Further developments of multiple centrality correctors

Marco Colombo Jacek Gondzio

School of Mathematics
The University of Edinburgh

EURO 2006, Reykjavik

Primal–dual interior point methods
- Introduction
- Mehrotra's predictor-corrector algorithm
- Multiple centrality correctors

Pitfalls and workarounds
- Practical pitfalls
- Weight of the corrector direction
- Computational experience

Conclusions

Linear programming and optimality conditions

- Linear programming problem and KKT conditions

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array} \qquad \begin{bmatrix} Ax - b \\ A^T y + s - c \\ XSe \end{bmatrix} = 0 \quad x, s \geq 0$$

Linear programming and optimality conditions

- Linear programming problem and KKT conditions

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array} \qquad \begin{bmatrix} Ax - b \\ A^T y + s - c \\ XSe \end{bmatrix} = 0 \quad x, s \geq 0$$

- Perturb the complementarity conditions

$$XSe = \mu e$$

Linear programming and optimality conditions

- Linear programming problem and KKT conditions

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array} \qquad \begin{bmatrix} Ax - b \\ A^T y + s - c \\ XSe \end{bmatrix} = 0 \quad x, s \geq 0$$

- Perturb the complementarity conditions

$$XSe = \mu e$$

- Solve the perturbed KKT conditions with Newton's method

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} b - Ax \\ c - A^T y - s \\ -XSe + \mu e \end{bmatrix}$$

Mehrotra's predictor-corrector algorithm

- Exploit linearity: solve independently for two right-hand sides

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \underbrace{\begin{bmatrix} b - Ax \\ c - A^T y - s \\ -XSe \end{bmatrix}}_{predictor} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \mu e \end{bmatrix}}_{corrector}$$

Predictor direction

- Set $\mu = 0$ and solve for the direction Δ_a
- Evaluate the allowed stepsizes

$$\alpha_p = \max_{\alpha} : x + \alpha \Delta_a x \geq 0 \qquad \alpha_d = \max_{\alpha} : s + \alpha \Delta_a s \geq 0$$

Predictor direction

- Set $\mu = 0$ and solve for the direction Δ_a
- Evaluate the allowed stepsizes

$$\alpha_p = \max_{\alpha} : x + \alpha \Delta_a x \geq 0 \qquad \alpha_d = \max_{\alpha} : s + \alpha \Delta_a s \geq 0$$

- Predicted complementarity gap

$$g_a = (x + \alpha_p \Delta_a x)^T (s + \alpha_d \Delta_a s)$$

- Estimate the centering term

$$\mu = \left(\frac{g_a}{x^T s} \right)^3 \frac{x^T s}{n}$$

Mehrotra's corrector direction

- Error in taking a full step in the predictor

$$(X+\Delta_a X)(S+\Delta_a S)e = XSe + \underbrace{(S\Delta_a x + X\Delta_a s)}_{-XSe} + \Delta_a X \Delta_a S e$$

Mehrotra's corrector direction

- Error in taking a full step in the predictor

$$(X+\Delta_a X)(S+\Delta_a S)e = XSe + \underbrace{(S\Delta_a x + X\Delta_a s)}_{-XSe} + \Delta_a X \Delta_a S e$$

- Consider a second order term and solve for Δ_c

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mu e - \Delta_a X \Delta_a S e \end{bmatrix}$$

Mehrotra's corrector direction

- Error in taking a full step in the predictor

$$(X+\Delta_a X)(S+\Delta_a S)e = XSe + \underbrace{(S\Delta_a x + X\Delta_a s)}_{-XSe} + \Delta_a X \Delta_a Se$$

- Consider a second order term and solve for Δ_c

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mu e - \Delta_a X \Delta_a Se \end{bmatrix}$$

- Find the stepsizes in the combined direction $\Delta = \Delta_a + \Delta_c$

Multiple centrality correctors

- Try to increase the stepsizes: $\tilde{\alpha} = \min(\alpha + \delta, 1)$

Multiple centrality correctors

- Try to increase the stepsizes: $\tilde{\alpha} = \min(\alpha + \delta, 1)$
- Move the trial point in the neighbourhood

$$\tilde{v}_i = x_i(\tilde{\alpha})s_i(\tilde{\alpha}) \in \mathcal{N}_s(\gamma) = \{x_i s_i : \gamma\mu \le x_i s_i \le \frac{1}{\gamma}\mu,\ \gamma \in (0,1)\}$$

Multiple centrality correctors

- Try to increase the stepsizes: $\tilde{\alpha} = \min(\alpha + \delta, 1)$
- Move the trial point in the neighbourhood

$$\tilde{v}_i = x_i(\tilde{\alpha})s_i(\tilde{\alpha}) \in \mathcal{N}_s(\gamma) = \{x_i s_i : \gamma\mu \le x_i s_i \le \frac{1}{\gamma}\mu,\ \gamma \in (0,1)\}$$

- Define an achievable target

$$t_i = \begin{cases} 0 & \text{if } \tilde{v}_i \in [\gamma\mu, \frac{1}{\gamma}\mu] \\ \gamma\mu - \tilde{v}_i & \text{if } \tilde{v}_i < \gamma\mu \\ \frac{1}{\gamma}\mu - \tilde{v}_i & \text{if } \tilde{v}_i > \frac{1}{\gamma}\mu \end{cases} \qquad rhs = \begin{bmatrix} 0 \\ 0 \\ t \end{bmatrix}$$

Multiple centrality correctors

- Try to increase the stepsizes: $\tilde{\alpha} = \min(\alpha + \delta, 1)$
- Move the trial point in the neighbourhood

$$\tilde{v}_i = x_i(\tilde{\alpha})s_i(\tilde{\alpha}) \in \mathcal{N}_s(\gamma) = \{x_i s_i : \gamma\mu \le x_i s_i \le \frac{1}{\gamma}\mu,\ \gamma \in (0,1)\}$$

- Define an achievable target

$$t_i = \begin{cases} 0 & \text{if } \tilde{v}_i \in [\gamma\mu, \frac{1}{\gamma}\mu] \\ \gamma\mu - \tilde{v}_i & \text{if } \tilde{v}_i < \gamma\mu \\ \frac{1}{\gamma}\mu - \tilde{v}_i & \text{if } \tilde{v}_i > \frac{1}{\gamma}\mu \end{cases} \qquad rhs = \begin{bmatrix} 0 \\ 0 \\ t \end{bmatrix}$$

- The number of correctors allowed depends on

$$\frac{\textit{Factorization effort}}{\textit{Backsolve effort}}$$

Multiple centrality correctors

- Try to increase the stepsizes: $\tilde{\alpha} = \min(\alpha + \delta, 1)$
- Move the trial point in the neighbourhood

$$\tilde{v}_i = x_i(\tilde{\alpha})s_i(\tilde{\alpha}) \in \mathcal{N}_s(\gamma) = \{x_i s_i : \gamma\mu \le x_i s_i \le \frac{1}{\gamma}\mu,\ \gamma \in (0,1)\}$$

- Define an achievable target

$$t_i = \begin{cases} 0 & \text{if } \tilde{v}_i \in [\gamma\mu, \frac{1}{\gamma}\mu] \\ \gamma\mu - \tilde{v}_i & \text{if } \tilde{v}_i < \gamma\mu \\ \frac{1}{\gamma}\mu - \tilde{v}_i & \text{if } \tilde{v}_i > \frac{1}{\gamma}\mu \end{cases} \qquad rhs = \begin{bmatrix} 0 \\ 0 \\ t \end{bmatrix}$$

- The number of correctors allowed depends on

$$\frac{\textit{Factorization effort}}{\textit{Backsolve effort}}$$

- A corrector is accepted if $\hat{\alpha} \ge \alpha + \rho\delta$

Practical pitfalls

- The stepsizes in the affine-scaling direction can be very short, especially if the point is badly centered
- Mehrotra's corrector is computed on the basis of full step in affine scaling direction
- Sometimes the magnitude of the corrector is much larger than the magnitude of the predictor (Cartis, 2005)
- Short steps in the combined direction may be produced

Weighting the corrector direction

Recent developments:

- Cartis (2005) suggests weighting the corrector by α^2, based on a quadratic approximation of a local path from the current point to a target on the central path (PDSOC)

Weighting the corrector direction

Recent developments:

- Cartis (2005) suggests weighting the corrector by α^2, based on a quadratic approximation of a local path from the current point to a target on the central path (PDSOC)
- Generalize Mehrotra's scheme

$$\Delta^\omega = \Delta_a + \omega \Delta_c$$

Weighting the corrector direction

Recent developments:

- Cartis (2005) suggests weighting the corrector by α^2, based on a quadratic approximation of a local path from the current point to a target on the central path (PDSOC)
- Generalize Mehrotra's scheme

$$\Delta^\omega = \Delta_a + \omega\Delta_c$$

- Salahi, Peng and Terlaky (2005) propose $\omega = \alpha$ and safeguards based on centering

Finding the best weight

- Find the corrector direction Δ_c

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \mu e - \Delta_a X \Delta_a S e \end{bmatrix}$$

- Set $\omega \in [\alpha_p \alpha_d, 1]$ and compute

$$\Delta^\omega = \Delta_a + \omega \Delta_c$$

- Do a linesearch to find the optimal $\hat{\omega}_p$ and $\hat{\omega}_d$ that maximize the stepsizes in Δ^ω

Weighted scheme for multiple centrality correctors

- Stronger emphasis on the symmetric neighbourhood

$$\mathcal{N}_s(\gamma) = \{x_i s_i : \gamma\mu \le x_i s_i \le \frac{1}{\gamma}\mu,\ \gamma \in (0,1)\}$$

- Use more aggressive centrality correctors

$$\hat{\delta} = 3\delta$$

- Find the best weight $\hat{\omega}$ for each corrector
- Better control on when we should stop correcting (no fixed number of correctors)

Computational experience

- Initial comparison between PCx code (Czyzyk, Mehrotra, Wright) and HOPDM code (Gondzio)
- Different linear algebra in PCx and HOPDM accounts for different choices concerning multiple centrality correctors
- Analysis done on number of iterations and number of backsolves
- Time comparison between HOPDM and HOPDM-ω on larger problems

Computational results I

Results from a collection of 101 problems (Netlib and Kennington)

	PCx	HOPDM	HOPDM-ω	Change
Iterations	2114	1871	1445	-22%
Backsolves	4849	6043	5717	-5%
Backsolves/iter.	2.29	3.23	3.95	+22%

- Decrease in iteration count but increase in backsolves
- We are accepting more multiple centrality correctors than normally we would do

Computational results II

Time comparison (in seconds) on larger problems

Problem	HO	HO-ω	Diff	Problem	HO	HO-ω	Diff
mod2	20.5	21.6	5.3%	world	26.3	23.4	-11.2%
world3	31.1	27.4	-11.7%	world4	73.2	56.1	-23.3%
world6	39.3	32.7	-16.6%	world7	43.1	36.0	-16.5%
worldl	43.9	36.8	-16.2%	route	40.9	33.7	-17.4%
ulevi	9.0	9.5	5.6%	ulevimin	16.5	16.4	-0.4%
dbir1	162.1	146.5	-9.7%	dbir2	208.9	156.1	-25.3%
pcb3000	1.1	1.1	2.7%	rlfprim	15.6	15.0	-3.5%
rlfdual	71.1	49.7	-30.0%	neos1	169.1	141.8	-16.1%
neos2	113.8	86.1	-24.4%	neos3	132.0	120.5	-8.7%
neos	1785.8	1386.5	-22.4%	watson-1	138.6	166.2	19.9%
sgpf5y6	49.5	64.4	30.0%	stormG2	1661.5	1623.1	-2.3%
rail507	9.7	10.1	3.4%	rail516	7.5	5.8	-22.4%
rail582	9.6	9.6	-0.7%	rail2586	1029.3	566.8	-44.9%
rail4284	2779.6	978.4	-64.8%	fome11	407.2	265.2	-34.9%
fome12	766.9	508.6	-33.7%	fome13	1545.0	990.6	-35.9%

Conclusions

- The theoretical suggestion of using $\omega = \alpha_p \alpha_d$ provides a reliable lower bound, but is too restrictive
- Doing a line search in order to find the best weight pays off in terms of reduction of iteration count
- The quality of the points allows for more aggressive multiple centrality correctors
- The computational experience validates the heuristic choices, with savings in number of iterations and in computing time

Reference

- http://www.maths.ed.ac.uk/~gondzio/reports/mcjgMCC.pdf