

ali_1でやっていること

1. 区画内のbvxxを取り出し(bt0, bt1)
2. bt0とbt1をフィルタリング
 - 2.1. angle cut, ph cut, vol cut, ghost cut
3. BaseTrackAlign::Exec
 - 3.1. bt0とbt1でリンクレットを作る ConnectPlate
 - 3.2. 2Dヒストグラムに入れる Fill()
 - 3.3. Peakを見つける PeakDetection()
 - 3.4. Peakのセンター、サイズを決める PeakValidation()
 - 3.5. CalcAffineForAngle()
 - 3.6. CalcAffine()
 - 3.7. CalcGap()
 - 3.8. もう一回 CalcAffine()

rc

使うパラメーター

```
[LocalAlign]
AngleCut      = 0 -0.40 0.40 -0.40 0.40
Enable_Affine = 1
Enable_GapTune    = 1
SearchArea  = 50.0          # +/-SearchAreaの範囲内のピークを探す。
SearchAngle  = 0.020
Significance  = 5.0
PHCUT        = 1 0.1 18
PHSUMCUT     = 0 0.1 0
VolCut       = 1 0.1 3
GhostFilter   = 1.0 0.007
BinWidth     = 6.0 0.015
PeakSizeMin  = 1.0 0.000    # minimum peak size ( $1+$2*gap ) to be searched
PeakSizeMax  = 50.0        # maximum peak size to be serarched
Gap          = 0.01 50.0 1.0      # 探索範囲 +/- ( 50.0 + 0.01xnominal-dz ) を 1.0micron ステップ
IsVerbose    = 0
AffineFitMode= 1    # 0 = full affine fit / 1 = shrink+rotation+shift
Zproj        = 0.5  # 最近接ベース面間距離を Zproj : 1-Zproj に内分する z で alignment 探索を行う。default = 0.5
```

```
//linkletを作るとき
ErrPos = SearchArea + PeakSizeMax
ErrAng = SearchAngle
ErrDist = 0. //使っていない
ErrShur = 0. //使っていない

GapPitch = Gap[2]
GapMin = -( Gap[1]+ fabs(dz)*Gap[0] ); //dzはz0 とz1 の差
GapMax = ( Gap[1] + fabs(dz)*Gap[0] );
BinWidth = sqrt( BinWidth[0]^2 + (BinWidth[1]*dz*0.5)^2 )
PeakSizeMin = sqrt( PeakSizeMin[0]^2 + (PeakSizeMin[1]*dz*0.5)^2 );
```

内部で使う変数

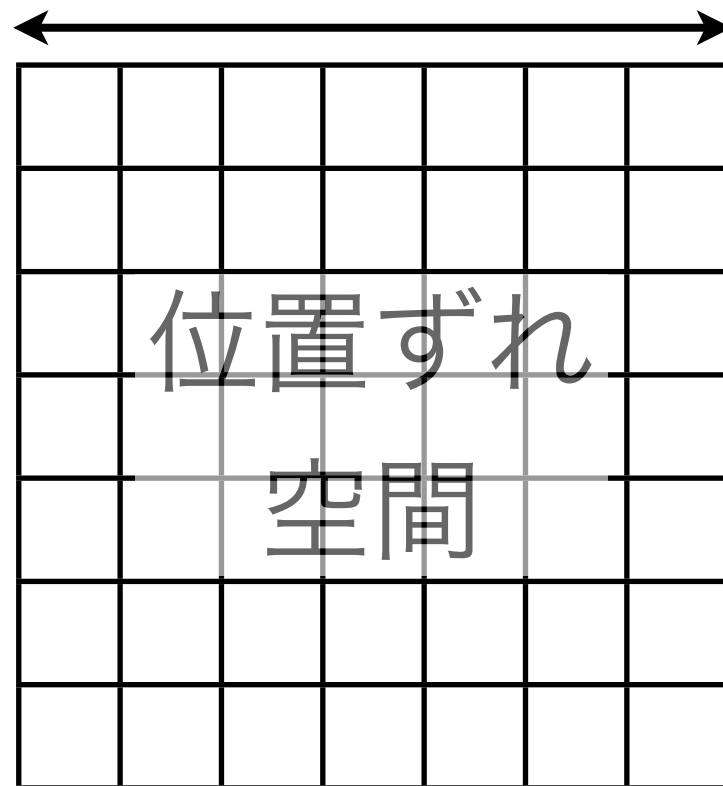
3.1 bt0とbt1でリンクレットを作る

- bt1を接続する初期のアライメントで補正。
- bt0, bt1を接続z面まで外挿
- PhSumでカット
- 角度ずれ つなぎwindow
 - $\text{fabs}(ax1 - ax0) < \text{ErrAng} + \text{fabs}(ax) * \text{ErrShur} + \text{ErrDist}$
 - (ayも)
- 位置ずれ つなぎwindow
 - $\text{fabs}(x1 - x0) < \text{ErrPos} + \text{PeakSizeMax} + \text{fabs}(ax) * \text{ErrGap}$
 - (yも)
 - $\text{ErrGap} = 0.0$ //ErrGapを代入する記述が無い。たぶんdefault値

3.2 2Dヒストグラムに入れる

- Nbin x Nbinのヒストグラムを用意
 - $\text{Nbin} = (\text{int})((\text{SearchArea} + \text{PeakSizeMax}) * 2. / \text{BinWidth}) + 1$
- linkletを全部詰める

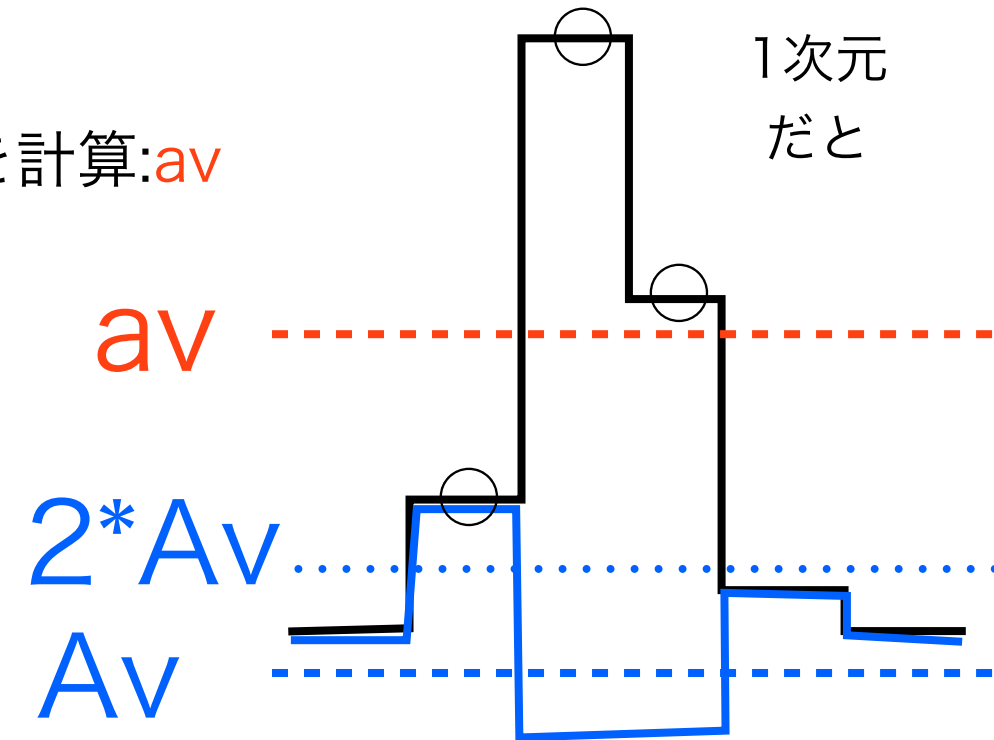
$\pm (\text{SearchArea} + \text{PeakSizeMax})$



Nbin個

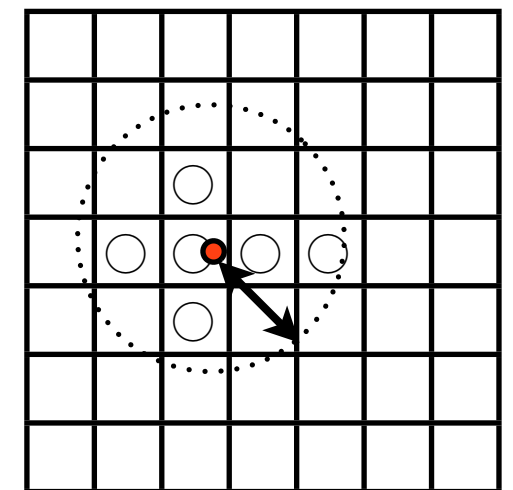
3.3 Peakを見つける

- 2Dヒストグラム(Nbin x Nbin)の全binの高さ平均を計算: av
- 2Dヒストグラムの av より低いbinの高さ平均: Av
- 2Dヒストグラムの $2*Av$ より高いBinの中身を登録



- Peakのセンターを仮決める
 - 登録した他のbinセンターを見て、ずれが半径PeakSizeMaxよりも小さければ、そのbin内のエントリーで加重平均し、Peakのセンター座標を再計算。
 - 登録したBinを全部checkするまでPeakのセンターを再計算する。

半径=PeakSizeMax



- boolMultiPeak==1だと、登録が残っている場合、multiPeakを探す。(デフォルトはboolMultiPeak=0)

3.4 Peakのセンター、サイズ(幅)を決める

補正後のlinkletデータは
ba_???_pos0_pos1.sumとして出力出来る

- PeakFilter() : Peakのサイズを仮決める。
 - linkletの位置ずれ空間で、peakセンターから $\pm 2 * \text{PeakSizeMax}$ のデータを保持。
 - PeakSizeMaxからPeakSizeMinまで $(\text{PeakSizeMax} - \text{PeakSizeMin}) / 10$ のステップで10段階(ただしstepの最小は0.1)でsigmaが最大になるPeakのサイズを探す。
 - peakセンターからPeakSizeの半径以内をシグナル
 - その外側から半径の2倍以内をバックグラウンド
 - $\text{background} = \text{バックグラウンド数} \div 3$
 - $\text{signal} = \text{シグナル数} - \text{background}$
 - $\text{sigma} = \text{signal} / \sqrt{\text{background}}$
- PeakRefine() : Peakのセンター、サイズを計算し直す。
 - peakの中のlinkletデータで今のセンター座標、サイズを使ってをセンターを計算しなおす。x2回やる。
 - 今のセンター、サイズを使ってRMSxとRMSyを計算する。
 - linkletの位置ずれ空間で、peakセンターからサイズ内のデータを保持。
 - 現状のsize から PeakSizeMinまで $(\text{現状のsize} - \text{PeakSizeMin}) / 5$ のステップで5段階(ただしstepの最小は0.1)でsigmaが最大になるPeakのサイズを探す。
 - peakセンターからPeakSizeの半径以内をシグナル
 - その外側から半径の2倍以内をバックグラウンド
 - $\text{background} = \text{バックグラウンド数} \div 3$
 - $\text{signal} = \text{シグナル数} - \text{background}$
 - $\text{sigma} = \text{signal} / \sqrt{\text{background}}$
 - rmsやらsignificanceやらを再計算

3.5 CalcAffineForAngle()

- linkletデータのうち、Peakのセンターから半径size以内の集団をつかう。
- 角度ずれセンター計算、RMS計算 ①
 - 角度ずれ $dax^2 + day^2 < SeachAngle$ だったらセンター、RMS計算に入れる。
- 角度ずれセンター計算、RMS計算 ②
 - 角度ずれ $dax^2 + day^2 < RMS$ だったらセンター、RMS計算に入れる。
 - これを3回やる。
- 角度ずれセンター計算、RMS計算 ③
 - 角度ずれ $dax^2 + day^2 < RMS * 1.5$ だったらセンター、RMS計算に入れる。
- センター(ShiftAX, ShiftAY)を採用。

3.6 CalcAffine

- linkletデータのうち、Peakのセンターから半径size以内の集団をつかう。
- fittingしてabcdpqをだす。(中身は未確認)
- PeakRefine()してsignificanceを再計算
 - もとのよりもsignificanceが良くなれば、置き換えて、そのabcdpqを採用。 return 1
 - もとのpeakよりもsignificanceが良くならなければ、return 0 終了
- return 0にならないければ最大3回繰り返す。

3.7 CalcGap

- linkletデータのうち、Peakのセンターから半径size以内の集団をつかう。
- GapMinからGapMaxまでGapPitchのステップでdzを変化させる。
- PeakRefine()してsignificanceを計算。
- 一番良くなったdzを採用。