The conic advantage in MINLP

MINLP Virtual Workshop 2021, June 28-29

Henrik A. Friberg

www.mosek.com

LP NLP

$$\text{LP} \subseteq \text{Conic} \subseteq \text{NLP}$$

$$\text{LP} \subseteq \text{Conic} \subseteq \text{NLP}$$

- Lagrangian duality theory can be used to derive conic optimization as the natural extension.

$$\text{LP} \subseteq \text{Conic} \subseteq \text{NLP}$$

- Best practices in MINLP leads to conic optimization.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g_i(x) \leq 0, & \forall i = 1, \ldots, k, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g_i(x) \leq 0, & \forall i = 1, \ldots, k, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

1. Migrate to a linear objective function

From Belotti, Kirches, Leyffer, Linderoth, Luedtke and Mahajan[1]:

$$\min\ x_1 + x_2^2 \quad \longrightarrow \quad \min\ x_1 + t, \quad t \geq x_2^2$$

👍 Avoid interior solutions (which cannot be separated by cuts).

[1] Belotti, Kirches, Leyffer, Linderoth, Luedtke and Mahajan (2013): Mixed-integer nonlinear optimization, Acta Numerica, 22.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & g_i(x) \le 0, & \forall i = 1, \dots, k, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

1. Migrate to a linear objective function

From Belotti, Kirches, Leyffer, Linderoth, Luedtke and Mahajan[1]:

$$\min\ x_1 + x_2^2 \quad \longrightarrow \quad \min\ x_1 + t, \quad t \ge x_2^2$$

Avoid interior solutions (which cannot be separated by cuts).

[1] Belotti, Kirches, Leyffer, Linderoth, Luedtke and Mahajan (2013): Mixed-integer nonlinear optimization, Acta Numerica, 22.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & g_i(x) \le 0, & \forall i = 1, \ldots, k, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

2. Distinguish convexity

- Avoid unnecessary spatial branching.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & g_i(x) \le 0, & \forall i = 1, \ldots, k, \\ & g_i^{\mathrm{c}}(x) \le 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

2. Distinguish convexity

Avoid unnecessary spatial branching.

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & g_i(x) \le 0, & \forall i = 1, \dots, k, \\
& g_i^c(x) \le 0, & \forall i = 1, \dots, k^c, \\
& x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

3. Implement your model using built-in atoms

Lack of callback functions imply:

- Easier to debug the MINLP solver.
- No callback overhead.
- No fear of ill-defined boundary points. For example,
  - The value of $\log(x)$ near $x = 0$.
  - The gradient of $\|x\|_2$ near $x = 0$.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & g_i(x) \leq 0, & \forall i = 1, \dots, k, \\ & g_i^c(x) \leq 0, & \forall i = 1, \dots, k^c, \\ & x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

3. Implement your model using built-in atoms

Possible atoms:

$$t \leq \log(x),$$

(hypograph of logarithm),

$$t \geq \sqrt{x_1^2 + x_2^2},$$

(epigraph of 2-norm).

Use of such built-in atoms imply:

- 👍 Solvers can specialize for stability and high performance.
- 👎 You are limited to a predefined set of atoms.

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
 & g_i(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k, \\
 & g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
 & l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

3. Implement your model using built-in atoms

Possible atoms:

$$t \leq \log(x),$$
(hypograph of logarithm),

$$t \geq \sqrt{x_1^2 + x_2^2},$$
(epigraph of 2-norm).

Use of such built-in atoms imply:

- 👍 Solvers can specialize for stability and high performance.
- 👎 You are limited to a predefined set of atoms.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\ & l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

3. Implement your model using built-in atoms

Possible atoms:

$$t \leq \log(x),$$
(hypograph of logarithm),

$$t \geq \sqrt{x_1^2 + x_2^2},$$
(epigraph of 2-norm).

Use of such built-in atoms imply:

- 👍 Solvers can specialize for stability and high performance.
- 👎 You are limited to a predefined set of atoms.

Best practices for MINLP

$$
\begin{aligned}
\text{minimize} \quad & c^T x \\
\text{subject to} \quad & Ax = b, \\
& g_i(x^{(i)}) \le 0, && \forall i = 1, \ldots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, && \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \le x \le u,\ x_j \in \mathbb{Z}, && \forall j \in \mathcal{I}.
\end{aligned}
$$

3. Implement your model using built-in atoms

To summarize, the set of atoms should be

- Numerically stable,
- Distinguish convexity,
- Versatile (so they can cover a wide range of applications).

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

- Ceria and Soares[1] characterized the closed convex hull of the union of convex sets using perspective transformation.
  - E.g., to be used on a disjunction $[x \leq 0] \vee [x \geq 1]$.

[1] Ceria and Soares (1999): Convex programming for disjunctive optimization, Mathematical Programming, 86.

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\
& l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

- Günlük and Linderoth[2] tightened a bunch of common sets with perspective transformation and promoted it as a useful tool for MINLPs with binary on-off relationships.

[2] Günlük and Linderoth (2012): Perspective reformulation and applications, Mixed Integer Nonlinear Programming, 154.

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \le 0, & \forall i = 1, \ldots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\ & l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \ge 0$:

$$t \le \log(x),$$

(hypograph of logarithm),

$$t \ge \sqrt{x_1^2 + x_2^2},$$

(epigraph of 2-norm),

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\ & l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \geq 0$:

$$t \leq \log(x),$$
(hypograph of logarithm),

$$t \geq \sqrt{x_1^2 + x_2^2},$$
(epigraph of 2-norm),

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\ & l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \ge 0$:

$$t/s \le \log(x/s),$$
(hypograph of logarithm),

$$t \ge \sqrt{x_1^2 + x_2^2},$$
(epigraph of 2-norm),

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \leq 0, & \forall i = 1, \dots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\
& l \leq x \leq u, \; x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \geq 0$:

$$t \leq s \log(x/s),$$

(exponential cone),

$$t \geq \sqrt{x_1^2 + x_2^2},$$

(epigraph of 2-norm),

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\
& l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \ge 0$:

$$t \le s \log(x/s),$$

(exponential cone),

$$t \ge \sqrt{x_1^2 + x_2^2},$$

(epigraph of 2-norm),

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \le 0, & \forall i = 1, \ldots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \le x \le u, \ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \ge 0$:

$$t \le s \log(x/s),$$
(exponential cone),

$$t \ge \sqrt{x_1^2 + x_2^2},$$
(quadratic cone),

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\ & l \le x \le u, \ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Support $\tilde{g}(x, s) = g(x/s)$ for $s \ge 0$:

$$t \le s \log(x/s),$$

(exponential cone),

$$t \ge \sqrt{x_1^2 + x_2^2},$$

(quadratic cone),

Did you notice the abuse of notation?

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \le 0, & \forall i = 1, \ldots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \le s\, g(x/s),\ s \ge 0$$

we write:

$$t \le s\, g(x/s),\ s > 0$$

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \leq 0, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\ & l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \leq s\, g(x/s),\ \ s \geq 0$$

we write:

$$\{(t, s, x)\ :\ t \leq s\, g(x/s),\ \ s > 0\}$$

Best practices for MINLP

$$\begin{array}{lll} \text{minimize} & c^T x & \\ \text{subject to} & Ax = b, & \\ & g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\ & g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\ & l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}. \end{array}$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \le s\, g(x/s),\ s \ge 0$$

we write:

$$\mathrm{cl}\,\{(t, s, x)\ :\ t \le s\, g(x/s),\ s > 0\}$$

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& g_i(x^{(i)}) \le 0, & \forall i = 1, \dots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, & \forall i = 1, \dots, k^{\mathrm{c}}, \\
& l \le x \le u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \le s\, g(x/s),\ s \ge 0$$

we write:

$$\mathcal{K}_g = \operatorname{cl}\{(t, s, x)\ :\ t \le s\, g(x/s),\ s > 0\}$$

Best practices for MINLP

$$
\begin{aligned}
\text{minimize} \quad & c^T x \\
\text{subject to} \quad & Ax = b, \\
& g_i(x^{(i)}) \le 0, && \forall i = 1, \dots, k, \\
& g_i^{\mathrm{c}}(x^{(i)}) \le 0, && \forall i = 1, \dots, k^{\mathrm{c}}, \\
& l \le x \le u, \; x_j \in \mathbb{Z}, && \forall j \in \mathcal{I}.
\end{aligned}
$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \le s\, g(x/s), \; s \ge 0$$

we write:

$$\mathcal{K}_g = \mathrm{cl}\left\{(t, s, x) \;:\; t \le s\, g(x/s), \; s > 0\right\}$$

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& x^{(i)} \in \mathcal{K}_i, & \forall i = 1, \ldots, k, \\
& x^{(i)} \in \mathcal{K}_i^{\mathrm{c}}, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \leq x \leq u, \; x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

4. Implement solvers to support perspective transformation

Hence, rather than:

$$t \leq s\, g(x/s), \; s \geq 0$$

we write:

$$\mathcal{K}_g = \mathrm{cl}\,\{(t, s, x) \; : \; t \leq s\, g(x/s), \; s > 0\}$$

Best practices for MINLP

$$
\begin{array}{lll}
\text{minimize} & c^T x & \\
\text{subject to} & Ax = b, & \\
& x^{(i)} \in \mathcal{K}_i, & \forall i = 1, \ldots, k, \\
& x^{(i)} \in \mathcal{K}_i^{\mathrm{c}}, & \forall i = 1, \ldots, k^{\mathrm{c}}, \\
& l \leq x \leq u,\ x_j \in \mathbb{Z}, & \forall j \in \mathcal{I}.
\end{array}
$$

Journey complete!

1. Migrate to a linear objective function.
2. Distinguish convexity.
3. Implement your model using built-in atoms.
4. Implement solvers to support perspective transformation.

$\Longrightarrow$ **Mixed-integer general conic optimization.**

Resources for mixed-integer conic optimization

- Günlük and Linderoth (2012): Perspective reformulation and applications, Mixed Integer Nonlinear Programming, 154.
- Lodi, Tanneau and Vielma (2020): Disjunctive cuts for Mixed-Integer Conic Optimization.
- Belotti, Goez, Polik, Ralphs and Terlaky (2017): A complete characterization of disjunctive conic cuts for mixed integer second order cone optimization, Discrete Optimization, 24.
- Shahabsafa, Goez and Terlaky (2018): On pathological disjunctions and redundant disjunctive conic cuts. Operations Research Letters, 46(5).
- Coey, Lubin, Vielma (2020): Outer approximation with conic certificates for mixed-integer convex problems, Mathematical Programming Computation, 12.

Resources for mixed-integer conic optimization

- Günlük and Linderoth (2012): Perspective reformulation and applications, Mixed Integer Nonlinear Programming, 154.
- Lodi, Tanneau and Vielma (2020): Disjunctive cuts for Mixed-Integer Conic Optimization.
- Belotti, Goez, Polik, Ralphs and Terlaky (2017): A complete characterization of disjunctive conic cuts for mixed integer second order cone optimization, Discrete Optimization, 24.
- Shahabsafa, Goez and Terlaky (2018): On pathological disjunctions and redundant disjunctive conic cuts. Operations Research Letters, 46(5).
- Coey, Lubin, Vielma (2020): Outer approximation with conic certificates for mixed-integer convex problems, Mathematical Programming Computation, 12.

You are limited to a predefined set of atoms.

Important questions

- What cones to support?
- How to decompose nonlinear constraints into conic atoms?

What cones to support?

- Numerically stable + distinguish convexity + versatile.
- Support perspective transformation

What cones to support?

- Numerically stable + distinguish convexity + versatile.
- ~~Support perspective transformation~~ (trivial).

What cones to support?

- Numerically stable + distinguish convexity + versatile.
- ~~Support perspective transformation~~ (trivial).

Idea: **Convex + low degree of nonlinearity (i.e., quadratic-like), so Newton-type methods are expected to solve the relaxations faster.**

What cones to support?

- Numerically stable + distinguish convexity + versatile.
- ~~Support perspective transformation~~ (trivial).

Idea: **Convex + low degree of nonlinearity (i.e., quadratic-like), so Newton-type methods are expected to solve the relaxations faster.**

Seminal work by Nesterov and Todd[3]

- Coined a property called "self-scaled".
- Such cones achieve the best known global convergence rate among Newton-type methods.
- Characterized as the set of symmetric cones.

[3] Nesterov and Todd (1997): Self-Scaled Barriers and Interior-Point Methods for Convex Programming, Mathematics of Operations Research, 22.

What cones to support?
The MOSEK selection

3 nonlinear symmetric cones:

- Quadratic: $\{(t,x)\ :\ t \geq \|x\|_2\}$.
- Rotated quadratic: $\{(t,s,x)\ :\ 2ts \geq \|x\|_2^2,\ t \geq 0,\ s \geq 0\}$.
- Semidefinite: $X \succeq 0$.

2 nonsymmetric cones:

- Exponential: $\mathrm{cl}\{(t,s,x)\ :\ t \geq s\exp(x/s),\ s > 0\}$.
- Power: $\{(t,s,x)\ :\ t^{\alpha}s^{1-\alpha} \geq \|x\|_2,\ t \geq 0,\ s \geq 0\}$ for any parameter $0 < \alpha < 1$.

Observation:

- **Almost** all convex optimization problems appearing in practice can be formulated using those 5 cones.

What cones to support?

The MOSEK selection

3 nonlinear symmetric cones:

- Quadratic: $\{(t, x) \; : \; t \geq \|x\|_2\}$.
- Rotated quadratic: $\{(t, s, x) \; : \; 2ts \geq \|x\|_2^2, \; t \geq 0, \; s \geq 0\}$.
- Semidefinite: $X \succeq 0$.

2 nonsymmetric cones:

- Exponential: $\mathrm{cl}\{(t, s, x) \; : \; t \geq s \exp(x/s), \; s > 0\}$.
- Power: $\{(t, s, x) \; : \; t^\alpha s^{1-\alpha} \geq \|x\|_2, \; t \geq 0, \; s \geq 0\}$ for any parameter $0 < \alpha < 1$.

Observation:

- Lubin et. al.[4]: all convex instances in MINLPLIB 2.0 (i.e., 333) are representable using these cones.

---

[4] Lubin, Yamangil, Bent and Vielma (2018): Polyhedral approximation in mixed-integer convex optimization, Mathematical Programming, 172(1).

How to decompose nonlinear constraints?

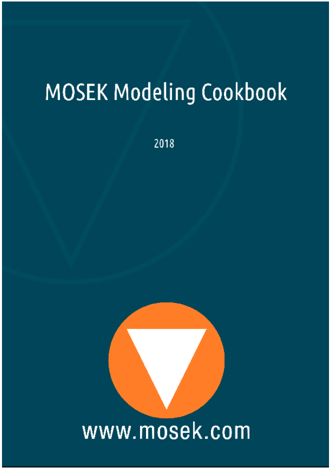
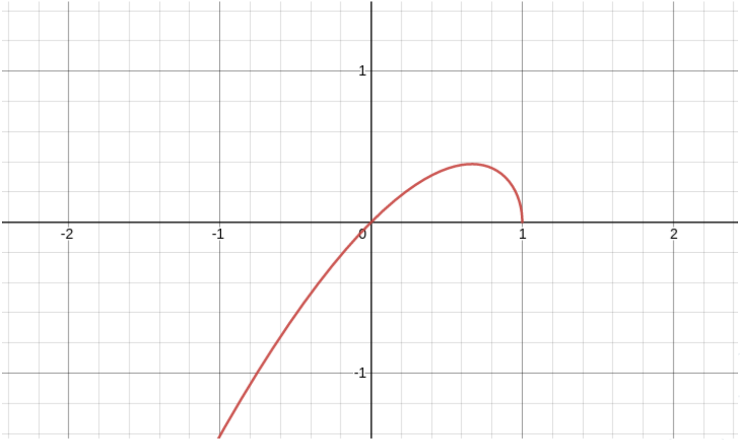


Figure: The MOSEK modeling cookbook (`www.mosek.com/documentation`)

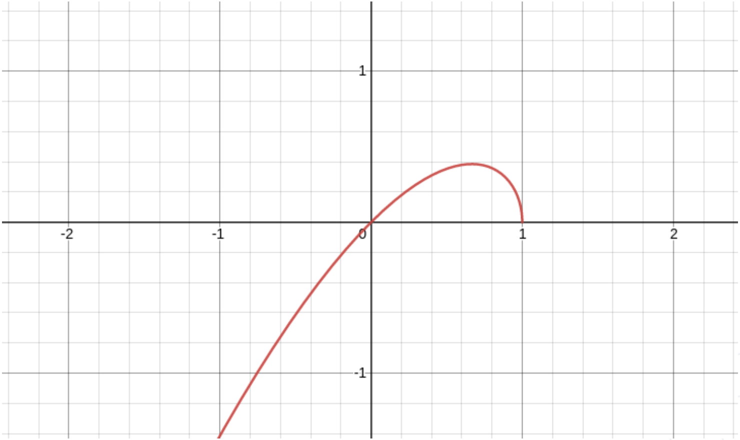
How to decompose nonlinear constraints?

1. $t \leq x\sqrt{1-x}$



How to decompose nonlinear constraints?

1. $t \leq x\sqrt{1-x} = (x) \cdot (\sqrt{1-x})$.



How to decompose nonlinear constraints?

1. $t \leq x\sqrt{1-x} = (x) \cdot (\sqrt{1-x})$.

2. $t \leq y_1 y_2$ is not possible.
   (Hessian matrix of $y_1 y_2$ is indefinite everywhere).

How to decompose nonlinear constraints?

1. $t \leq x\sqrt{1-x} = (x) \cdot (\sqrt{1-x})$.
2. $t \leq y_1 y_2$ is not possible.
   (Hessian matrix of $y_1 y_2$ is indefinite everywhere).
3. So relation is important, namely:

$$\left[y_2 = \sqrt{1-y_1}\right] \Leftrightarrow \left[y_1 = 1 - y_2^2\right] \Leftrightarrow \left[y_1 y_2 = y_2 - y_2^3\right].$$

How to decompose nonlinear constraints?

1. $t \leq x\sqrt{1-x} = (x) \cdot (\sqrt{1-x})$.

2. $t \leq y_1 y_2$ is not possible.
   (Hessian matrix of $y_1 y_2$ is indefinite everywhere).

3. So relation is important, namely:

$$\left[y_2 = \sqrt{1-y_1}\right] \Leftrightarrow \left[y_1 = 1 - y_2^2\right] \Leftrightarrow \left[y_1 y_2 = y_2 - y_2^3\right].$$

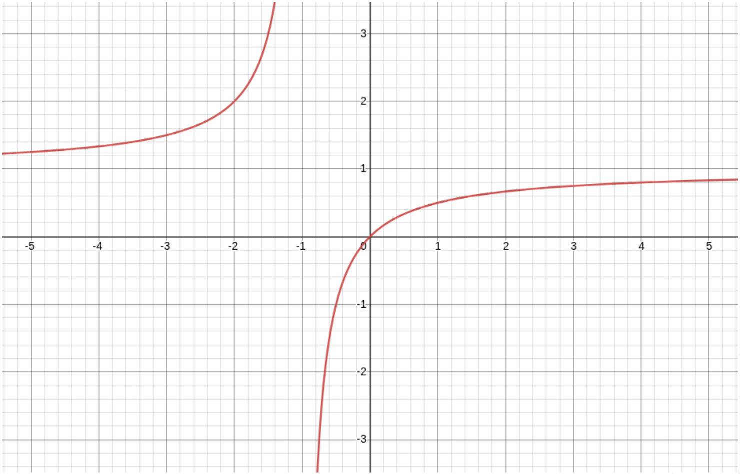
4. $t \leq \sqrt{1-x} - (1-x)^{3/2} = r_1 - r_2$, where

$$r_1 \leq \sqrt{1-x},$$
(rotated quadratic cone),

$$r_2 \geq (1-x)^{3/2},$$
(power cone).

How to decompose nonlinear constraints?

1. $t \le \dfrac{x}{x+1}$.



How to decompose nonlinear constraints?

1. $t \leq \dfrac{x}{x+1}$.
2. $t \leq y_1/y_2$ is not possible.
   (Hessian matrix of y_1/y_2 is indefinite everywhere).

How to decompose nonlinear constraints?

1. $t \le \dfrac{x}{x+1}$.

2. $t \le y_1/y_2$ is not possible.
   (Hessian matrix of y_1/y_2 is indefinite everywhere).

3. So relation is important, namely:

$$[y_2 = y_1 + 1] \Leftrightarrow [y_1 = y_2 - 1] \Leftrightarrow [y_1/y_2 = 1 - 1/y_2] \,.$$

How to decompose nonlinear constraints?

1. $t \leq \dfrac{x}{x+1}$.

2. $t \leq y_1/y_2$ is not possible.
(Hessian matrix of y_1/y_2 is indefinite everywhere).

3. So relation is important, namely:

$$[y_2 = y_1 + 1] \Leftrightarrow [y_1 = y_2 - 1] \Leftrightarrow [y_1/y_2 = 1 - 1/y_2]\,.$$

4. $t \leq 1 - \dfrac{1}{x+1} = 1 - r$, where

$$r \geq \frac{1}{x+1},$$

(rotated quadratic cone),

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \operatorname{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4}\tilde{a}^T D \tilde{a}\right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \mathrm{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4} \tilde{a}^T D \tilde{a} \right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \mathrm{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4} \tilde{a}^T D \tilde{a} \right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \mathrm{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4} \tilde{a}^T D \tilde{a} \right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```
Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \operatorname{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4} \tilde{a}^T D \tilde{a} \right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \operatorname{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^T Q x + a^T x + b \quad = \quad u^T D u + \left(b - \frac{1}{4} \tilde{a}^T D \tilde{a} \right),$$

in terms of $u = G^T x + \frac{1}{2} D\tilde{a}$.

```
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

Signed sum-of-squares decomposition

Let Q be a real symmetric matrix, such that $Q = GDG^T$ for a signature matrix D. For any $a \in \operatorname{colspan}(G)$, let $G\tilde{a} = a$ whereby

$$x^TQx + a^Tx + b \quad = \quad u^TDu + \left(b - \frac{1}{4}\tilde{a}^TD\tilde{a}\right),$$

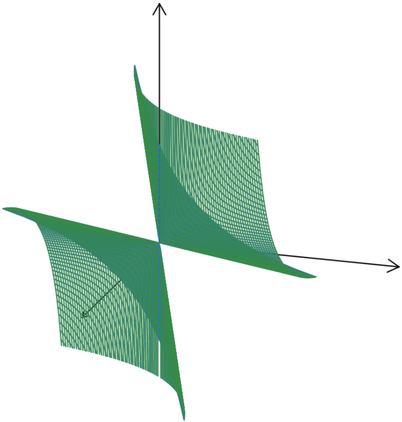
in terms of $u = G^Tx + \frac{1}{2}D\tilde{a}$.

```julia
using LinearAlgebra, GenericLinearAlgebra
D,G=eigen(Hermitian(BigFloat.(Q)));
nz=(abs.(D).>1e-12); G=G[:,nz]; D=Diagonal(D[nz]);
G*=sqrt.(abs.(D)); D=sign.(D);
@assert(norm(G*D*G'-Q,Inf)<=1e-12)
```

Julia 1.4

How to decompose nonlinear constraints?

1. $xy + yz - zx \leq 0$.
   (Hessian matrix of $xy + yz - zx$ is indefinite everywhere, but we only need a sublevel set!)



How to decompose nonlinear constraints?

1. $xy + yz - zx \leq 0$.
   (Hessian matrix of $xy + yz - zx$ is indefinite everywhere, but we only need a sublevel set!)

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$\frac{1}{2}(x+y)^2 + \frac{1}{6}(-x+y+2z)^2 \leq \frac{2}{3}(x-y+z)^2,$$

   but a more elegant decomposition is

$$x^2 + y^2 + z^2 \leq (x-y+z)^2.$$

How to decompose nonlinear constraints?

1. $xy + yz - zx \leq 0$.
   (Hessian matrix of $xy + yz - zx$ is indefinite everywhere, but we only need a sublevel set!)

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$\frac{1}{2}(x+y)^2 + \frac{1}{6}(-x+y+2z)^2 \leq \frac{2}{3}(x-y+z)^2,$$

   but a more elegant decomposition is

$$x^2 + y^2 + z^2 \leq (x-y+z)^2.$$

How to decompose nonlinear constraints?

1. $xy + yz - zx \leq 0$.
   (Hessian matrix of $xy + yz - zx$ is indefinite everywhere, but we only need a sublevel set!)

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$\frac{1}{2}(x+y)^2 + \frac{1}{6}(-x+y+2z)^2 \leq \frac{2}{3}(x-y+z)^2,$$

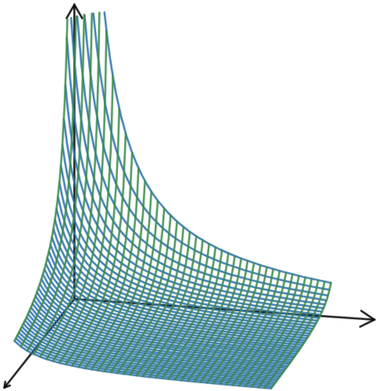
   but a more elegant decomposition is

$$x^2 + y^2 + z^2 \leq (x-y+z)^2.$$

3. If either $x - y + z \geq 0$ or $x - y + z \leq 0$ we can reformulate using a quadratic cone. Otherwise, you can branch on this disjunction and reformulate in each child node.

How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$



How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$t \geq \frac{1}{(\frac{1}{2}x_1 + \frac{1}{2}x_2)^2 - (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 - 1^2}$$

How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$t \geq \frac{1}{(\frac{1}{2}x_1 + \frac{1}{2}x_2)^2 - (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 - 1^2}$$

$$\Longleftrightarrow$$

$$t^{-1} + (\tfrac{1}{2}x_1 - \tfrac{1}{2}x_2)^2 + 1^2 \leq (\tfrac{1}{2}x_1 + \tfrac{1}{2}x_2)^2.$$

How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$t \geq \frac{1}{(\frac{1}{2}x_1 + \frac{1}{2}x_2)^2 - (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 - 1^2}$$

$$\iff$$

$$t^{-1} + (\tfrac{1}{2}x_1 - \tfrac{1}{2}x_2)^2 + 1^2 \leq (\tfrac{1}{2}x_1 + \tfrac{1}{2}x_2)^2.$$

How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$t \geq \frac{1}{(\frac{1}{2}x_1 + \frac{1}{2}x_2)^2 - (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 - 1^2}$$

$$\Longleftrightarrow$$

$$t^{-1} + (\tfrac{1}{2}x_1 - \tfrac{1}{2}x_2)^2 + 1^2 \leq (\tfrac{1}{2}x_1 + \tfrac{1}{2}x_2)^2.$$

3. $$r^2 \geq t^{-1},$$
   (power cone),

How to decompose nonlinear constraints?

1. $t \geq \dfrac{1}{x_1 + x_2 + x_1 x_2}$

2. The eigenvalue-based signed sum-of-squares decomposition reformulates this constraint as

$$t \geq \frac{1}{(\frac{1}{2}x_1 + \frac{1}{2}x_2)^2 - (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 - 1^2}$$

$$\iff$$

$$t^{-1} + (\tfrac{1}{2}x_1 - \tfrac{1}{2}x_2)^2 + 1^2 \leq (\tfrac{1}{2}x_1 + \tfrac{1}{2}x_2)^2.$$

3. $r^2 \geq t^{-1}$, (power cone), $\quad r^2 + (\frac{1}{2}x_1 - \frac{1}{2}x_2)^2 + 1^2 \leq (\frac{1}{2}x_1 + \frac{1}{2}x_2)^2$. (quadratic cone).

Summary

- Conic optimization is a sweet-spot in nonlinear optimization.
  - Derivable from LP to preserve nice duality theory.
  - Derivable from MINLP to avoid computational disadvantages.
- No issues with smoothness and differentiability.
- Models can be solved efficiently. Symmetric cones achieve the best known global convergence rate among Newton-type methods.

Further information

- The listed references on mixed-integer conic optimization.
- The MOSEK Modeling Cookbook.
- The MOSEK Optimization Suite (www.mosek.com).
  - Trial and free academic license.
  - Interfaces to C, Java, .NET, Julia, Matlab, R, Python, ...
- The applied optimization tutorials (www.github.com/MOSEK/Tutorials).