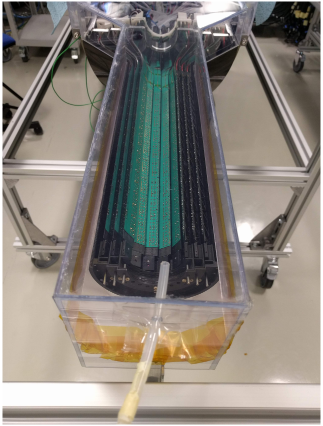
Cosmics

Taejun Kim

5^{th} Jan, 2020

ALICE

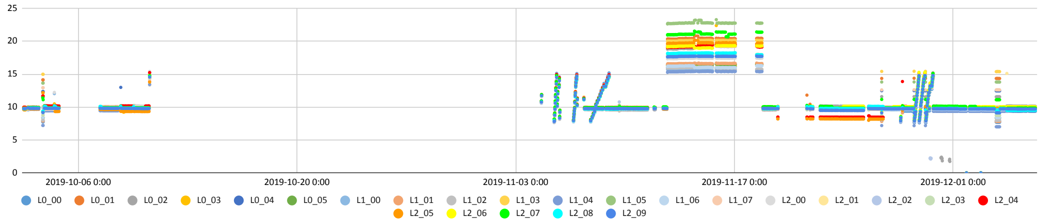
Inner Barrel Top and Bottom

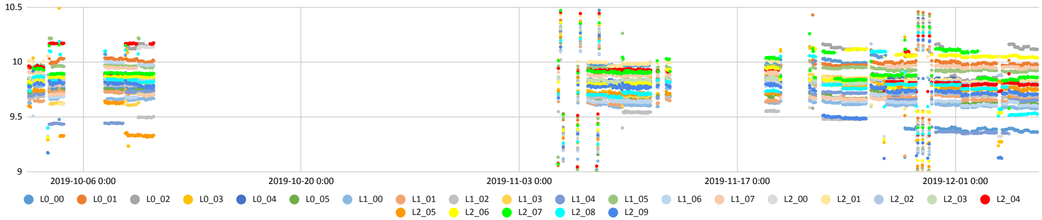


IB-TOP
On data taking. Since 3 layers are synchronized, one can reconstruct track.

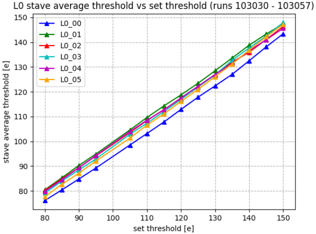
IB-BOTTOM
A Few data taking runs were done. But layers are not synchronized yet.

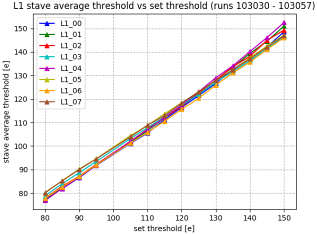
MEAN THRESHOLD

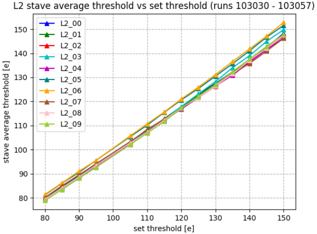




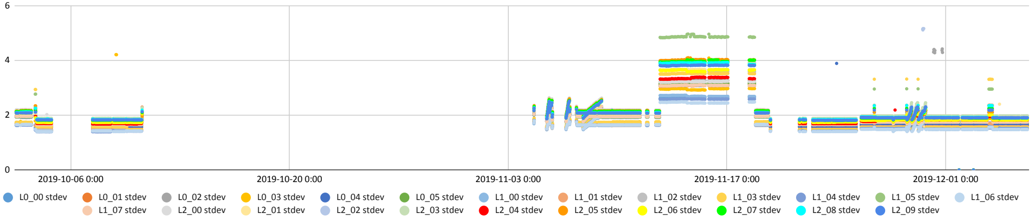
THRESHOLD SET AND MEASURED VALUE

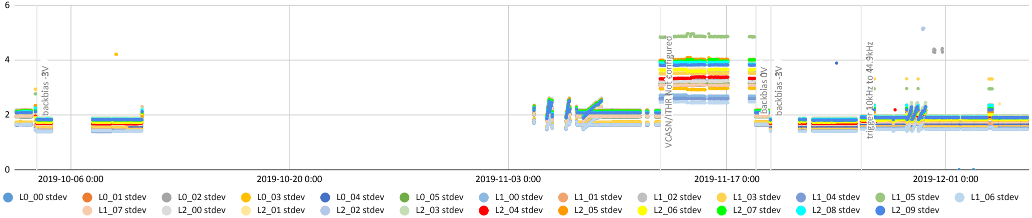




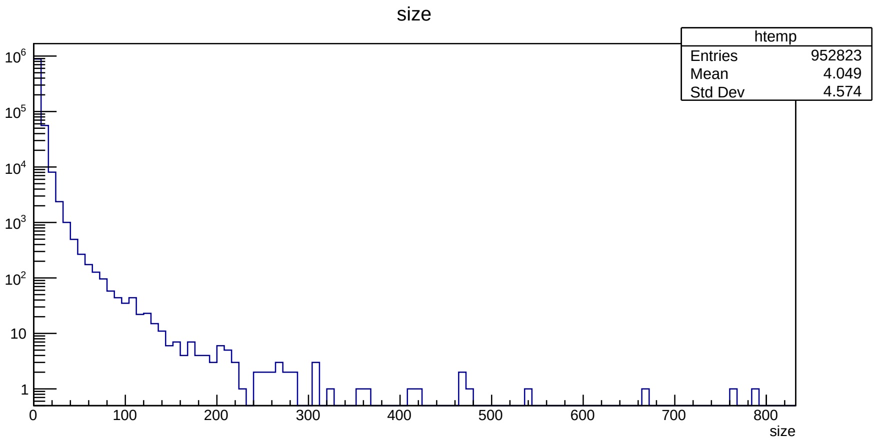


THRESHOLD STANDARD DEVIATION





From FakeHitRate data, we search clusters and tracks.

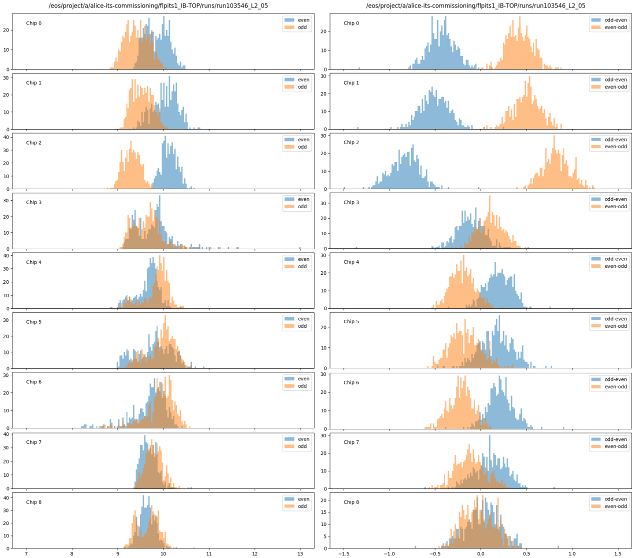


Size of cluster which constitute tracks.

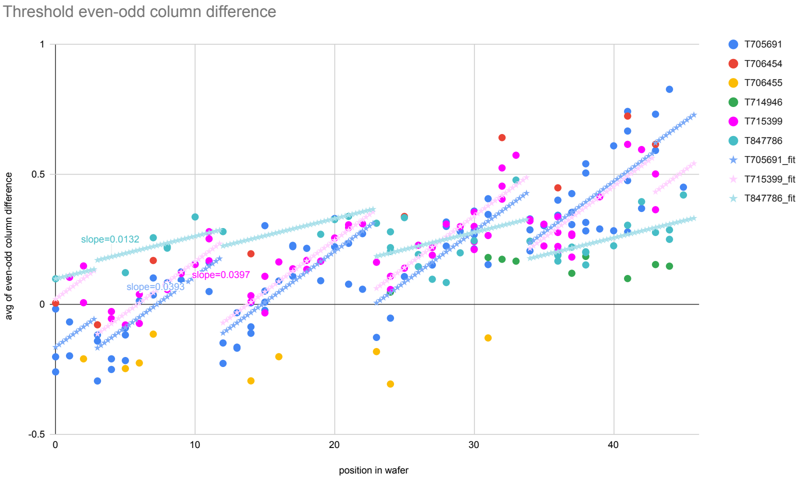
- Alignment of ITS from cosmic data.
- Accumulating IB-TOP cosmic data
- Trigger synchronization of IB-BOTTOM
- Monitoring thresholds
- Optimizing trigger and timing condition.
- Inspection of firmware.

Backup

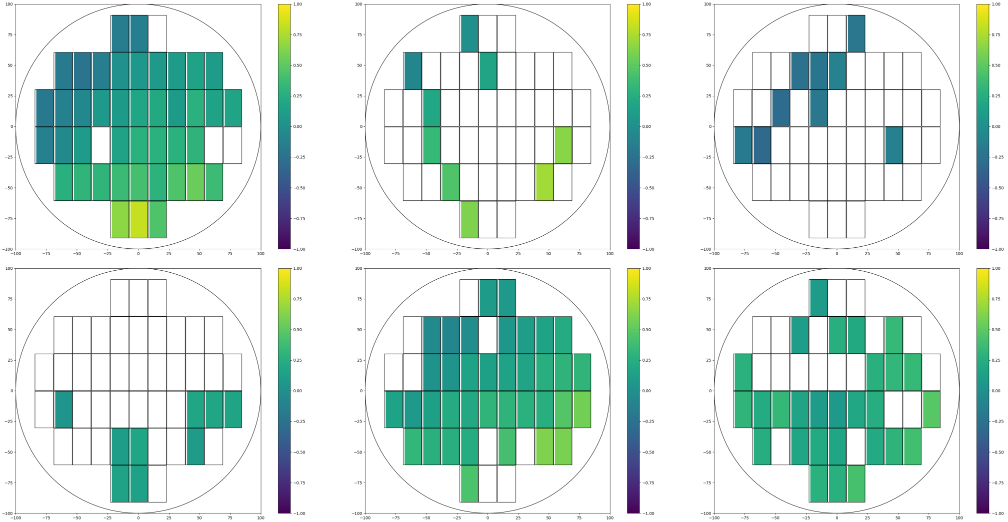
EVEN-ODD COLUMN THRESHOLD DIFFERENCE



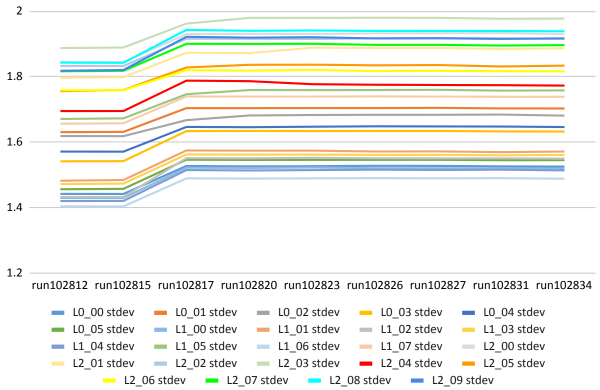
EVEN-ODD COLUMN THRESHOLD DIFFERENCE



EVEN-ODD COLUMN THRESHOLD DIFFERENCE

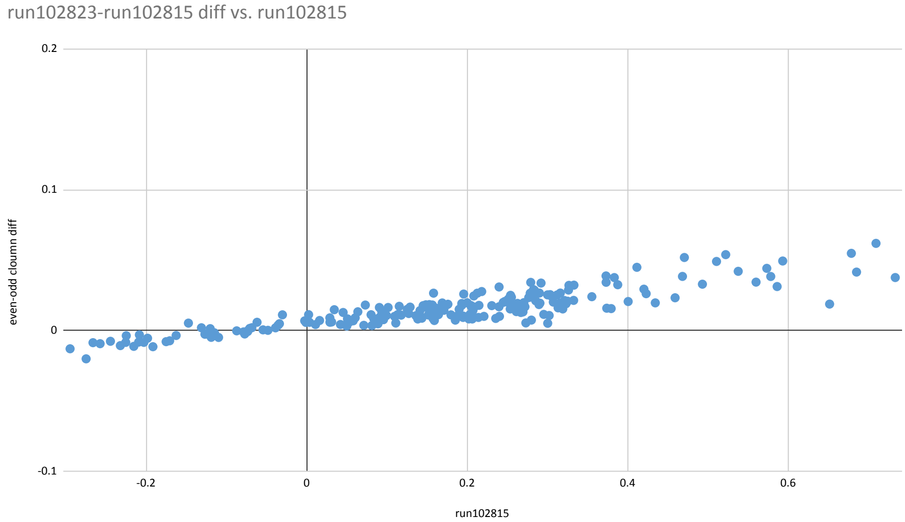


TRIGGER FREQUENCY CHANGE FROM 10KHz TO 44.9KHz



Standard deviation is increased when frequency is increased.

EVEN-ODD DIFFERENCE AT TRIGGER FREQUENCY CHANGE



We update track parameters.

$$X' = X'(X) = X'(\mu_X) + \left.\frac{\partial X'}{\partial X}\right|_{X=\mu_X} (X - \mu_X) = \mu_{X'} + F \cdot (X - \mu_X)$$

We update covriance matrix of track parameters,

$$C' = \mathrm{Cov}[X', X'] = \mathrm{E}\left[(X' - \mu_{X'})(X' - \mu_{X'})^T\right] = \mathrm{E}\left[F \cdot (X - \mu_X)(X - \mu_X)^T \cdot F^T\right] = FCF^T$$

Track parameters are updated in following way where $\Delta = x_{new} - x_{current}$,
ϕ =azimuthal angle and θ =altitude angle

$$y' = y + \Delta \tan\phi + \frac{\Delta^2}{2R\cos^3\phi}$$

$$z' = z + \Delta \tan\theta \sec\phi + \frac{\Delta^2 \tan\theta \sin\phi}{2R\cos^3\phi}$$

$$(\sin\phi)' = \sin\phi + \frac{\Delta}{R}$$

$$\begin{pmatrix} dy' \\ dz' \\ d(\sin\theta)' \\ d(\tan\theta)' \\ d(1/p_t)' \end{pmatrix} = \begin{pmatrix} 1 & 0 & \frac{\Delta}{\cos^3\phi} & 0 & \frac{p_t\Delta^2}{2R\cos^3\phi} \\ '' & 1 & \frac{\Delta\tan\theta\sin\phi}{\cos^3\phi} & \frac{\Delta}{\cos\phi} & \frac{p_t\Delta^2\tan\theta\sin\phi}{2R\cos^3\phi} \\ '' & '' & 1 & 0 & \frac{p_t\Delta}{R} \\ '' & '' & '' & 1 & 0 \\ '' & '' & '' & '' & 1 \end{pmatrix} \begin{pmatrix} dy \\ dz \\ d(\sin\theta) \\ d(\tan\theta) \\ d(1/p_t) \end{pmatrix}$$

$$F = \begin{pmatrix} 1 & 0 & \frac{\Delta}{\cos^3\phi} & 0 & \frac{p_t\Delta^2}{2R\cos^3\phi} \\ '' & 1 & \frac{\Delta\tan\theta\sin\phi}{\cos^3\phi} & \frac{\Delta}{\cos\phi} & \frac{p_t\Delta^2\tan\theta\sin\phi}{2R\cos^3\phi} \\ '' & '' & 1 & 0 & \frac{p_t\Delta}{R} \\ '' & '' & '' & 1 & 0 \\ '' & '' & '' & '' & 1 \end{pmatrix}$$

Measurements is parametrized by $M_k = HX_k + \delta_k$, where $H = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$ and $\mathrm{Cov}[\delta_k, \delta_k] = V_k$. At k^{th} cluster, we update track parameters using measurement M_k and its uncertainty V_K.

$$K_k = C_k^{k-1} H^T \left(V_k + H C_k^{k-1} H^T \right)^{-1}$$

$$X_k = X_k^{k-1} + K_k (M_k - H X_k)$$

$$C_k = C_k^{k-1} - K_k H C_k^{k-1}$$

This is how Kalman gain matrix K_k is derived.

$$\chi_+^2 \approx \left(X_k^* - X_k^{k-1}\right)^T \left(C_k^{k-1}\right)^{-1} \left(X_k^* - X_k^{k-1}\right)$$
$$+ \left(M_k - H_k X_k^{k-1} - H_k \left(X_k^* - X_k^{k-1}\right)\right)^T (V_k)^{-1} \left(M_k - H_k X_k^{k-1} - H_k \left(X_k^* - X_k^{k-1}\right)\right)$$

To optimize increment of χ^2,

$$\frac{\partial \chi_+^2}{\partial X_k^*} = 0$$

$$\frac{\partial \left(\boldsymbol{B}\boldsymbol{x} + \boldsymbol{b}\right)^T \boldsymbol{C} \left(\boldsymbol{D}\boldsymbol{x} + \boldsymbol{d}\right)}{\partial \boldsymbol{x}} = \boldsymbol{B}^T \boldsymbol{C} \left(\boldsymbol{D}\boldsymbol{x} + \boldsymbol{d}\right) + \boldsymbol{D}^T \boldsymbol{C}^T \left(\boldsymbol{B}\boldsymbol{x} + \boldsymbol{b}\right)$$

Then,

$$X_k = X_k^{k-1} + \left[\left(C_k^{k-1}\right)^{-1} + H_k^T (V_k)^{-1} H_k\right]^{-1} H_k^T (V_k)^{-1} \left(M_k - H_k X_k^{k-1}\right)$$

Cluster Measurement uncertainty V_k

V_k is pre-defined at "AliITSClusterParam::GetError"