Linear Algebra Software in Nonlinear Optimization

Elizabeth Wong

Department of Mathematics
University of California, San Diego

Joint work with:
Philip E. Gill & Michael A. Saunders

SIAM Meeting on Applied Linear Algebra,
Atlanta, October 26–30, 2015.
Introduction

Optimization methods require the solution of many systems:

\[ Kv = r, \quad \text{with} \quad K = \begin{pmatrix} H & J^T \\ J & -G \end{pmatrix}. \]

- \( H \) is \( p \times p \) and symmetric.
- \( J \) is \( m \times p \).
- \( G \) is \( m \times m \) and symmetric positive semidefinite.
- \( J \) and \( H \) are \textit{sparse} with \( p \) and \( m \) \textit{large}. 
In some cases we must reject the KKT matrix $K$ and solve with

$$\tilde{K} = \begin{pmatrix} H + E & J^T \\ J & -G \end{pmatrix},$$

for some positive semidefinite $E$.

If $K$ is symmetric, then the matrix inertia of $K$ is the ordered triple

$$\text{In}(K) = (i_P, i_N, i_0),$$

where

$$\begin{cases} i_P = \# \text{ positive eigenvalues of } K \\ i_N = \# \text{ negative eigenvalues of } K \\ i_0 = \# \text{ zero eigenvalues of } K \end{cases}$$
\[ K = \begin{pmatrix} H & J^T \\ J & -G \end{pmatrix}, \quad H \ p \times \ p \quad J \ m \times \ p \]

\( G \) is positive semidefinite (typically, \( G \) is a positive-definite diagonal).

The matrix must have \textit{correct inertia}

\[ \text{In}(K) = (p, m, 0). \]

\[ \text{In}(K) = \text{In}(H + J^T G^{-1} J) \quad + \quad (0, m, 0) \quad \text{when} \ G \neq 0 \]
\[ \text{In}(K) = \text{In}(Z^T H Z) \quad + \quad (m, m, 0) \quad \text{when} \ G = 0 \]

\( Z^THZ \) is the reduced Hessian, with the columns of \( Z \) forming a basis for the nullspace of \( J \).
Otherwise, we must find
\[ \bar{K} = \begin{pmatrix} H + E & J^T \\ J & -G \end{pmatrix}, \]
with \( m \) negative eigenvalues.

The “least” \( E \) is known, but not easily computed (Cheng and Higham ’98).
Result (Inertia-controlling factorization)

There exists a permutation $P$ and a positive semidefinite $E$ such that

$$\ln(\bar{K}) = (p, m, 0) \quad \text{and} \quad P\bar{K}P^T = LDL^T$$

where $L$ is unit lower-triangular and $D$ is block-diagonal.

If $\ln(K) = (p, m, 0)$, then $E = 0$, i.e., $\bar{K} = K$.

(Forsgren '92, Forsgren & G, '98).

This method is based on deferring certain pivots during the $LDL^T$ factorization.

- Interfering with the pivot order increases “fill” in $L$.
- $\|L\|$ cannot be bounded independently of $K$.
- We want a method that can exploit “black box” software
Identify a subset of the columns of $J$ to form a square $m \times m$ matrix $B$ such that
\[
(B \quad -G) \quad \text{has full row rank.}
\]
The remaining columns of $J$ form the $m \times n_s$ matrix $S$.

\[
K = \begin{pmatrix}
H & J^T \\
J & -G
\end{pmatrix} \sim \begin{pmatrix}
H_{BB} & B^T & H_{BS}^T \\
B & -G & S \\
H_{BS} & S^T & H_{SS}
\end{pmatrix}.
\]

The smaller KKT matrix in the $(1, 1)$ block of $K$

\[
K_{BB} = \begin{pmatrix}
H_{BB} & B^T \\
B & -G
\end{pmatrix}
\]

is nonsingular and has $m$ negative eigenvalues.
Identifying $B$

Given the KKT matrix, we create $2 \times 2$ “tile” matrices of the form

$$T_{ij} = \begin{pmatrix} h & a \\ \bar{a} & -g \end{pmatrix},$$

where $h$ is an element of $H$, $a$ and $\bar{a}$ are elements of $A$, and $g$ is an element of $G$.

These tiles form a symmetric $2m \times 2m$ “checkerboard” matrix $T$ of tiles from $K_{BB}$, i.e.,

$$T = \begin{pmatrix} T_{11} & T_{12} & T_{13} & \cdots \\ T_{12}^T & T_{22} & T_{23} & \cdots \\ T_{13}^T & T_{23}^T & T_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \sim \begin{pmatrix} H_{BB} & B^T \\ B & -G \end{pmatrix} = K_{BB}$$
A permutation matrix $\Pi$ is defined such that the upper left-hand corner of a symmetrically permuted version of $K$ consists of

$$C = \Pi^T K \Pi = \begin{pmatrix} T & F^T \\ F & M \end{pmatrix}$$

with $T$ nonsingular. Then

$$C / T \equiv T - FM^{-1}F^T.$$

is the \textit{Schur complement} of $T$ in $C$. 
\[ C = \begin{pmatrix} T & F^T \\ F & M \end{pmatrix} \quad T \sim \begin{pmatrix} H_{BB} & B^T \\ B & -G \end{pmatrix} \]

- If we compute the LDL\(^T\) factors of \( C \), the permuted and tiled version \( K \), in such a way that \( T \) is eliminated first, the remaining Schur complement must be positive definite for \( \ln(K) \) to be correct.
- If the Schur complement is not positive definite, then we can implicitly modify \( H \) based on the LDL\(^T\) factors such that \( \bar{K} \) will have correct inertia.
- Eliminating \( T \) first allows us to safely modify the factors (modifications are restricted to \( H \)).
Nonlinear optimization

Our interest is in *sequential quadratic programming (SQP)* methods to solve nonlinear optimization problem:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad c(x) - s = 0, \quad x \geq 0, \quad s \geq 0.
\end{align*}
\]

This involves solving a sequence of *quadratic subproblems (QP)*

\[
\begin{align*}
\text{minimize} & \quad g^T(x - x_0) + \frac{1}{2}(x - x_0)^TH(x - x_0) \\
\text{subject to} & \quad c + A(x - x_0) - s = 0, \quad x \geq 0, \quad s \geq 0.
\end{align*}
\]

- \(H\) is a positive-semidefinite approximation or the exact Hessian of the Lagrangian function.
- \(A\) is the \(m \times n\) Jacobian of \(c(x)\) at \(x_0\).
The majority of the work in solving the QP subproblem involves solving a sequence of linear systems with a KKT matrix defined by a set of *basis variables*:

\[
K_B = \begin{pmatrix} H_B & A_B^T \\ A_B & 0 \end{pmatrix}
\]

where \( H_B \) is symmetric and \( n_B \times n_B \), and \( A_B \) is \( m \times n_B \) with rank \( m \).

Basis variables must define \( K_B \) such that \( K_B \) has *correct inertia*

\[
\text{In}(K_B) = (n_B, m, 0) = \text{In}(Z^THZ) + (m, m, 0).
\]
$K_B$ and its associated set of basis variables are rejected if $K_B$ does not have the correct inertia.

- QP method will maintain correct inertia.
- In the SQP context, an initial set of basis variables is always provided.

⇒ Apply the algorithm to find an initial set of basis variables such that $K_B$ has correct inertia.
Given an initial set of basis variables, we identify the \textit{basic} and \textit{superbasic} variables:

\[ A_B \sim (B \quad S) \]

with \( B \) \( m \times m \) and nonsingular, and \( S \) an \( m \times n_s \) matrix. To find \( B \), we compute the \textit{LU} of \( A_B^T \).

Compute the \textit{LDL}^T factors of \( K_{BB} \) to get a pivot ordering

\[ K_{BB} = \begin{pmatrix} H_{BB} & B^T \\ B & 0 \end{pmatrix} \quad \ln(K_B) = (m, m, 0). \]

Form \( K_B \) such that

\[
K_B = \begin{pmatrix} H_{BB} & B^T & \hline H_{BS}^T \\ B & 0 & \hline S^T \\ H_{BS} & S & \hline H_{SS} \end{pmatrix} \sim \begin{pmatrix} H_B & A_B^T \\ A_B & 0 \end{pmatrix}.
\]

The Schur complement of \( K_B \) is:

\[
H_{SS} - (H_{BS} \quad S^T) K_B^{-1} \begin{pmatrix} H_{BS}^T \\ S \end{pmatrix} = Z^T H Z.
\]
Compute a restricted ordering for $K_B$ such that the ordering from $K_{BB}$ is preserved (J. Hogg, RAL).

Compute the $LDL^T$ factors of $K_B$ using this restricted ordering.

- $Z^T H Z$ is implicitly formed and factored.
- Sparsity in $Z^T H Z$ is exploited.
- Once $K_{BB}$ has been eliminated, it is safe to modify the factors.

Given the factors of $K_B$, we can

- remove any superbasics that cause incorrect inertia, or

- modify the appropriate components of $H$ so that $K_B$ will have correct inertia.
  
  $\Rightarrow$ modifications to $H$ affect the reduced Hessian $Z^T H Z$. 
## Some numerical results

<table>
<thead>
<tr>
<th>Problem</th>
<th>$m$</th>
<th>Initial $n_S$</th>
<th>Final $n_S$</th>
<th># elts in $K$</th>
<th># elts in $L$ (restricted) (unrestricted)</th>
<th>Density of $L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVGASA</td>
<td>10</td>
<td>8</td>
<td>8</td>
<td>75</td>
<td>254 241</td>
<td>63% 59%</td>
</tr>
<tr>
<td>DUALC1</td>
<td>10</td>
<td>2</td>
<td>2</td>
<td>1294</td>
<td>1975 1954</td>
<td>2% 2%</td>
</tr>
<tr>
<td>DUALC2</td>
<td>229</td>
<td>2</td>
<td>2</td>
<td>1378</td>
<td>2101 2080</td>
<td>2% 2%</td>
</tr>
<tr>
<td>GMNCAE1</td>
<td>300</td>
<td>175</td>
<td>23</td>
<td>18973</td>
<td>37752 35957</td>
<td>19% 18%</td>
</tr>
<tr>
<td>GMNCAE2</td>
<td>1050</td>
<td>175</td>
<td>175</td>
<td>43499</td>
<td>100824 80059</td>
<td>4% 3%</td>
</tr>
<tr>
<td>GOULDQQP1</td>
<td>18</td>
<td>32</td>
<td>23</td>
<td>114</td>
<td>429 363</td>
<td>41% 35%</td>
</tr>
<tr>
<td>HS118</td>
<td>17</td>
<td>15</td>
<td>15</td>
<td>104</td>
<td>513 415</td>
<td>42% 34%</td>
</tr>
<tr>
<td>QPCBOEI1</td>
<td>351</td>
<td>384</td>
<td>384</td>
<td>4922</td>
<td>147737 23634</td>
<td>25% 4%</td>
</tr>
<tr>
<td>QPCBOEI2</td>
<td>166</td>
<td>143</td>
<td>143</td>
<td>1837</td>
<td>33671 7664</td>
<td>30% 7%</td>
</tr>
<tr>
<td>QPNBOEI2</td>
<td>166</td>
<td>143</td>
<td>44</td>
<td>1351</td>
<td>10617 5907</td>
<td>15% 8%</td>
</tr>
<tr>
<td>QPNSTAIR</td>
<td>356</td>
<td>467</td>
<td>72</td>
<td>3098</td>
<td>24011 11747</td>
<td>8% 4%</td>
</tr>
</tbody>
</table>

HSL_MA97 + J. Hogg’s restricted ordering code
Conclusions

- When $G \neq 0$, symmetric indefinite solvers are the only viable option.
- In practice, $K$ is likely to have correct inertia except for the early iterations, so we can afford to use a vanilla $\text{LDL}^T$ first.
- The method described is effectively deferring all pivots except “HG” pivots. The inertia-controlling factorization defers fewer pivots but needs a better implementation.


Software information at: [http://ccom.ucsd.edu/~optimizers](http://ccom.ucsd.edu/~optimizers)