Decomposition and Substitution

The base class for all linear solvers in Rts, *RLinearSolver*, contains the C++ code that implements the approach described in the following. Namely, a popular technique for solving for **x** in the system of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{1}$$

is to split **A** into two triangular parts. Several algorithms exist to achieve that, such as LU decomposition and Cholesky decomposition. The advantage of this approach is understood by considering the decomposition

$$\mathbf{A} = \mathbf{L}\mathbf{U} \tag{2}$$

where **L** is a lower-tringular matrix and **U** is an upper-triangular matrix, both having diagonal elements that are typically different from zero and unity. Using this decomposition, Eq. (1) reads

$$\mathbf{L}\underbrace{\mathbf{U}\mathbf{x}}_{\mathbf{y}} = \mathbf{L}\mathbf{y} = \mathbf{b} \tag{3}$$

where the auxiliary vector

$$\mathbf{y} = \mathbf{U}\mathbf{x} \tag{4}$$

has been defined. Assuming **L** and **U** have already been determined, the vector **y** is determined by solving Eq. (3), i.e., by the forward substitution algorithm

$$y_i = \frac{b_i - \sum_{j=1}^{i-1} \left(L_{ij} y_j \right)}{L_{ii}} \qquad \text{for} \quad i = 1, \ \dots \ , n \tag{5}$$

where n is the number of equations. Once **y** is determined, **x** is determined by solving Eq. (4), i.e., by the backward substitution algorithm

$$x_i = \frac{y_i - \sum_{j=i+1}^{n} \left(U_{ij} x_j \right)}{U_{ii}} \qquad \text{for} \quad i = n, \ \dots \ , 1 \tag{6}$$

If the Cholesky decomposition is used then $\mathbf{U}=\mathbf{L}^{\mathrm{T}}$ and Eq. (6) is readily modified to accommodate that situation by setting $U_{ij}=L_{ji}$ (notice the switch of indices) and $U_{ii}=L_{ii}$. If the system of equations needs to be solved repeated with the same **A**-matrix but different right-hand-sides **b** then the decomposition is done once while the forward/backward substitution is repeated for each **b**-vector.