FAST ALGORITHMS FOR TRIANGULAR FACTORIZATION OF MATRICES

notation : * everything is to be understood in block-sense (transpose, row, symmetry ...) unless otherwise said.

* ^{T} : block transpose
  ^{t} : transpose element by element
  $^\tau$: transpose of the elements in the blocks

$T = t\tau = \tau t$
$\tau = Tt = tT$
$t = T\tau = \tau T$
$(-1)T \neq T(-1)$

Justification

NORMAL EQUATIONS

$$[I\ A_{m1} \cdots A_{mm}] \begin{bmatrix} R_0 & R_1 & \cdots & R_m \\ R_{-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & R_1 \\ R_{-m} & \cdots & R_{-1} & R_0 \end{bmatrix} = [R_m^\varepsilon\ 0 \cdots 0]$$

$$[B_{mm} \cdots\ B_{m1} I] \begin{bmatrix} & & \\ & & \\ & & \end{bmatrix} = [0 \cdots 0\ R_m^r]$$

$$\begin{cases} A_{m+1} R_{m+1} = R_m^\varepsilon \\ B_{m+1} R_{m+1} = R_m^r \end{cases} \rightarrow \begin{cases} A_{m+1} = R_m^\varepsilon R_{m+1}^{-1} = R_m^\varepsilon (R_{m+1}^{-1})_{1\cdot} \\ B_{m+1} = R_m^r R_{m+1}^{-1} = R_m^r (R_{m+1}^{-1})_{m+1\cdot} \end{cases}$$

Cholesky factorization ($R^t = R$)

$$R_m = L_m D_m L_m^t$$

$$B_m \sim (R_m^{-1})_{m\cdot} = (L_m^{-t} D_m^{-1} L_m^{-1})_{m\cdot}$$

$$\sim (L_m^{-1})_{m\cdot}$$

$$R_m = \bar{U}_m \bar{D}_m \bar{U}_m^t$$

$$A_m \sim (R_m^{-1})_{1\cdot} = \bar{U}_m^{-t} \bar{D}_m^{-1} \bar{U}_m^{-1}$$

$$\sim (\bar{U}_m^{-1})_{1\cdot}$$

Bierman calls this a U-D factorization

if $\tilde{L} \triangleq L D^{1/2}$ and $\tilde{U} = \bar{U} \bar{D}^{1/2}$ then
$R = \tilde{L} \tilde{L}^t$ and $R = \tilde{U} \tilde{U}^t$
$\tilde{L}$ and $\tilde{U}$ are called square roots of R

thus Levinson computes L^{-1} and $\bar{U}^{-1}$ row by row

Problems to solve ($\mathcal{R}$ Toeplitz)

	solve the system	
$\mathcal{R}\,x = c$ $c^T \triangleq [\gamma_0\ \gamma_1 \cdots]$ or $c^T \triangleq [\cdots\ \gamma_1\ \gamma_0]$	$\mathcal{S}$	$\mathcal{R}^T x^* = c$ $c^T \triangleq [\gamma_0\ \gamma_1 \cdots]$ or $c^T \triangleq [\cdots\ \gamma_1\ \gamma_0]$
	compute	
$\mathcal{R}^{-1}$	$\mathcal{I}$	$(\mathcal{R}^T)^{-1}$
	compute	
$\mathcal{R} = \mathcal{L}\,\mathcal{D}\,\mathcal{U}$ $\mathcal{R}^{-1} = \mathcal{U}^{-1}\,\mathcal{D}^{-1}\,\mathcal{L}^{-1}$ $\mathcal{R} = \bar{\mathcal{U}}\,\bar{\mathcal{D}}\,\bar{\mathcal{L}}$ $\mathcal{R}^{-1} = \bar{\mathcal{L}}^{-1}\,\bar{\mathcal{D}}^{-1}\,\bar{\mathcal{U}}^{-1}$		$\mathcal{R}^T = \mathcal{L}^*\,\mathcal{D}^*\,\mathcal{U}^*$ $(\mathcal{R}^T)^{-1} = \mathcal{U}^{*-1}\,\mathcal{D}^{*-1}\,\mathcal{L}^{*-1}$ $\mathcal{R}^T = \bar{\mathcal{U}}^*\,\bar{\mathcal{D}}^*\,\bar{\mathcal{L}}^*$ $(\mathcal{R}^T)^{-1} = \bar{\mathcal{L}}^{*-1}\,\bar{\mathcal{D}}^{*-1}\,\bar{\mathcal{U}}^{*-1}$

A Toeplitz matrix is persymmetric [↗↘]
(= symmetry with respect to the cross diagonal)

$$E = \begin{bmatrix} & & I \\ & \iddots & \\ I & & \end{bmatrix} \qquad E\,E = I$$

Then: $\boxed{E\,\mathcal{R}^T E = \mathcal{R}}$ $\qquad$ $\boxed{E\,\mathcal{R}^{-1} E = (\mathcal{R}^T)^{-1}}$

These are the only two relations the solutions of the above mentioned problems rely upon

($\mathcal{R}$ persymmetric $\Rightarrow$ $\mathcal{R}^{-1}$ persymmetric : only in SCALAR case)

define

$$r_m^T \triangleq [R_{-1}\ R_{-2}\ \cdots\ R_{-m}] \qquad \overset{*}{r}{}_m^T \triangleq [R_1\ R_2\ \cdots\ R_m]$$

then we have

$$E R^T E = R \qquad E R^{-1} E = (R^T)^{-1}$$

$$R_{m+1} \triangleq \left[\begin{array}{c|c} R_0 & \overset{*}{r}{}_m^T \\ \hline r_m & R_m \end{array}\right] = \left[\begin{array}{c|c} R_m & \hat{\overset{*}{r}}_m \\ \hline \hat{r}_m^T & R_0 \end{array}\right] \qquad R_{m+1}^T \triangleq \left[\begin{array}{c|c} R_0 & r_m^T \\ \hline \overset{*}{r}_m & R_m^T \end{array}\right] = \left[\begin{array}{c|c} R_m^T & \hat{r}_m \\ \hline \hat{\overset{*}{r}}_m^T & R_0 \end{array}\right]$$

$$R_{m+1}^{-1} \triangleq \left[\begin{array}{c|c} Q_m & f_m^T \\ \hline h_m & M_m \end{array}\right] = \left[\begin{array}{c|c} \overset{*}{M}{}'_m & \hat{\overset{*}{h}}_m \\ \hline \hat{\overset{*}{f}}_m^T & \overset{*}{Q}_m \end{array}\right] \qquad (R_{m+1}^T)^{-1} \triangleq \left[\begin{array}{c|c} \overset{*}{Q}_m & \overset{*}{f}{}_m^T \\ \hline \overset{*}{h}_m & \overset{*}{M}_m \end{array}\right] = \left[\begin{array}{c|c} M'_m & \hat{\overset{*}{h}}_m \\ \hline \hat{f}_m^T & Q_m \end{array}\right]$$

$$\overset{*}{M}{}' = E \overset{*}{M} E \qquad M' = EME$$

define

$$\lambda_m = Q_m^{-1} \qquad \overset{*}{\lambda}_m = \overset{*}{Q}_m^{-1}$$

$$F_m = Q_m^{-1} f_m = \lambda_m f_m \qquad \overset{*}{F}_m = \overset{*}{Q}_m^{-1} \overset{*}{f}_m = \overset{*}{\lambda}_m \overset{*}{f}_m$$

$$H_m = h_m Q_m^{-1} = h_m \lambda_m \qquad \overset{*}{H}_m = \overset{*}{h}_m \overset{*}{Q}_m^{-1} = \overset{*}{h}_m \overset{*}{\lambda}_m$$

① INVERSION (I)

Writing down $R_{m+1} R_{m+1}^{-1} = I_{m+1}$ and $R_{m+1}^T (R_{m+1}^T)^{-1} = I_{m+1}$ will give you after some algebra the <u>algorithm 1</u> (without the S-part) - (F, H, $\overset{*}{F}$, $\overset{*}{H}$ definitions appear in a natural way)

② SOLUTION OF THE SYSTEMS (S')

$R\,x = c$	$R^T x^* = c$
$c^T \triangleq [\gamma_0\ \gamma_1 \cdots\ ]$	$c^T \triangleq [\gamma_0\ \gamma_1 \cdots\ ]$
$R_{m+1}\left(\begin{bmatrix} x_{m+1} \end{bmatrix} - \begin{bmatrix} x_m \\ 0 \end{bmatrix}\right)$ $= \begin{bmatrix} c_m \\ \gamma_m \end{bmatrix} - \left[\begin{array}{c\|c} R_m & \hat{r}_m^* \\ \hline \hat{r}_m^T & R_0 \end{array}\right] \begin{bmatrix} x_m \\ 0 \end{bmatrix}$ $= \begin{bmatrix} c_m \\ \gamma_m \end{bmatrix} - \begin{bmatrix} R_m x_m \\ \hat{r}_m^T x_m \end{bmatrix} \triangleq \begin{bmatrix} 0 \\ \mu_m \end{bmatrix}$	$R_{m+1}^T\left(\begin{bmatrix} x_{m+1}^* \end{bmatrix} - \begin{bmatrix} x_m^* \\ 0 \end{bmatrix}\right)$ $= \begin{bmatrix} c_m \\ \gamma_m \end{bmatrix} - \left[\begin{array}{c\|c} R_m^T & \hat{r}_m^* \\ \hline r_m^{*T} & R_0 \end{array}\right] \begin{bmatrix} x_m^* \\ 0 \end{bmatrix}$ $= \begin{bmatrix} c_m \\ \gamma_m \end{bmatrix} - \begin{bmatrix} R_m^T x_m^* \\ \hat{r}_m^{*T} x_m^* \end{bmatrix} \triangleq \begin{bmatrix} 0 \\ \mu_m^* \end{bmatrix}$
$\mu_m = \gamma_m - \hat{r}_m^{*T} x_m$	$\mu_m^* = \gamma_m - \hat{r}_m^{*T} x_m^*$
$\begin{bmatrix} x_{m+1} \end{bmatrix} - \begin{bmatrix} x_m \\ 0 \end{bmatrix} = R_{m+1}^{-1} \begin{bmatrix} 0 \\ \mu_m \end{bmatrix} = \begin{bmatrix} \hat{H}_m^* \\ Q_m^* \end{bmatrix} \mu_m$ $= \begin{bmatrix} \hat{H}_m^* \\ I \end{bmatrix} \theta_m$	$\begin{bmatrix} x_{m+1}^* \end{bmatrix} - \begin{bmatrix} x_m^* \\ 0 \end{bmatrix} = \cdots$
$\theta_m = Q_m^* \mu_m = \lambda_m^{*-1} \mu_m$	$\theta_m^* = \lambda_m^{-1} \mu_m^*$
$x_{m+1} = \begin{bmatrix} x_m \\ 0 \end{bmatrix} + \begin{bmatrix} \hat{H}_m^* \\ I \end{bmatrix} \theta_m$ $\theta_m = \lambda_m^{*-1} (\gamma_m - \hat{r}_m^T x_m)$ (1)	$x_{m+1}^* = \begin{bmatrix} x_m^* \\ 0 \end{bmatrix} + \begin{bmatrix} \hat{H}_m \\ I \end{bmatrix} \theta_m^*$ $\theta_m^* = \lambda_m^{-1} (\gamma_m - \hat{r}_m^{*T} x_m^*)$ (1)
$c^T \triangleq [\cdots\ \gamma_2\ \gamma_1\ \gamma_0]$	$c^T \triangleq [\cdots\ \gamma_2\ \gamma_1\ \gamma_0]$
$x_{m+1} = \begin{bmatrix} 0 \\ x_m \end{bmatrix} + \begin{bmatrix} I \\ H_m \end{bmatrix} \theta_m$ $\theta_m = \lambda_m^{-1} (\gamma_m - r_m^{*T} x_m)$ (2)	$x_{m+1}^* = \begin{bmatrix} 0 \\ x_m^* \end{bmatrix} + \begin{bmatrix} I \\ H_m^* \end{bmatrix} \theta_m^*$ $\theta_m^* = \lambda_m^{*-1} (\gamma_m - r_m^T x_m^*)$ (2)

$x^T R = c^T$

$c^T = [\cdots \gamma_1 \gamma_0]$

$$x_{m+1} = \begin{bmatrix} 0 \\ x_m \end{bmatrix} + \theta_m \begin{bmatrix} I \\ F_m \end{bmatrix}$$

$$\theta_m = (\gamma_m - x_m^T r_m)\lambda_m^{-1}$$

(3)

$\overset{*}{x}{}^T R^T = c^T$

$c^T = [\cdots \gamma_1 \gamma_0]$

$$\overset{*}{x}_{m+1} = \begin{bmatrix} 0 \\ \overset{*}{x}_m \end{bmatrix} + \overset{*}{\theta}_m \begin{bmatrix} I \\ \overset{*}{F}_m \end{bmatrix}$$

$$\overset{*}{\theta}_m = (\gamma_m - \overset{*}{x}_m^T \overset{*}{r}_m)\overset{*}{\lambda}_m^{-1}$$

(3)

$x^T R = c^T$

$c^T = [\gamma_0 \gamma_1 \cdots]$

$$x_{m+1} = \begin{bmatrix} x_m \\ 0 \end{bmatrix} + \theta_m \begin{bmatrix} \hat{\overset{*}{F}} \\ I \end{bmatrix}$$

$$\theta_m = (\gamma_m - x_m^T \hat{\overset{*}{r}}_m)\overset{*}{\lambda}_m^{-1}$$

(4)

$\overset{*}{x}{}^T R^T = c^T$

$c^T = [\gamma_0 \gamma_1 \cdots]$

$$\overset{*}{x}_{m+1} = \begin{bmatrix} \overset{*}{x}_m \\ 0 \end{bmatrix} + \overset{*}{\theta}_m \begin{bmatrix} \hat{F}_m \\ I \end{bmatrix}$$

$$\overset{*}{\theta}_m = (\gamma_m - \overset{*}{x}_m^T \hat{r}_m)\lambda_m^{-1}$$

(4)

If $c^T = -[R_1 R_2 \cdots]$ then

$x_m^{(3)} = F_m$; $\theta_m^{(3)} = \alpha_m$ (L) | $\overset{*}{x}_m^{(1)} = \overset{*}{H}_m$; $\overset{*}{\theta}_m^{(1)} = \overset{*}{\beta}_m$

If $c^T = -[R_{-1} R_{-2} \cdots]$ then

$x_m^{(1)} = H_m$; $\theta_m^{(1)} = \beta_m$ | $\overset{*}{x}_m^{(3)} = \overset{*}{F}$; $\overset{*}{\theta}_m^{(3)} = \overset{*}{\alpha}_m$ (L)

These are again the normal equations and in the symmetric case Levinson's algorithm is found again.

③ TRIANGULAR FACTORS OF R^{-1} and $(R^T)^{-1}$

$$R = \mathcal{L}\,\mathcal{D}\,\mathcal{U}$$

$$R_{m+1}^{-1} = \mathcal{U}_{m+1}^{-1}\,\mathcal{D}_{m+1}^{-1}\,\mathcal{L}_{m+1}^{-1}$$

$$\begin{bmatrix} \overset{*}{M}'_m & \hat{h}_m \\ \hat{f}_m^T & \overset{*}{Q}_m \end{bmatrix} = \begin{bmatrix} I & \mathcal{U}_m^{-1} & b_m \\ & I & \\ & & I \end{bmatrix} \begin{bmatrix} \mathcal{D}_m^{-1} & \\ & \delta_m^{-1} \end{bmatrix} \begin{bmatrix} I & & \\ & \mathcal{L}_m^{-1} & I \\ & a_m^T & I \end{bmatrix}$$

$$\begin{aligned} \delta_m^{-1} &= \overset{*}{Q}_m = \overset{*}{\lambda}_m^{-1} \\ b_m &= \hat{h}_m \delta_m = \hat{\overset{*}{H}}_m \\ a_m &= \delta_m \hat{f}_m = \hat{\overset{*}{F}}_m \end{aligned}$$

$\mathcal{U}_m^{-1} =$  ; $\mathcal{L}_m^{-1} =$ 

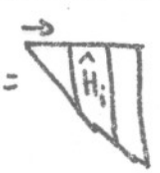
$$\mathcal{D}_m^{-1} = \mathrm{diag}\,(\overset{*}{\lambda}_0^{-1} \cdots \overset{*}{\lambda}_{m-1}^{-1})$$

$$R^T = \overset{*}{\mathcal{L}}\,\overset{*}{\mathcal{D}}\,\overset{*}{\mathcal{U}}$$

$$(R_{m+1}^T)^{-1} = \overset{*}{\mathcal{U}}_m^{-1}\,\overset{*}{\mathcal{D}}_{m+1}^{-1}\,\overset{*}{\mathcal{L}}_{m+1}^{-1}$$

$$\begin{bmatrix} M'_m & \hat{h}_m \\ \hat{f}_m^T & Q_m \end{bmatrix} = \begin{bmatrix} I & & \\ & \overset{*}{b}_m & \\ & I & I \end{bmatrix} \begin{bmatrix} \overset{*}{\mathcal{D}}_m^{-1} & \\ & \overset{*}{\delta}_m^{-1} \end{bmatrix} \begin{bmatrix} I & & \\ & I & \\ & \overset{*}{a}_m^T & I \end{bmatrix}$$

$$\begin{aligned} \overset{*}{\delta}_m^{-1} &= Q_m = \lambda_m^{-1} \\ \overset{*}{b}_m &= \hat{h}_m \overset{*}{\delta}_m = \hat{H}_m \\ \overset{*}{a}_m &= \overset{*}{\delta}_m \hat{f}_m = \hat{F}_m \end{aligned}$$

$\overset{*}{\mathcal{U}}_m^{-1} =$  ; $\overset{*}{\mathcal{L}}_m^{-1} =$ 

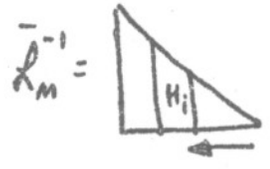
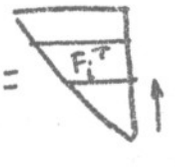
$$\overset{*}{\mathcal{D}}_m^{-1} = \mathrm{diag}\,(\lambda_0^{-1} \cdots \lambda_{m-1}^{-1})$$

$$R = \bar{\mathcal{U}}\,\bar{\mathcal{D}}\,\bar{\mathcal{L}}$$

$$R_{m+1}^{-1} = \bar{\mathcal{L}}_{m+1}^{-1}\,\bar{\mathcal{D}}_{m+1}^{-1}\,\bar{\mathcal{U}}_{m+1}^{-1}$$

$$\begin{bmatrix} Q_m & f_m^T \\ h_m & M_m \end{bmatrix} = \begin{bmatrix} I & & \\ & \bar{a}_m & I \\ & & I \end{bmatrix} \begin{bmatrix} \bar{\delta}_m^{-1} & \\ & \bar{\mathcal{D}}_m^{-1} \end{bmatrix} \begin{bmatrix} I & \bar{b}_m^T & \\ & I & \\ & & I \end{bmatrix}$$

$$\begin{aligned} \bar{\delta}_m^{-1} &= Q_m = \lambda_m^{-1} \\ \bar{b}_m &= \delta_m f_m = F_m \\ \bar{a}_m &= h_m \delta_m = H_m \end{aligned}$$

$\bar{\mathcal{L}}_m^{-1} =$  ; $\bar{\mathcal{U}}_m^{-1} =$ 

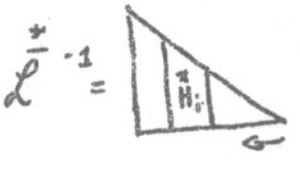
$$\bar{\mathcal{D}}_m^{-1} = \mathrm{diag}\,(\lambda_{m-1}^{-1} \cdots \lambda_0^{-1})$$

$$R^T = \overset{*}{\bar{\mathcal{U}}}\,\overset{*}{\bar{\mathcal{D}}}\,\overset{*}{\bar{\mathcal{L}}}$$

$$(R_{m+1}^T)^{-1} = \overset{*}{\bar{\mathcal{L}}}_{m+1}^{-1}\,\overset{*}{\bar{\mathcal{D}}}_{m+1}^{-1}\,\overset{*}{\bar{\mathcal{U}}}_{m+1}^{-1}$$

$$\begin{bmatrix} \overset{*}{Q}_m & \overset{*}{f}_m^T \\ \overset{*}{h}_m & \overset{*}{M}_m \end{bmatrix} = \begin{bmatrix} I & & \\ & \overset{*}{a}_m & I \\ & & I \end{bmatrix} \begin{bmatrix} \overset{*}{\delta}_m^{-1} & \\ & \overset{*}{\mathcal{D}}_m^{-1} \end{bmatrix} \begin{bmatrix} I & \overset{*}{b}_m^T & \\ & I & \\ & & I \end{bmatrix}$$

$$\begin{aligned} \overset{*}{\delta}_m^{-1} &= \overset{*}{Q}_m = \overset{*}{\lambda}_m^{-1} \\ \overset{*}{b}_m &= \overset{*}{\delta}_m \overset{*}{f}_m = \overset{*}{F}_m \\ \overset{*}{a}_m &= \overset{*}{h}_m \overset{*}{\delta}_m = \overset{*}{H}_m \end{aligned}$$

$\overset{*}{\bar{\mathcal{L}}}_m^{-1} =$  ; $\overset{*}{\bar{\mathcal{U}}} =$ 

$$\overset{*}{\bar{\mathcal{D}}}_m^{-1} = \mathrm{diag}\,(\overset{*}{\lambda}_{m-1}^{-1} \cdots \overset{*}{\lambda}_0^{-1})$$

④ TRIANGULAR FACTORS OF R and R^T

$$R = \mathcal{L}\,\mathcal{D}\,\mathcal{U} = \mathcal{L}\,\mathcal{T} = \begin{bmatrix} I & & \\ & I & \\ & & \ddots \end{bmatrix}$$

$$\mathcal{L}_{m+1}^{-1} R_{m+1} = \mathcal{T}_{m+1}$$

$$\begin{bmatrix} I & & \\ & \mathcal{L}_m^{-1} & \\ & a_m^T & I \end{bmatrix} \begin{bmatrix} R_m & \hat{r}_m^* \\ r_m^T & R_0 \end{bmatrix} = \begin{bmatrix} \mathcal{T} & \\ & P_{m+1} \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \cdots & & P_{1,m+1} \\ & P_{22} \cdots & P_i & \vdots \\ & & & P_{m+1,m+1} \end{bmatrix}$$

$$i \to \begin{bmatrix} 1 & & \\ & \mathcal{L}_{m+1}^{-1} & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix} = \begin{bmatrix} P_{m+1} \end{bmatrix} \leftarrow i$$

$$\underbrace{[a_i^T I 0 \cdots 0]}_{\hat{F}_i^{*T}} \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix} = P_{i+1,m+1} \qquad i = 0, 1, \cdots n$$

$$\hat{F}_{i+1}^* = \begin{bmatrix} 0 \\ \hat{F}_i^* \end{bmatrix} - \alpha_i^* \begin{bmatrix} I \\ F_i \end{bmatrix}$$

$$P_{i+2,m+1} = \underbrace{[0\ \hat{F}_i^T 0 \cdots 0] \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix}}_{P_{i+1,m}} - \alpha_i^* \underbrace{[I\ F_i^T 0 \cdots 0] \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix}}_{\| \Delta\ q_{i+1,m}} \qquad i = 0, 1, \cdots, m-1$$

$$P_{1,m+1} = q_{1,m} = R_m$$

$$q_{i+2,m} = [I\ F_{i+1}^T 0 \cdots 0] \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix} = \underbrace{[I\ F_i^T 0 \cdots 0] \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix}}_{q_{i+1,m}} - \alpha_i \underbrace{[0\ \hat{F}_i^{*T} I 0 \cdots 0] \begin{bmatrix} \hat{r}_m^* \\ R_0 \end{bmatrix}}_{P_{i+1,m}}$$

$$F_{i+1} = \begin{bmatrix} F_i \\ 0 \end{bmatrix} - \alpha_i \begin{bmatrix} \hat{F}_i^* \\ I \end{bmatrix}$$

This is a recursion to compute $\mathcal{T}$

since $\mathcal{T} \triangleq \mathcal{D}\,\mathcal{U}$ we have: $\mathcal{U} = \begin{bmatrix} u_i & & \\ & \ddots & \\ & & \end{bmatrix}$; $u_i = \begin{bmatrix} \delta_0^{-1} & & \\ & \ddots & \\ & & \delta_{i-1}^{-1} \end{bmatrix} \begin{bmatrix} P_i \end{bmatrix}$

$\mathcal{L}$ can be obtained in a similar way:

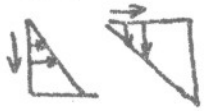
$$R_{m+1} \mathcal{U}_{m+1}^{-1} = \mathcal{L}_{m+1}\,\mathcal{D}_{m+1} \triangleq W_{m+1} = \begin{bmatrix} w_i & & \\ & \ddots & \\ & & \end{bmatrix}$$

and by a dual approach one can find

$$R = \bar{\mathcal{U}}\,\bar{\mathcal{D}}\,\bar{\mathcal{L}} = \bar{\mathcal{T}}\,\bar{\mathcal{L}} = \bar{\mathcal{U}}\,\bar{W}$$

Summary (algorithm 2)

$$R = \mathcal{L}\mathcal{D}\mathcal{U} = \mathcal{L}P = W\mathcal{U}$$



$$R = \bar{\mathcal{U}}\bar{\mathcal{D}}\bar{\mathcal{L}} = \bar{P}\bar{\mathcal{L}} = \bar{\mathcal{U}}\bar{W}$$



$q_0 = P_0 = [\phi]$

$$P_{n+1} = \begin{bmatrix} R_n \\ P_n \end{bmatrix} - \left[\begin{array}{c|c} 0 & \\ \hline \begin{bmatrix} \alpha_0^* & & \\ & \ddots & \\ & & \alpha_{n-1}^* \end{bmatrix} & \end{array}\right] q_n$$

$$q_n = \begin{bmatrix} R_n \\ z q_n \end{bmatrix} - \left[\begin{array}{c|c} 0 & \\ \hline \begin{bmatrix} \alpha_0 & & \\ & \ddots & \\ & & \alpha_{n-2} \end{bmatrix} & 0 \end{array}\right] \begin{bmatrix} P_n \end{bmatrix}$$

$$\alpha_{n-1} = q_{nn} (P_{nn})^{-1}$$

$$P_{nn} = \delta_{n-1} = R^{\varepsilon}_{n-1}$$

$q_{nn} = \mu_{n-1} = \nu_{n-1} = \dots = \bar{q}_{nn} = \bar{\nu}_{n-1} = \bar{\mu}_{n-1}$

$\bar{q}_0 = \bar{P}_0 = [\phi]$

$$\bar{P}_{n+1}^T = [R_n : \bar{P}_n^T] - [0 : \bar{q}_n^T \begin{bmatrix} \beta_0 & & \\ & \ddots & \\ & & \beta_{n-1} \end{bmatrix}]$$

$$\bar{q}_n^T = [R_n : z\bar{q}_n^T] - \bar{P}_n^T \left[\begin{array}{c|c} \begin{bmatrix} \beta_0^* & & \\ & \ddots & \\ & & \beta_{n-2}^* \end{bmatrix} & \\ \hline 0 & \end{array}\right]$$

$$\bar{\beta}_{n-1}^* = (\bar{P}_{nn})^{-1} \bar{q}_{nn}$$

$$\bar{P}_{nn} = \bar{\delta}_{n-1} = R^{\varepsilon}_{n-1}$$

$w_0 = c_0 = [\phi]$

$$w_{n+1}^T = [R_{-n} : w_n^T] - [0 : c_n^T \begin{bmatrix} \beta_0^* & & \\ & \ddots & \\ & & \beta_{n-1}^* \end{bmatrix}]$$

$$c_n^T = [R_{-n} : z c_n^T] - w_n^T \left[\begin{array}{c|c} 0 & \begin{bmatrix} \beta_0 & & \\ & \ddots & \\ & & \beta_{n-2} \end{bmatrix} \\ \hline 0 & \end{array}\right]$$

$$\beta_{n-1} = (w_{nn})^{-1} c_{nn}$$

$$w_{nn} = \delta_{n-1} = R^{\nearrow}_{n-1}.$$

$c_{nn} = \nu_{n-1} = \mu_{n-1} = \dots = \bar{c}_{nn} = \bar{\mu}_{n-1} = \bar{\nu}_{n-1}$

$\bar{w}_0 = \bar{c}_0 = [\phi]$

$$\bar{w}_{n+1} = \begin{bmatrix} R_{-n} \\ \bar{w}_n \end{bmatrix} - \left[\begin{array}{c|c} 0 & \\ \hline \begin{bmatrix} \alpha_0 & & \\ & \ddots & \\ & & \alpha_{n-1} \end{bmatrix} & \end{array}\right] \bar{c}_n$$

$$\bar{c}_n = \begin{bmatrix} R_{-n} \\ z\bar{c}_n \end{bmatrix} - \left[\begin{array}{c|c} 0 & \\ \hline \begin{bmatrix} \alpha_0^* & & \\ & \ddots & \\ & & \alpha_{n-2}^* \end{bmatrix} & 0 \end{array}\right] \begin{bmatrix} \bar{w}_n \end{bmatrix}$$

$$\bar{\alpha}_{n-1}^* = \bar{c}_{nn} (\bar{w}_{nn})^{-1}$$

$$\bar{w}_{nn} = \bar{\delta}_{n-1} = R^{\varepsilon}_{n-1}$$

$\bar{c}_{nn} = \bar{\mu}_{n-1} = \bar{\nu}_{n-1} = \dots = c_{nn} = \mu_{n-1} = \nu_{n-1}$

This is exactly the recursion used in ladder form algorithms to compute the reflection coefficients.

REMARKS

1. symmetry considerations

a) if R is block symmetric ($R = R^T$) ($R_i = R_{-i}$ $\forall i$)
then $a = \overleftarrow{a}$ $\forall a$ thus the left column of alg 1 coincides with its right column

b) if R is symmetric ($R^t = R$) ($R_i = R_{-i}^t$ $\forall i$)

then $\overleftarrow{r}_m = r_m^T$:

$H = F^t$ $\quad \lambda_i = \lambda_i^t$ $\quad \overleftarrow{\nu}_i = \mu_i = \nu_i^t = \overleftarrow{\mu}_i^t$ $\quad \alpha_i^t = \beta_i$

$\overleftarrow{H} = \overleftarrow{F}^T$ $\quad \overleftarrow{\lambda}_i = \overleftarrow{\lambda}_i^t$ $\quad \nu_i = \overleftarrow{\mu}_i = \overleftarrow{\nu}_i^t = \mu_i^t$ $\quad \overleftarrow{\alpha}_i^t = \beta_i$

and $\mu_i^t = \overleftarrow{\mu}_i$

alg 1 reduces to the LEVINSON algorithm.

c) if R is symmetric $R^t = R$ but has no other recursion in its elements then the classical cholesky method (eventually recursive version) is the best you can do ($\mathcal{O}(n^3)$ operations)
However the covariance method of LPC has a matrix with some recursivity. By Chandraselekar equations it is possible to derive a fast Cholesky algorithm.

d) the Hankel matrix problem is essentially the same as the Toeplitz case.

2. Some special cases

a) if R is a band matrix (Toeplitz) some further reduction in computational work is possible (TRENCH 74, DICKINSON 78)

b) Special type equations (FARDEN) $Rx = c$

$\left.\begin{matrix} R^t = R \\ \hat{c} = c \end{matrix}\right\} \Rightarrow \overleftarrow{x} = x$ thus $x_{i+2} = \begin{pmatrix} 0 \\ x_i \\ 0 \end{pmatrix} + \frac{\theta_i}{\lambda_i - \eta_i} \begin{pmatrix} 1 \\ e_i + \hat{e}_i \\ 1 \end{pmatrix}$

$$R_{i+2}^{-1} = \frac{1}{\lambda_{i+1}} \begin{pmatrix} 1 & \hat{e}_{i+1}^T \\ \hat{e}_{i+1} & M_{i+1} \end{pmatrix}$$

also BUTLER ASSP 1975

3. Positive definiteness

The principal minors should always be non-singular otherwise column pivoting is necessary (RISSANEN 1974)

$\triangle R \Pi = \triangledown \Pi$ or $\triangledown R \Pi = \triangle \Pi$ (Π: permutations)

4. The normal equations

$$[I\ A_{m1}\cdots A_{mm}] = [R_m^{\varepsilon}\ 0\cdots 0]\left[\begin{array}{c|c} Q_m & f_m^T \\ \hline f_m & M_m \end{array}\right]$$

$$= [R_m^{\varepsilon} Q_m \,;\, R_m^{\varepsilon} f_m]$$

$$\rightarrow R_m^{\varepsilon} = Q_m^{-1} = \bar{\delta}_m = \lambda_m = \overset{*}{\delta}_m$$

$$[A_{m1}\cdots A_{mm}] = Q_m^{-1} f_m^T = F_m^T$$

$$[B_{mm}\cdots B_{m1}\ I] = [0\cdots 0\ R_m^{r}]\left[\begin{array}{c|c} \overset{*}{M}_m & \overset{*}{f}_m \\ \hline \overset{*}{f}_m^T & \overset{*}{Q}_m \end{array}\right]$$

$$= [R_m^{r}\ \overset{*}{f}_m^T \,;\, R_m^{r}\ \overset{*}{Q}_m]$$

$$\rightarrow R_m^{r} = \overset{*}{Q}_m^{-1} = \overset{*}{\delta}_m = \overset{*}{\lambda}_m = \delta_m$$

$$[B_{mm}\cdots B_{m1}] = \overset{*}{F}^T$$

if $R_{m+1} = \bar{U}\ \bar{\vartheta}_{m+1}\ \bar{L}_{m+1}$

then $\bar{\vartheta}_{m+1} = \mathrm{diag}\,(R_m^{\varepsilon}\cdots R_0^{\varepsilon})$

$\bar{U}_{m+1}^{-1} =$

$$\begin{array}{l} I\ A_{m1}\cdots A_{mm} \\ \quad I\ A_{i1}\cdots A_{ii} \\ \qquad\qquad I \end{array}$$

$$A_{ii} = K_{i-1}^{a} = -\alpha_i$$

if $R_{m+1} = \bar{L}_{m+1}\ \vartheta_{m+1}\ U_{m+1}$

then $\vartheta_{m+1} = \mathrm{diag}\,(R_0^{r}\cdots R_m^{r})$

$\bar{L}_{m+1}^{-1} =$

$$\begin{array}{l} I \\ B_{ii}\cdots B_{i1}\ I \\ B_{mm}\cdots\qquad B_{m1}\ I \end{array}$$

$$B_{ii} = K_{i-1}^{b} = -\overset{*}{\alpha}_i$$

5. The use of the factorization of R

- As we saw it is essentially the ladder form algorithm (Durbin's extraction)
- R^{-1} factorization solved the AR problem
  R " solves " MA "

AR

$$\left[\begin{array}{ccc} I & A_{m1}\cdots & A_{mm} \\ & \ddots & \\ & & I \end{array}\right]\left[\begin{array}{c} y_m \\ \vdots \\ y_0 \end{array}\right] = \left[\begin{array}{c} \varepsilon_m \\ \vdots \\ \varepsilon_0 \end{array}\right]$$

$\bar{U}_{m+1}^{-1}$

$$\bar{U}_{m+1}^{-1}\ R_{m+1}\left(\bar{U}_{m+1}^{t}\right)^{-1} = \left[\begin{array}{ccc} R_m^{\varepsilon} & & \\ & \ddots & \\ & & R_0^{\varepsilon} \end{array}\right]$$

MA

$$\left[\begin{array}{ccc} C_{0m} & \cdots & C_{mm} \\ & \ddots & \\ & & C_{00} \end{array}\right]\left[\begin{array}{c} v_m \\ \vdots \\ v_0 \end{array}\right] = \left[\begin{array}{c} y_m \\ \vdots \\ y_0 \end{array}\right]$$

U_{m+1}

$$U_{m+1}\left[\begin{array}{ccc} R_m^{v} & & \\ & \ddots & \\ & & R_0^{v} \end{array}\right] U_{m+1}^{t} = R_{m+1}$$

$$\hat{y}_n = -\sum_{k=1}^{p} a_k y_{n-k} \qquad \rightarrow \min E = \|E\|^2$$

$$\varepsilon_n = y_n - \hat{y}_n = y_n + \sum_{k=1}^{p} a_k y_{n-k} \qquad \text{normal equations.}$$

given $\{ y_n ; S \le n \le T \}$ $\Rightarrow$ all quantities depend upon p, S, T

$$a^t \triangleq [1 \; a_1 \; a_2 \cdots a_p]$$

$$Y_n^t \triangleq [y_n^t \; y_{n-1}^t \cdots y_{n-p}^t]$$

$$y_{i,f} = \begin{bmatrix} y_i & \cdots & y_f \\ \vdots & \ddots & \\ y_{i-p} & \cdots & y_{f-p} \end{bmatrix} = [Y_i \; Y_{i+1} \cdots Y_f]$$

$$[\varepsilon_i \cdots \varepsilon_k \cdots \varepsilon_f] = a^t [Y_i \cdots Y_k \cdots Y_f] = a^t y_{if}$$

$$E = tr \sum_{k=i}^{f} \varepsilon_k \varepsilon_k^t = (4)\, y_{i,f} \qquad i = S, \; f = T \quad (y_k = 0 \; \forall k < S)$$

= pre windowing $\quad y_{i,f}^{(1)} = \begin{bmatrix} y_S & \cdots & y_T \\ 0 & \ddots & \vdots \end{bmatrix}$

(2) $y_{i,f}$ $\quad i = S+p, \; f = T$

$\rightarrow$ covariance method $\quad y_{i,f}^{(2)} = \begin{bmatrix} y_{S+p} & \cdots & y_T \\ \vdots & & \vdots \\ y_S & \cdots & y_{T-p} \end{bmatrix}$

(3) $y_{i,f}$ $\quad i = S, \; f = T+p$

$\rightarrow$ autocorrelation method $\quad y_{i,f}^{(3)} = \begin{bmatrix} y_S & & y_T & 0 \\ & \ddots & & \ddots \\ 0 & & y_S & & y_T \end{bmatrix}$

(4) y $\quad i = S+p, \; f = T+p$

$\rightarrow$ post windowing $\quad y_{if}^{(4)} = \begin{bmatrix} y_{S+p} & & y_T & 0 \\ & \ddots & & \ddots \\ y_S & & & & y_T \end{bmatrix}$

$$= tr(a^t \underbrace{y_{if} \, y_{if}^t}_{R} \, a)$$

only $R^{(3)}$ is Toeplitz.
but R = product of Toeplitz y

The stochastic equivalent of this is: $R^{(2)}$ is an approximation for the covariance matrix of a stationary process
$R^{(3)}$ is the approximation for an ergodic nonstationary proces (local stationarity) [Makhoul]
Some fast algorithms have been designed to solve the problem with an $R^{(2)}$ matrix

see | MORF - DICKINSON - KAILATH - VIEIRA (1977) IEEE Trans. ASSP 25(5) 429-433
MORF - VIEIRA - LEE : Ladder forms for identification and speech processing - CDC conference, New Orleans, Dec 7-9, 1977

CHANDRASEKHAR

⇒ A fast cholesky algorithm for some special symmetric non-Toeplitz (covariance method for LPC)

assume non stationarity but constant state space model

$$\begin{cases} x_{i+1} = \Phi x_i + \Gamma u_{i+1} \\ y_i = H x_i \end{cases} \qquad i \geq 0 \qquad \Phi, \Gamma, H, \text{cnst}$$

finite order model

$$R^y_{ij} = R_{ij} \triangleq E[y_i y_j^t] = \begin{cases} M \Phi^{j-i} N_j & j \leq i \\ N_i^t (\Phi^t)^{j-i} M^t & j \geq i \end{cases} \qquad N_i = \pi_i H^t \quad M = H$$

$$\pi_i = E[x_i x_i^t] \; ; \quad \pi_{i+1} = \Phi \pi_i \Phi^t + \Gamma Q \Gamma$$

$$Q\delta_{ij} \triangleq E[u_i u_j^t]$$

in the stationary case π_i is a constant

After some calculations one can get :

$$E[y_j \varepsilon_i^t] = \begin{cases} H \Phi^{j-i} K_i & j \geq i \\ 0 & j < i \text{ (orthogon.)} \end{cases} \qquad K_i : \text{kalman gain}$$

$$E[\varepsilon_i \varepsilon_j^t] \triangleq R_i^\varepsilon \delta_{ij}$$

$$\begin{cases} K_{i+1} = K_i + \Phi Y_i M_i Y_i^t \Phi^t H^t & ; K_0 = N_0 \\ M_{i+1} = M_i - [M_i Y_i^t \Phi^t H^t] (R^\varepsilon_{i+1})^{-1} [H \Phi Y_i M_i] & \\ R^\varepsilon_{i+1} = R^\varepsilon_i + [H \Phi Y_i] M_i [Y_i^t \Phi^t H^t] & ; R_0^\varepsilon = H N_0 \\ Y_{i+1} = (I - K_i (R_i^\varepsilon)^{-1} H) \Phi Y_i & \end{cases}$$

set $(M_i)^{-1} \triangleq -R_i^2$ and using $(A+BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1}+DA^{-1}B)^{-1}DA^{-1}$ then

$$\begin{cases} R^2_{i+1} = R^2_i - (Y_i^t \Phi^t H^t)(R^\varepsilon_{i+1})^{-1}(H \Phi Y_i) \\ R^\varepsilon_{i+1} = R^\varepsilon_i - (H \Phi Y_i)(R^2_i)^{-1}(Y_i^t \Phi^t H^t) \end{cases}$$

$$R^Y = R^{Y\epsilon} (R^\epsilon)^{-1} (R^{Y\epsilon})^t$$

$$(R^Y)_{ij} = R^Y_{ij} = E[y_i y_j^t] \;;\; (R^{Y\epsilon})_{ij} = E[y_i \epsilon_j^t] \;;\; (R^\epsilon)_{ij} = \delta_{ij} R^\epsilon_i = E[\epsilon_i \epsilon_j^t]$$

$$R_{m+1} = \mathcal{L}_{m+1}\, \mathcal{D}_{m+1}\, \mathcal{L}^t_{m+1}$$

R : symmetric but NON TOEPLITZ

$$\mathcal{L}_{j+1,i+1} \triangleq H \phi^{j-i} K_{i+1}$$

$$= \underbrace{H \phi^{j-i} K_i}_{\mathcal{L}_{ji}} + \underbrace{H \phi^{j-i} \phi Y_i}_{\| \Delta \; C_{ji}} \underbrace{M_i}_{(R^2_i)^{-1}} \underbrace{Y_i^t \phi^t H^t}_{C^t_{ii}}$$

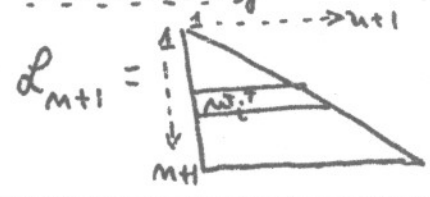
$$C_{j+1,i+1} \triangleq H \phi^{j-i+1} Y_{i+1}$$

$$= \underbrace{H \phi^{j-i+1} \phi Y_i}_{C_{j+1,i}} - \underbrace{H \phi^{j-i+1} K_i}_{\mathcal{L}_{j+1,i}} (R^\epsilon_i)^{-1} \underbrace{H \phi Y_i}_{C_{ii}}$$

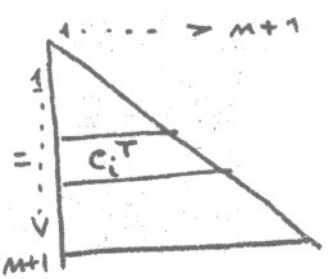
$$\mathcal{L}_{j0} \triangleq H \phi^j K_0 = H \phi^j N_0 = R_{j0}$$

$$C_{j0} \triangleq H \phi^{j+1} Y_0$$

renumbering



$$\mathcal{L}_{m+1} = \quad \omega^T_{m+1} = [\mathcal{L}_{m0} \cdots \mathcal{L}_{mm}] \;;\; \mathcal{C}_{m+1} =$$



then

$$\omega^T_{m+1} = [R_{m0} \vdots \omega^T_m] - [0 \vdots C^T_m \begin{bmatrix} \overset{*}{\beta}_0 \cdots \overset{*}{\beta}_{m-1} \end{bmatrix}]$$

$$\overset{*}{\beta}_{i-1} \triangleq (R^2_{i-1})^{-1} C^t_{ii} \;;\; \omega_{mm} = R^\epsilon_{m-1}$$

$$C^T_m = [\rho_m \vdots z e^T_m] - \omega^T_m \begin{bmatrix} 0 & \beta_0 \cdots \beta_{m-2} \\ & 0 \end{bmatrix}$$

$$\beta_{i-1} \triangleq (R^\epsilon_{i-1})^{-1} C_{ii} \;;\; C_{mm} = R^2_{m-1}$$

$$\rho_{j+1} \triangleq H \phi^{j+1} Y_0$$

(in the stationary case $Y_0 = N_0$ and $\rho_m = R_{m0}$)

these are relations of exactly the same form we had before

THE CHOLESKY ALGORITHM

1. If R has no other property but symmetry then the Cholesky algorithm is the best one can do (algorithm 3-4) $6(n^3)$ operations.
2. If R_n is a nested series of matrices then a recursive form of the Cholesky algorithm can be designed (alg 5) this is still of $6(n^3)$ operations
3. If R is symmetric and Toeplitz one can do in $6(n^2)$ oper. If there is some additional structure imposed it is possible to do it in $6(n^2)$ operations too (Chandrasekhar)
4. As we've seen: In the Toeplitz case we could find the reflection coëfficients in a column of an inverse triangular factor. If R is symmetric but non Toeplitz then the quantities appearing in the same places have some corresponding properties but they are not reflection coëfficients in the usual sense
   - they are <u>not</u> sufficient to generate the whole L^{-1} triangle (this is all the predictor coëfs)
   - they are <u>not</u> guaranteed to be less than 1 in magnitude (GIBSON)

5. Some generalization can be made to handle non positive definite symmetric matrices. This form is used to determine the rank and thus save storage and computational effort however care must been taken in this case.

RECURSIVE RATIONAL CHOLESKY ROW by ROW

$$R_{n+1} = \left[\begin{array}{c|c} R_n & r_n \\ \hline r_n^t & R_{nn} \end{array}\right] = \mathcal{L}_{n+1}\,\mathcal{D}_{n+1}\,\mathcal{L}_{n+1}^t = \begin{bmatrix} I & & \\ & \mathcal{L}_n & \\ & a_n^t & I \end{bmatrix} \left[\begin{array}{c|c} \mathcal{D}_n & \\ \hline & \delta_n \end{array}\right] \begin{bmatrix} I & & \\ & \mathcal{L}_n^t & a_n \\ & & I \end{bmatrix}$$

$$= \left[\begin{array}{c|c} \mathcal{L}_n \mathcal{D}_n \mathcal{L}_n^t & \mathcal{L}_n \mathcal{D}_n a_n \\ \hline a_n^t \mathcal{D}_n \mathcal{L}_n^t & a_n^t \mathcal{D}_n a_n + \delta_n \end{array}\right] \Rightarrow \boxed{\begin{aligned} a_n &= \mathcal{D}_n^{-1}\,\mathcal{L}_n^{-1}\,r_n \\ \delta_n &= R_{nn} - a_n^t\,\mathcal{D}_n\,a_n \end{aligned}}$$

$$\mathcal{L}_{n+1}^{-1} = \begin{bmatrix} I & & \\ & \mathcal{L}_n^{-1} & \\ & f_n^t & I \end{bmatrix} \quad ; \quad \mathcal{L}_{n+1}\,\mathcal{L}_{n+1}^{-1} = I \Rightarrow \boxed{f_n = -\mathcal{L}_n^{-t}\,a_n}$$

a combination of $\mathcal{L}$ and $\mathcal{L}^{-1}$ computation gives alg. 5

RECURSIVE RATIONAL CHOLESKY COLUMN BY COLUMN

$$R_{n+1} = \left[\begin{array}{c|c} R_n & r_n \\ \hline r_n^t & R_{nn} \end{array}\right] = \bar{\mathcal{U}}_{n+1}\,\bar{\mathcal{D}}_{n+1}\,\mathcal{U}_{n+1}^t = \begin{bmatrix} I & & \\ & \bar{\mathcal{U}}_n & b_n \\ & & I \end{bmatrix} \left[\begin{array}{c|c} \tilde{\mathcal{D}}_n & \\ \hline & \bar{\delta}_n \end{array}\right] \begin{bmatrix} I & & \\ & \tilde{\mathcal{U}}_n^t & \\ & b_n^t & I \end{bmatrix}$$

$$= \left[\begin{array}{c|c} \tilde{\mathcal{U}}_n \tilde{\mathcal{D}}_n \tilde{\mathcal{U}}_n^t + b_n \delta_n b_n^t & b_n \delta_n \\ \hline \delta_n b_n^t & \bar{\delta}_n \end{array}\right] \Rightarrow \boxed{\begin{aligned} \bar{\delta}_n &= R_{nn} \\ b_n &= r_n \bar{\delta}_n^{-1} \\ \tilde{\mathcal{U}}_n \tilde{\delta}_n \tilde{\mathcal{U}}_n^t &= \bar{\mathcal{U}}_n \bar{\mathcal{D}}_n \bar{\mathcal{U}}_n^t - b_n \bar{\delta}_n b_n^t \end{aligned}}$$

the computation of $\tilde{\mathcal{U}}_n$ and $\tilde{\mathcal{D}}_n$ from:

$$\tilde{\mathcal{U}}_n \tilde{\mathcal{D}}_n \tilde{\mathcal{U}}_n^t = \bar{\mathcal{U}}_n \bar{\mathcal{D}}_n \bar{\mathcal{U}}_n^t - r_n \bar{\delta}_n^{-1} \bar{\delta}_n \bar{\delta}_n^{-1} r_n^t = R_n - b_n \bar{\delta}_n b_n^t$$

$$= \bar{\mathcal{U}}_n \left(\bar{\mathcal{D}}_n - v_n \bar{\delta}_n^{-1} v_n^t \right) \bar{\mathcal{U}}_n^t \quad ; \boxed{\bar{\mathcal{U}}_n v_n = b_n}$$

$$= \underbrace{\bar{\mathcal{U}}_n \hat{\mathcal{U}}_n}_{\tilde{\mathcal{U}}_n}\ \tilde{\mathcal{D}}_n\ \underbrace{\hat{\mathcal{U}}_n^t \bar{\mathcal{U}}_n^t}_{\tilde{\mathcal{U}}_n^t}$$

$\hat{\mathcal{U}}_n, \tilde{\mathcal{D}}_n$ can easily be found (Gill et al , Fletcher et al)

$$\boxed{(\hat{\mathcal{U}}_n)_{j,i} = (v_n)_j\,\beta_i \quad ; \beta_i = \left[\bar{\delta}_n^{-1} - \sum_{k=j+1}^{n} \rho_k \tilde{\delta}_k \beta_k^t \right] (v_n)_i^t\ \tilde{\delta}_i^{-1}}$$

in the references you can find methods deriving directly in a recursive way $\tilde{\mathcal{U}}_n$, v_n, $\bar{\delta}_n$ implemented in HARWELL subroutine Library

GENERALIZED CHOLESKY FACTORIZATION OF SYMMETRIC INDEFINITE SYSTEMS

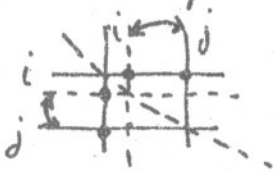
(Bunch & Parlett 1974)

If R is indefinite it will be possible in some cases to factorize a symmetric matrix by applying a symmetric permutation matrix Π, in the form

$$\Pi R \Pi^t = \begin{bmatrix} \ddots & & \\ & d_k & \\ & & \ddots \end{bmatrix} \qquad (*)$$

with $d_1, \dots, d_\ell > 0$, $d_{\ell+1} \dots d_k < 0$

By symmetric pivoting, an off-diagonal element can not be brought on the diagonal, the only thing to do is to bring it in a place $(i, i+1)$ or $(i+1, i)$



So $R = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ can not be factorized in form $(*)$
The numerical analogue of this is instability.
The following factorization is allways possible:

$$L = \begin{bmatrix} I & & \\ & \ddots & \\ & & \ddots \end{bmatrix} \qquad D = \begin{bmatrix} \ddots & & \\ & \ddots & \\ & & 0 \end{bmatrix}$$

Block pivoting.

$$R = \left[\begin{array}{c|c} P & C^t \\ \hline C & B \end{array}\right] \rightarrow R = L \left[\begin{array}{c|c} P & \\ \hline & B - CP^{-1}C^t \end{array}\right] L^t \; ; \; L = \left[\begin{array}{c|c} I & 0 \\ \hline CP^{-1} & I \end{array}\right]$$

if P is 2×2: $P = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}$

$$\text{then } CP^{-1} = \begin{bmatrix} \vdots & \vdots \\ R_{k1} & R_{k2} \\ \vdots & \vdots \end{bmatrix} = \frac{1}{R_{11}R_{22} - R_{21}^2} \begin{bmatrix} \vdots & \vdots \\ R_{k1}R_{22} - R_{k1}R_{21} & R_{k2}R_{11} - R_{k1}R_{21} \\ \vdots & \vdots \end{bmatrix}$$

Optimal pivoting:

$$R^{(n)} = R \rightarrow R^{(k)} \text{ where } R = L^{(k)} \begin{bmatrix} \ddots & 0 \\ 0 & R^{(k)} \end{bmatrix} (L^{(k)})^t$$

$$\mu_0^{(k)} = \max_{ij} |R_{ij}^{(k)}| = |R_{\bar{i}\bar{j}}^{(k)}|$$

$$\mu_1^{(k)} = \max_{j} |R_{jj}^{(k)}| = |R_{\bar{j}\bar{j}}^{(k)}|$$

then if $\mu_1^{(k)} \geq \mu_0^{(k)}$ then 1×1 pivot $R_{\bar{j}\bar{j}}$

else 2×2 pivot for which $\begin{cases} \nu_c^{(k)} = \max_{ij} |R_{ii}R_{jj} - R_{ij}^2| \\ \text{OR} \\ \nu_p^{(k)} = \max_{i} |R_{ku}R_{ii} - R_{ki}^2| \end{cases}$

implemented in LINSYS

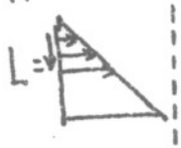
ALGORITHM 1 INVERSION - FACTORIZATION OF R^{-1} - SOLUTION OF $Rx = c$

$\lambda_0 = R_0$ $F_0 = [\phi]$ $H_0 = [\phi]$ $x_0 = [\phi]$	L L S	$\overset{*}{\lambda}_0 = R_0$ $\overset{*}{F}_0 = [\phi]$ $\overset{*}{H}_0 = [\phi]$ $\overset{*}{x}_0 = [\phi]$
$i = 0, 1, 2, \dots$		$i = 0, 1, 2, \dots$
$\mu_i = R_{i+1} + \hat{F}_i^T x_i^*$; $\alpha_i = \mu_i \overset{*}{\lambda}_i^{-1}$ $\nu_i = x_i^T \hat{H}_i + R_{-(i+1)}$; $\beta_i = \overset{*}{\lambda}_i^{-1} \nu_i$	L	$\overset{*}{\mu}_i = \hat{f}_i^T z_i + R_{-(i+1)}$; $\overset{*}{\alpha}_i = \overset{*}{\mu}_i \lambda_i^{-1}$ $\overset{*}{\nu}_i = R_{i+1} + \overset{*}{x}_i^T \hat{H}_i$; $\overset{*}{\beta}_i = \lambda_i^{-1} \overset{*}{\nu}_i$
$\theta_i = \overset{*}{\lambda}_i^{-1} (\gamma_i - x_i^T z_i)$ $x_{i+1} = \begin{bmatrix} z_i \\ 0 \end{bmatrix} + \begin{bmatrix} \hat{H}_i \\ I \end{bmatrix} \theta_i$	S S	$\overset{*}{\theta}_i = \lambda_i^{-1} (\gamma_i - \overset{*}{z}_i^T \overset{*}{x}_i)$ $\overset{*}{x}_{i+1} = \begin{bmatrix} \overset{*}{x}_i \\ 0 \end{bmatrix} + \begin{bmatrix} \hat{H}_i \\ I \end{bmatrix} \overset{*}{\theta}_i$
$F_{i+1} = \begin{bmatrix} F_i \\ 0 \end{bmatrix} - \alpha_i \begin{bmatrix} \hat{F}_i \\ I \end{bmatrix}$ $H_{i+1} = \begin{bmatrix} H_i \\ 0 \end{bmatrix} - \begin{bmatrix} \hat{H}_i \\ I \end{bmatrix} \beta_i$ $\lambda_{i+1} = (I - \alpha_i \overset{*}{\alpha}_i) \lambda_i = \lambda_i - \underbrace{\mu_i}_{\overset{*}{\nu}_i} \overset{*}{\lambda}_i^{-1} \underbrace{\overset{*}{\mu}_i}_{\overset{*}{\nu}_i}$ $= \lambda_i (I - \overset{*}{\beta}_i \beta_i)$	L L	$\overset{*}{F}_{i+1} = \begin{bmatrix} \overset{*}{F}_i \\ 0 \end{bmatrix} - \overset{*}{\alpha}_i \begin{bmatrix} \hat{F}_i \\ I \end{bmatrix}$ $\overset{*}{H}_{i+1} = \begin{bmatrix} \overset{*}{H}_i \\ 0 \end{bmatrix} - \begin{bmatrix} \hat{H}_i \\ I \end{bmatrix} \overset{*}{\beta}_i$ $\overset{*}{\lambda}_{i+1} = (I - \overset{*}{\alpha}_i \alpha_i) \overset{*}{\lambda}_i = \overset{*}{\lambda}_i - \underbrace{\overset{*}{\mu}_i}_{\overset{*}{\nu}_i} \lambda_i^{-1} \underbrace{\mu_i}_{\overset{*}{\nu}_i}$ $= \overset{*}{\lambda}_i (I - \beta_i \overset{*}{\beta}_i)$
$R_{i+1}^{-1} = \left[\begin{array}{c\|c} Q_i & f_i^T \\ \hline h_i & M_i \end{array}\right]$ $\rightarrow \begin{cases} Q_i = \lambda_i^{-1} \\ f_i = Q_i F_i \\ h_i = H_i Q_i \\ M_i = (R_{i+1}^{-1})_{k,l} + (\underbrace{H_i f_i^T}_{h_i F_i^T} - \underbrace{\hat{H}_i \hat{F}_i^T}_{\hat{H}_i \hat{f}_i^T})_{k,l} \end{cases}$ $(R_{i+1}^{-1})_{k+1,l+1} =$	I ↓	$(R_{i+1}^T)^{-1} = \left[\begin{array}{c\|c} \overset{*}{Q}_i & \overset{*}{f}_i^T \\ \hline \overset{*}{f}_i & \overset{*}{M}_i \end{array}\right]$ $\rightarrow \begin{cases} \overset{*}{Q}_i = \overset{*}{\lambda}_i^{-1} \\ \overset{*}{f}_i = \overset{*}{Q}_i \overset{*}{F}_i \\ \overset{*}{h}_i = \overset{*}{H}_i \overset{*}{Q}_i \\ \overset{*}{M}_i = (R_{i+1}^T)_{k,l}^{-1} + (\underbrace{\overset{*}{H}_i \overset{*}{f}_i^T}_{\overset{*}{h}_i \overset{*}{F}_i^T} - \underbrace{\hat{H}_i \hat{f}_i^T}_{\hat{H}_i \hat{f}_i^T})_{k,l} \end{cases}$ $(R_{i+1}^T)_{k+1,l+1}^{-1} =$

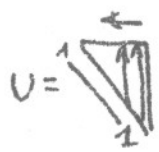
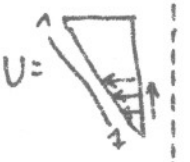
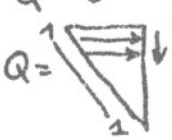
ALGORITHM 2 TRIANGULAR FACTORIZATION OF $\mathcal{R}$

$\mathcal{R} = \mathcal{L}\mathcal{D}\mathcal{U} = \mathcal{L}\mathcal{P} = \mathcal{W}\mathcal{U}$		$\mathcal{R} = \bar{\mathcal{U}}\bar{\mathcal{D}}\bar{\mathcal{L}} = \bar{\mathcal{U}}\bar{\mathcal{W}} = \bar{\mathcal{P}}\bar{\mathcal{L}}$
if $i>0$ then $\begin{cases} q_{1i} = R_i \\ k = 2(1)i \\ \quad \left[q_{ki} = q_{k-1,i} - \alpha_{k-2} p_{k-1,i} \right] \end{cases}$	$\mathcal{P}$	if $i>0$ then $\begin{cases} \bar{q}_{i1} = R_i \\ k = 2(1)i \\ \quad \left[\bar{q}_{ik} = \bar{q}_{i,k-1} - \bar{p}_{i,k-1} \overset{*}{\beta}_{k-2} \right] \end{cases}$
$\alpha_{i-1} = q_{ii} (p_{ii})^{-1}$		$\overset{*}{\beta}_{i-1} = (\bar{p}_{ii})^{-1} \bar{q}_{ii}$
$\begin{cases} p_{1,i+1} = R_i \\ k = 2(1) i+1 \\ \quad \left[p_{k,i+1} = p_{k-1,i} - \overset{*}{\alpha}_{k-2} q_{k-1,i} \right] \end{cases}$		$\begin{cases} \bar{p}_{i+1,1} = R_i \\ k = 2(1) i+1 \\ \quad \left[\bar{p}_{i+1,k} = \bar{p}_{i,k-1} - \bar{q}_{i,k-1} \beta_{k-2} \right] \end{cases}$
$\delta_i = p_{i+1,i+1} \quad (= R_i^z)$	$\mathcal{D}$	$\bar{\delta}_i = \bar{p}_{i+1,i+1} \quad (= R_i^{\varepsilon})$
$\begin{cases} k = 1(1) i+1 \\ \quad \left[u_{k,i+1} = \delta_{k+1}^{-1} p_{k,i+1} \right] \end{cases}$	$\mathcal{U}$	$\begin{cases} k = 1(1) i+1 \\ \quad \left[\bar{u}_{i+1,k} = \bar{p}_{i+1,k} \bar{\delta}_{k-1}^{-1} \right] \end{cases}$
if $i>0$ then $\begin{cases} c_{1i} = R_{-i} \\ k = 2(1)i \\ \quad \left[c_{ik} = c_{i,k-1} - w_{i,k-1} \beta_{k-2} \right] \end{cases}$	$\mathcal{W}$	if $i>0$ then $\begin{cases} \bar{c}_{1i} = R_{-i} \\ k = 2(1)i \\ \quad \left[\bar{c}_{ki} = \bar{c}_{k-1,i} - \overset{*}{\alpha}_{k-2} \bar{w}_{k-1,i} \right] \end{cases}$
$\beta_{i-1} = (w_{ii})^{-1} c_{ii}$		$\overset{*}{\alpha}_{i-1} = \bar{c}_{ii} (\bar{w}_{ii})^{-1}$
$\begin{cases} w_{i+1,1} = R_{-i} \\ k = 2(1) i+1 \\ \quad \left[w_{i+1,k} = w_{i,k-1} - c_{i,k-1} \overset{*}{\beta}_{k-2} \right] \end{cases}$		$\begin{cases} \bar{w}_{1,i+1} = R_{-i} \\ k = 2(1) i+1 \\ \quad \left[\bar{w}_{k,i+1} = w_{k-1,i} - \alpha_{k-2} \bar{c}_{k-1,i} \right] \end{cases}$
$\delta_i = w_{i+1,i+1} \quad (= R_i^z)$	$\mathcal{D}$	$\bar{\delta}_i = \bar{w}_{i+1,i+1} \quad (= R_i^{\varepsilon})$
$\begin{cases} k = 1(1) i+1 \\ \quad \left[l_{i+1,k} = w_{i+1,k} \delta_{k-1}^{-1} \right] \end{cases}$	$\mathcal{L}$	$\begin{cases} k = 1(1) i+1 \\ \quad \left[\bar{l}_{k,i+1} = \bar{\delta}_{k-1}^{-1} \bar{w}_{k,i+1} \right] \end{cases}$
$\mathcal{R}^T = \overset{*}{\mathcal{U}} \overset{*}{\mathcal{D}} \overset{*}{\mathcal{L}} = \overset{*}{\mathcal{P}} \overset{*}{\mathcal{L}} = \overset{*}{\mathcal{U}} \overset{*}{\mathcal{W}}$		$\mathcal{R}^T = \overset{*}{\mathcal{L}} \overset{*}{\mathcal{D}} \overset{*}{\bar{\mathcal{U}}} = \overset{*}{\mathcal{L}} \overset{*}{\bar{\mathcal{P}}} = \overset{*}{\mathcal{W}} \overset{*}{\bar{\mathcal{U}}}$

ALGORITHM 3 CHOLESKY (scalar case)

numbering = order of comp.	$R = L\,L^T$	numbering ≠ order of computation	$R = U\,U^T$
$R = L\,L^T$ 	$j = 1(1)n$ $l_{jj} = \sqrt{r_{jj} - \sum_{k=1}^{j-1} l_{jk}^2}$ $i = j+1(1)n$ $l_{ij} = \frac{1}{l_{jj}}\left(r_{ij} - \sum_{k=1}^{j-1} l_{ik}\,l_{jk}\right)$	$R = U\,U^T$ 	$i = n(-1)1$ $u_{ii} = \sqrt{r_{ii} - \sum_{k=i+1}^{n} u_{ik}}$ $j = i-1(-1)1$ $u_{ji} = \frac{1}{u_{ii}}\left(r_{ij} - \sum_{k=i+1}^{n} u_{ik}\,u_{jk}\right)$
OR		OR	
$R = L\,L^T$ 	$j = 1(1)n$ $i = 1(1)j-1$ $l_{ji} = \frac{1}{l_{ii}}\left(r_{ij} - \sum_{k=1}^{i-1} l_{ik}\,l_{jk}\right)$ $l_{jj} = \sqrt{r_{jj} - \sum_{k=1}^{j-1} l_{ik}^2}$	$R = U\,U^T$ 	$i = n(-1)1$ $j = n(-1)i+1$ $u_{ij} = \frac{1}{u_{ii}}\left(r_{ji} - \sum_{k=j+1}^{n} u_{jk}\,u_{ik}\right)$ $u_{ii} = \sqrt{r_{ii} - \sum_{k=i+1}^{n} u_{ik}^2}$
$Ly = b$	$i = 1(1)n$ $y_i = \frac{1}{l_{ii}}\left(b_i - \sum_{k=1}^{i-1} l_{ik}\,y_k\right)$	$Uy = b$	$i = n(-1)1$ $y_i = \frac{1}{u_{ii}}\left(b_i - \sum_{k=i+1}^{n} u_{ik}\,y_k\right)$
$L^T x = y$	$i = n(-1)1$ $x_i = \frac{1}{l_{ii}}\left(y_i - \sum_{k=i+1}^{n} l_{ki}\,x_k\right)$	$U^T x = y$	$i = 1(1)n$ $x_i = \frac{1}{u_{ii}}\left(y_i - \sum_{k=1}^{i-1} u_{ki}\,x_k\right)$
$P = L^{-1}$ 	$i = 1(1)n$ $p_{ii} = 1/l_{ii}$ $j = i-1(-1)1$ $p_{ij} = -\frac{1}{l_{jj}}\left(\sum_{k=j+1}^{i} p_{ik}\,l_{kj}\right)$	$Q = U^{-1}$ 	$i = 1(1)n$ $q_{ii} = 1/u_{ii}$ $j = i+1(1)n$ $q_{ij} = -\frac{1}{u_{jj}}\left(\sum_{k=i}^{j-1} q_{ik}\,u_{kj}\right)$
OR		OR	
$P =$ 	$i = 1(1)n$ $p_{ii} = 1/l_{ii}$ $j = i+1(1)n$ $p_{ji} = -\frac{1}{l_{jj}}\left(\sum_{k=i}^{j-1} l_{jk}\,p_{ki}\right)$	$Q =$ 	$i = 1(1)n$ $q_{ii} = 1/u_{ii}$ $j = i-1(-1)1$ $q_{ji} = -\frac{1}{u_{jj}}\left(\sum_{k=j+1}^{i} u_{jk}\,q_{ki}\right)$
$S = R^{-1}$	$i = 1(1)n$ $j = i(1)n$ $s_{ij} = s_{ji} = \sum_{k=j}^{n} p_{ki}\,p_{kj}$	$S = R^{-1}$	$i = 1(1)n$ $j = i(1)n$ $s_{ij} = s_{ji} = \sum_{k=1}^{i} q_{ik}\,q_{jk}$

ALGORITHM 4 RATIONAL CHOLESKY (scalar case)

order of comp.	$R = L D L^T$	order of comp.	$R = U D U^T$
$R = LDL^T$ 	$j = 1(1)n$ $d_j = r_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 d_k$ $i = j+1(1)n$ $l_{ij} = \frac{1}{d_j}\left(r_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk} d_k\right)$	$R = UDU^T$ 	$i = n(-1)1$ $d_i = r_{ii} - \sum_{k=i+1}^{n} u_{ik}^2 d_k$ $j = i-1(-1)1$ $u_{ji} = \frac{1}{d_i}\left(r_{ij} - \sum_{k=i+1}^{n} u_{ik} u_{jk} d_k\right)$
OR		OR	
$R = LDL^T$ 	$j = 1(1)n$ $i = 1(1)j-1$ $l_{ji} = \frac{1}{d_i}\left(r_{ij} - \sum_{k=1}^{i-1} l_{ik} l_{jk} d_k\right)$ $d_j = r_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 d_k$	$R = UDU^T$ 	$i = n(-1)1$ $j = n(-1)i+1$ $u_{ji} = \frac{1}{d_i}\left(r_{ji} - \sum_{k=j+1}^{n} u_{jk} u_{ik} d_k\right)$ $d_i = r_{ii} - \sum_{k=i+1}^{n} u_{ik}^2 d_k$
$Ly = b$	$i = 1(1)n$ $y_i = b_i - \sum_{k=1}^{i-1} l_{ik} y_k$	$Uy = b$	$i = n(-1)1$ $y_i = b_i - \sum_{k=i+1}^{n} u_{ik} y_k$
$L^T D x = y$	$i = n(-1)1$ $x_i = \frac{1}{d_i}\left(y_i - \sum_{k=i+1}^{n} l_{ki} x_k\right)$	$U^T D x = y$	$i = 1(1)n$ $x_i = \frac{1}{d_i}\left(y_i - \sum_{k=1}^{i-1} u_{ki} x_k\right)$
$P = L^{-1}$ 	$i = 1(1)n$ $j = i-1(-1)1$ $P_{ij} = -\sum_{k=j+1}^{i} P_{ik} l_{kj}$	$Q = U^{-1}$ 	$i = 1(1)n$ $j = i+1(1)n$ $q_{ij} = -\sum_{k=i}^{j-1} q_{ik} u_{kj}$
OR		OR	
$P =$ 	$i = 1(1)n$ $j = i+1(1)n$ $P_{ji} = -\sum_{k=i}^{j-1} l_{jk} P_{ki}$	$Q =$ 	$i = 1(1)n$ $j = i-1(-1)1$ $q_{ji} = -\sum_{k=j+1}^{i} u_{jk} q_{ki}$
$S = \vec{R}^{-1}$	$i = 1(1)n$ $j = i(1)n$ $s_{ij} = s_{ji} = \sum_{k=j}^{n} P_{ki} P_{kj} / d_k$	$S = \vec{R}^{-1}$	$i = 1(1)n$ $j = i(1)n$ $s_{ij} = s_{ji} = \sum_{k=1}^{i} q_{ik} q_{jk} d_k$

ALGORITHM 5 RECURSIVE RATIONAL CHOLESKY ROW by ROW

$$R = \mathcal{L} \mathcal{D} \mathcal{L}^t \quad ; \Lambda = \mathcal{L}^{-1}$$

$i = 1, 2, \ldots$

$j = 1\,(1)\,i-1$

$$l_{ij} = \delta_j^{-1} \sum_{k=1}^{j} \Lambda_{jk} r_{ki}$$

$$\delta_i = R_{ii} - \sum_{k=1}^{i-1} l_{ik} \delta_k l_{ik}$$

$$l_{ii} = I$$

$$\Lambda_{ii} = I$$

$j = 1\,(1)\,i-1$

$$\Lambda_{ij} = - \sum_{k=1}^{i} \Lambda_{i+1-k,j} \, l_{ik}$$

ALGORITHM 6 RECURSIVE RATIONAL CHOLESKY COLUMN by COLUMN

$$R_{m+1} = \bar{\mathcal{L}}_{m+1} \bar{\mathcal{D}}_{m+1} \bar{\mathcal{L}}_{m+1}^t$$



recursive implementation details of this and other methods are found in the literature
Gill et al – Fletcher - Powell

$m = 1, 2, \ldots$

$$\bar{\delta}_m = R_{mm}$$

$$b_m = r_m \bar{\delta}_m^{-1}$$

solve $\bar{\mathcal{L}}_m v_m = b_m$

factorize $\bar{\mathcal{L}} \bar{\mathcal{D}} \bar{\mathcal{L}}_m^t - b_m \bar{\delta}_m b_m^t = \tilde{\mathcal{L}}_m \tilde{\mathcal{D}}_m \tilde{\mathcal{L}}_m^t$

$$\tilde{\mathcal{L}}_m = \bar{\mathcal{L}}_m \hat{\mathcal{L}}_m$$

$$(\hat{\mathcal{L}}_m)_{ji} = (v_m)_j \beta_i \qquad \beta_i = \Big[\underbrace{\bar{\delta}_m^{-1} - \sum_{k=j+1}^{m} \beta_k \tilde{\delta}_k \beta_k^t}_{\alpha_i} \Big] (v_m)_i^t \, \tilde{\delta}_i^{-1}$$

$$\tilde{\delta}_i = \bar{\delta}_i - (v_m)_i \alpha_i (v_m)_i^t$$

$$\bar{\mathcal{L}}_{m+1} = \begin{array}{|c|c|} \hline \tilde{\mathcal{L}}_m & b_m \\ \hline & I \\ \hline \end{array} \quad ; \quad \bar{\mathcal{D}}_{m+1} = \begin{array}{|c|c|} \hline \tilde{\mathcal{D}}_m & \\ \hline & \tilde{\delta}_m \\ \hline \end{array}$$

(SOME) REFERENCES

On Hankel and Toeplitz matrices

1. AKAIKE H. (1973) - Block Toeplitz matrix inversion
   SIAM J. Appl. Math 24 234-241

2. BAREISS E.H. (1969) - Numerical solution of linear equations with Toeplitz and vector Toeplitz matrices
   Numer. Math. 13 (5) 404-424

3. FARDEN D.C. (1977) - The solution of a special set of hermitian Toeplitz linear equations. T.O.M.S. 3 (2) 159-163

4. TRENCH W.F. (1964) - An algorithm for the inversion of finite Toeplitz matrices J. SIAM 12 515-522

5. TRENCH W.F. (1974) - Inversion of Toeplitz band matrices
   Math. Comp. 28 (128) 1089-1095

6. ZOHAR S. (1969) - Toeplitz matrix inversion: the algorithm of W.F. Trench J. ACM 16 592-601

7. ZOHAR S. (1974) - The solution of a Toeplitz set of linear equations J. ACM 21 (2) 272-276

On Cholesky Factorization

1. BIERMAN G.J. (1977) - Factorization methods for discrete sequential estimation, Ac. Press, Math. in sc. & engng vol 128

2. BUNCH JR - PARLETT B N (1974) - Direct methods for solving symmetric indefinite systems of linear equations
   SIAM J. Num. Anal 8 (4) 639-655

3. CHAN C - TWEEDALE A - OLESKY D - WONG, S. (1974) - LINSYS: linear systems package (fortran programs) Univ. of Victoria.

4. GIBSON J.D (1977) - On reflection coefficients and the Cholesky decomposition IEEE Trans. ASSP febr. '77, 93-95

5. FLETCHER R - POWELL M.J.D. (1974) - On the modification of LDL^t factorizations - Math. Comp. 28 (128) 1067-1087

6. GILL P. MURRAY W - SAUNDERS M (1975) - Methods for computing and modifying the LDV factors of a matrix -
   Math. of Comp. 29 (132) 1051-1077.

7. WILKINSON J.H. - REINSCH C (1971) - Handbook for automatic computation vol II linear algebra, Springer verlag.

8. HARWELL SUBROUTINE LIBRARY - Theoretical Physics division, Atomic Energy Research Establishment (UKAERE), Harwell Berkshire