rtklib 深入解读

1　模糊度固定策略（rtkpos/resamb_LAMBDA）

1.1　模糊度固定的准则

$$(\bar{N}-\bar{N}^0)^{\mathrm{T}}\mathbf{Q}_{\bar{N}}^{-1}(\bar{N}-\bar{N}^0)=\min \quad \longleftrightarrow \quad \text{模糊度搜索}$$

由于这是整数最小二乘（ILSE），我们无法通过经典的最小二乘公式快速确定，因此，我们必须为要每个模糊度参数划定一个固定的范围，逐个搜索寻求最优解

$$\begin{aligned} f &=(\bar{N}-\bar{N}^0)^{\mathrm{T}}(\bar{L}^{\mathrm{T}}\bar{D}\bar{L})^{-1}(\bar{N}-\bar{N}^0)=(\bar{N}-\bar{N}^0)^{\mathrm{T}}\bar{L}^{-1}\bar{D}^{-1}(\bar{L}^{\mathrm{T}})^{-1}(\bar{N}-\bar{N}^0) \\ &=[(\bar{L}^{\mathrm{T}})^{-1}(\bar{N}-\bar{N}^0)]^{\mathrm{T}}\bar{D}^{-1}[(\bar{L}^{\mathrm{T}})^{-1}(\bar{N}-\bar{N}^0)]=\min \end{aligned}$$

$$f=(\mathbf{Z}-\bar{\mathbf{N}}^0)^{\mathrm{T}}\bar{\mathbf{D}}^{-1}(\mathbf{Z}-\bar{\mathbf{N}}^0)=\min$$

自底向上的序贯解算

$$\bar{\mathbf{N}}^0=\begin{pmatrix}\bar{\mathbf{N}}_1^0\\ \bar{\mathbf{N}}_2^0\\ \vdots\\ \bar{\mathbf{N}}_n^0\end{pmatrix},\mathbf{Z}=\begin{pmatrix}\bar{\mathbf{N}}_1+\sum_{j=2}^{n}h_{1j}(\bar{\mathbf{N}}_j-\bar{\mathbf{N}}_j^0)\\ \bar{\mathbf{N}}_2+\sum_{j=3}^{n}h_{2j}(\bar{\mathbf{N}}_j-\bar{\mathbf{N}}_j^0)\\ \vdots\\ \bar{\mathbf{N}}_{n-1}+h_{n-1n}(\bar{\mathbf{N}}_n-\bar{\mathbf{N}}_n^0)\\ \bar{\mathbf{N}}_n\end{pmatrix}$$

$$f=\frac{(Z_1-\bar{N}_1^0)^2}{\bar{d}_1}+\frac{(Z_2-\bar{N}_2^0)^2}{\bar{d}_2}+\cdots+\frac{(Z_n-\bar{N}_n^0)^2}{\bar{d}_n}\le \mathrm{r}^2$$

$$\begin{cases} A_i=Z_i-\sqrt{\bar{d}_i r^2-\sum_{k=i+1}^{n}\frac{(Z_k-\bar{N}_k^0)^2}{\bar{d}_k}\bar{d}_i} \\ B_i=Z_i+\sqrt{\bar{d}_i r^2-\sum_{k=i+1}^{n}\frac{(Z_k-\bar{N}_k^0)^2}{\bar{d}_k}\bar{d}_i} \end{cases} \qquad \text{剪枝：}\sum_{k=i+1}^{n}\frac{(Z_k-\bar{N}_k^0)^2}{\bar{d}_k}>r^2$$

1.2　LAMBDA 降相关方法

目的：受限于模糊度浮点解的精度以及强相关性，搜索备选节点数量会很大，此方法可以在搜索椭球容积不变的情况下降低浮点解之间的相关性同时降低模糊度参数的方差，从而达到压缩搜索椭球、快速固定的目的

$$z=Z^T a \quad Q_z=Z^T Q_a Z \quad \breve{a}=Z^{-T}\breve{z}$$

准则：

1. Z 行列式的绝对值为 1
2. Z 中每个元素必须为整数
3. $\bar{L}$ 尽可能的对角占优，即非对角线元素小于等于 0.5
4. 方差 $\bar{D}$ 尽可能地按照降序排序

1.2.1 整数高斯变换

$$\mathbf{L}' = \mathbf{L}\cdot\mathbf{Z}_{ij} = \begin{pmatrix} 1 & & & & & & & \\ l_{21} & 1 & & & & & & \\ \vdots & \vdots & \ddots & & & & & \\ \vdots & \vdots & \cdots & 1 & & & & \\ \vdots & \vdots & \cdots & \vdots & \ddots & & & \\ l_{i1} & l_{i2} & \cdots & \boxed{l_{ij}} & \cdots & 1 & & \\ \vdots & \vdots & \cdots & \vdots & -(l_{ij}) & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nj} & \cdots & \cdots & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -(l_{ij}) & \ddots & \\ & & & & 1 \end{pmatrix}$$

自上而下降相关

1.2.2 条件方差降序

降序的判定标准：$d_i + l'^2_{i+1,i} d_{i+1} \le d_{i+1}$

$$\mathbf{P}_{i,i+1} = \begin{pmatrix} \mathbf{I}_{i-1} & & \\ & \mathbf{P} & \\ & & \mathbf{I}_{n-(i+1)} \end{pmatrix}, \mathbf{P} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\bar{\mathbf{Q}}_N = \mathbf{P}^{\mathrm{T}}_{i,i+1}\mathbf{Q}'_N\mathbf{P}_{i,i+1} = \bar{\mathbf{L}}^{\mathrm{T}}\bar{\mathbf{D}}\bar{\mathbf{L}} = \begin{bmatrix} \mathbf{L}^{\mathrm{T}}_{11} & \bar{\mathbf{L}}^{\mathrm{T}}_{21} & \mathbf{L}^{\mathrm{T}}_{31} \\ & \bar{\mathbf{L}}^{\mathrm{T}}_{22} & \bar{\mathbf{L}}^{\mathrm{T}}_{32} \\ & & \mathbf{L}^{\mathrm{T}}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{D}_{11} & & \\ & \bar{\mathbf{D}}_{22} & \\ & & \mathbf{D}_{33} \end{bmatrix} \begin{bmatrix} \mathbf{L}_{11} & & \\ \bar{\mathbf{L}}_{21} & \bar{\mathbf{L}}_{22} & \\ \mathbf{L}_{31} & \bar{\mathbf{L}}_{32} & \mathbf{L}_{33} \end{bmatrix}$$

$$\bar{L}_{32} = \begin{bmatrix} l_{i+2,i+1} & l_{i+2,i} \\ l_{i+3,i+1} & l_{i+3,i} \\ \vdots & \vdots \\ l_{n,i+1} & l_{n,i} \end{bmatrix} \qquad \bar{D}_{22} = \begin{bmatrix} d'_i & 0 \\ 0 & d'_{i+1} \end{bmatrix} = \begin{bmatrix} \boxed{\dfrac{d_i d_{i+1}}{d_i + l^2_{i+1,i} d_{i+1}}} \ge d_i & 0 \\ 0 & \boxed{d_i + l^2_{i+1,i} d_{i+1}} \end{bmatrix} \le d_{i+1}$$

$$\bar{L}_{21} = \begin{bmatrix} \dfrac{-l_{i+1,i}}{\dfrac{d_i}{d_i + l^2_{i+1,i} d_{i+1}}} & \dfrac{1}{\dfrac{l_{i+1,i} d_{i+1}}{d_i + l^2_{i+1,i} d_{i+1}}} \end{bmatrix} L_{21} \qquad \bar{L}_{22} = \begin{bmatrix} 1 & 0 \\ l'_{i+1,i} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \dfrac{l_{i+1,i} d_{i+1}}{d_i + l^2_{i+1,i} d_{i+1}} & 1 \end{bmatrix}$$

降序后依然判定依据的证明：

$$\frac{d_i d_{i+1}}{d_i + l_{i+1,i}^2 d_{i+1}} + \frac{l_{i+1,i}^2 d_{i+1}^2}{d_i + l_{i+1,i}^2 d_{i+1}} \le d_i + l_{i+1,i}^2 d_{i+1}$$

$$d_i + l_{i+1,i}^2 \cdot d_{i+1} \le \frac{\left(d_i + l_{i+1,i}^2 d_{i+1}\right)^2}{d_{i+1}} \le d_{i+1}$$

1.2.3　LAMBDA 降相关步骤

由于降序会影响 L 矩阵左边，而右边基本不受影响，依然在小于 0.5 的范畴，因此降相关和降序不能分开进行，并且应该按照自右向左，自上向下交替的顺序进行。同时，为了确保，相邻方差最大程度地交换，我们先降相关，然后再判断该列是否需要降序。

1.3　mLAMBDA 搜索策略

之前的整数模糊度搜索算法里通过某种方式确定 r 的取值，后续的所有搜索都在固定半径的超椭圆区域中进行，这种固定半径搜索方式的效率依赖于 r 取值的合理性。mLAMBDA 方法根据误差分布理论和计算实践，优先从备选组中心往两端搜索，在此过程不断收缩搜索半径，从而达到提高搜索效率的目的。

```
function: Optis = MSEARCH(L, D, ẑ, p)
maxDist = +∞        // maxDist: current χ^2
k = n; dist(k) = 0
endSearch = false
count = 0        // count: the number of candidates
Initialize an n × n zero matrix S
    // S will be used for computing z̄(k)
z̄(n) = ẑ(n)        // see Eq. (17)
z(n) = ⌊z̄(n)⌉; y = z̄(n) − z(n)
step(n) = sgn(y); imax = p
while endSearch = false
    newDist = dist(k) + y^2/D(k, k)
        // newDist = Σ_{j=k}^{n} (z_j − z̄_j)^2/d_j
    if newDist < maxDist
        if k ≠ 1        //Case 1: move down
            k = k − 1
            dist(k) = newDist
                //dist(k) = Σ_{j=k+1}^{n} (z_j − z̄_j)^2/d_j
            S(k, 1:k) = S(k + 1, 1:k)
              +(z(k+1) − z̄(k+1))L(k+1, 1:k)
            // S(k, 1 : k) = Σ_{j=k+1}^{n} (z_j − z̄_j)L(j, 1:k)
            z̄(k) = ẑ(k) + S(k, k)      // see Eqn. (17)
            z(k) = ⌊z̄(k)⌉; y = z̄(k) − z(k)
            step(k) = sgn(y)
        else
            //Case 2: store the found candidate
             //and try next valid integer
            if count < p − 1
                // store the first p − 1 initial points
                count = count + 1
                Optis(:, count) = z(1 : n)
                fun(count) = newDist // store f(z)
            else
                Optis(:, imax) = z(1 : n)
                fun(imax) = newDist
                imax = arg max_{1≤i≤p} fun(i)
                maxDist = fun(imax)
            end
            z(1) = z(1) + step(1)
            //next valid integer
            y = z̄(1) − z(1)
            step(1) = −step(1) − sgn(step(1))
            // cf. Eqs. (25) and (26)
        end
    else        //Case 3: exit or move up
        if k = n
            endSearch = true
        else
            k = k + 1        // move up
            z(k) = z(k) + step(k)
                // next valid integer
            y = z̄(k) − z(k)
            step(k) = −step(k) − sgn(step(k))
                // cf. Eqs. (25) and (26)
        end
    end
end
```

2 双差固定解的推导

$$y = \begin{bmatrix} H_l & H_r \end{bmatrix} \cdot \begin{bmatrix} x^a \\ x^o \end{bmatrix} \qquad x_{float} = \begin{bmatrix} Q_R & Q_{RN} \\ Q_{NR} & Q_N \end{bmatrix} \begin{bmatrix} H_l^{\,T} \\ H_r^{\,T} \end{bmatrix} P \cdot y$$

$$Q_N = \left(A_{22} - A_{21} A_{11}^{-1} A_{12} \right)^{-1}$$

$$Q_{RN} = -A_{11}^{-1} A_{12} Q_N$$

$$Q_R = \left(A_{11} - A_{12} A_{22}^{-1} A_{21} \right)^{-1}$$

$$A_{11} = H_l^{\,T} P H_l \quad A_{12} = H_l^{\,T} P H_r \quad A_{21} = H_r^{\,T} P H_l \quad A_{22} = H_r^{\,T} P H_r$$

$$x_{float}^a = \left(Q_R H_l^{\,T} + Q_{RN} H_r^{\,T} \right) P y$$

$$x_{float}^o = \left(Q_{NR} H_l^{\,T} + Q_N H_r^{\,T} \right) P y$$

$$y - H_r x^o = H_l x^a \quad A_{11}^{-1} A_{12} = -Q_{RN} Q_N^{-1} \quad A_{11}^{-1} = Q_R - Q_{RN} Q_N^{-1} Q_{NR}$$

$$\begin{aligned} x_{fix} &= A_{11}^{-1} H_l^{\,T} P \left(y - H_r x^o \right) = A_{11}^{-1} H_l^{\,T} P y - A_{11}^{-1} A_{12} x^o \\ &= \left(Q_R - Q_{RN} Q_N^{-1} Q_{NR} \right) H_l^{\,T} P y + Q_{RN} Q_N^{-1} x^o \\ &= \left(Q_R H_l^{\,T} + Q_{RN} H_r^{\,T} \right) P y - Q_{RN} Q_N^{-1} \left(\left(Q_{NR} H_l^{\,T} + Q_N H_r^{\,T} \right) P y - x^o \right) \\ &= x_{float}^a - Q_{RN} Q_N^{-1} \left(x_{float}^o - x^o \right) \end{aligned}$$

$$Q_{x_{fix}} = Q_R - Q_{RN} Q_N^{-1} Q_{NR}$$