A NEW PROPOSAL TO MODIFY CONTROLLED CHOLESKY FACTORIZATION TO IMPROVE INTERIOR POINT METHOD

Manolo Rodriguez Heredia   Aurelio Ribeiro Leite Oliveira

DMA, IMECC, UNICAMP

April, 2015

Primal–dual interior point methods

Consider the linear programming problem

$$
(\mathrm{P})\begin{cases} \min & c^T x \\ \text{s.t.} & A\,x = b \\ & x \geq 0 \end{cases}
$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, $x \in \mathbb{R}^n$ and $m \leq n$.

$$
(\mathrm{D})\begin{cases} \max & y^T b \\ \text{s.t.} & A^T y + z = c \\ & z \geq 0 \\ & y \in \mathbb{R}^m \end{cases}
$$

where $z \in \mathbb{R}^n$.

Interior Point Method

- The Karush–Kuhn–Tucker conditions.
- Mehrotra's predictor-corrector method.
- In practice

$$\begin{bmatrix} -\Theta^{-1} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_d - X^{-1} r_a \\ r_p \end{bmatrix} \tag{1}$$

  where $\Theta = Z^{-1} X$.

$$(A \Theta A^T) \Delta y = A \Theta (r_d - X^{-1} r_a) + r_p \,. \tag{2}$$

- Hybrid preconditioner.

Hybrid preconditioner

This approach assumes the existence of two phases during interior point iterations.

$$M^{-1}(A\Theta A^T)M^{-T}\bar{y} = M^{-1}\big(A\Theta(r_d - X^{-1}r_a) + r_p\big). \tag{3}$$

where $\bar{y} = M^T \Delta y$.

Phases

1. The controlled Cholesky factorization preconditioner;
2. The splitting preconditioner.

Controlled Cholesky factorization preconditioner

Consider:

- The Cholesky factorization $A\Theta A^T = LL^T = \tilde{L}\tilde{L}^T + R$
  Defining $E = L - \tilde{L}$

$$\tilde{L}^{-1}(A\Theta A^T)\tilde{L}^{-T} = (\tilde{L}^{-1}L)(\tilde{L}^{-1}L)^T = (I + \tilde{L}^{-1}E)(I + L^{-1}E)^T .$$

- The problem

$$\text{minimize} \, \|E\|_F^2 = \sum_{j=1}^{m} c_j \quad \text{with} \quad c_j = \sum_{i=1}^{m} |\ell_{ij} - \tilde{\ell}_{ij}|^2 .$$

Now c_j can be split in two summations:

$$c_j = \sum_{j=1}^{t_j+\eta} |\ell_{i_k j} - \tilde{\ell}_{i_k j}|^2 + \sum_{i=t_j+\eta+1}^{m} |\ell_{ij}|^2 .$$

Controlled Cholesky factorization preconditioner improvements

- Choice of entries by value
- Generalization of improved ICF
- Versatile preconditioner

η	M	Storage
$-m$	$\text{diag}(A\Theta A^T)^{-1/2}$	less than $A\Theta A^T$
0	$\tilde{L}$	equal to $A\Theta A^T$
m	L	more than $A\Theta A^T$

- Predictable storage
- Avoiding loss of positive definiteness by exponential shift

Avoiding loss of positive definiteness by exponential shift

Manteufell et. al., proved that if the coefficient matrix V is symmetric positive definite then there is a constant $\sigma > 0$ such that an incomplete factorization $V + \sigma I$ exists.

CCF uses an exponential shift $\sigma_i = 5 \cdot 10^{-4} \cdot 2^{i-1}$ in order to have smaller diagonal perturbations.

New values

- $\sigma_i = 5 \cdot 10^{-(i+1)}$ (named CCF10);
- $\sigma_i = \text{tol} + 5 \cdot 10^{-4} \cdot 2^{i-1}$ (named CCFe);
- $\sigma_i = \text{tol} + 5 \cdot 10^{-4} \cdot 2^{i-1} + \alpha$ (named CCFp).

where $\text{tol} = 10^{-8}$.

Idea

Algorithm 1: LDL^T Fatorization

Input: A matrix simétric definide positive A and n the order the A

Output: The matrix triangular L and the matrix diagonal D

for $j \leftarrow 1$ ***to*** n **do**

$$d_j = a_{jj} - \sum_{k=1}^{j-1} \ell_{jk}^2 d_k$$

for $i \leftarrow j+1$ ***to*** n **do**

$$\ell_{ij} = \frac{1}{d_j}\left(a_{ij} - \sum_{k=1}^{j-1} \ell_{ik} d_k \ell_{jk}\right)$$

return $L,\ D$;

Idea

Let be A scaled, for $j = 1, \ldots, n$

$$d_j = 1 - \sum_{k=1}^{j-1} \ell_{jk}^2 d_k \quad \text{or} \quad d_j = \mathrm{K} - d_{j-1}\ell_{j(j-1)}^2 \,,$$

where $\mathrm{K} = 1 - \sum_{k=1}^{j-2} \ell_{jk}^2 d_k$. So

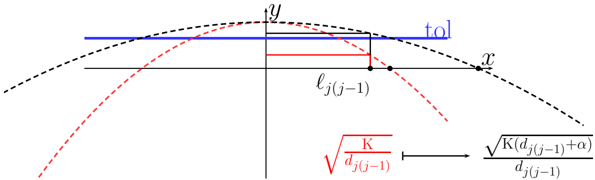
$$y = \mathrm{K} - d_{j-1}x^2 \,. \tag{4}$$

where $y = d_j$ e $x = \ell_{j(j-1)}$.
If $d_j <$ tol, we propose a new method for calculate σ using this parabola. Then we a new parabola such that

$$d_j^{\text{new}} > \text{tol} \,.$$

Idea

$$\text{A new parabola}: \quad y^{\text{new}} = K^{\text{new}} - \frac{d_{j-1}}{d_{j-1}+\alpha} d_{j-1} x^2 \,,$$



where and $K^{\text{new}} = \begin{cases} K & \text{if } K > 0; \\ K + (tol - d_j) & \text{else.} \end{cases}$

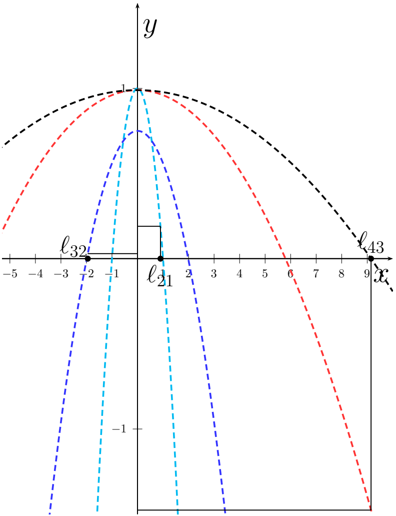
We want $d_j^{\text{new}} \geq \text{tol}$. So we need

$$\alpha \begin{cases} \geq \frac{d_{j-1}(\text{tol}-d_j)}{d_{j-1}\ell^2_{j(j-1)} - (\text{tol}-d_j)} & \text{if } K > 0; \\ > 0 & \text{else.} \end{cases}$$

Example

$$V = \begin{bmatrix} 1 & 0.9 & 0.5 & 0.1 & 0 \\ 0.9 & 1 & 0.08 & 0.1 & 0 \\ 0.5 & 0.008 & 1 & 0.3 & 0.09 \\ 0.1 & 0.1 & 0.3 & 1 & 0.9 \\ 0 & 0 & 0.09 & 0.9 & 1 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0.9 & 0.19 & 0 & 0 & 0 \\ 0.5 & -1.9474 & 0.02945 & 0 & 0 \\ 0.1 & 0.052632 & 9.1502 & -1.4763 & 0 \end{bmatrix}$$

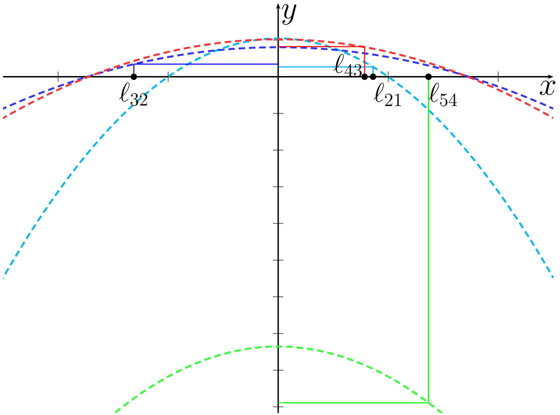


Example

$$\alpha_1 = \frac{d_3(\text{tol} - d_4)}{d_3\ell_{43}^2 - (\text{tol} - d_3)} = 0.043756\,.$$

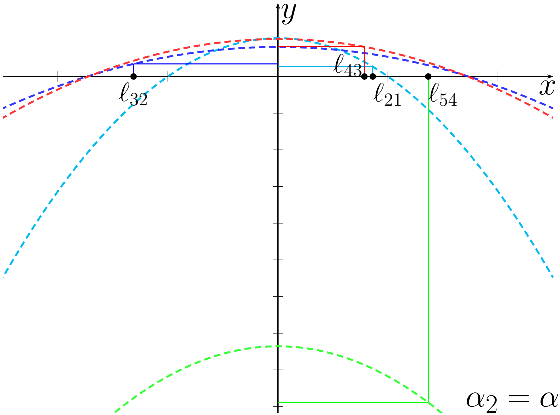
$$V + \alpha_1 I = \begin{bmatrix} 1.043756 & 0.9 & 0.5 & 0.1 & 0 \\ 0.9 & 1.043756 & 0.08 & 0.1 & 0 \\ 0.5 & 0.008 & 1.043756 & 0.3 & 0.09 \\ 0.1 & 0.1 & 0.3 & 1.043756 & 0.9 \\ 0 & 0 & 0.09 & 0.9 & 1.043756 \end{bmatrix}$$

Example



$$L = \begin{bmatrix} 1.043756 & 0 & 0 & 0 & 0 \\ 0.86227 & 0.26771 & 0 & 0 & 0 \\ 0.47904 & -1.31161 & 0.34368 & 0 & 0 \\ 0.09581 & 0.05145 & 0.78608 & 0.8211 & 0 \\ 0.86227 & -2.89882 & -3.95426 & 1.36644 & -8.88887 \end{bmatrix}$$

Example



$$\alpha_2 = \alpha_1 + 0.43756$$

$$L = \begin{bmatrix} 1.043756 & 0 & 0 & 0 & 0 \\ 0.86227 & 0.26771 & 0 & 0 & 0 \\ 0.47904 & -1.31161 & 0.34368 & 0 & 0 \\ 0.09581 & 0.05145 & 0.78608 & 0.8211 & 0 \\ 0.86227 & -2.89882 & -3.95426 & 1.36644 & -8.88887 \end{bmatrix}$$

Example

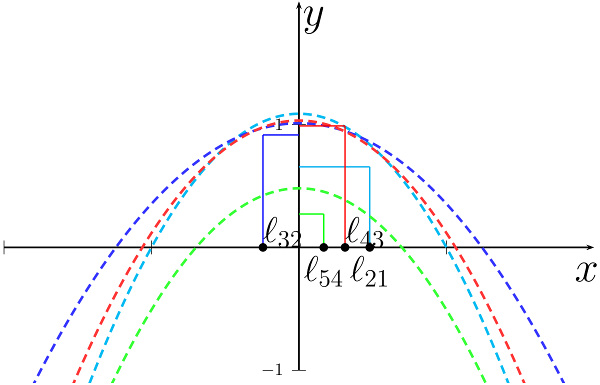
With

$$\alpha_2 = \alpha_1 + 0.043756$$

A new matrix is

$$V + \alpha_2 I = \begin{bmatrix} 1.08751 & 0.9 & 0.5 & 0.1 & 0 \\ 0.9 & 1.08751 & 0.08 & 0.1 & 0 \\ 0.5 & 0.008 & 1.08751 & 0.3 & 0.09 \\ 0.1 & 0.1 & 0.3 & 1.08751 & 0.9 \\ 0 & 0 & 0.09 & 0.9 & 1.08751 \end{bmatrix}$$

Example



$$L = \begin{bmatrix} 1.08751 & 0 & 0 & 0 & 0 \\ 0.47997 & 0.65554 & 0 & 0 & 0 \\ 0.26665 & -0.24406 & 0.91514 & 0 & 0 \\ 0.05333 & 0.07933 & 0.31255 & 0.98866 & 0 \\ 0.47997 & -0.65897 & -0.27910 & 0.16801 & 0.27168 \end{bmatrix}$$

Numerical experiments

NETLIB problems

	Iteration					**Time performance**				
Problem	CCF	CCF10	CCFe	CCFp	CCFβ	CCF	CCF10	CCFe	CCFp	CCFβ
bnl2	37	37	37	37	37	1.87	3.11	2.45	2.85	3.77
degen3	16	16	16	16	16	5.40	4.09	4.88	4.31	3.89
nesm	31	31	31	31	31	1.62	1.65	1.56	1.67	1.82
scsd8-2b-64	7	7	7	7	7	1.37	1.29	1.29	1.32	1.40
scsd8-2c-64	7	7	7	7	7	1.35	1.26	1.26	1.32	1.36
scsd8-2r-432	18	18	18	18	18	10.30	10.05	10.05	10.41	10.30
stocfor2	21	23	21	21	21	1.20	1.41	1.20	1.20	1.47
stocfor3	32	32	32	32	32	87.88	78.78	89.08	102.16	80.67
BL	38	39	38	38	38	18.04	16.16	17.50	15.43	38.61

QAP problems

	Iteration					Time performance				
Problem	CCF	CCF10	CCFe	CCFp	CCFβ	CCF	CCF10	CCFe	CCFp	CCFβ
els19	31	31	31	31	31	43.47	43.83	43.53	46.49	44.92
chr22b	29	29	29	29	29	19.76	19.35	18.55	20.14	17.15
chr25a	29	28	28	28	29	42.48	41.61	42.00	41.73	36.24
scr15	24	24	24	24	24	7.76	6.63	7.70	6.66	6.33
scr20	21	21	22	21	21	60.98	55.73	59.40	56.19	54.35
rou20	24	24	24	23	24	755.41	625.90	845.45	625.16	663.19

KENNINGTON problems

Problem	Iteration					Time performance				
	CCF	CCF10	CCFe	CCFp	CCFβ	CCF	CCF10	CCFe	CCFp	CCFβ
cre-a	27	27	27	27	27	7.63	5.44	6.81	5.53	4.61
cre-b	43	43	43	43	43	43.28	52.84	42.86	53.66	51.98
cre-c	27	26	26	26	26	5.74	4.45	5.56	4.61	3.07
cre-d	42	41	41	43	43	27.91	33.23	27.82	34.28	32.85
ex05	39	39	39	65	65	5.85	12.09	5.61	13.29	11.99
ex09	45	51	45	51	51	51.01	61.97	45.92	64.46	59.75
ken11	23	22	22	22	23	10.17	13.78	9.78	12.49	15.34
ken13	29	29	28	32	29	93.64	102.12	88.38	100.18	141.62
ken18	41	41	40	42	38	1014.74	1287.95	943.40	1286.81	1096.73
pds-06	39	39	39	40	38	8.19	7.37	8.17	7.26	38.98
pds-10	47	47	47	46	47	18.48	14.70	17.41	14.33	168.17
pds-20	60	62	61	62	59	214.59	213.91	209.32	229.33	928.28
pds-40	78	77	78	79	77	395.70	376.30	407.84	370.89	4521.67
pds-60	84	83	84	83	84	1076.86	1004.52	1082.16	1053.24	11160.55
pds-80	83	83	83	83	83	1509.34	1339.18	1551.40	1351.95	15848.94
pds-100	87	85	85	87	83	2573.59	2419.53	2322.75	2507.77	28371.93

The Bibliography

- Bocanegra S, Campos F. F. Oliveira A. R. L. 2007. Using a hybrid preconditioner for solving large-scale linear systems arising from interior point methods. Computational Optimization and Applications (1-2): 149-164. Special issue on Linear Algebra Issues arising in Interior Point Methods.
- F.F.Campos. Analysis of conjugate Gradients - type methods for solving linear equations. Phd tesis,Oxford UniversityComputing Laboratory, Oxford, 1995.
- A.R L.Oliveira, D.C. Sorensen, A new class of preconditioners for large-scale linear systems from interior point methods for linear programming, Linear Algebra and Its Applications 394(2005), pp. 1-24.
- S. J. Wright, Primal Dual Interior Point Methods, SIAM Publications, SIAM, Philadelphia,PA, USA, 1997