# An Efficient Affine-Scaling Algorithm for Hyperbolic Programming

Jim Renegar

– joint work with Mutiara Sondjaja

Euclidean space

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 $t \mapsto p(x + t\,e)$ 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 $p(X + t\,E)$ 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 ...

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

A *hyperbolic program* is an optimization problem of the form

$$\left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+ \end{array}\right\} \text{HP}$$

closure of $\Lambda_{++}$

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

" $O(\sqrt{n})$ 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)

$$\left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+ \end{array}\right\} \text{HP}$$

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

---

For example, to obtain a relaxation of SDP ...

Fix $n$, and for $1 \le k \le n$ 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.

$$\left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+ \end{array}\right\} \text{HP}$$

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

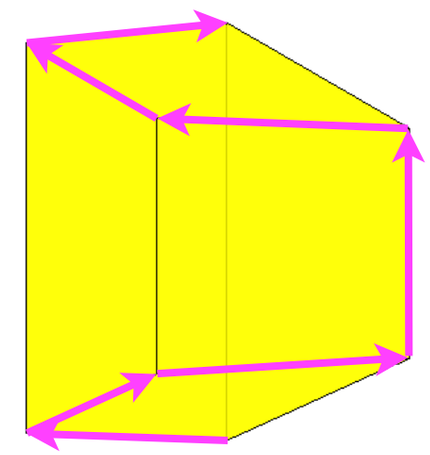
$$\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \bar{B}_e(e, r) \end{array}$$

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:

| | | | |
|---|---|---|---|
| Monteiro, Adler and Resende | 1990 | LP, and convex QP | use "scaling points" and "V-space" |
| Jansen, Roos and Terlaky | 1996 | LP | |
| | 1997 | PSD LCP-problems | |
| Sturm and Zhang | 1996 | SDP | use ellipsoidal cones rather than ellipsoids |
| Chua | 2007 | symmetric cone programming | |

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

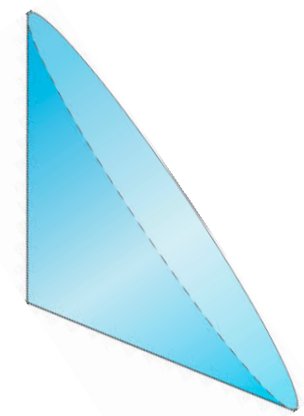
*Our framework shares some strong connections*
*to the one developed by Chek Beng Chua, to whom we are indebted.*

$$\left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+ \end{array}\right\} \text{HP}$$

For $e \in \Lambda_{++}$ and $0 < \alpha < \sqrt{n}$ , let $\boxed{K_e(\alpha) := \{x : \langle e, x\rangle_e \geq \alpha \, \|x\|_e\}}$

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

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



$$\boxed{\left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+ \end{array}\right\} \text{HP} \quad \longrightarrow \quad \left.\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K_e(\alpha) \end{array}\right\} \text{QP}_e(\alpha)}$$

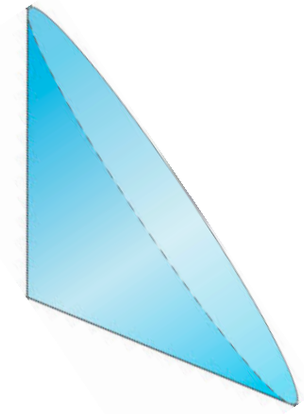
**Definition:** $\text{Swath}(\alpha) = \{e \in \Lambda_{++} : \; Ae = b \;\text{ and }\; \text{QP}_e(\alpha) \text{ has an optimal solution}\}$

**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.

$$\left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+\end{array}\right\}\text{HP} \quad \longrightarrow \quad \left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K_e(\alpha)\end{array}\right\}\text{QP}_e(\alpha)$$

$\{x : \langle e, x\rangle_e \geq \alpha\|x\|_e\}$



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

Let $x_e(\alpha)$ = 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

---

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

– thus, $K_e(\alpha)$ is a relaxation of HP

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

---

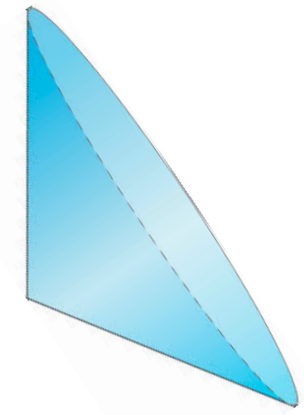
Current iterate: $e \in \Lambda_{++}$
Next iterate will be $e'$ , a convex combination of $e$ and $x_e(\alpha)$

$$e' = \tfrac{1}{1+t}\big(e + t\, x_e(\alpha)\big)$$

The choice of $t$ is made through duality ...

$$\left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+\end{array}\right\}\text{HP} \quad \longrightarrow \quad \left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K_e(\alpha)\end{array}\right\}\text{QP}_e(\alpha)$$

$\{x : \langle e, x\rangle_e \geq \alpha\|x\|_e\}$

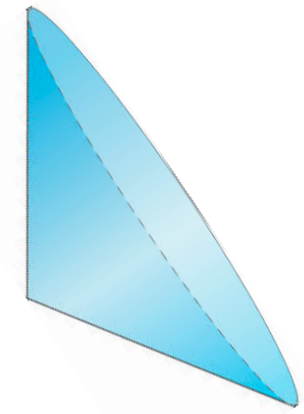


**Definition:** $\text{Swath}(\alpha) = \{e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution}\}$

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

$$\left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+\end{array}\right\}\text{HP} \quad \longrightarrow \quad \left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K_e(\alpha)\end{array}\right\}\text{QP}_e(\alpha)$$

$\{x : \langle e, x\rangle_e \geq \alpha\|x\|_e\}$

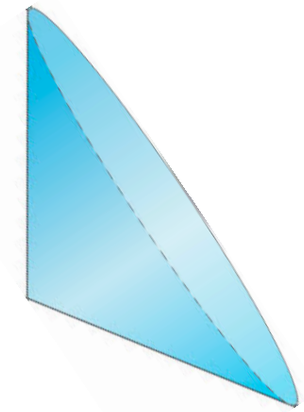


**Definition:** $\text{Swath}(\alpha) = \{e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution}\}$

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

$$\left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in \Lambda_+\end{array}\right\}\text{HP} \quad \longrightarrow \quad \left.\begin{array}{ll}\min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K_e(\alpha)\end{array}\right\}\text{QP}_e(\alpha)$$

$\{x : \langle e, x\rangle_e \geq \alpha\|x\|_e\}$

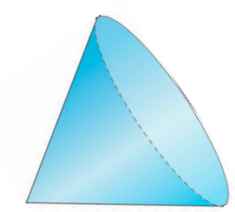


**Definition:** $\text{Swath}(\alpha) = \{e \in \Lambda_{++} : Ae = b \text{ and } \text{QP}_e(\alpha) \text{ has an optimal solution}\}$

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

---

$$\left.\begin{array}{ll}\max & b^T y \\ \text{s.t.} & A^* y + s = c \\ & s \in \Lambda_+^*\end{array}\right\}\text{HP}^* \quad \longrightarrow \quad \left.\begin{array}{ll}\max & b^T y \\ \text{s.t.} & A^* y + s = c \\ & x \in K_e(\alpha)^*\end{array}\right\}\text{QP}_e(\alpha)^*$$



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 $\text{HP}^*$

**duality gap:** $\text{gap}_e := \langle c, e\rangle - b^T y_e$

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

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

$$\text{primal-dual feasible pair:} \quad e \text{ for HP}, \quad (y_e, s_e) \text{ for HP}^*$$

---

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

$$e(t) = \tfrac{1}{1+t}\big(e + t\, x_e\big)$$

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:

- $e(t) \in \Lambda_{++}$
- $s_e \in \text{int}(K_{e(t)}(\alpha)^*)$

  – consequently, both $e(t)$ is strictly feasible for $\text{QP}_{e(t)}(\alpha)$
  and $(y_e, s_e)$ is strictly feasible for $\text{QP}_{e(t)}(\alpha)^*$

  – hence, $e(t) \in \text{Swath}(\alpha)$

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

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

$$\text{primal-dual feasible pair:} \qquad e \text{ for HP}\,, \qquad (y_e, s_e) \text{ for HP}^*$$

---

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

$$e(t) = \tfrac{1}{1+t}\big(e + t\, x_e\big)$$

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 <u>primal</u> objective value
  if, say, $\|x_e\|_e \leq \sqrt{n}$

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

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

$$\text{primal-dual feasible pair:} \qquad e \text{ for HP}\,, \qquad (y_e, s_e) \text{ for HP}^*$$

---

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

$$e(t) = \tfrac{1}{1+t}\big(e + t\, x_e\big)$$

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:

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

  – which implies $s_e$ is "deep within" $\quad K_{e(t)}(\alpha)^*$

    – and hence $(y_e, s_e)$ is "very strongly" feasible for $\text{QP}_{e(t)}(\alpha)^*$

$$e(t) = \tfrac{1}{1+t}\big(e + t\, x_e\big)$$

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 $\mathrm{QP}^*_{e(t)}$

---

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

On the other hand, we show

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

*In this manner we establish the Main Theorem* ...

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

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

**Main Thm:**

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

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

$$e(t) = \tfrac{1}{1+t}\big(e + t\, x_e\big)$$

"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:**

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

$$\text{Let} \quad E(t) = \tfrac{1}{1+t}\,(E + t\,X_E)$$

Here is the quadratic polynomial: $\quad q(t) := \mathbf{tr}\left( ((E + t\,X_E)S_E)^2 \right)$

**Prop:**

$$(t \geq 0) \ \wedge \ (E(t) \succ 0) \quad \Rightarrow \quad \min\left\{\beta : S_E \in K_{E(t)}(\beta)^*\right\} \ = \ \sqrt{n - 1/q(t)}$$

**Prop:** The minimizer $t$ of $q$ satisfies

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

*Corollary:* $\quad S_E$ *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:**

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

$$\text{Let} \quad e(t) = \tfrac{1}{1+t}\,(e + t\,x_e)$$

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

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

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 $p : \mathcal{E} \to R$ 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

$$T(e) = I \quad \text{and} \quad p(x) \;=\; p(e)\,\det\left(T(x)\right) \quad \text{for all } x \in L\ .$$

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

$$\gamma \mapsto p(x_e + \gamma\, e) \quad = \sum_{i=1}^{n} a_i \gamma^i$$

- 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 "$O(n\,L)$ iterations"
rather than "$O(\sqrt{n}\,L)$ iterations."

I.S. Litvinchev,
"A circular cone relaxation primal interior point algorithm for LP,"
*Optimization* 52 (2003) 529-540.