{j_1 < \ldots < j_k} \lambda_{j_1} \cdots \lambda_{j_k} \)

- elementary symmetric polynomial of degree \( k \)

Then \( X \mapsto \sigma_k(\lambda(X)) \)

is a hyperbolic polynomial in direction \( E = I \) of degree \( k \),
and its hyperbolicity cone contains \( \mathbb{S}^{n \times n}_{++} \)

These polynomials can be evaluated efficiently via the FFT.

Perhaps relaxing SDP’s in this and related ways will allow larger SDP’s to be approximately solved efficiently.

The relaxations easily generalize to all hyperbolic programs.
\[
\begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b \\
& \quad x \in \Lambda_+ \\
\end{align*}
\tag{HP}
\]

barrier function, \( f(x) = -\ln p(x) \)
its gradient \( g(x) \)
and Hessian \( H(x) \) positive-definite for all \( x \in \Lambda_{++} \)

“local inner product at \( e \in \Lambda_{++} \)” \( \langle u, v \rangle_e := \langle u, H(e)v \rangle \)

- the induced norm: \( \|v\|_e = \sqrt{\langle v, v \rangle_e} \)
- “Dikin ellipsoids”: \( \bar{B}_e(e, r) = \{x : \|x - e\|_e \leq r\} \)

The gist of the original affine-scaling method due to Dikin is simply:

Given a strictly feasible point \( e \) for HP and an appropriate value \( r > 0 \),
move from \( e \) to the optimal solution \( e_+ \) for

\[
\begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b \\
& \quad x \in \bar{B}_e(e, r) \\
\end{align*}
\]

Dikin focused on linear programming
and chose \( r = 1 \) (giving the largest Dikin ellipsoids contained in \( \mathbb{R}^n_+ \))
also: Vanderbei, Meketon and Freedman (1986)
In the mid-1980’s, there was considerable effort trying to prove that Dikin’s affine-scaling method runs in polynomial-time (perhaps with choice $r < 1$).

The efforts mostly ceased when in 1986, Shub and Megiddo showed that the “infinitesimal version” of the algorithm can come near all vertices of a Klee-Minty cube.

Nevertheless, several algorithms with spirit similar to Dikin’s method have been shown to halve the duality gap in polynomial time:

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Year</th>
<th>Algorithm(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monteiro, Adler and Resende</td>
<td>1990</td>
<td>LP, and convex QP</td>
</tr>
<tr>
<td>Jansen, Roos and Terlaky</td>
<td>1996</td>
<td>LP</td>
</tr>
<tr>
<td></td>
<td>1997</td>
<td>PSD LCP-problems</td>
</tr>
<tr>
<td>Sturm and Zhang</td>
<td>1996</td>
<td>SDP</td>
</tr>
<tr>
<td>Chua</td>
<td>2007</td>
<td>symmetric cone programming</td>
</tr>
</tbody>
</table>

These algorithms are primal-dual methods and rely heavily on the cones being self-scaled.

Our framework shares some strong connections to the one developed by Chek Beng Chua, to whom we are indebted.
\[
\min \left\{ \langle c, x \rangle \mid Ax = b, x \in \Lambda_+ \right\}
\]

For \( e \in \Lambda_{++} \) and \( 0 < \alpha < \sqrt{n} \), let

\[
K_e(\alpha) := \{ x : \langle e, x \rangle_e \geq \alpha \| x \|_e \}
\]

- this happens to be the smallest cone containing the Dikin ellipsoid \( B_e(e, \sqrt{n - \alpha^2}) \)

Keep in mind that the cone grows in size as \( \alpha \) decreases.

\[
\begin{array}{c}
\min \left\{ \langle c, x \rangle \mid Ax = b, x \in \Lambda_+ \right\} \\
\rightarrow \quad \min \left\{ \langle c, x \rangle \mid Ax = b, x \in K_e(\alpha) \right\}
\end{array}
\]

**Definition:** \( \text{Swath}(\alpha) = \{ e \in \Lambda_{++} : Ae = b \text{ and } QP_e(\alpha) \text{ has an optimal solution} \} \)

**Prop:** \( \text{Swath}(0) = \text{Central Path} \)

Thus, \( \alpha \) can be regarded as

a measure of the proximity of points in \( \text{Swath}(\alpha) \) to the central path.
Let $x \in \Lambda_+$ be the optimal solution of $\text{QP}_e(\alpha)$ (assuming $e \in \text{Swath}(\alpha)$). The main work in computing $x_e(\alpha)$ lies in solving a system of linear equations:

$$
\begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b, \quad x \in \Lambda_+ \\
\end{align*}
\quad \rightarrow
\begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b, \quad x \in K_e(\alpha) \\
\end{align*}
\{x : \langle e, x \rangle_e \geq \alpha \|x\|_e\}
$$

**Definition:** $\text{Swath}(\alpha) = \{e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution}\}$

Let $x_e(\alpha)$ be the optimal solution of $\text{QP}_e(\alpha)$ (assuming $e \in \text{Swath}(\alpha)$), then $x_e(\alpha)$ is obtained by solving a system of linear equations.

We assume $0 < \alpha < 1$, in which case $\Lambda_+ \subseteq K_e(\alpha)$

- thus, $K_e(\alpha)$ is a relaxation of $\text{HP}$

- hence, optimal_value_of_HP $\geq \langle c, x_e(\alpha) \rangle$

**Current iterate:** $e \in \Lambda_{++}$

Next iterate will be $e'$, a convex combination of $e$ and $x_e(\alpha)$

$$
e' = \frac{1}{1+t} (e + t x_e(\alpha))$$

The choice of $t$ is made through duality ...
\[
\begin{align*}
&\min \langle c, x \rangle \\
&\text{s.t. } Ax = b \\
&\quad x \in \Lambda_+ \quad \{\text{HP}\} \quad \longrightarrow \quad \{\text{QP}_e(\alpha)\}
\end{align*}
\]

**Definition:** \(\text{Swath}(\alpha) = \{ e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution} \} \)

Let \(x_e(\alpha) = \text{optimal solution of } \text{QP}_e(\alpha) \quad (\text{assuming } e \in \text{Swath}(\alpha)) \)
\[
\min \langle c, x \rangle \quad \text{s.t.} \quad Ax = b, \quad x \in \Lambda_+ \quad \rightarrow \quad \min \langle c, x \rangle \quad \text{s.t.} \quad Ax = b, \quad x \in K_e(\alpha) \quad \text{QP}_e(\alpha)
\]
\[
\{x : \langle e, x \rangle \geq \alpha \|x\|_e\}
\]

**Definition:** Swath(\(\alpha\)) = \{\(e \in \Lambda_{++} : Ae = b\) and QP\(_e(\alpha)\) has an optimal solution\}

Let \(x_e = \text{optimal solution of QP}_{e}(\alpha)\) (assuming \(e \in \text{Swath}(\alpha)\))
\[
\begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b \\
x & \in \Lambda_+ \quad \text{HP} \\
\end{align*} \quad \rightarrow \quad \begin{align*}
\min & \quad \langle c, x \rangle \\
\text{s.t.} & \quad Ax = b \\
x & \in K_e(\alpha) \quad \text{QP}_e(\alpha)
\end{align*}
\]

\[
\{x : \langle e, x \rangle \geq \alpha \| x \|_e\}
\]

**Definition:** Swath(\(\alpha\)) = \(\{e \in \Lambda_{++} : Ae = b\) and \(\text{QP}_e(\alpha)\) has an optimal solution\)

Let \(x_e = \) optimal solution of \(\text{QP}_e(\alpha)\) \(\) (assuming \(e \in \text{Swath}(\alpha)\))

\[
\begin{align*}
\max & \quad b^T y \\
\text{s.t.} & \quad A^* y + s = c \\
s & \in \Lambda_+^* \quad \text{HP}^* \\
\end{align*} \quad \rightarrow \quad \begin{align*}
\max & \quad b^T y \\
\text{s.t.} & \quad A^* y + s = c \\
x & \in K_e(\alpha)^* \quad \text{QP}_e(\alpha)^*
\end{align*}
\]

First-order optimality conditions for \(x_e\)

yield optimal solution \((y_e, s_e)\) for \(\text{QP}_e(\alpha)^*\)

Moreover, \((y_e, s_e)\) is feasible for \(\text{HP}^*\) because \(\Lambda_+ \subseteq K_e(\alpha)\) and hence \(K_e(\alpha)^* \subseteq \Lambda_+^*\)

**primal-dual feasible pair:** \(e\) for HP, \((y_e, s_e)\) for HP*

**duality gap:** \(\text{gap}_e := \langle c, e \rangle - b^T y_e\)
Swath(α) = \{e ∈ Λ_{++} : Ae = b \ and \ QP_{e}(α) \ has \ an \ optimal \ solution\}

\[x_e = \text{optimal solution of } QP_{e}(α) \quad (\text{assuming } e ∈ \text{Swath}(α))\]

primal-dual feasible pair: \(e\) for HP, \((y_e, s_e)\) for HP*

Current iterate: \(e \in Λ_{++}\)
Next iterate will be a convex combination of \(e\) and \(x_e\):

\[e(t) = \frac{1}{1+t}(e + tx_e)\]

Want \(t\) to be large so as to improve primal objective value,
but also want \(e(t) ∈ \text{Swath}(α)\)

We choose \(t\) to be the minimizer of a particular quadratic polynomial,
and thereby ensure that:

- \(e(t) ∈ Λ_{++}\)
- \(s_e ∈ \text{int}(K_{e(t)}(α)^*)\)

- consequently, both \(e(t)\) is strictly feasible for \(QP_{e(t)}(α)\)
  and \((y_e, s_e)\) is strictly feasible for \(QP_{e(t)}(α)^*\)

- hence, \(e(t) ∈ \text{Swath}(α)\)
Swath(\(\alpha\)) = \{e \in \Lambda_{++} : Ae = b\text{ and } \text{QP}_e(\alpha)\text{ has an optimal solution}\}

\[x_e = \text{optimal solution of } \text{QP}_e(\alpha) \text{ (assuming } e \in \text{Swath}(\alpha))\]

primal-dual feasible pair: \(e\) for HP, \((y_e, s_e)\) for HP*

Current iterate: \(e \in \Lambda_{++}\)
Next iterate will be a convex combination of \(e\) and \(x_e\):

\[e(t) = \frac{1}{1+t} (e + t x_e)\]

Want \(t\) to be large so as to improve primal objective value,
but also want \(e(t) \in \text{Swath}(\alpha)\)

**We choose \(t\) to be the minimizer of a particular quadratic polynomial,**
and thereby ensure that:

- \(t \geq \frac{1}{2} \alpha/\|x_e\|_e\)

  - and thus ensure good improvement in the primal objective value
    
    if, say, \(\|x_e\|_e \leq \sqrt{n}\)
Swath(\(\alpha\)) = \{e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution}\}

\(x_e = \text{optimal solution of } \text{QP}_e(\alpha) \text{ (assuming } e \in \text{Swath}(\alpha))\)

primal-dual feasible pair: \(e\) for HP, \((y_e, s_e)\) for HP*

Current iterate: \(e \in \Lambda_{++}\)
Next iterate will be a convex combination of \(e\) and \(x_e\):

\[e(t) = \frac{1}{1+t}(e + t x_e)\]

Want \(t\) to be large so as to improve primal objective value, but also want \(e(t) \in \text{Swath}(\alpha)\)

We choose \(t\) to be the minimizer of a particular quadratic polynomial, and thereby ensure that:

- \(s_e \in K_{e(t)}(\beta)^*\) where \(\beta = \alpha \sqrt{\frac{1+\alpha}{2}}\)

- which implies \(s_e\) is “deep within” \(K_{e(t)}(\alpha)^*\)

- and hence \((y_e, s_e)\) is “very strongly” feasible for \(\text{QP}_{e(t)}(\alpha)^*\)
We choose $t$ to be the minimizer of a particular quadratic polynomial, and thereby ensure that:

1. There is “good” improvement in primal objective value if $\|x_e\|_e \leq \sqrt{n}$
2. $(y_e, s_e)$ is “very strongly” feasible for QP$_{e(t)}^*$

Sequence of iterates: $e_0, e_1, e_2, \ldots$

- write $x_i$ and $(y_i, s_i)$ rather than $x_{e_i}$ and $(y_{e_i}, s_{e_i})$

If $i > 0$, then

1. $\|x_i\|_{e_i} \leq \sqrt{n} \Rightarrow \langle c, e_{i+1} \rangle \ll \langle c, e_i \rangle$
2. $(y_{i-1}, s_{i-1})$ is “very strongly” feasible for QP$_{e_i}^*$

On the other hand, we show

3. $(\|x_i\|_{e_i} \geq \sqrt{n}) \wedge (2.) \Rightarrow b^T y_i \gg b^T y_{i-1}$

In this manner we establish the Main Theorem ...
Main Thm:

- The primal objective value improves monotonically, and so does the dual objective value.
- If $i, k \geq 0$, then

\[ \frac{\text{gap}_{e_i}}{\text{gap}_{e_{i+k}}} \geq \left( 1 + \alpha \sqrt{\frac{1 - \alpha}{8n}} \right)^{k-1} \]

\[ e(t) = \frac{1}{1+t} (e + tx_e) \]

“We choose $t$ to be the minimizer of a particular quadratic polynomial”

In fact, the theorem holds if one simply chooses $t = \frac{1}{2} \alpha / \| x_i \|_{e_i}$, but choosing $t$ to be the minimizer can result in steps that are far longer.

So what is the particular quadratic polynomial?
\[ K_e(\alpha) := \{ x : \langle e, x \rangle_e \geq \alpha \| x \|_e \} \]

**Special Case of SDP:**

\[ E \in \mathbb{S}^{n \times n}_{++}, \quad X_E \text{ optimal for } \text{QP}_E(\alpha), \quad (y_E, S_E) \text{ optimal for } \text{QP}_E(\alpha) \]

Let \[ E(t) = \frac{1}{1+t} \left( E + t X_E \right) \]

Here is the quadratic polynomial:
\[
q(t) := \text{tr} \left( \left( (E + t X_E) S_E \right)^2 \right)
\]

**Prop:**

\[(t \geq 0) \land (E(t) \succ 0) \Rightarrow \min \{ \beta : S_E \in K_{E(t)}(\beta)^* \} = \sqrt{n - 1/q(t)}\]

**Prop:** The minimizer \( t \) of \( q \) satisfies

\[
t > \frac{1}{2} \alpha / \| X_E \|_E, \quad E(t) \succ 0 \quad \text{and} \quad q(t) \leq \alpha \sqrt{\frac{1+\alpha}{2}}.
\]

**Corollary:** \( S_E \) is “deep within” \( K_{E(t)}(\alpha)^* \) for the minimizer \( t \) of \( q \)
\[ K_e(\alpha) := \{ x : \langle e, x \rangle_e \geq \alpha \| x \|_e \} \]

**Hyperbolic Programming in General:**

\[ e \in \Lambda_{++}, \quad x_e \text{ optimal for } \text{QP}_e(\alpha), \quad (y_e, s_e) \text{ optimal for } \text{QP}_e(\alpha)^* \]

Let \[ e(t) = \frac{1}{1+t} (e + t x_e) \]

As happened for SDP, we would like to have an easily computable function \( \tilde{q} \) for which

\[ (t \geq 0) \land (e(t) \in \Lambda_{++}) \quad \Rightarrow \quad \min \{ \beta : s_e \in K_{e(t)}(\beta)^* \} = \sqrt{n - 1/\tilde{q}(t)} \]

However, in general the resulting function \( \tilde{q} \) need not be a quadratic polynomial, nor do we see any reason that it necessarily be efficiently computable

– in fact, about the most we know is that \( \tilde{q} \) is semi algebraic.

But we do know how to obtain a quadratic polynomial \( q \) which serves as an appropriate upper bound to the function \( \tilde{q} \)

– we do this by leveraging our SDP result with the (very deep) Helton-Vinnikov Theorem for hyperbolicity cones.
Helton-Vinnikov Theorem:

If $p : \mathcal{E} \to \mathbb{R}$ is hyperbolic in direction $e$ and of degree $n$,
and if $L$ is a 3-dimensional subspace of $\mathcal{E}$ containing $e$,
then there exists a linear transformation $T : L \to \mathbb{S}^{n \times n}$ satisfying

$$T(e) = I \quad \text{and} \quad p(x) = p(e) \det(T(x)) \quad \text{for all } x \in L.$$
Luckily, the resulting quadratic polynomial always can be efficiently computed:

- First compute the five leading coefficients \( a_n, a_{n-1}, a_{n-2}, a_{n-3}, a_{n-4} \) of the univariate polynomial

\[
\gamma \mapsto p(x_e + \gamma e) = \sum_{i=1}^{n} a_i \gamma^i
\]

- Then compute

\[
\kappa_1 = \frac{a_{n-1}}{a_n} , \quad \kappa_2 = \left( \frac{a_{n-1}}{a_n} \right)^2 - \frac{a_{n-2}}{a_n} , \quad \kappa_3 = \left( \frac{a_{n-1}}{a_n} \right)^3 - \frac{3}{2} \frac{a_{n-1}}{a_n} \frac{a_{n-2}}{a_n} + \frac{1}{2} \frac{a_{n-3}}{a_n}
\]

\[
\text{and} \quad \kappa_4 = \left( \frac{a_{n-1}}{a_n} \right)^4 - 2 \left( \frac{a_{n-1}}{a_n} \right)^2 \frac{a_{n-2}}{a_n} + \frac{1}{2} \left( \frac{a_{n-2}}{a_n} \right)^2 + \frac{2}{3} \frac{a_{n-1}}{a_n} \frac{a_{n-3}}{a_n} - \frac{1}{6} \frac{a_{n-4}}{a_n}
\]

- The desired quadratic polynomial is \( t \mapsto at^2 + bt + c \) where

\[
a = \kappa_1^2 \kappa_2 - 2 \alpha^2 \kappa_1 \kappa_3 + \alpha^4 \kappa_4 ,
\]

\[
b = 2 \alpha^4 \kappa_3 - 2 \kappa_1^3 \quad \text{and} \quad c = (n - \alpha^2) \kappa_1^2
\]
**Epilogue:** Recently we learned of a paper on linear programming in which quadratic cones relaxing the non-negative orthant are used in devising a polynomial-time algorithm, albeit one with complexity \(O(nL)\) iterations rather than \(O(\sqrt{n}L)\) iterations.

I.S. Litvinchev,