

# Kalman filter based processor for track reconstruction in MarlinTPC

LI Bo

Center for High Energy Physics

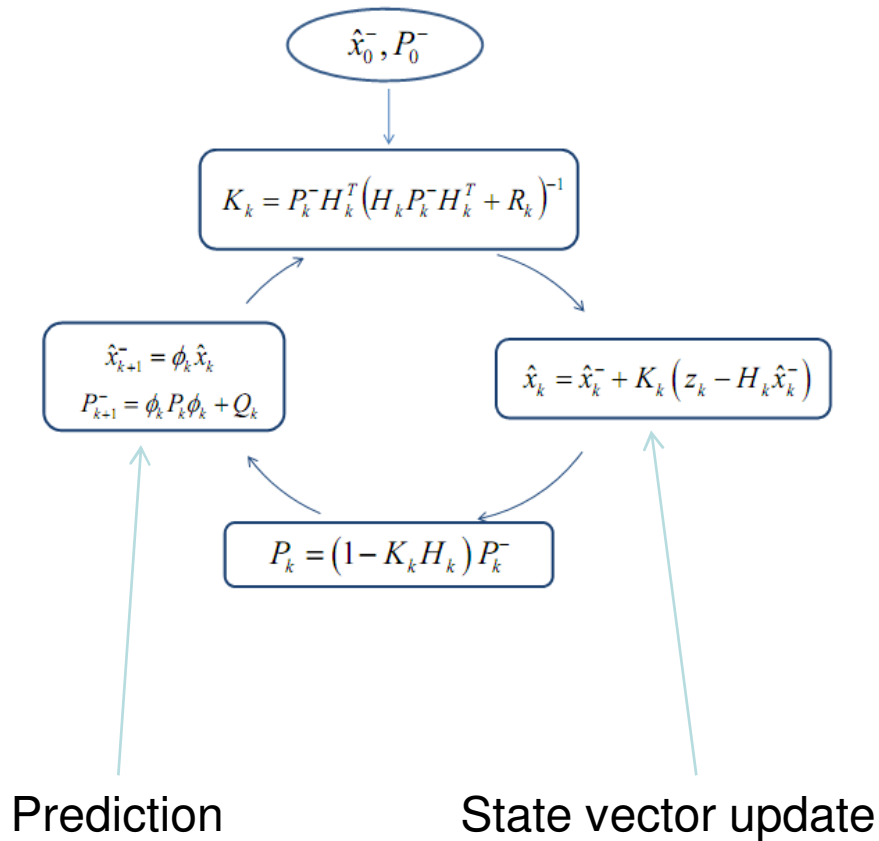
Tsinghua University

# Why Kalman filter

In this new track reconstruction processor, Kalman filter is used. Using this algorithm, we can:

- Easily deal with non-homogeneous magnetic field, multiple scattering, energy loss;
- Carry out track finding and track fitting simultaneously;
- Make computation expense less.

# Kalman filter for track reconstruction algorithm

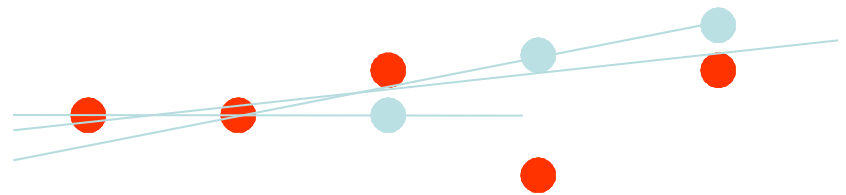


For straight track :  $y = ax + b$

state vector  $X = \begin{bmatrix} a \\ b \end{bmatrix}$

transfer matrix  $\phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

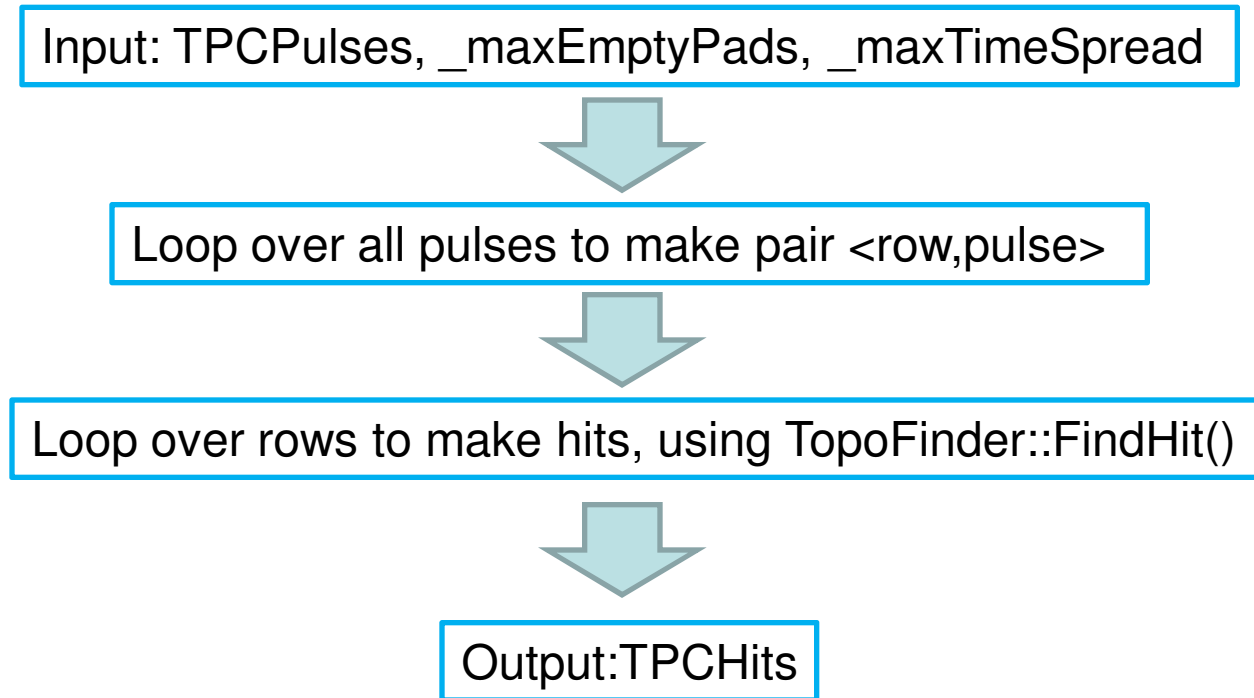
projection matrix  $H = \begin{bmatrix} x & 1 \end{bmatrix}$



Helix track model will be used as the next step. In this talk, I will use straight line model.

# Hit finding

**TemporaryHitMakingProcessor** find hits on each row from pulses.



This processor will be replaced by RowBasedHitFinderProcessor.

# Kalman filter class

**MyKalTrack** is a class which contains Kalman filter algorithm and track hits.

K.F.	{	AddAndFilter();	//calculate new state vector
		GetDeltaChi2();	//get chi2 increment
		GetStateVec();	//get state vector
Hits	{	GetHits();	//get hits collection
		GetNumbetOfHits();	//get track hit number

These methods are called by

**TrackMakingKalmanFilterProceesor** in track reconstruction.

# Track reconstruction

TrackMakingKalmanFilterProcessor searches tracks from input hits. This processor is independent of track model, and we can easily replace track model.

Input:TPCHits, \_minTrackHits, \_maxSkipRows



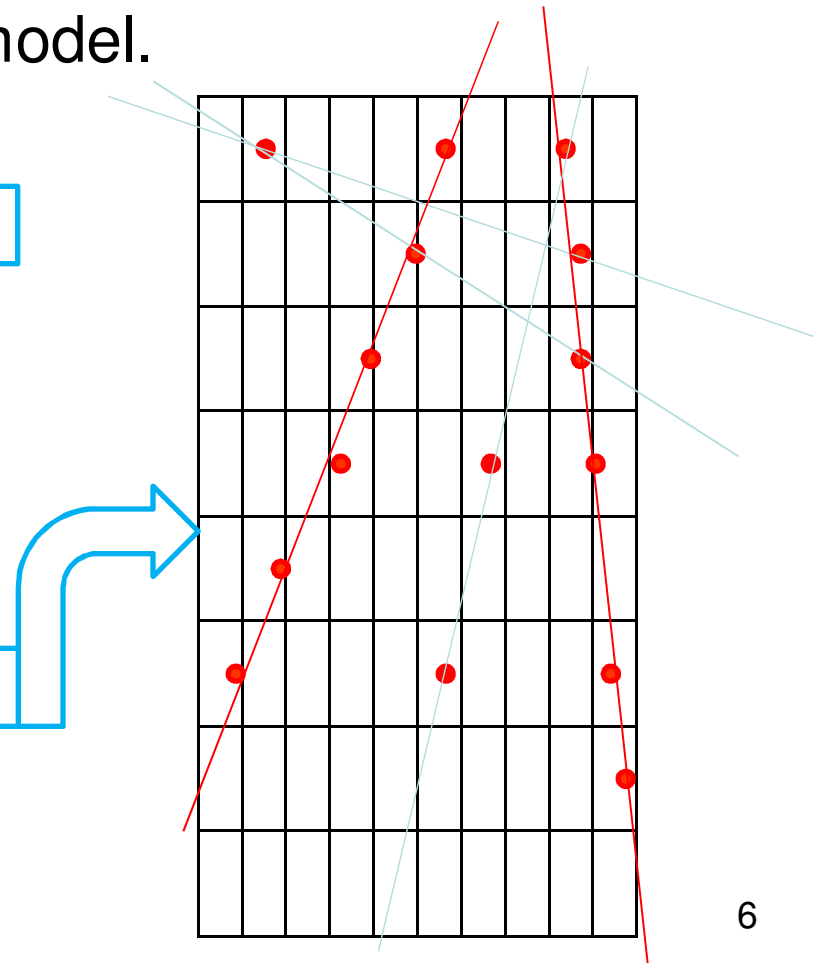
Arrange hits as pair<row, hit>



Loop over rows to search proper hits for a track



Output:TPCTracks

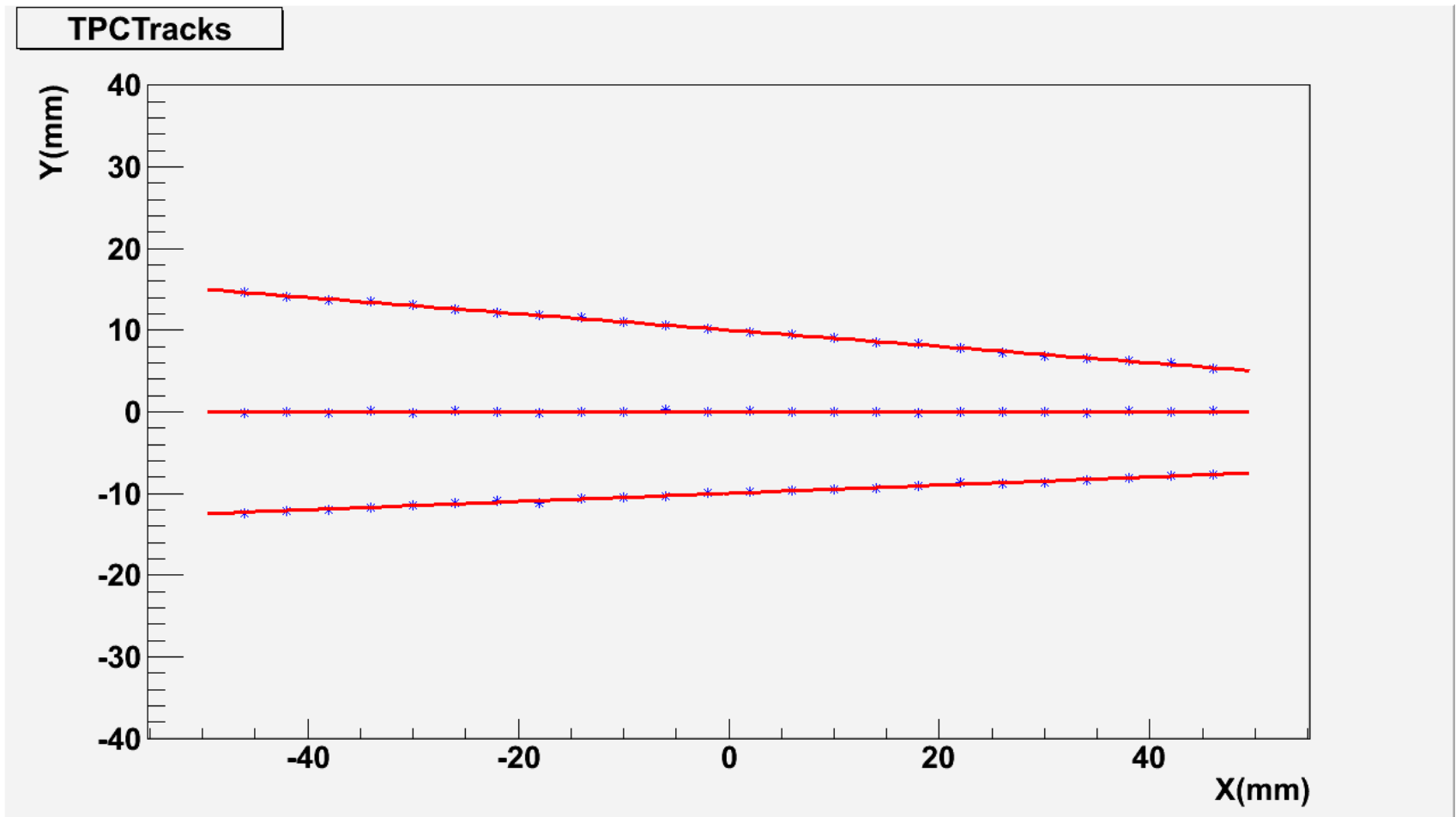


# Steering file

```
<marlin>
<execute>
  <processor name="MyConditionsProcessor"/>
  <processor name="MyTrackerRawDataToDataConverterProcessor"/>
  <processor name="MyADCPulseConverterProcessor"/>
  <processor name="MyChannelMapperProcessor"/>
  <processor name="MyTemporayHitMakingProcessor"/>
  <processor name="MyTrackMakingKalmanFilterProcessor"/>
  <processor name="MyLCIOOutputProcessor"/>
</execute>
...
...
<processor name=" MyTemporayHitMakingProcessor " type=" TemporayHitMakingProcessor ">
  <!--Name of the input collection-->
  <parameter name="InputCollectionName" type="string" lciolnType="TrackerPulse" value="TPCPulses"/>
  <!--Name of the output collection-->
  <parameter name="OutputTrackerHits" type="string" lcioutType="TrackerHit" value="TPCHits"/>
</processor>

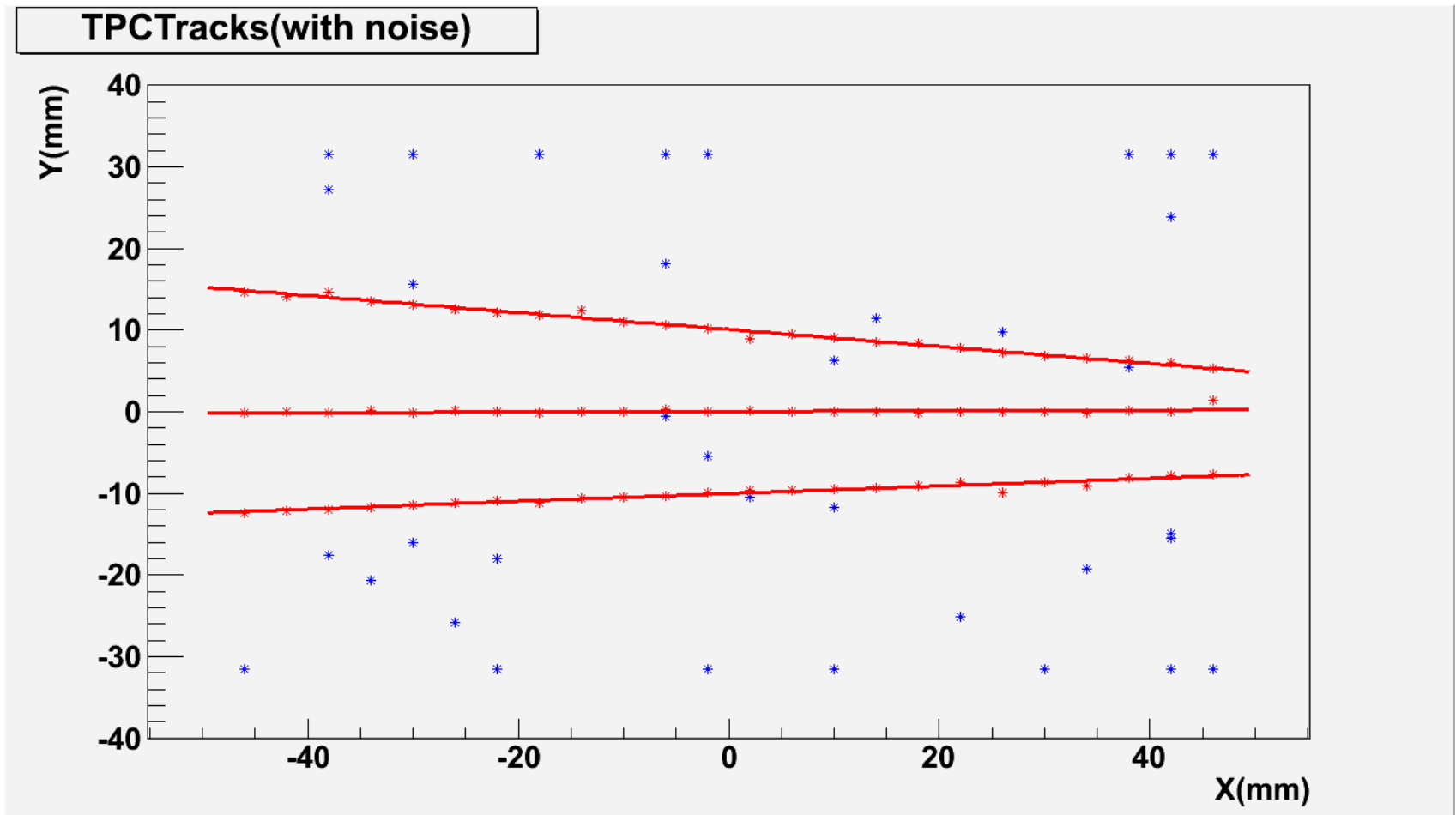
<processor name="MyTrackMakingKalmanFilterProcessor" type="TrackMakingKalmanFilterProcessor">
  <!--Name of the input collection-->
  <parameter name="InputCollectionName" type="string" lciolnType="TrackerHit" value="TPCHits"/>
  <!--Name of the output collection-->
  <parameter name="OutputTracks" type="string" lcioutType="Track" value="TPCTracks"/>
</processor>
...
...
```

# Result for MC data





# Result for MC data with noise



# Summary

- Because of Kalman filter's advantages in track reconstruction, Kalman filter based processor has been created in MarlinTPC;
- Processor is tested for MC Data with/without noise.

# Plan

- First public version will be released on MarlinTPC SVN;
- Use KalTest for MyKalTrack;
- Multi module treatment;
- LP1 Asian GEM data reconstruction check.

Thanks for your attention!