Cholesky Decompositions

Thursday, 5 February 2026 20:53

Problems

We like ALTRO. Recall that we find the feedback gain using:
$\delta u_k = -(Q_{uu} + \rho I)^{-1}(Q_{ux}\delta x_k + Q_u)$
We may also remember that:

$$Q_{uu} = \frac{\partial^2 l}{\partial u_k^2} + B_k^\top P_{k+1} B_k + \frac{\partial c_k}{\partial u_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial u_k}$$

$$Q_{ux} = \frac{\partial^2 l}{\partial u_k \partial x_k} + B_k^\top P_{k+1} A_k + \frac{\partial c_k}{\partial u_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial x_k}$$

$$Q_u = \frac{\partial l}{\partial u_k} + B_k^\top p_{k+1} + \lambda_k \frac{\partial c_k}{\partial u_k} + \frac{\partial c_k}{\partial u} I_{\mu,k} c_k$$

As the constraints costs increase over successive outer loops, so do Q_{uu}, Q_{ux}, Q_u. This is further exacerbated by Q_{uu}, Q_{ux}, Q_u depending on P_{k+1}, p_{k+1}, which depend on Q_{uu}, Q_{ux}, Q_u of the future timestep. Overall the matrices grow in size.

Eventually, either after enough constraint increases, or given a long-enough trajectory
$P_k = Q_{xx} + K_k^\top Q_{uu} K_k + K_k^\top Q_{ux} + Q_{xu} K_k$
will run into numerical issues where the addition/subtraction of such large matrices results in floating point loss.

Cholesky Decomposition

QR-Decomposition P_N

The terminal cost is:

$$P_N = \frac{\partial^2 l_f(x_N)}{\partial x^2} + \frac{\partial c(x_N)}{\partial x}^\top I_{\mu,N} \frac{\partial c(x_N)}{\partial x}$$

$$P_N = \begin{bmatrix} \sqrt{Q_{xx,N}} \\ \sqrt{I_{\mu,N}} \frac{\partial c(x_N)}{\partial x} \end{bmatrix}^\top \begin{bmatrix} \sqrt{Q_{xx,N}} \\ \sqrt{I_{\mu,N}} \frac{\partial c(x_N)}{\partial x} \end{bmatrix}$$

$P_N = M^\top M$
$P_N = (QR)^\top (QR)$
$P_N = R^\top Q^\top Q R$
Since Q is unitary ($Q^\top = Q^{-1}$), property of QR-decomposition:
$P_N = R^\top R$

Consequently: $S_N = R = \sqrt{P_N} = QR\left(\begin{bmatrix} \sqrt{Q_{xx,N}} \\ \sqrt{I_{\mu,N}} \frac{\partial c(x_N)}{\partial x} \end{bmatrix}\right)$

QR-Decomposition Q_{xx}

$$Q_{xx} = \frac{\partial^2 l}{\partial x_k^2} + A_k^\top P_{k+1} A_k + \frac{\partial c_k}{\partial x_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial x_k}$$

$$Q_{xx} = \begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial x_k^2}} \\ S_{k+1} A_k \\ \sqrt{I_{\mu,k}} \frac{\partial c_k}{\partial x_k} \end{bmatrix}^\top \begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial x_k^2}} \\ S_{k+1} A_k \\ \sqrt{I_{\mu,k}} \frac{\partial c_k}{\partial x_k} \end{bmatrix}$$

Using same principles:

$$Z_{xx} = \sqrt{Q_{xx}} = QR\left(\begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial x_k^2}} \\ S_{k+1} A_k \\ \sqrt{I_{\mu,k}} \frac{\partial c_k}{\partial x_k} \end{bmatrix}\right)$$

QR-Decomposition Q_{uu}

$$Q_{uu} = \frac{\partial^2 l}{\partial u_k^2} + B_k^\top P_{k+1} B_k + \frac{\partial c_k}{\partial u_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial u_k}$$

$$Q_{uu} = \begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial u_k^2}} \\ S_{k+1}B_k \\ \sqrt{I_{\mu,k}}\frac{\partial c_k}{\partial u_k} \end{bmatrix}^\top \begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial u_k^2}} \\ S_{k+1}B_k \\ \sqrt{I_{\mu,k}}\frac{\partial c_k}{\partial u_k} \end{bmatrix}$$

$$Z_{uu} = \sqrt{Q_{uu}} = QR\left(\begin{bmatrix} \sqrt{\frac{\partial^2 l}{\partial u_k^2}} \\ S_{k+1}B_k \\ \sqrt{I_{\mu,k}}\frac{\partial c_k}{\partial u_k} \end{bmatrix}\right)$$

Other QR-Decompositions

Unfortunately, the form of Q_{xu}, Q_{ux} means it is not possible to do a Cholesky Decomposition in this manner.

$$Q_{xu} = \frac{\partial^2 l}{\partial x_k \partial u_k} + A_k^\top P_{k+1} B_k + \frac{\partial c_k}{\partial x_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial u_k}$$
$$Q_{ux} = \frac{\partial^2 l}{\partial u_k \partial x_k} + B_k^\top P_{k+1} A_k + \frac{\partial c_k}{\partial u_k}^\top I_{\mu,k} \frac{\partial c_k}{\partial x_k}$$

Using Cholesky Decomposition

We may now use S_k, Z_{xx}, Z_{uu} for two important formulas:

Feedback Control Law

Recall

$$\delta u_k = -Q_{uu}^{-1}(Q_{ux}\delta x_k + Q_u)$$
$$\delta u_k = K_k \delta x_k + d_k$$

$$K_k = -Q_{uu}^{-1}Q_{ux} = -Z_{uu}^{-1}Z_{uu}^{-\top}Q_{ux}$$
$$d_k = -Q_{uu}^{-1}Q_u = -Z_{uu}^{-1}Z_{uu}^{-\top}Q_u$$

Observation: Here, the regularization term is not used!

Gradient and change in cost-to-go

$$\Delta V_k = Q_u^\top d_k + \frac{1}{2} d_k^\top Q_{uu} d_k$$
$$\Delta V_k = Q_u^\top d_k + \frac{1}{2} (Z_{uu} d_k)^\top Z_{uu} d_k$$

$$p = Q_x + K_k^\top Q_{uu} d_k + K_k^\top Q_u + Q_{xu} d_k$$
$$p = Q_x + (Z_{uu}K_k)^\top (Z_{uu} d_k) + K_k^\top Q_u + Q_{xu} d_k$$

Recursive Formula for P_k

Finally, we are getting to what our original problem was. Let's have at it.

$$P_k = Q_{xx} + K_k^\top Q_{uu} K_k + K_k^\top Q_{ux} + Q_{xu} K_k$$
$$P_k = \begin{bmatrix} I \\ K_k \end{bmatrix}^\top \begin{bmatrix} Q_{xx} & Q_{xu} \\ Q_{ux} & Q_{uu} \end{bmatrix} \begin{bmatrix} I \\ K_k \end{bmatrix}$$
$$P_k = \begin{bmatrix} I \\ K_k \end{bmatrix}^\top \begin{bmatrix} Z_{xx}^\top & 0 \\ C^\top & D^\top \end{bmatrix} \begin{bmatrix} Z_{xx} & C \\ 0 & D \end{bmatrix} \begin{bmatrix} I \\ K_k \end{bmatrix}$$
$$P_k = \begin{bmatrix} Z_{xx} + CK \\ DK \end{bmatrix}^\top \begin{bmatrix} Z_{xx} + CK \\ DK \end{bmatrix}$$

with

$$C = Z_{xx}^{-\top} Q_{xu}$$
$$D = \sqrt{Z_{uu}^\top Z_{uu} - Q_{ux} Z_{xx}^{-1} Z_{xx}^{-\top} Q_{xu}} = \boldsymbol{DownDate}(Z_{uu}, Z_{xx}^{-\top} Q_{xu})$$

We can use the same trick as above to find:

$$S_k = \sqrt{P_k} = QR\left(\begin{bmatrix} Z_{xx} + CK \\ DK \end{bmatrix}\right)$$