Update LU factors

10/11/2013
Zack
Update LU Factors

- Replace one column (or one row)
  \[ \tilde{A} = A + (\bar{a} - a)e_r^T \]

- Add a rank-1 matrix \( \sigma vw^T \)
  \[ \tilde{A} = A + \sigma vw^T \]
BGR update

\[
\tilde{A} = A + (\tilde{a} - a)e_r^T
= L(U + (\tilde{u} - u)e_r^T)
= LU'
\]

the last nonzero element of \(\tilde{u}\) will be in row \(l\), and to restore triangularity we need to find an \(LU\) factorization of \(U'\).
BGR update

\[ U' = \]

\[ U'' = P^T U' P = \]

let \( P \) be a cyclic permutation that moves the \( r \)th column and row of \( U' \) to position \( l \) and shifts the intervening columns and rows forward.
BGR update

• Ideally the existing diagonals in rows $r : l - 1$ will be large enough to serve as pivots, but to ensure stability we must allow row interchanges.

\[
U'' \equiv P^T U' P =
\]

![Diagram showing row interchanges](image)
FT update

- the FT update is equivalent to the BGR update with the restriction that \( l = m \) and row interchanges are not allowed in the factorization \( U' \).

- numerical stability of the FT update is not completely reliable. Nevertheless, the ease of implementation means that the FT update has been adopted in virtually all sparse implementations of the simplex method, except LA05, LA15, and LUSOL.
FT update

• The reason why FT update works well in practice
BLU update

\[
\begin{pmatrix}
A_0 & V \\
E^T & 0
\end{pmatrix}
\begin{pmatrix}
y \\
z
\end{pmatrix}
= \begin{pmatrix}
b \\
0
\end{pmatrix}
\]

V = (v_1, v_2 ...) have replaced columns \( r_1, r_2, ... \) of \( A_0 \)

columns of \( E \) are columns \( r_1, r_2, ... \) of the \( m \times m \) identity
BLU update

• Example:

\[
\begin{pmatrix}
  a_{11} & a_{12} & a_{13} & a_{14} & v_{11} & v_{12} \\
  a_{21} & a_{22} & a_{23} & a_{24} & v_{21} & v_{22} \\
  a_{31} & a_{32} & a_{33} & a_{34} & v_{31} & v_{23} \\
  a_{41} & a_{42} & a_{43} & a_{44} & v_{41} & v_{24} \\
  1 &  &  &  & 1 & \\
  1 &  &  &  & &
\end{pmatrix}
\begin{pmatrix}
  x_1 \\
  x_2 \\
  x_3 \\
  x_4 \\
  x_5 \\
  x_6
\end{pmatrix}
= 
\begin{pmatrix}
  b_1 \\
  b_2 \\
  b_3 \\
  b_4 \\
  0 \\
  0
\end{pmatrix}
\]
BLU update

• block-triangular factorization

\[
\begin{pmatrix}
A_0 & V \\
E^T & \end{pmatrix} = \begin{pmatrix}
L_0 & \\
Z^T & I \\
\end{pmatrix}\begin{pmatrix}
U_0 & Y \\
& C \\
\end{pmatrix}
\]

Where \( L_0 Y = V \), \( Z^T U_0 = E^T \), and \( C = -Z^T Y \)
BLU update

\[
\begin{bmatrix}
L_0 \\
Z^T
\end{bmatrix}
\begin{bmatrix}
I \\
Y
\end{bmatrix}
\begin{bmatrix}
y \\
z
\end{bmatrix}
=
\begin{bmatrix}
b \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
L_0 \\
Z^T
\end{bmatrix}
\begin{bmatrix}
w \\
w'
\end{bmatrix}
=
\begin{bmatrix}
b \\
0
\end{bmatrix}
\]

Where \( L_0w = b, w' = -Z^Tw \)
BLU update

\[
\begin{bmatrix}
U_0 & Y \\
C & Y
\end{bmatrix}
\begin{bmatrix}
y \\
z
\end{bmatrix}
= 
\begin{bmatrix}
w \\
w'
\end{bmatrix}
\]

\(Cz = w', U_0y = w - Yz\)
BLU update

• we can solve $Ax = b$ by the following sequence:

Solve $L_0 w = b$
Solve $C z = -Z^T w$
Solve $U_0 y = w - Y z$
Extract $x$ from the appropriate parts of $\begin{pmatrix} y \\ z \end{pmatrix}$
BLU update

The vectors in $Y$ were often rather dense, but it did save time to store them and avoid double solves with $A_0$.

On most problems, the BLU updates were found to be faster than LUSOL implementation of the Bartels-Golub-Reid update.
Rank-1 update

\[ \overline{A} = A + \sigma w^T = L(U + \sigma cw^T) = L \begin{pmatrix} c & U \end{pmatrix} \begin{pmatrix} \sigma w^T \\ I \end{pmatrix} \]

\[ (c \ U) = \tilde{L} \tilde{U} \quad \tilde{U} = (\beta e_i \ U) \]

\[ \overline{A} = \tilde{L} \tilde{L}(\tilde{U} + \beta \sigma e_i w^T) = \tilde{L} \tilde{L} \tilde{U} \]
Thank you