Some New Path-Following Algorithms for Convex Quadratic Programming

Dennis L. Bricker & Yi-Chih Hsieh
Dept. of Industrial Engineering
University of Iowa
stmt. of the QP problem
In path-following methods for convex quadratic programming, one must solve systems of equations of the form:

\[
\begin{align*}
Ax - y &= b \\
-Qx + A^T w + s &= c \\
XSe &= \mu e \\
WYe &= \mu e
\end{align*}
\]

This system consists of both linear and nonlinear equations, and are frequently solved using Newton’s method.
The motivation for our current work was a presentation by Scott Burns (U. of Illinois) on the “Monomial Method” for solving certain systems of nonlinear equations.


Arithmetic–Geometric Mean Inequality
Condensation of Posynomials
Posynomial Approximation of Signomials
The "Monomial Method" for Solving Systems of Nonlinear Equations
A "toy" LCP Example
Application to Path-Following Algorithm
Computational Experience
The Arithmetic-Geometric Mean Inequality

Simplest case: Given two positive numbers $a$ and $b$, their arithmetic mean $\frac{1}{2}a + \frac{1}{2}b$ is greater than or equal to their geometric mean $\sqrt{ab}$.

i.e., $\frac{1}{2}a + \frac{1}{2}b \geq a^{\frac{1}{2}}b^{\frac{1}{2}}$

with equality if and only if $a = b$. \(\sqsubseteq\)
Arithmetic-Geometric Mean Inequality

\[ \frac{1}{2} a + \frac{1}{2} b \geq a^{\frac{1}{2}} b^{\frac{1}{2}} \]

For example, let \( a = 2 \) & \( b = 8 \). Then this inequality is

\[ 5 = \frac{1}{2} \times 2 + \frac{1}{2} \times 8 \geq \sqrt{2 \times 8} = 4 \]

\[ \text{Arithmetic mean} \quad \text{Geometric Mean} \]

If \( a = 4 \) & \( b = 9 \),

\[ 6.5 = \frac{1}{2} \times 4 + \frac{1}{2} \times 9 \geq \sqrt{4 \times 9} = 6 \]

\[ \text{Arithmetic mean} \quad \text{Geometric Mean} \]
Arithmetic-Geometric Mean Inequality

\[
\frac{1}{2} a + \frac{1}{2} b \geq a^{\frac{1}{2}} b^{\frac{1}{2}}
\]

\[\square\]

**Proof.** Let \( \alpha \) & \( \beta \) be real numbers and \( a = \alpha^2 \geq 0 \)

\[ b = \beta^2 \geq 0 \]

Then \( (\alpha - \beta)^2 = \alpha^2 - 2\alpha\beta + \beta^2 \geq 0 \)

\[ \implies \alpha^2 + \beta^2 \geq 2\alpha\beta \]

\[ \implies \frac{1}{2} \alpha^2 + \frac{1}{2} \beta^2 \geq \alpha\beta \implies \frac{1}{2} a + \frac{1}{2} b \geq \sqrt{ab} \]
The Arithmetic-Geometric Mean Inequality

The General Case: Let \( x_1, x_2, \ldots, x_n > 0 \)
and \( \delta_1, \delta_2, \ldots, \delta_n \geq 0 \) and \( \sum_{i=1}^{n} \delta_i = 1 \)

Then \[
\sum_{i=1}^{n} \delta_i x_i \geq \prod_{i=1}^{n} x_i^{\delta_i}
\]

with equality if & only if \( x_1 = x_2 = \ldots = x_n \)
The Arithmetic–Geometric Mean Inequality

\[ \sum_{i=1}^{n} \delta_i x_i \geq \prod_{i=1}^{n} x_{1}^{\delta_i} \]

If we let \( n=2 \), and \( \delta_i = \frac{1}{2} \), then we obtain the earlier inequality,

\[ \frac{1}{2} a + \frac{1}{2} b \geq a^{\frac{1}{2}} b^{\frac{1}{2}} \]
The Arithmetic-Geometric Mean Inequality

Writing $u_i = \delta_i x_i$, we get

$$\sum_i u_i \geq \prod_i \left(\frac{u_i}{\delta_i}\right)^{\delta_i}$$

Equivalent form:

where $\delta_1, \delta_2, \ldots, \delta_n \geq 0$ and $\sum_{i=1}^{n} \delta_i = 1$

with equality if & only if $\frac{u_1}{\delta_1} = \frac{u_2}{\delta_2} = \ldots = \frac{u_n}{\delta_n}$
Condensation of Posynomials

\[ g(x_1, x_2, \ldots, x_m) = \sum_{i=1}^{n} c_i \prod_{j=1}^{m} x_j^{a_{ij}} \]

where \( c_i > 0 \) and \( a_{ij} \) are real numbers.
Recall the A-G Mean Inequality:

\[
\sum_i u_i \geq \prod_i \left(\frac{u_i}{\delta_i}\right)^{\delta_i}
\]

Letting \( u_i = c_i \prod_j x_j^{a_{ij}} \), we obtain

\[
g(x) = \sum_i c_i \prod_j x_j^{a_{ij}} \geq \prod_i \left[ \frac{\prod_j x_j^{a_{ij}}}{\delta_i} \right]^{\delta_i} = C(\delta) \prod_j x_j^{\alpha_{ij}(\delta)}
\]

where \( C(\delta) = \prod_i \left(\frac{c_i}{\delta_i}\right)^{\delta_i} \), \( \alpha_{ij}(\delta) = \sum_i a_{ij}\delta_i \).
That is, we obtain a monomial approximation (lower bound) of the posynomial,

\[ g(x) = \sum_i c_i \prod_j x_j^{a_{ij}} \geq C(\delta) \prod_j x_j^{\alpha_{ij}(\delta)} \]

where \( C(\delta) = \prod_i \left( \frac{c_i}{\delta_i} \right)^{\delta_i} \), \( \alpha_{ij}(\delta) = \sum_i a_{ij} \delta_i \)

which is exact when

\[ \frac{c_1 \prod_j x_j^{a_{1j}}}{\delta_1} = \frac{c_2 \prod_j x_j^{a_{2j}}}{\delta_2} = \ldots = \frac{c_n \prod_j x_j^{a_{nj}}}{\delta_n} \]
Signomial Functions

g(x_1, x_2, \ldots, x_m) = \sum_{i=1}^{n} c_i \prod_{j=1}^{m} x_j^{a_{ij}}

Condensation has long been used in solving Signomial GP problems (which are essentially nonconvex) by means of a sequence of approximating Posynomial GP problems (which are essentially convex problems).
Example

Minimize $x_1$

subject to

$$(x_1 - 2)^2 + (x_2 - 4)^2 \geq 4$$

$$(x_1 - 3)^2 + (x_2 - 3)^2 \leq 4$$

$X$ is outside a circle centered at $(2, 4)$ with radius 2

$X$ is within a circle centered at $(3, 3)$ with radius 2
Minimize $x_1$
subject to
\[(x_1 - 2)^2 + (x_2 - 4)^2 \geq 4\]
\[(x_1 - 3)^2 + (x_2 - 3)^2 \leq 4\]
Reformulation as a GP problem

\[(X_1 - 2)^2 + (X_2 - 4)^2 \geq 4\]

\[\Rightarrow (x_1^2 - 4x_1 + 4) + (x_2^2 - 8x_2 + 16) \geq 4\]

\[\Rightarrow -x_1^2 + 4x_1 - x_2^2 + 8x_1 \leq 16\]

The constraint becomes the signomial constraint

\[\Rightarrow \frac{X_1}{4} + \frac{X_2}{2} - \frac{X_1^2}{16} - \frac{X_2^2}{16} \leq 1\]
Reformulation as a GP problem

\[(X_1 - 3)^2 + (X_2 - 3)^2 \leq 4\]

\[\Rightarrow (x_1^2 - 6x_1 + 9) + (x_2^2 - 6x_2 + 9) \leq 4\]

\[\Rightarrow x_1^2 - 6x_1 + x_2^2 + 14 \leq 6x_2\]

The constraint becomes the signomial constraint

\[\Rightarrow \frac{X_1^2X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} - X_1X_2^{-1} \leq 1\]
Signomial Geometric Program

Minimize $X_1$

subject to

$$\frac{X_1}{4} + \frac{X_2}{2} - \frac{X_1^2}{16} - \frac{X_2^2}{16} \leq 1$$

$$\frac{X_1^2X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} - X_1X_2^{-1} \leq 1$$

$$X_1 > 0, \ X_2 > 0$$
To condense the signomial constraint
\[ \frac{X_1}{4} + \frac{X_2}{2} - \frac{X_1^2}{16} - \frac{X_2^2}{16} \leq 1 \]

we first write it in the form
\[ \frac{X_1}{4} + \frac{X_2}{2} \leq 1 + \frac{X_1^2}{16} + \frac{X_2^2}{16} \]

\[ \Rightarrow \frac{X_1}{4} + \frac{X_2}{2} \leq 1 \Rightarrow \frac{0.25X_1 + 0.5X_2}{1 + 0.0625X_1 + 0.0625X_2} \leq 1 \]
We next condense the denominator of

\[
\frac{0.25X_1 + 0.5X_2}{1 + 0.0625 X_1^2 + 0.0625 X_2^2} \leq 1
\]

into a single term. Let's use the point \(X_0 = (4,5)\) at which the terms of the denominator are

\[
1 + 1 + 1.5626 = 3.5625
\]

Then

\[
\delta_1 = \delta_2 = \frac{1}{3.5625} = 0.2807 \quad \text{and} \quad \delta_3 = \frac{1.5625}{3.5625} = 0.4386
\]
\[ \delta_1 = \delta_2 = 0.2807, \quad \delta_3 = 0.4386 \]

Coefficient:

\[
C(\delta) = \prod_{i=1}^{3} \left( \frac{c_i}{\delta_i} \right)^{\delta_i}
\]

\[
C(\delta) = \left( \frac{1}{0.2807} \right)^{0.2807} \left( \frac{0.0625}{0.2807} \right)^{0.2807} \left( \frac{0.0625}{0.4386} \right)^{0.4386}
\]

\[= 0.3987\]
\[ \delta_1 = \delta_2 = 0.2807, \quad \delta_3 = 0.4386 \]

**Exponents:**

\[ a_j(\delta) = \sum_{i=1}^{3} a_{ij} \delta_i \]

\[ a_1 = 0\delta_1 + 2\delta_2 + 0\delta_3 = 2(0.2807) = 0.5614 \]

\[ a_2 = 0\delta_1 + 0\delta_2 + 2\delta_3 = 2(0.4386) = 0.8772 \]
\[ C(\delta) = 0.3987 \]
\[ a_1 = 0.5614 \]
\[ a_2 = 0.8772 \]

Condensed denominator is
\[ 0.3987 X_1^{0.5614} X_2^{0.8772} \]

*monomial!*
Geometric Inequality implies

\[ 1 + 0.0625X_1^2 + 0.0625X_2^2 \geq 0.3987X_1^{0.5614}X_2^{0.8772} \]

and so

\[ \frac{0.25X_1 + 0.5X_2}{1 + 0.0625X_1^2 + 0.0625X_2^2} \leq \frac{0.25X_1 + 0.5X_2}{0.3987X_1^{0.5614}X_2^{0.8772}} \]
\[
\frac{0.25X_1 + 0.5X_2}{0.3987X_1^{0.5614}X_2^{0.8772}} = \frac{\text{posynomial}}{\text{monomial}} = \text{posynomial}
\]

\[
= \frac{0.25}{0.3987} X_1^{1-0.5614} X_2^{-0.8772} + \frac{0.5}{0.3987} X_1^{-0.5614} X_2^{1-0.8772}
\]

\[
= 0.627 X_1^{0.4386} X_2^{-0.6772} + 1.254 X_1^{-0.5614} X_2^{0.1228}
\]

which is a posynomial!
If we constrain this posynomial so as to be \( \leq 1 \), then by the geometric inequality, the original signomial should also be \( \leq 1 \).

That is, any \( X \) feasible in the posynomial constraint derived by condensation will also be feasible in the signomial constraint:

\[
\frac{0.25X_1 + 0.5X_2}{1 + 0.0625 X_1^2 + 0.0625 X_2^2} \\
\leq 0.627 X_1^{0.4386} X_2^{-0.8772} + 1.254 X_1^{-0.5614} X_2^{0.1228} \leq 1
\]
The second signomial constraint may be condensed in a similar fashion:

\[
\frac{X_1^2 X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} - X_1 X_2^{-1} \leq 1
\]

\[
\Rightarrow \quad \frac{X_1^2 X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} \leq 1 + X_1 X_2^{-1}
\]

\[
\Rightarrow \quad \frac{X_1^2 X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} \leq 1 + X_1 X_2^{-1}
\]
\[
\frac{X_1^2X_2^{-1}}{6} + \frac{X_2}{6} + \frac{7X_2^{-1}}{3} \leq 1
\]

At (4,5), the denominator is \(1 + 0.8 = 1.8\), so
\[
\delta_1 = \frac{1}{1.8} = 0.555, \quad \delta_2 = \frac{0.8}{1.8} = 0.444
\]

can be condensed (using \(\delta_1 = 0.555, \delta_2 = 0.444\)) into the posynomial constraint
\[
0.08385X_1^{1.555}X_2^{-0.555} + 0.08385X_1^{-0.444}X_2^{1.444} + 1.174X_1^{-0.444}X_2^{-0.555} \leq 1
\]
The signomial GP problem is therefore approximated by the posynomial problem:

\[ \text{Minimize } X_1 \]

subject to

\[ 0.627 X_1^{0.4386} X_2^{-0.8772} + 1.254 X_1^{-0.5614} X_2^{0.1228} \leq 1 \]

\[ 0.08385 X_1^{1.555} X_2^{-0.555} + 0.08385 X_1^{-0.444} X_2^{1.444} + 1.174 X_1^{-0.444} X_2^{-0.555} \leq 1 \]

\[ X_1 > 0, \quad X_2 > 0 \]
We wish to find a (positive) solution of the following system of nonlinear (signomial) equations:

\[ g_k(x) = \sum_i \sigma_{ik} c_{ik} \prod_j x_j^{a_{ijk}} = 0, \ k=1, \cdots, N \]

where \( \sigma_{ik} \in [+1, -1], \ c_{ik} > 0 \)

**Example:**

\[
\begin{align*}
2.5 x_1^{-1.5} + 15 x_1^{8/3} x_2^{-2} - 30x_2 &= 0 \\
77 + 9 x_2^{-1} - 28x_1 x_2 - 4x_1^{-3} &= 0
\end{align*}
\]
Define the index sets of the positive & negative terms of each equation:

\[ T^+_k = \{ i \mid \sigma_{ik} > 0 \} \quad \& \quad T^-_k = \{ i \mid \sigma_{ik} < 0 \} \]

Then separate each signomial into positive & negative parts:

\[ g_k(x) = P_k(x) - Q_k(x) \]

where

\[ P_k(x) = \sum_{i \in T^+_k} c_{ik} \prod_{j} x^{a_{ijk}} \quad \& \quad Q_k(x) = \sum_{i \in T^-_k} c_{ik} \prod_{j} x^{a_{ijk}} \]
\[ g_k(x) = P_k(x) - Q_k(x) = 0 \]
\[ \Rightarrow \quad P_k(x) = Q_k(x) \]
\[ \Rightarrow \quad \frac{P_k(x)}{Q_k(x)} = 1 \]

Each of the posynomials \( P_k(x) \) and \( Q_k(x) \) are then condensed into monomial approximations \( \overline{P}_k(x) \) and \( \overline{Q}_k(x) \), respectively, and the ratio of the two monomials is also a monomial!
Each nonlinear equation is then approximated by a monomial equation

\[
\frac{P_k(x)}{Q_k(x)} \approx \frac{P_k(x)}{\bar{Q}_k(x)} = C_k(\delta) \prod_j x_j^{\alpha_{jk}(\delta)} = 1
\]

for some choice of the weights (\(\delta\)).

By taking the logarithms of both sides and making the change of variable \(z_j = \ln x_j\),

we get the linear equation

\[
\sum_j \alpha_{jk}(\delta) z_j = -C_k(\delta)
\]
0. Select an initial starting point $x^\circ$.
1. Evaluate the weights of all the terms:
   \[ \delta_{ik} = \frac{c_{ik} \prod_j (x^\circ)_j^{a_{ijk}}}{p_k(x^\circ)} \quad \forall \ i \in T_k^+ \quad \text{and} \quad \delta_{ik} = \frac{c_{ik} \prod_j (x^\circ)_j^{a_{ijk}}}{q_k(x^\circ)} \quad \forall \ i \in T_k^- \]

2. Evaluate $C_k(\delta)$ and $\alpha_{kj}(\delta)$
3. Solve the linear system of equations in $z$.
4. Exponentiate $z$ to obtain $x'$ (yielding $x' > 0$!)
5. Test for convergence, e.g.,
   \[ \|x^\circ - x'\| \leq \varepsilon \]
   If the test fails, replace $x^\circ$ with $x'$ and return to step 1.
It can be shown that the "Monomial" Method is equivalent to Newton's Method applied to

\[
\ln \left[ \frac{P_k(e^x)}{Q_k(e^x)} \right] = 0, \quad k=1,...,N
\]
Standard Newton

\[ \text{P}(x) - \text{Q}(x) = 0 \]

Newton-Central

\[ \text{P}(e^2) - \text{Q}(e^2) = 0 \]

\[ \frac{\text{P}(e^2)}{\text{Q}(e^2)} = 1 \]

Monomial

\[ \ln \left[ \frac{\text{P}(e^2)}{\text{Q}(e^2)} \right] = 0 \]

These all have the property that they will exactly follow the central path and yield strictly positive iterates!
A "toy" LCP:

\[ y = Mx + q, \quad xy = 0 \]

\[
\begin{cases}
xy = \mu \\
y = x + 1
\end{cases}
\]

i.e., one "complementarity" equation
one linear equation
\[
\begin{align*}
\begin{cases}
  xy &= \mu = 0.75 \\
  y &= x + 1
\end{cases}
\end{align*}
\]

In general, the solution is

\[
x(\mu) = -\frac{1}{2} + \sqrt{\frac{1}{4} + \mu}
\]

\[
y(\mu) = \frac{1}{2} + \sqrt{\frac{1}{4} + \mu}
\]
Monomial Method

Note that the "complementarity" equation is already in monomial form.

The linear equation is approximated by a monomial as follows:

\[ P - Q = (x+1) - y = 0 \]

\[ \Rightarrow \frac{P}{Q} = \frac{x+1}{y} = 1 \]

\[ x + 1 \geq \left( \frac{x}{\delta_1} \right)^{\delta_1} \left( \frac{1}{\delta_2} \right)^{\delta_2} = \delta_1^{\delta_1} \delta_2^{\delta_2} x^{\delta_1} \]

where the weights are:

\[ \delta_1 = \frac{x^\circ}{x^\circ + 1}, \quad \delta_2 = \frac{1}{x^\circ + 1} \]
The nonlinear system: \[
\begin{align*}
xy &= \mu \\
y &= x + 1
\end{align*}
\]
is approximated by the linear system:

\[
\begin{align*}
\ln x + \ln y &= \ln \mu \\
\delta_1 \ln x - \ln y &= \delta_1 \ln \delta_1 + \delta_1 \ln \delta_1
\end{align*}
\]

that is,

\[
\begin{align*}
z_x + z_y &= \ln \mu \\
\delta_1 z_x - z_y &= \delta_1 \ln \delta_1 + \delta_1 \ln \delta_1
\end{align*}
\]

where \(z_x = \ln x\), \(z_y = \ln y\)
In the **Monomial Method**, then, we solve
\[
\begin{bmatrix}
1 & -1 \\
x^0 & x^0 + 1
\end{bmatrix}
\begin{bmatrix}
z_x \\
z_y
\end{bmatrix}
= 
\begin{bmatrix}
\ln \mu \\
C
\end{bmatrix}
\]
and update \( x^0 \leftarrow \exp(z_x) \) & \( y^0 \leftarrow \exp(z_y) \)

while in **Newton's Method**, we solve
\[
\begin{bmatrix}
y^0 & x^0 \\
1 & -1
\end{bmatrix}
\begin{bmatrix}
\Delta_x \\
\Delta_y
\end{bmatrix}
= 
\begin{bmatrix}
\mu - x^0 y^0 \\
y^0 - x^0 - 1
\end{bmatrix}
\]
and update \( x^0 \leftarrow x^0 + \Delta_x \) & \( y^0 \leftarrow y^0 + \Delta_y \)
Still another algorithm may be obtained by applying Newton’s Method after making the logarithmic transformation:

\[
\begin{align*}
    z_x + z_y &= \ln \mu \\
    e^{z_x} - e^{z_y} &= -1
\end{align*}
\]

which requires solving

\[
\begin{bmatrix}
    1 & 1 \\
    \ln x^\circ - \ln y^\circ
\end{bmatrix}
\begin{bmatrix}
    dz_x \\
    dz_y
\end{bmatrix}
= \begin{bmatrix}
    \mu - x^\circ y^\circ \\
    y^\circ - x^\circ - 1
\end{bmatrix}
\]

and updating

\[
z_x^\circ \leftarrow z_x^\circ + dz_x \quad \& \quad z_y^\circ \leftarrow z_y^\circ + dz_y
\]

\emph{Newton–Central}
Newton’s Method

\[ \mu = 10^{-8} \]

**stopping criterion** \[ |\mu - xy| + |y - x - 1| \leq 10^{-8} \]

Starting point: \((100, 10)\)

<table>
<thead>
<tr>
<th>k</th>
<th>(x^k)</th>
<th>(y^k)</th>
<th>(\mu - x^k y^k)</th>
<th>(y^{k-1} - x^k)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1E2</td>
<td>1E1</td>
<td>-1E3</td>
<td>-9.1E1</td>
</tr>
<tr>
<td>1</td>
<td>8.18182E0</td>
<td>9.18182E0</td>
<td>-7.5124E1</td>
<td>-1.77636E-15</td>
</tr>
<tr>
<td>2</td>
<td>3.85531E0</td>
<td>4.85531E0</td>
<td>-1.87187E1</td>
<td>8.88178E-16</td>
</tr>
<tr>
<td>3</td>
<td>1.70635E0</td>
<td>2.70635E0</td>
<td>-4.618E0</td>
<td>4.44089E-16</td>
</tr>
<tr>
<td>4</td>
<td>6.59832E-1</td>
<td>1.65983E0</td>
<td>-1.09521E0</td>
<td>OEO</td>
</tr>
<tr>
<td>5</td>
<td>1.8769E-1</td>
<td>1.18769E0</td>
<td>-2.22918E-1</td>
<td>2.22045E-16</td>
</tr>
<tr>
<td>6</td>
<td>2.5613E-2</td>
<td>1.02561E0</td>
<td>-2.6269E-2</td>
<td>OEO</td>
</tr>
<tr>
<td>7</td>
<td>6.24066E-4</td>
<td>1.00062E0</td>
<td>-6.24445E-4</td>
<td>OEO</td>
</tr>
<tr>
<td>8</td>
<td>3.9896E-7</td>
<td>1E0</td>
<td>-3.88961E-7</td>
<td>OEO</td>
</tr>
<tr>
<td>9</td>
<td>1.000002E-8</td>
<td>1E0</td>
<td>-1.5129E-13</td>
<td>OEO</td>
</tr>
</tbody>
</table>
**Monomial Method**

\[ \mu = 10^{-8} \]

*Stopping criterion*

\[ |\mu - xy| + |y - x - 1| \leq 10^{-8} \]

*Starting point: (100, 10)*

<table>
<thead>
<tr>
<th>k</th>
<th>( x^k )</th>
<th>( y^k )</th>
<th>( \mu - x^k y^k )</th>
<th>( y^{k-1} - x^{k-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1E2</td>
<td>1E1</td>
<td>-1E3</td>
<td>-9.1E1</td>
</tr>
<tr>
<td>1</td>
<td>9.28919E-5</td>
<td>1.07652E-4</td>
<td>1.8198E-23</td>
<td>-9.99985E-1</td>
</tr>
<tr>
<td>2</td>
<td>1.000076E-8</td>
<td>9.99245E-1</td>
<td>8.27181E-24</td>
<td>-7.55405E-4</td>
</tr>
<tr>
<td>3</td>
<td>1E-8</td>
<td>1E0</td>
<td>1.98523E-23</td>
<td>-2.88658E-15</td>
</tr>
</tbody>
</table>
An Infeasible Path-Following Algorithm using the Newton-Central Method

Equations to be approximately solved at each iteration

\[
\begin{align*}
Ax - y &= b \\
-Qx + A^Tw + s &= c \\
XSe &= \mu e \\
WYe &= \mu e
\end{align*}
\]

The logarithmic transformation is made, so that the complementarity equations are linearized, and the linear equations become nonlinear:

\[ P(e^2) - Q(e^2) = 0 \]
An Infeasible Path-Following Algorithm using the Monomial Method

Equations to be approximately solved at each iteration

\[
\begin{align*}
Ax - y &= b \\
-Qx + ATw + s &= c \\
XSe &= \mu e \\
WYe &= \mu e
\end{align*}
\]

The linear equations are approximated by monomial equations, and the logarithmic transformation is then made to linearize all the constraints.
0 Start with any interior solution \((x^0, y^0, s^0, w^0) > 0\)
set \(k = 0\), and choose 3 tolerances \(\varepsilon_1, \varepsilon_2, \varepsilon_3 > 0\)

1 Compute \(\mu^k = \sigma \frac{x^k s^k + y^k w^k}{n + m}\), for \(0 < \sigma < 1\)
\[ t_p^k = b + y^k - Ax^k, \quad t_d^k = Qx^k + c - A^T w^k - s^k \]

2 If \(\mu^k \leq \varepsilon_1, \frac{t_p^k}{\|b\| + 1} \leq \varepsilon_2, \quad \frac{t_d^k}{\|Qx^k + c\| + 1} \leq \varepsilon_3\)
then stop & accept the current iterate as optimal.
3. Evaluate the weights
4. Compute coefficients & rhs of linear system
5. Solve linear system & return to step 1.

Properties of the sequence generated by this algorithm:
- exactly on the central trajectory
- strictly positive
- converges if bounded and the algorithm does not fail
Computational Experience
• Random subproblems with two variables, three constraints, and known solutions were randomly generated and used to build larger problems.

• Separability was eliminated by performing a linear transformation.

• For each problem size, ten random test problems were tested.

• Initial solutions for Newton & Newton-Central algorithm are randomly generated but ON the central trajectory.

• Initial solutions for Monomial algorithm are randomly generated but not on central trajectory.
<table>
<thead>
<tr>
<th>Number of Subproblems</th>
<th>Separable Problems</th>
<th>Nonseparable Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>Q</td>
</tr>
<tr>
<td>M=2</td>
<td>50%</td>
<td>25%</td>
</tr>
<tr>
<td>M=4</td>
<td>25%</td>
<td>12.5%</td>
</tr>
<tr>
<td>M=8</td>
<td>12.5%</td>
<td>6.25%</td>
</tr>
<tr>
<td>M=12</td>
<td>8.33%</td>
<td>4.17%</td>
</tr>
<tr>
<td># variables</td>
<td>2M</td>
<td></td>
</tr>
<tr>
<td># constraints</td>
<td>3M</td>
<td></td>
</tr>
</tbody>
</table>
Adjustment of factor $\sigma^k$

Standard Newton Algorithm:

$$\sigma^{k+1} = \begin{cases} 
\min (0.95, 1.3\sigma^k) & \text{if } \frac{\mu^{k+1}}{\mu^k} < 1 \\
\max (0.2, 0.7\sigma^k) & \text{otherwise}
\end{cases}$$

Newton-Central & Monomial Algorithms:

$$\sigma^{k+1} = \begin{cases} 
\min (0.95, 1.3\sigma^k) & \text{if } \frac{\text{error}^{k+1}}{\text{error}^k} < 1 \\
\max (0.2, 0.7\sigma^k) & \text{otherwise}
\end{cases}$$

$$\text{error}^k = \frac{t_p^k}{\|b\| + 1} + \frac{t_d^k}{\|Qx^k+c\| + 1}$$
Iterations vs # subproblems

Separable Problems

- Newton
- Newton(Central)
- Monomial
CPU vs \# subproblems

Separable Problems

- Newton
- Newton(Central)
- Monomial

M=2  M=4  M=8  M=12
Iterations vs. # subproblems

Nonseparable Problems

<table>
<thead>
<tr>
<th>M</th>
<th>Newton</th>
<th>Newton(Central)</th>
<th>Monomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>24</td>
<td>18.1</td>
<td>29.9</td>
</tr>
<tr>
<td>4</td>
<td>27.9</td>
<td>19.8</td>
<td>29.9</td>
</tr>
<tr>
<td>8</td>
<td>29.8</td>
<td>22.4</td>
<td>23.2</td>
</tr>
<tr>
<td>12</td>
<td>27.6</td>
<td>22.4</td>
<td>23.2</td>
</tr>
</tbody>
</table>
CPU vs # subproblems

Nonseparable Problems

- Newton
- Newton(Central)
- Monomial