# Progress on solution of OPF problems

Daniel Bienstock, Columbia University

## Optimal Power Flow

- Used to operate electrical power transmission systems ("grids")
- Primary goal: economic and secure operation
- Run as often as every five minutes
- Inputs for the computation: the current state of the grid, and estimates of demands ("loads") in the next time window
- First proposed by Carpentier in 1962

## Brief introduction

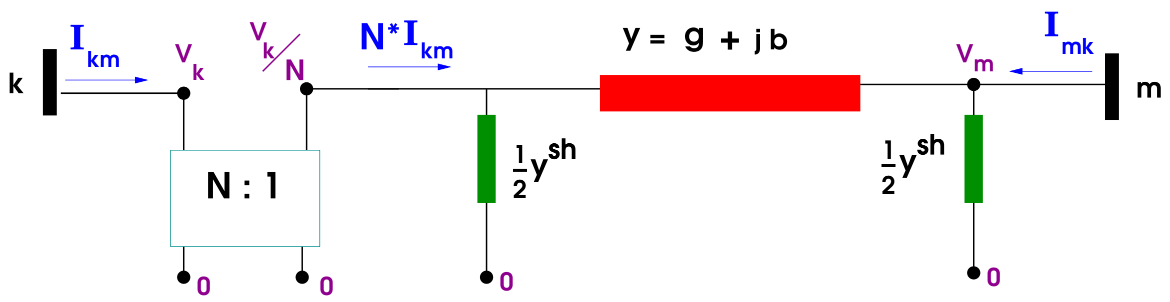
- Grid modeled as a network; nodes = "buses", edges = "lines"
- **Steady-state** operation: each bus $k$ has a voltage (potential energy)

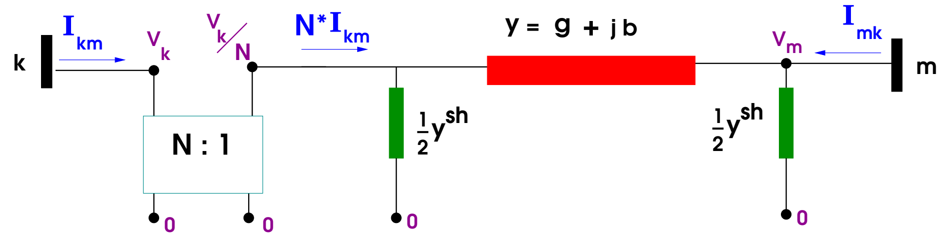
$$V_k = |V_k| e^{j\theta_k}$$

- Each line $\{k, m\}$ has physical attributes: e.g. *resistance* $r$, *reactance* $x$, *shunt admittance* $y^{sh}$

$$z \doteq r + jx, \quad \text{(series impedance)}$$
$$y \doteq z^{-1} = g + jb, \quad \text{(admittance)}$$
$$g = \frac{r}{r^2 + x^2} \quad \text{and} \quad b = -\frac{x}{r^2 + x^2},$$

- A **transformer** with $N \doteq \tau e^{j\sigma}$ scales voltages by $N$.






$$(N = \tau e^{j\sigma})$$

$$V = \begin{pmatrix} V_k \\ V_m \end{pmatrix} = \begin{pmatrix} |V_k|e^{j\theta_k} \\ |V_m|e^{j\theta_m} \end{pmatrix} = \begin{pmatrix} e_k + jf_k \\ e_m + jf_m \end{pmatrix} \quad \text{(voltages at } k \text{ and } m\text{)}$$

$$I = \begin{pmatrix} I_{km} \\ I_{mk} \end{pmatrix} \quad \text{(complex current injections at } k \text{ and } m\text{)}$$

$$S = \begin{pmatrix} S_{km} \\ S_{mk} \end{pmatrix} = \begin{pmatrix} P_{km} + jQ_{km} \\ P_{mk} + jQ_{mk} \end{pmatrix} \quad \text{(complex power injections at } k \text{ and } m\text{)}$$

Then

$$S_{km} = V_k I_{km}^*, \quad S_{mk} = V_m I_{mk}^* \quad \text{and} \quad I = \mathbb{Y} V,$$

where

$$\mathbb{Y} = \begin{pmatrix} (y + \frac{y^{sh}}{2})\frac{1}{\tau^2} & -y\frac{1}{\tau e^{-j\sigma}} \\ -y\frac{1}{\tau e^{j\sigma}} & y + \frac{y^{sh}}{2} \end{pmatrix}.$$

$$S_{km} = P_{km} + jQ_{km}$$

$$P_{km} = |V_k|^2 g - |V_k||V_m| g \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

(active power injected by $k$ into $km$)

$$Q_{km} = -|V_k|^2 b + |V_k||V_m| b \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

(reactive power injected by $k$ into $km$)

$$(\theta_{km} \doteq \theta_k - \theta_m)$$

$$S_{km} = P_{km} + jQ_{km}$$

$$P_{km} = |V_k|^2 g - |V_k||V_m| g \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

(active power injected by $k$ into $km$)

$$Q_{km} = -|V_k|^2 b + |V_k||V_m| b \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

(reactive power injected by $k$ into $km$)

$$(\theta_{km} \doteq \theta_k - \theta_m)$$

$$P_k \doteq \sum_{km} P_{km} \quad \text{total active power injection by } k$$

$$Q_k \doteq \sum_{km} Q_{km} \quad \text{total reactive power injection by } k$$

## OPF problem, simple version

Choose $|V_k|$ and $\theta_k$ for each bus $k$, so that

$$
\begin{aligned}
\text{min} \quad & \sum_{g \in \mathbb{G}} F_g(P_g) \\
\text{s.t.} \quad & L_k \ \leq \ P_k \ \leq \ U_k \qquad \text{all } k \\
& V_k^{min} \ \leq \ |V_k| \ \leq \ V_k^{max} \qquad \text{all } k \\
& |S_{km}| \ \leq \ S_{km}^{max} \qquad \text{all } km \\
& |\theta_{km}| \ \leq \ \theta_{km}^{max} \qquad \text{all } km\text{, sometimes}
\end{aligned}
$$

$F_g$ convex quadratic, usually.

## OPF problem, simple version

Choose $|V_k|$ and $\theta_k$ for each bus $k$, so that

$$\begin{aligned}
\min \quad & \sum_{g \in \mathbb{G}} F_g(P_g) \\
\text{s.t.} \quad & L_k \ \leq \ P_k \ \leq \ U_k \qquad \text{all } k \\
& V_k^{min} \ \leq \ |V_k| \ \leq \ V_k^{max} \qquad \text{all } k \\
& |S_{km}| \ \leq \ S_{km}^{max} \qquad \text{all } km \\
& |\theta_{km}| \ \leq \ \theta_{km}^{max} \qquad \text{all } km\text{, sometimes}
\end{aligned}$$

$F_g$ convex quadratic, usually.

In principle, this is a difficult, nonconvex optimization problem

## How does the industry handle this problem?

- Techniques borrowed from convex optimization, i.e. logarithmic barrier methods
- Sequential linearization
- Other heuristics
- If everything fails, change the problem
- Some software is quite old
- Works very well on routine problems – runs in seconds
- May not work well on grids under distress

$$P_{km} = |V_k|^2 g - |V_k||V_m| g \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

$$P_{km} \;=\; |V_k|^2 g - |V_k||V_m| g \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

## Practical adaptation for routine operation

- $r = 0$ for each line (zero resistance).
  So $g = \frac{r}{r^2+x^2} = 0$, $b = -\frac{x}{r^2+x^2} = -x^{-1}$
- $|V_k| = 1$ for all buses $k$ (after scaling)
- $\theta_k - \theta_m \;\approx\; 0$ for all lines $km$, so $\sin(\theta_k - \theta_m) \;\approx\; \theta_k - \theta_m$
- Only focus on active power

$$P_{km} \;=\; |V_k|^2 g - |V_k||V_m| g \cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

$$\approx\; \frac{\theta_k - \theta_m}{x} \;=\; y(\theta_k - \theta_m)$$

$$P_{km} \;=\; |V_k|^2 g - |V_k||V_m| g\cos\theta_{km} - |V_k||V_m| b \sin\theta_{km}$$

**"DC Approximation"**

- $r = 0$ for each line (zero resistance).
  So $g = \frac{r}{r^2+x^2} = 0$, $b = -\frac{x}{r^2+x^2} = -x^{-1}$
- $|V_k| = 1$ for all buses $k$ (after scaling)
- $\theta_k - \theta_m \approx 0$ for all lines $km$, so $\sin(\theta_k - \theta_m) \approx \theta_k - \theta_m$
- Only focus on active power

$$\begin{aligned} P_{km} &= |V_k|^2 g - |V_k||V_m| g\cos\theta_{km} - |V_k||V_m| b \sin\theta_{km} \\ &\approx \frac{\theta_k - \theta_m}{x} \;=\; y(\theta_k - \theta_m) \end{aligned}$$

So, get

$$\begin{aligned} \min \quad & \sum_{g\in\mathbb{G}} F_g(P_g) \\ \text{s.t.} \quad & \sum_{km} y_{km}(\theta_k - \theta_m) \;=\; P_k \quad \text{all } k \\ & L_k \;\le\; P_k \;\le\; U_k \quad \text{all } k, \quad |y_{km}(\theta_k - \theta_m)| \;\le\; U_{km}^{max} \quad \text{all } km \end{aligned}$$

## OPF using rectangular coordinates

$$V = \begin{pmatrix} V_k \\ V_m \end{pmatrix} = \begin{pmatrix} |V_k|e^{j\theta_k} \\ |V_m|e^{j\theta_m} \end{pmatrix} = \begin{pmatrix} e_k + jf_k \\ e_m + jf_m \end{pmatrix} \quad \text{(voltages at } k \text{ and } m\text{)}$$

$$I = \begin{pmatrix} I_{km} \\ I_{mk} \end{pmatrix} \quad \text{(complex current injections at } k \text{ and } m\text{)}$$

$$S = \begin{pmatrix} S_{km} \\ S_{mk} \end{pmatrix} = \begin{pmatrix} P_{km} + jQ_{km} \\ P_{mk} + jQ_{mk} \end{pmatrix} \quad \text{(complex power injections at } k \text{ and } m\text{)}$$

Then

$$S_{km} = V_k I_{km}^*, \quad S_{mk} = V_m I_{mk}^* \quad \text{and} \quad I = \mathbb{Y}V,$$

where

$$\mathbb{Y} = \begin{pmatrix} (y + \frac{y^{sh}}{2})\frac{1}{\tau^2} & -y\frac{1}{\tau e^{-j\sigma}} \\ -y\frac{1}{\tau e^{j\sigma}} & y + \frac{y^{sh}}{2} \end{pmatrix}.$$

## OPF using rectangular coordinates

$$V = \begin{pmatrix} V_k \\ V_m \end{pmatrix} = \begin{pmatrix} |V_k|e^{j\theta_k} \\ |V_m|e^{j\theta_m} \end{pmatrix} = \begin{pmatrix} e_k + jf_k \\ e_m + jf_m \end{pmatrix} \quad \text{(voltages at } k \text{ and } m\text{)}$$

$$I = \begin{pmatrix} I_{km} \\ I_{mk} \end{pmatrix} \quad \text{(complex current injections at } k \text{ and } m\text{)}$$

$$S = \begin{pmatrix} S_{km} \\ S_{mk} \end{pmatrix} = \begin{pmatrix} P_{km} + jQ_{km} \\ P_{mk} + jQ_{mk} \end{pmatrix} \quad \text{(complex power injections at } k \text{ and } m\text{)}$$

Then

$$S_{km} = V_k I_{km}^*, \quad S_{mk} = V_m I_{mk}^* \quad \text{and} \quad I = \mathbb{Y}V,$$

where

$$\mathbb{Y} = \begin{pmatrix} (y + \frac{y^{sh}}{2})\frac{1}{\tau^2} & -y\frac{1}{\tau e^{-j\sigma}} \\ -y\frac{1}{\tau e^{j\sigma}} & y + \frac{y^{sh}}{2} \end{pmatrix}.$$

→ $P_{km}$ and $Q_{km}$ are *bilinear* functions of $e_k, e_m, f_k, f_m$, e.g.

$$P_{km} = e_k g(e_k - e_m) - e_k b(f_k - f_m) + f_k g(f_k - f_m) + f_k b(e_k - e_m)$$

in the *no shunt, no transformer case.*

**OPF in rectangular coordinates, simple case**

Choose $|e_k|$ and $f_k$ for each bus $k$, so that

$$
\begin{aligned}
K^{OPF} \;=\; \min \quad & \sum_{g \in \mathbb{G}} F_g(P_g) \\
\text{s.t.} \quad & w^T A_k w \;=\; P_k, \qquad \text{all } k \\
& w^T B_k w \;=\; Q_k, \qquad \text{all } k \\
& \text{box constraints on } P_k,\, Q_k, \text{ for all } k \\
& V_k^{min} \;\le\; w^T M_k w \;\le\; V_k^{max} \qquad \text{all } k
\end{aligned}
$$

Here $w = (e_1, e_2, \ldots, e_n, f_1, f_2, \ldots, f_n)^T$.

## OPF in rectangular coordinates, II

$$
\begin{aligned}
K^{OPF} \; = \; \min \quad & w^T F w \\
\text{s.t.} \quad & L_k \; \leq \; w^T A^k w \; \leq \; U_k, \qquad k = 1, 2, \ldots m \\
& w \in \mathbb{R}^n.
\end{aligned}
$$

Here, $F \succeq 0$.

## OPF in rectangular coordinates, II

$$
\begin{aligned}
K^{OPF} \;=\; \min \quad & w^T F w \\
\text{s.t.} \quad & L_k \;\le\; w^T A^k w \;\le\; U_k, \qquad k = 1, 2, \ldots m \\
& w \in \mathbb{R}^n.
\end{aligned}
$$

Here, $F \succeq 0$. A **quadratically constrained, quadratic program**.

**OPF in rectangular coordinates, II**

$$\begin{aligned} K^{OPF} \; = \; \min \quad & w^T F w \\ \text{s.t.} \quad & L_k \; \leq \; w^T A^k w \; \leq \; U_k, \qquad k = 1, 2, \ldots m \\ & w \in \mathbb{R}^n. \end{aligned}$$

Here, $F \succeq 0$. A **quadratically constrained, quadratic program**.

Write $W \; = \; ww^T \in \mathbb{R}^{n \times n}$. Then $W \succeq 0$, rank 1. So:

$$\begin{aligned} K^{OPF} \; = \; \min \quad & \sum_{i,j} F_{ij} W_{ij} \\ \text{s.t.} \quad & L_k \; \leq \; \sum_{ij} A^k_{ij} W_{ij} \; \leq \; U_k, \qquad k = 1, 2, \ldots m \end{aligned}$$

A *linear* program ?

## OPF in rectangular coordinates, III

$$
\begin{aligned}
K^{OPF} \;=\; \min \quad & w^T F w \\
\text{s.t.} \quad & L_k \;\leq\; w^T A^k w \;\leq\; U_k, \qquad k = 1, 2, \ldots m \\
& w \in \mathbb{R}^n.
\end{aligned}
$$

Here, $F \succeq 0$. A **quadratically constrained, quadratic program**.

**Semidefinite relaxation:**

$$
\begin{aligned}
K^{sdp} \;=\; \min \quad & \sum_{i,j} F_{ij} W_{ij} \\
\text{s.t.} \quad & L_k \;\leq\; \sum_{ij} A^k_{ij} W_{ij} \;\leq\; U_k, \qquad k = 1, 2, \ldots m \\
& W \succeq 0.
\end{aligned}
$$

A relaxation $\rightarrow$ proves a *lower bound* on $K^{OPF}$.

## OPF in rectangular coordinates, III

$$
\begin{aligned}
K^{OPF} \ = \ \min \quad & w^T F w \\
\text{s.t.} \quad & L_k \ \leq \ w^T A^k w \ \leq \ U_k, \qquad k = 1, 2, \ldots m \\
& w \in \mathbb{R}^n.
\end{aligned}
$$

Here, $F \succeq 0$. A **quadratically constrained, quadratic program**.

**Semidefinite relaxation:**

$$
\begin{aligned}
K^{sdp} \ = \ \min \quad & \sum_{i,j} F_{ij} W_{ij} \\
\text{s.t.} \quad & L_k \ \leq \ \sum_{ij} A^k_{ij} W_{ij} \ \leq \ U_k, \qquad k = 1, 2, \ldots m \\
& W \succeq 0.
\end{aligned}
$$

A relaxation $\rightarrow$ proves a *lower bound* on $K^{OPF}$.

Relaxation is *exact*, i.e. $K^{sdp} = K^{OPF}$ if optimal $W$ has rank 1.

## Interesting developments

→ Lavaei and Low (many coauthors), 2010 –

- SDP relaxation is often exact (rank-1 solution)

## Interesting developments

$\longrightarrow$ Lavaei and Low (many coauthors), 2010 –

- SDP relaxation is often exact (rank-1 solution)
- May require perturbing resistances to small positive values

## Interesting developments

→ Lavaei and Low (many coauthors), 2010 –

- SDP relaxation is often exact (rank-1 solution)
- May require perturbing resistances to small positive values
- More nuanced view: the SDP solution is often of small rank

## Interesting developments

$\longrightarrow$ Lavaei and Low (many coauthors), 2010 –

- SDP relaxation is often exact (rank-1 solution)
- May require perturbing resistances to small positive values
- More nuanced view: the SDP solution is often of small rank
- But $K^{sdp} \approx K^{OPF}$, often! (?)
- Positive results may concern grids that are not under stress
- And how do we extract a good, feasible rank-1 solution from the SDP solution?

## Interesting developments

→ Lavaei and Low (many coauthors), 2010 –

- SDP relaxation is often exact (rank-1 solution)
- May require perturbing resistances to small positive values
- More nuanced view: the SDP solution is often of small rank
- But $K^{sdp} \approx K^{OPF}$, often! (?)
- Positive results may concern grids that are not under stress
- And how do we extract a good, feasible rank-1 solution from the SDP solution?
- Despite quibbles, this is a very interesting development!
- Many groups trying to "make it work"
- Practical challenges?

## Practical challenges?

- Solving large SDPs is **not** easy!

## Practical challenges?

- Solving large SDPs is **not** easy!
- Matrix completion theorem (Laurent): SDPs involving sparse constraints can be kept sparse – but involves a combinatorial computation
- Lavaei, Molzahn, others(?) (2014) – Polish grid (about 3000 buses) can be solved in 300 - 900 seconds using matrix completion.

## Practical challenges?

- Solving large SDPs is **not** easy!
- Matrix completion theorem (Laurent): SDPs involving sparse constraints can be kept sparse – but involves a combinatorial computation
- Lavaei, Molzahn, others(?) (2014) – Polish grid (about 3000 buses) can be solved in 300 - 900 seconds using matrix completion.
- And how about extracting a low rank near-optimal, feasible solution to OPF?
- And what do we do when there is a large *duality gap*:

$$K^{sdp} << K^{OPF} \quad ?$$

## Our current work

- Cutting-plane algorithms for computing *linear* approximations to *lifted* formulations to OPF
- Goal is to compute tight lower bounds, fast, with linear formulations
- Only linear (not conic) can be extended to handle important features, e.g. binary variables to model optional line switching or generator commitment

- Sample result: near optimal lower bound on Polish grid in $\approx 5$ seconds.

**Quadratically constrained, quadratic programming:**

$$\begin{array}{lll} \min & f_0(x) & \\ \text{s.t.} & f_i(x) \ \leq \ 0, & 1 \leq i \leq m \\ & x \in \mathbb{R}^n & \end{array}$$

Here,

$$f_i(x) \ = \ x^T M_i x + c_i^T x + d_i$$

is a general quadratic

Each $M_i$ is $n \times n$, wlog symmetric

# Folklore result: QCQP is NP-hard

## Folklore result: QCQP is NP-hard

Let $w_1, w_2, \ldots, w_n$ be **integers**, and consider:

$$\begin{aligned} W^* \doteq \min \; & -\sum_i x_i^2 \\ \text{s.t.} \; & \sum_i w_i\, x_i = 0, \\ & -1 \le x_i \le 1, \;\; 1 \le i \le n. \end{aligned}$$

## Folklore result: QCQP is NP-hard

Let $w_1, w_2, \ldots, w_n$ be **integers**, and consider:

$$\begin{aligned} W^* \doteq \min \; & -\sum_i x_i^2 \\ \text{s.t.} \; & \sum_i w_i \, x_i = 0, \\ & -1 \leq x_i \leq 1, \quad 1 \leq i \leq n. \end{aligned}$$

$\boldsymbol{W^* = -n}$, iff there exists a subset $J \subseteq \{1, \ldots, n\}$ with

$$\sum_{j \in J} w_j \; = \; \sum_{j \notin J} w_j$$

Take any $\{-1, 1\}$-linear program

Take any $\{-1,1\}$-linear program

$$\begin{aligned} \min \ & c^T x \\ \text{s.t. } & Ax = b \\ & x \in \{-1,1\}^n. \end{aligned}$$

Take any $\{-1,1\}$-linear program

$$\begin{aligned} \min \ & c^T x \\ \text{s.t.} \ & Ax = b \\ & x \in \{-1,1\}^n. \end{aligned}$$

$\rightarrow$

$$\begin{aligned} \min \ & c^T x \ - \ \boldsymbol{M} \sum_j x_j^2 \\ \text{s.t.} \ & Ax = b \\ & -1 \le x_j \le 1, \ \ 1 \le j \le n. \end{aligned}$$

(and many other similar transformations)

## Observation

Any instance of QCQP

$$\begin{array}{lll} \min & f_0(x) & \\ \text{s.t.} & f_i(x) \;\leq\; 0, & 1 \leq i \leq m \\ & x \in \mathbb{R}^n, & \end{array}$$

with a **fixed** number of distinct bilinear terms can be solved in polynomial time.

## Even more general

Solving systems of polynomial equations:

**Problem:** given polynomials $p_i : \mathbb{R}^n \to \mathbb{R}$, for $1 \leq i \leq m$
find $x \in \mathbb{R}^n$ s.t. $p_i(x) = 0, \forall\, i$

## Even more general

Solving systems of polynomial equations:

**Problem:** given polynomials $p_i : \mathbb{R}^n \to \mathbb{R}$, for $1 \leq i \leq m$
find $x \in \mathbb{R}^n$ s.t. $p_i(x) = 0, \forall\, i$

**Example:** find a root for $3v^6w - v^4 + 7 = 0$.

## Even more general

Solving systems of polynomial equations:

**Problem:** given polynomials $p_i : \mathbb{R}^n \to \mathbb{R}$, for $1 \leq i \leq m$
find $x \in \mathbb{R}^n$ s.t. $p_i(x) = 0, \forall\, i$

**Example:** find a root for $3v^6w - v^4 + 7 = 0$.

Equivalent to the system on variables $v, v_2, v_4, v_6, w, y$ and $c$:

$$
\begin{aligned}
c^2 &= 1 \\
v^2 - cv_2 &= 0 \\
v_2^2 - cv_4 &= 0 \\
v_2 v_4 - cv_6 &= 0 \\
v_6 w - cy &= 0 \\
3cy - cv_4 &= -7
\end{aligned}
$$

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns
be found **approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found **approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- On the average?
- Uniform algorithm?

## “Approximately”

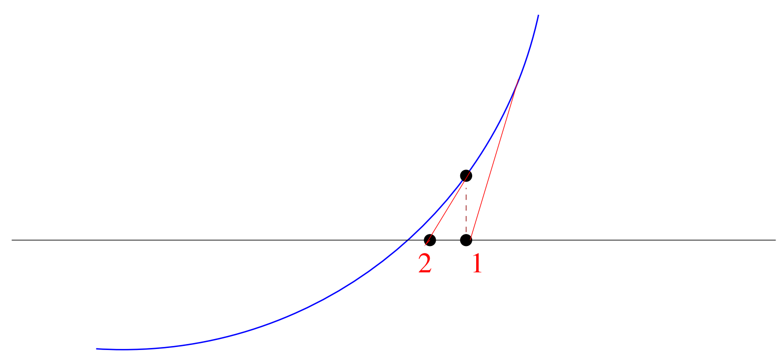
## “Approximately”

**Q:** How do practitioners ~~and other lesser folk~~ solve systems of nonlinear equations?

# “Approximately”

**Q:** How do practitioners ~~and other lesser folk~~ solve systems of nonlinear equations?

**A:** Newton-Raphson, of course!

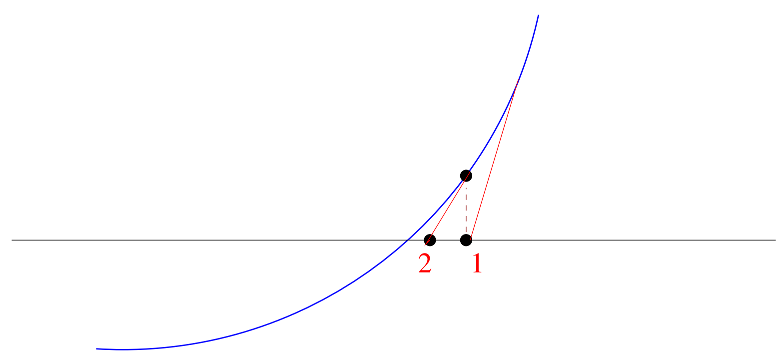



→ If we start near a solution, quadratic convergence

# “Approximately”

**Q:** How do practitioners ~~and other lesser folk~~ solve systems of nonlinear equations?

**A:** Newton-Raphson, of course!




→ If we start near a solution, quadratic convergence

**“Approximate”** solution to a system of polynomials:

a point in the region of quadratic convergence (to a solution)

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found
**approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- **On the average?**
- Uniform algorithm?

## “On the average” in polynomial time

A QCQP could be **quite** difficult!

e.g., a unique feasible solution, which additionally is an irrational vector

## “On the average” in polynomial time

A QCQP could be **quite** difficult!
e.g., a unique feasible solution, which additionally is an irrational vector

**but** a “nearby” problem instance could be much easier

## “On the average” in polynomial time

A QCQP could be **quite** difficult!
e.g., a unique feasible solution, which additionally is an irrational vector

**but** a “nearby” problem instance could be much easier

- View a problem as a vector in an appropriate space

## “On the average” in polynomial time

A QCQP could be **quite** difficult!
e.g., a unique feasible solution, which additionally is an irrational vector

**but** a “nearby” problem instance could be much easier

- View a problem as a vector in an appropriate space
- Endow that space with an appropriate metric
  (Bombieri-Weyl Hermitian product)

## **“On the average” in polynomial time**

A QCQP could be **quite** difficult!
e.g., a unique feasible solution, which additionally is an irrational vector

**but** a “nearby” problem instance could be much easier

- View a problem as a vector in an appropriate space
- Endow that space with an appropriate metric
  (Bombieri-Weyl Hermitian product)
- In that space, uniformly sample a ball (of appropriate radius) around a given problem

## “On the average” in polynomial time

A QCQP could be **quite** difficult!
e.g., a unique feasible solution, which additionally is an irrational vector

**but** a “nearby” problem instance could be much easier

- View a problem as a vector in an appropriate space
- Endow that space with an appropriate metric (Bombieri-Weyl Hermitian product)
- In that space, consider the set of problems given by a ball (of appropriate radius) around a given problem
- We want the algorithm to run in polynomial time, on average, in that ball

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found
**approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- On the average?
- **Uniform algorithm?** When is an algorithm non-uniform?

## Smale's $17^{th}$ problem

| |
|---|
| Can a zero of $n$ polynomial equations on $n$ unknowns be found<br>**approximately**,<br>**on the average** in polynomial time,<br>with a **uniform** algorithm? |

(but we are cheating)

- Approximately?
- On the average?
- **Uniform algorithm?** When is an algorithm non-uniform?

Blum, Shub, Smale (89), Blum, Cucker, Shub, Smale (98)

**First version:** A non-uniform algorithm specifies the existence of an algorithm *for each input size*.

As such, we cannot write a "program" that implements the algorithm.

It is more a proof of existence of an algorithm for each input size.

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found **approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- On the average?
- **Uniform algorithm?** When is an algorithm non-uniform?

Blum, Shub, Smale (89), Blum, Cucker, Shub, Smale (98)
Bürgisser, Cucker (2012)

**Second version:** A uniform algorithm

- allows operations over real numbers
- at unit cost per operation
- with infinite precision

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found
**approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- On the average?
- **Uniform algorithm?** When is an algorithm non-uniform?

Blum, Shub, Smale (89), Blum, Cucker, Shub, Smale (98)
Bürgisser, Cucker (2012)

**Second version:** A uniform algorithm

- allows operations over real numbers
- at unit cost per operation
- with infinite precision

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found
**approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Approximately?
- On the average?
- **Uniform algorithm?** When is an algorithm non-uniform?

Blum, Shub, Smale (89), Blum, Cucker, Shub, Smale (98)
Bürgisser, Cucker (2012)

**Second version:** A uniform algorithm

- allows operations over real numbers
- at unit cost per operation
- with infinite precision
- **Not!** the usual bit-model of computation

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found **approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Beltrán and Pardo (2009) – a randomized (Las Vegas) uniform algorithm that computes an approximate zero in *expected* polynomial time
- Bürgisser, Cucker (2012) – a deterministic $O(n^{\log \log n})$ (uniform) algorithm for computing approximate zeros
- **Techniques:** Homotopy (path-following method solving a sequence of problems), Newton's method

## Smale's $17^{th}$ problem

Can a zero of $n$ polynomial equations on $n$ unknowns be found **approximately**,
**on the average** in polynomial time,
with a **uniform** algorithm?

(but we are cheating)

- Beltrán and Pardo (2009) – a randomized (Las Vegas) uniform algorithm that computes an approximate zero in *expected* polynomial time
- Bürgisser, Cucker (2012) – a deterministic $O(n^{\log \log n})$ (uniform) algorithm for computing approximate zeros
- **Techniques:** Homotopy (path-following method solving a sequence of problems), Newton's method

**But we are cheating:** All of this is over $\mathbb{C}^n$, not $\mathbb{R}^n$

## Smale's $17^{th}$ problem

> Can a zero of $n$ polynomial equations on $n$ unknowns be found
> **approximately**,
> **on the average** in polynomial time,
> with a **uniform** algorithm?

(but we are cheating)

- Beltrán and Pardo (2009) – a randomized (Las Vegas) uniform algorithm that computes an approximate zero in *expected* polynomial time
- Bürgisser, Cucker (2012) – a deterministic $O(n^{\log \log n})$ (uniform) algorithm for computing approximate zeros
- **Techniques:** Homotopy (path-following method solving a sequence of problems), Newton's method

**But we are cheating:** All of this is over $\mathbb{C}^n$, not $\mathbb{R}^n$

**So what can be done over the reals?**

Take any $\{-1,1\}$-linear program

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \in \{-1,1\}^n. \end{aligned}$$

$\rightarrow$

$$\begin{aligned} \min \quad & c^T x \; - \; \boldsymbol{M} \sum_j x_j^2 \\ \text{s.t.} \quad & Ax = b \\ & -1 \le x_j \le 1, \quad 1 \le j \le n. \end{aligned}$$

- Fixed number of linear constraints?
- Fixed number of quadratic constraints?
- Non-convex quadratic constraints?

# The S-Lemma

Let $\boldsymbol{f}, \boldsymbol{g} : \mathbb{R}^{\boldsymbol{n}} \rightarrow \mathbb{R}$ be quadratic polynomials.

Suppose there exists $\bar{\boldsymbol{x}} \in \mathbb{R}^{\boldsymbol{n}}$ such that $\boldsymbol{g}(\bar{\boldsymbol{x}}) > \mathbf{0}$. Then

$$\boldsymbol{f}(\boldsymbol{x}) \geq \mathbf{0} \quad \text{whenever} \quad \boldsymbol{g}(\boldsymbol{x}) \geq \mathbf{0}$$

if and only if there exists $\boldsymbol{\gamma} \geq \mathbf{0}$ such that

$$\boldsymbol{f}(\boldsymbol{x}) \geq \boldsymbol{\gamma} \boldsymbol{g}(\boldsymbol{x}) \quad \text{for all} \quad \boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}.$$

Yakubovich (1971), also much earlier, related work

**Corollary:** Can solve

$$\min\{\boldsymbol{f}(\boldsymbol{x}) : \boldsymbol{g}(\boldsymbol{x}) \geq \mathbf{0}\}$$

in polynomial time (using semidefinite programming)

**Note:** duality may not hold if there is more than one quadratic constraint

# Special case: the trust-region subproblem

$$\min\{f(x) \; : \; g(x) \leq 0\}$$

can be solved in polynomial time, where $f$, $g$ quadratics, $g$ strictly convex

Scale, rotate, translate:

$$\min\{f(x) \; : \; \|x\| \leq 1\}$$

can be solved in poly time $\rightarrow$ $\log \epsilon^{-1}$

Y. Ye (1992) $\rightarrow$ $\log \log \epsilon^{-1}$

How about *extensions* of the trust-region subproblem?

## Sturm-Zhang (2003)

Where $f(x)$ is a quadratic,

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \|x\| \le 1 \\ & a^T x \le b \qquad (\textbf{one} \text{ linear side constraint}) \end{aligned}$$

can be solved in polynomial time, as can

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \|x\| \le 1 \\ & \|x - x^0\| \le r_0 \qquad (\textbf{one} \text{ additional convex ball constraint}) \end{aligned}$$

## Ye-Zhang (2003)

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & \|x\| \le 1 \\ & a_i^T x \le b_i \quad i = 1, 2 \\ & (a_1^T x - b_1)(a_2^T x - b_2) = 0 \end{aligned}$$

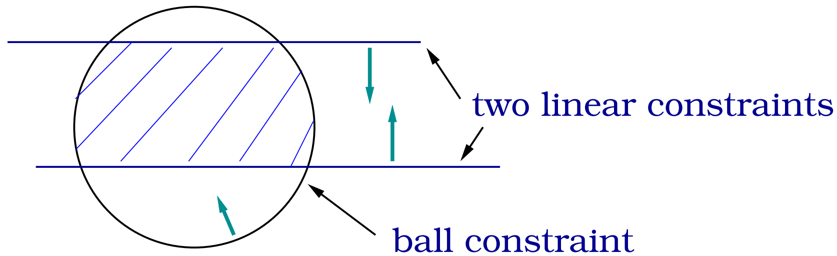
(two linear side constraints, but at least one binding)

# Anstreicher-Burer (2012)

In polynomial time, one can solve a problem of the form

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x\| \leq 1 \\ & a_i^T x \leq b_i \quad i = 1, 2 \end{aligned}$$

**provided** the two linear constraints are parallel:

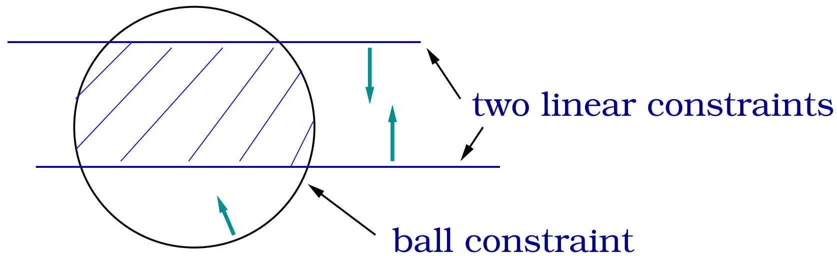


## Anstreicher-Burer (2012)

In polynomial time, one can solve a problem of the form

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x\| \le 1 \\ & a_i^T x \le b_i \quad i = 1, 2 \end{aligned}$$

**provided** the two linear constraints are parallel:



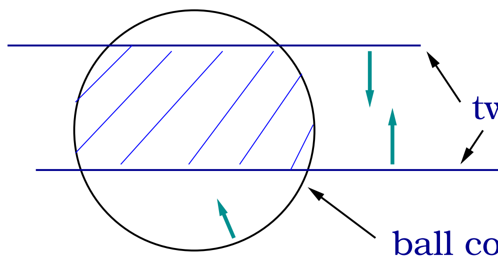

$$\rightarrow \min \{ x^T Q x + c^T x \, : \, l \le x_1 \le u, \; \|x\| \le 1 \}$$

# Anstreicher-Burer (2012)

In polynomial time, one can solve a problem of the form

$$\begin{aligned} \min \quad & x^TQx + c^Tx \\ \text{s.t.} \quad & \|x\| \le 1 \\ & a_i^Tx \le b_i \quad i = 1,2 \end{aligned}$$

**provided** the two linear constraints are parallel:




$$\rightarrow \min\{\, x^TQx + c^Tx \,:\, l \le x_1 \le u,\ \|x\| \le 1 \,\}$$

restate as:

$$\begin{aligned} \min \quad & \sum_{i,j} q_{ij}X_{ij} \ + \ c^Tx \\ \text{s.t.} \quad & X_{11} + lu \le (l+u)x_1 \\ & \|X_{.1} - lx\| \le x_1 - l \\ & \|ux - X_{.1}\| \le u - x_1 \\ & \sum_j X_{jj} \le 1 \quad , \quad X \succeq xx^T \end{aligned}$$

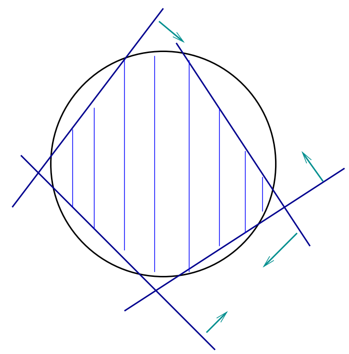
**Lemma:** This problem has an optimal solution with $X = xx^T$. Also: Ye-Zhang

## Burer-Yang (2012)

In polynomial time, one can solve a problem of the form

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x\| \leq 1 \\ & a_i^T x \leq b_i \quad 1 \leq i \leq m \end{aligned}$$

**if** no two linear inequalities are simultaneously binding in the feasible region

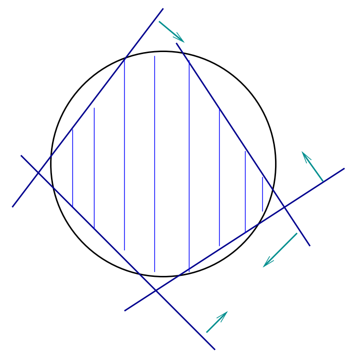


## Burer-Yang (2012)

In polynomial time, one can solve a problem of the form

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x\| \le 1 \\ & a_i^T x \le b_i \quad 1 \le i \le m \end{aligned}$$

**if** no two linear inequalities are simultaneously binding in the feasible region



**Lemma:** the following problem has an optimal solution with $X = xx^T$.

$$\begin{aligned} \min \quad & \sum_{i,j} q_{ij} X_{ij} \; + \; c^T x \\ \text{s.t.} \quad & X_{11} + lu \le (l+u)x_1 \\ & \|b_i x - X a_i\| \le b_i - a_i^T x \qquad i \le m \\ & b_i b_j - b_j a_i^T x - b_i a_j^T x + a_i^T X a_j \le 0 \qquad i < j \le m \\ & \sum_j X_{jj} \le 1 \;\; , \;\; X \succeq xx^T \end{aligned}$$

**This talk** (B. and Alex Michalka, SODA 2014)

$$
\begin{aligned}
\min \quad & x^T Q x + c^T x \\
\text{s.t.} \quad & \|x - \mu_h\| \le r_h, \quad h \in S, \\
& \|x - \mu_h\| \ge r_h, \quad h \in K, \\
& x \in P \doteq \{\, x \in \mathbb{R}^n \, : \, Ax \le b \,\}
\end{aligned}
$$

**Theorem.**
For each fixed $|S|$, $|K|$ can be solved in polynomial time if either

**(1)** $|S| \ge 1$ and polynomially large number of faces of $P$ intersect

$$\bigcap_{h \in S} \{x \in \mathbb{R}^n \, : \, \|x - \mu_h\| \le r_h\},$$

or

**(2)** $|S| = 0$ and the number of rows of $A$ is bounded.

**This talk** (B. and Alex Michalka, SODA 2014)

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x - \mu_h\| \leq r_h, \quad h \in S, \\ & \|x - \mu_h\| \geq r_h, \quad h \in K, \\ & x \in P \doteq \{\, x \in \mathbb{R}^n \,:\, Ax \leq b \,\} \end{aligned}$$

**Theorem.**

For each fixed $|S|$, $|K|$ can be solved in polynomial time if either

**(1)** $|S| \geq 1$ and polynomially large number of faces of $P$ intersecting

$$\bigcap_{h \in S} \{x \in \mathbb{R}^n \,:\, \|x - \mu_h\| \leq r_h\},$$

or

**(2)** $|S| = 0$ and the number of rows of $A$ is bounded.

Anstreicher-Burer: Case (1) with 3 faces of $P$ meeting the feasible region.

Burer-Yang: Case (1) with $m + 1$ faces of $P$ meeting the feasible region.

## More precise statement for case (1)

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x - \mu_h\| \leq r_h, \quad h \in S, \\ & \|x - \mu_h\| \geq r_h, \quad h \in K, \\ & x \in P \doteq \{ x \in \mathbb{R}^n \,:\, Ax \leq b \} \end{aligned}$$

**Theorem.**
For each fixed $|S| \geq 1$, $|K|$ there is an algorithm that solves the problem, to tolerance $0 < \epsilon < 1$ in time

**(a)** Polynomial in the number of bits in the data and $\log \epsilon^{-1}$

**(b)** Linear in the number of faces of $P$ that intersect

$$\bigcap_{h \in S} \{x \in \mathbb{R}^n \,:\, \|x - \mu_h\| \leq r_h\}.$$

Not hard **Lemma**

Given a collection of balls $B_h \subset \mathbb{R}^n$ $(h \in S)$

and a polyhedron

$$P \;=\; \{x \in \mathbb{R}^n \,:\, Ax \leq b\},$$

there is an algorithm that lists the faces of $P$ that intersect $\bigcap_{h\in S} B_h$

In time

**(a)** polynomial in the number of bits in the data

**(b)** linear in the number of intersecting faces

## Basic Idea

$$\min\{x^TQx+c^Tx \;:\; \|x-\mu_h\| \leq r_h,\ h \in S, \quad \|x-\mu_h\| \geq r_h,\ h \in K, \quad Ax \leq b\}$$

## Basic Idea

$$\min\{x^TQx+c^Tx\ :\ \|x-\mu_h\| \le r_h,\ h\in S,\quad \|x-\mu_h\| \ge r_h,\ h\in K,\quad Ax\le b\}$$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \le b}$, such that

$$\|x^*-\mu_h\| = r_h \quad \forall\, h\in S^= \cup K^=,\qquad a_i^Tx^* = b_i \quad \forall\, i\in I^=$$

$$\|x^*-\mu_h\| < r_h \quad \forall\, h\in S - S^=,\qquad \|x^*-\mu_h\| > r_h \quad \forall\, h\in K-K^=$$

$$a_i^Tx^* < b_i \quad \forall\, i\notin I^=.$$

## Basic Idea

$$\min\{x^TQx+c^Tx \ : \ \|x-\mu_h\| \le r_h,\ h \in S, \quad \|x-\mu_h\| \ge r_h,\ h \in K, \quad Ax \le b\}$$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \le b}$, such that

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$
$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$
$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

$\boldsymbol{(S^=,\ K^=,\ I^=)}$: an **optimal** triple.

## Basic Idea

$\min\{x^T Q x + c^T x \;:\; \|x - \mu_h\| \leq r_h,\ h \in S, \quad \|x - \mu_h\| \geq r_h,\ h \in K, \quad Ax \leq b\}$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \leq b}$, such that

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

$\boldsymbol{(S^=,\ K^=,\ I^=)}$: an **optimal** triple. $x^*$: **tight** for $(S^=,\ K^=,\ I^=)$

## Basic Idea

$\min\{x^TQx+c^Tx \,:\, \|x-\mu_h\| \le r_h,\ h \in S, \quad \|x-\mu_h\| \ge r_h,\ h \in K, \quad Ax \le b\}$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \le b}$, such that

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

$\boldsymbol{(S^=,\ K^=,\ I^=)}$: an **optimal** triple. $x^*$: **tight** for $(S^=,\ K^=,\ I^=)$

Algorithm will **guess** $\boldsymbol{(S^=,\ K^=,\ I^=)}$ (actually, **compute** $\boldsymbol{I^=}$).

## Basic Idea

$$\min\{x^TQx+c^Tx \;:\; \|x-\mu_h\| \leq r_h,\; h \in S, \quad \|x-\mu_h\| \geq r_h,\; h \in K, \quad Ax \leq b\}$$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \leq b}$, such that

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

$(\boldsymbol{S^=},\ \boldsymbol{K^=},\ \boldsymbol{I^=})$: an **optimal** triple. $x^*$: **tight** for $(S^=,\ K^=,\ I^=)$

Algorithm will **guess** $(\boldsymbol{S^=},\ \boldsymbol{K^=},\ \boldsymbol{I^=})$ (actually, **compute** $\boldsymbol{I^=}$).

For each enumerated triple $(\boldsymbol{\hat{S}}, \boldsymbol{\hat{K}}, \boldsymbol{\hat{I}})$, it will (in polynomial time) either

**(a)** Compute a finite set of vectors tight for $(\hat{S}, \hat{K}, \hat{I})$, one of which must be $\boldsymbol{x^*}$ if the guess is right, **or**

## Basic Idea

$$\min\{x^TQx+c^Tx \;:\; \|x-\mu_h\| \le r_h,\ h \in S, \quad \|x-\mu_h\| \ge r_h,\ h \in K, \quad Ax \le b\}$$

Let $\boldsymbol{x^*}$ be optimal. Trivial: there exist (possibly empty) subsets

$\boldsymbol{S^=}$ of $\boldsymbol{S}$, $\boldsymbol{K^=}$ of $\boldsymbol{K}$, and $\boldsymbol{I^=}$ of the rows of $\boldsymbol{Ax \le b}$, such that

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

$\boldsymbol{(S^=, K^=, I^=)}$: an **optimal** triple. $x^*$: **tight** for $(S^=, K^=, I^=)$

Algorithm will **guess** $\boldsymbol{(S^=, K^=, I^=)}$ (actually, **compute** $\boldsymbol{I^=}$).

For each enumerated triple $\boldsymbol{(\hat{S}, \hat{K}, \hat{I})}$, it will (in polynomial time) either

**(a)** Compute a finite set of vectors tight for $(\hat{S}, \hat{K}, \hat{I})$, one of which must be $\boldsymbol{x^*}$ if the guess is right, **or**

**(b)** Prove that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal, there is a different **optimal** triple $\boldsymbol{(\tilde{S}, \tilde{K}, \tilde{I})}$ with

$$\boldsymbol{\tilde{S} \supseteq \hat{S},\ \tilde{K} \supseteq \hat{K},\ \tilde{I} \supseteq \hat{K}} \quad \text{and} \quad |\boldsymbol{\tilde{S}}| + |\boldsymbol{\tilde{K}}| + |\boldsymbol{\tilde{I}}| > |\boldsymbol{\hat{S}}| + |\boldsymbol{\hat{K}}| + |\boldsymbol{\hat{I}}|.$$

## Geometry, 1

**Notation**. Given a ball $B = \{x \in \mathbb{R}^n \,:\, \|x - \hat{\mu}_i\| \leq \hat{r}\}$,

$$\boldsymbol{\partial B \doteq \{x \in \mathbb{R}^n \,:\, \|x - \hat{\mu}_i\| = \hat{r}\}}$$

## Geometry, 1

**Notation**. Given a ball $B = \{x \in \mathbb{R}^n : \|x - \hat{\mu}_i\| \leq \hat{r}\}$,

$$\boldsymbol{\partial B \doteq \{x \in \mathbb{R}^n : \|x - \hat{\mu}_i\| = \hat{r}\}}$$

**Lemma**. Let $B_i = \{x \in \mathbb{R}^n : \|x - \mu_i\| \leq r_i\}$, $i = 1, 2$, be **distinct** and **intersecting**.

## Geometry, 1

**Notation**. Given a ball $B = \{x \in \mathbb{R}^n \,:\, \|x - \hat{\mu}_i\| \leq \hat{r}\}$,

$$\boldsymbol{\partial B \;\doteq\; \{x \in \mathbb{R}^n \,:\, \|x - \hat{\mu}_i\| = \hat{r}\}}$$

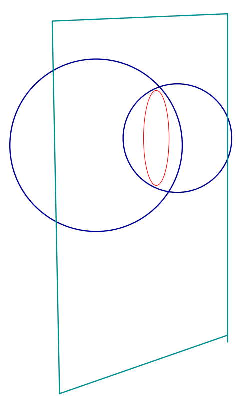
**Lemma**. Let $B_i = \{x \in \mathbb{R}^n \,:\, \|x - \mu_i\| \leq r_i\}$, $i = 1, 2$, be **distinct** and **intersecting**.

There exists an $(n-1)$-dim hyperplane $\boldsymbol{H}$, a point $\boldsymbol{v} \in \boldsymbol{H}$, and $\boldsymbol{r} \geq \mathbf{0}$ such that

$$\boldsymbol{\partial B_1 \cap \partial B_2 \;=\; \{x \in H \,:\, \|x - v\| = r\}}$$

and

$$\boldsymbol{\partial B_i \cap H \;=\; \{x \in H \,:\, \|x - v\| = r\}, \quad i = 1, 2}$$



## Geometry, 1

**Corollary** Given balls $B_i$, $i \in I$, not all equal, with

$$\bigcap_{i \in I} B_i \neq \emptyset,$$

there exists an $(n-t)$-dim hyperplane $\boldsymbol{H}$ ( $\boldsymbol{t} \geq \mathbf{1}$), $\boldsymbol{v} \in \boldsymbol{H}$ and $\boldsymbol{r} \geq \mathbf{0}$ s.t.

$$\bigcap_{i \in I} \boldsymbol{\partial B_i} = \{\boldsymbol{x} \in \boldsymbol{H} : \|\boldsymbol{x} - \boldsymbol{v}\| = \boldsymbol{r}\}$$

# Geometry, 1

**Corollary** Given balls $B_i$, $i \in I$, not all equal, with

$$\bigcap_{i \in I} B_i \neq \emptyset,$$

there exists an $(n-t)$-dim hyperplane $\boldsymbol{H}$ ( $\boldsymbol{t} \geq \mathbf{1}$), $\boldsymbol{v} \in \boldsymbol{H}$ and $\boldsymbol{r} \geq \mathbf{0}$ s.t.

$$\bigcap_{i \in I} \partial B_i \; = \; \{x \in H \; : \; \|x - v\| = r\}$$

**Implication:** When guessing an optimal triple $(S^=, K^=, I^=)$

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

# Geometry, 1

**Corollary** Given balls $B_i$, $i \in I$, not all equal, with

$$\bigcap_{i \in I} B_i \neq \emptyset,$$

there exists an $(n-t)$-dim hyperplane $\boldsymbol{H}$ ( $\boldsymbol{t} \geq \mathbf{1}$), $\boldsymbol{v} \in \boldsymbol{H}$ and $\boldsymbol{r} \geq \mathbf{0}$ s.t.

$$\bigcap_{i \in I} \partial \boldsymbol{B}_i = \{ \boldsymbol{x} \in \boldsymbol{H} \, : \, \| \boldsymbol{x} - \boldsymbol{v} \| = \boldsymbol{r} \}$$

**Implication:** When guessing an optimal triple $(\boldsymbol{S}^=, \boldsymbol{K}^=, \boldsymbol{I}^=)$

$$\|x^* - \mu_h\| = r_h \quad \forall\, h \in S^= \cup K^=, \quad a_i^T x^* = b_i \quad \forall\, i \in I^=$$

$$\|x^* - \mu_h\| < r_h \quad \forall\, h \in S - S^=, \quad \|x^* - \mu_h\| > r_h \quad \forall\, h \in K - K^=$$

$$a_i^T x^* < b_i \quad \forall\, i \notin I^=.$$

we

**(1)** Restrict to a lower dimensional space

**(2)** Obtain a single, binding, ball constraint

## The original problem:

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \mu_h\| &\le r_h, \quad h \in S, \\ \|x - \mu_h\| &\ge r_h, \quad h \in K, \\ a_i^T x &\le b_i, \quad i \in I \end{aligned}$$

## The original problem:

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \mu_h\| &\leq r_h, \quad h \in S, \\ \|x - \mu_h\| &\geq r_h, \quad h \in K, \\ a_i^T x &\leq b_i, \quad i \in I \end{aligned}$$

**Given a guess, this becomes** (ignoring the non-binding constraints):

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \hat{\mu}\| &= \hat{r}, \\ x &\in H \end{aligned}$$

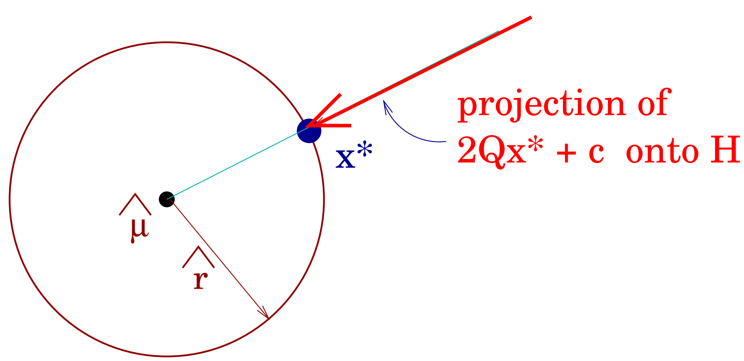
**The original problem:**

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \mu_h\| \ & \le \ r_h, \quad h \in S, \\ \|x - \mu_h\| \ & \ge \ r_h, \quad h \in K, \\ a_i^T x \ & \le \ b_i, \quad i \in I \end{aligned}$$

**Given a guess, this becomes** (ignoring the non-binding constraints):

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \hat{\mu}\| \ & = \ \hat{r}, \\ x \ & \in \ H \end{aligned}$$

**Almost correct:** first-order condition restricted to $H$



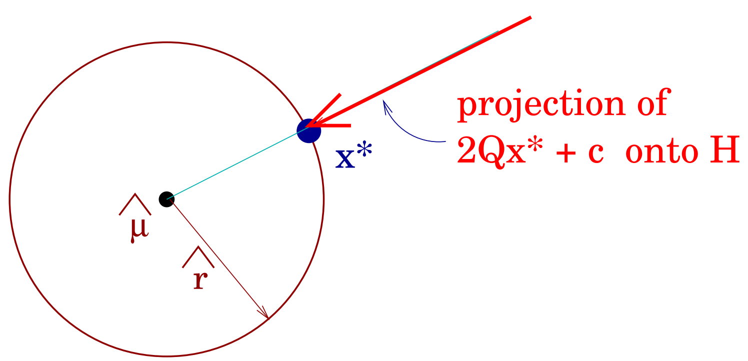
## The original problem:

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \mu_h\| &\leq r_h, \quad h \in S, \\ \|x - \mu_h\| &\geq r_h, \quad h \in K, \\ a_i^T x &\leq b_i, \quad i \in I \end{aligned}$$

**Given a guess, this becomes** (ignoring the non-binding constraints):

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ s.t. \quad \|x - \hat{\mu}\| &= \hat{r}, \\ x &\in H \end{aligned}$$

**Almost correct:** first-order condition restricted to $H$




**Better:** Use projected quadratic representation

**Theorem** (abridged).

Given a triple $\boldsymbol{(\hat{S}, \hat{K}, \hat{I})}$ there is polynomially computable list of points $\boldsymbol{x^j}$, $\boldsymbol{(j \in J)}$ tight for the triple, such that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal, then either

**Theorem** (abridged).

Given a triple $(\hat{\boldsymbol{S}}, \hat{\boldsymbol{K}}, \hat{\boldsymbol{I}})$ there is polynomially computable list of points $\boldsymbol{x^j}$, $(\boldsymbol{j \in J})$ tight for the triple, such that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal,
then either

**(1)** $\boldsymbol{x^* = x^j}$ for some $j \in J$, **or**

**Theorem** (abridged).

Given a triple $(\hat{\boldsymbol{S}}, \hat{\boldsymbol{K}}, \hat{\boldsymbol{I}})$ there is polynomially computable list of points $\boldsymbol{x^j}$, $(\boldsymbol{j} \in \boldsymbol{J})$ tight for the triple, such that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal, then either

**(1)** $\boldsymbol{x^*} = \boldsymbol{x^j}$ for some $j \in J$, **or**

**(2)** There exists *infeasible* $\boldsymbol{y}$ and a Jordan curve $\boldsymbol{\Theta}$ joining $\boldsymbol{y}$ and $\boldsymbol{x^*}$, s.t.

$$\boldsymbol{z^T Q z} + \boldsymbol{c^T z} = \boldsymbol{x^{*T} Q x^*} + \boldsymbol{c^T x^*} \quad \forall\, \boldsymbol{z} \in \boldsymbol{\Theta}$$

$$\boldsymbol{z} \ \textbf{tight for} \ (\hat{\boldsymbol{S}}, \hat{\boldsymbol{K}}, \hat{\boldsymbol{I}}) \quad \forall\, \boldsymbol{z} \in \boldsymbol{\Theta}$$

**Theorem** (abridged).

Given a triple $(\hat{\boldsymbol{S}}, \hat{\boldsymbol{K}}, \hat{\boldsymbol{I}})$ there is polynomially computable list of points $\boldsymbol{x^j}$, $(\boldsymbol{j} \in \boldsymbol{J})$ tight for the triple, such that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal, then either

**(1)** $\boldsymbol{x^*} = \boldsymbol{x^j}$ for some $j \in J$, **or**

**(2)** There exists *infeasible* $\boldsymbol{y}$ and a Jordan curve $\boldsymbol{\Theta}$ joining $\boldsymbol{y}$ and $\boldsymbol{x}^*$, s.t.

$$\boldsymbol{z^T Q z} + \boldsymbol{c^T z} = \boldsymbol{x^{*T} Q x^*} + \boldsymbol{c^T x^*} \quad \forall\, \boldsymbol{z} \in \boldsymbol{\Theta}$$

$$\boldsymbol{z} \;\; \textbf{tight for} \;\; (\hat{\boldsymbol{S}}, \hat{\boldsymbol{K}}, \hat{\boldsymbol{I}}) \quad \forall\, \boldsymbol{z} \in \boldsymbol{\Theta}$$

**Implication:** In case (2), there is a different **optimal** triple $(\tilde{\boldsymbol{S}}, \tilde{\boldsymbol{K}}, \tilde{\boldsymbol{I}})$ with

$\tilde{\boldsymbol{S}} \supseteq \hat{\boldsymbol{S}}, \; \tilde{\boldsymbol{K}} \supseteq \hat{\boldsymbol{K}}, \; \tilde{\boldsymbol{I}} \supseteq \hat{\boldsymbol{K}}$ and $|\tilde{\boldsymbol{S}}| + |\tilde{\boldsymbol{K}}| + |\tilde{\boldsymbol{I}}| > |\hat{\boldsymbol{S}}| + |\hat{\boldsymbol{K}}| + |\hat{\boldsymbol{I}}|$.

**Theorem** (abridged).

Given a triple $(\hat{S}, \hat{K}, \hat{I})$ there is polynomially computable list of points $x^j$, $(j \in J)$ tight for the triple, such that if $(\hat{S}, \hat{K}, \hat{I})$ is optimal, then either

**(1)** $x^* = x^j$ for some $j \in J$, **or**

**(2)** There exists *infeasible* $y$ and a Jordan curve $\Theta$ joining $y$ and $x^*$, s.t.

$$z^T Q z + c^T z = x^{*T} Q x^* + c^T x^* \quad \forall z \in \Theta$$

$$z \text{ tight for } (\hat{S}, \hat{K}, \hat{I}) \quad \forall z \in \Theta$$

**Implication:** In case (2), there is a different **optimal** triple $(\tilde{S}, \tilde{K}, \tilde{I})$ with

$\tilde{S} \supseteq \hat{S}$, $\tilde{K} \supseteq \hat{K}$, $\tilde{I} \supseteq \hat{K}$ and $|\tilde{S}| + |\tilde{K}| + |\tilde{I}| > |\hat{S}| + |\hat{K}| + |\hat{I}|$.

**The trust-region subproblem:**.

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x - \mu\| \le r \end{aligned}$$

**The trust-region subproblem:**.

$$\begin{aligned} \min \quad & x^T Q x + c^T x \\ \text{s.t.} \quad & \|x - \mu\| \leq r \end{aligned}$$

**Generalization: CDT (Celis-Dennis-Tapia) problem**

$$\begin{aligned} \min \quad & x^T Q_0 x + c_0^T x \\ \text{s.t.} \quad & x^T Q_1 x + c_1^T x + d_1 \ \leq \ 0 \\ & x^T Q_2 x + c_2^T x + d_2 \ \leq \ 0 \end{aligned}$$

where $\boldsymbol{Q_1 \succ 0}$, $\boldsymbol{Q_2 \succ 0}$

## Even more general

Barvinok (STOC 1992):

For each fixed $\boldsymbol{p} \geq \mathbf{1}$, there is a polynomial-time algorithm for deciding feasibility of a system

$$\begin{aligned} x^T M_i x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

where the $M_i$ are general matrices.

## Even more general

Barvinok (STOC 1992):

For each fixed $\boldsymbol{p \geq 1}$, there is a polynomial-time algorithm for deciding feasibility of a system

$$\begin{aligned} x^T M_i\, x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

where the $M_i$ are general matrices.

- **Non-constructive.** Algorithm says "yes" or "no."
- **Computational model?** Uniform algorithm? "Real-RAM"?

## A (better?) alternative: $\epsilon$-feasibility

For each fixed $\boldsymbol{p \geq 1}$, given a system

$$\begin{aligned} x^T M_i x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

and given $\boldsymbol{0 < \epsilon < 1}$, either

- **Prove** that the system is **infeasible**, or

- **Output** $\boldsymbol{\hat{x} \in \mathbb{R}^n}$ with

$$\begin{aligned} -\epsilon \leq x^T M_i \leq \epsilon, \quad 1 \leq i \leq p, \\ 1 - \epsilon \leq \|\hat{x}\| \leq 1 + \epsilon, \end{aligned}$$

in time polynomial in the data and in $\log \epsilon^{-1}$.

## A (better?) alternative: $\epsilon$-feasibility

For each fixed $\boldsymbol{p \geq 1}$, given a system

$$\begin{aligned} x^T M_i x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

and given $\mathbf{0} < \boldsymbol{\epsilon} < \mathbf{1}$, either

- **Prove** that the system is **infeasible**, or
- **Output** $\hat{\boldsymbol{x}} \in \mathbb{R}^{\boldsymbol{n}}$ with

$$\begin{aligned} -\epsilon \leq x^T M_i \leq \epsilon, &\quad 1 \leq i \leq p, \\ 1 - \epsilon \leq \|\hat{x}\| \leq 1 + \epsilon, \end{aligned}$$

in time polynomial in the data and in $\log \epsilon^{-1}$.

**Two issues:** Constructiveness, and $\epsilon$-feasibility

## Modification to Barvinok's result

Assume that for each fixed $\boldsymbol{p \geq 1}$, there is an algorithm that given a system

$$\begin{aligned} x^T M_i x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

and given $\boldsymbol{0 < \epsilon < 1}$, either

- **Proves** that the system is **infeasible**, or

- **Proves** that is $\boldsymbol{\epsilon}$-feasible,

in time polynomial in the data and in $\log \epsilon^{-1}$.

(so still nonconstructive)

## Modification to Barvinok's result

Assume that for each fixed $\boldsymbol{p \geq 1}$, there is an algorithm that given a system

$$\begin{aligned} x^T M_i x &= 0, \quad 1 \leq i \leq p, \\ \|x\| &= 1, \quad x \in \mathbb{R}^n \end{aligned}$$

and given $\boldsymbol{0 < \epsilon < 1}$, either

- **Proves** that the system is **infeasible**, or

- **Proves** that is $\boldsymbol{\epsilon}$-feasible,

in time polynomial in the data and in $\log \epsilon^{-1}$.

(so still nonconstructive)

**Assuming such an algorithm exists ...**

**Theorem** (2014).

Assume that an algorithm for $\boldsymbol{\epsilon}$-feasibility as indicated above exists.

**Theorem** (2014).

Assume that an algorithm for $\boldsymbol{\epsilon}$-feasibility as indicated above exists.

For each fixed $m \geq 1$ there is a polynomial-time algorithm that, given an optimization problem

$$\begin{aligned} \mathbf{min} \quad & \boldsymbol{f_0(x)} \doteq \boldsymbol{x^T Q_0 x + c_0^T x} \\ \text{s.t.} \quad & \boldsymbol{x^T Q_i x + c_i^T x + d_i} \leq \mathbf{0} \quad \mathbf{1} \leq \boldsymbol{i} \leq \boldsymbol{m}, \end{aligned}$$

where $\boldsymbol{Q_1} \succ \mathbf{0}$, and $\mathbf{0} < \boldsymbol{\epsilon} < \mathbf{1}$, either

**(1)** proves that the problem is infeasible,

**or**

**(2)** computes an $\boldsymbol{\epsilon}$-feasible vector $\boldsymbol{\hat{x}}$ such that there exists no feasible $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}$ with $\boldsymbol{f_0(x)} < \boldsymbol{f(\hat{x})} - \boldsymbol{\epsilon}$.

The complexity of the algorithm is polynomial in the number of bits in the data and in $\mathbf{log}\, \boldsymbol{\epsilon}^{-1}$

**Mathematical Programming C**

**Mathematical Programming C**

The best optimization journal !