Investigation into Vertex Detector Resolution

N. B. Sinev
University of Oregon, Eugene

Outline of the talk

- About pixsim package
- Is fast version accurate ?
- Is chronopixel fast enough for CLIC ?
- Do we need analog readout ?
- Parameters affecting impact parameters resolution.
- Use of pixsim package with seed tracker.
- Conclusions

Detailed simulation of a pixilated sensor

- To understand effect of different technical solutions on the vertex detector performance as a part of ILC detector, I have developed pixilated sensor simulation package (pixsim). This package is a part of org.lcsim package which is a reconstruction and analysis package for simulation studies for the international linear collider developed by a group of SLAC scientists. It allows detailed simulation of signal formation in the silicon pixel detector and it's processing by front end electronics and reconstruction software.

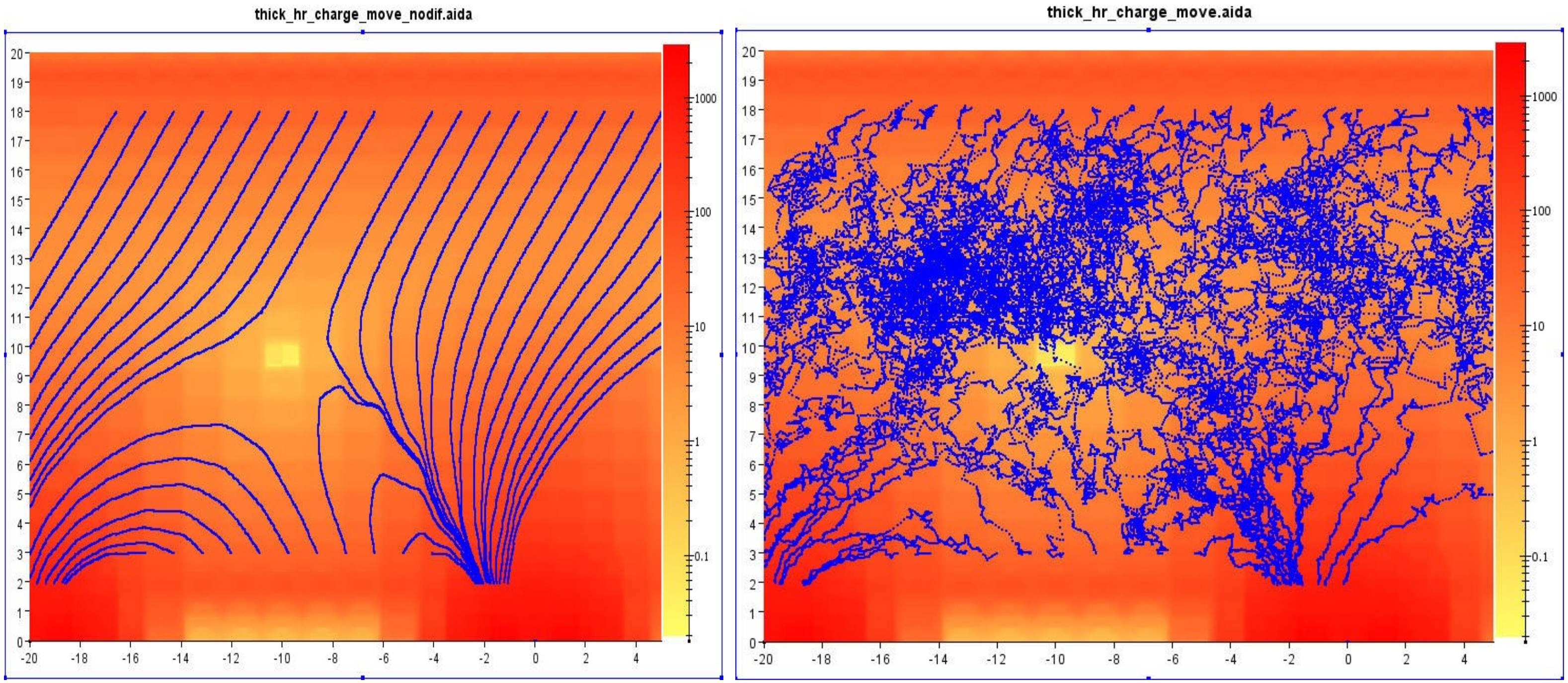


Illustration of charge carrier movement simulation. On the left diffusion was turned off, on the right – real movement, which includes diffusion. Background color shows the strength of electric field. Magnetic field of 5 Tesla is perpendicular to picture plane.

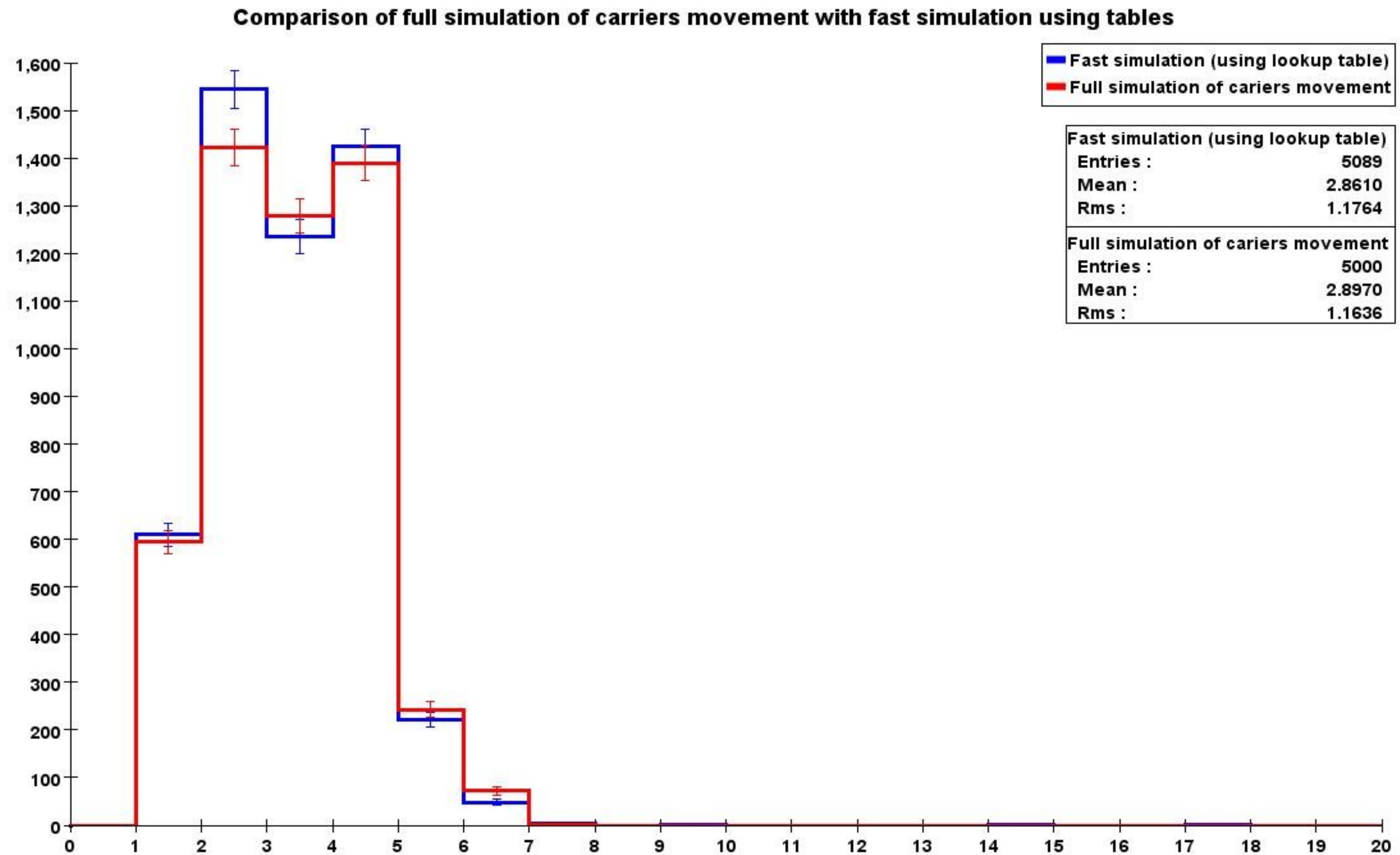
Simulation process

- First we need to define all parameters, affecting charge carriers movement inside sensor. For this we have the class describing the parameters of semiconductor medium used in sensor, and classes, describing electric and magnetic fields. For our sensors, magnetic field can be assumed uniform inside sensor volume, but electric field should be described in details. We are using TCAD simulation to get 3-dimensional map of the electric field.
- For simulating of charge carriers generation by charged particle traversing sensitive volume of the sensor, we are using method, developed by H.Bichsel. It allows calculation of the probability of particular energy loss in the ionizing collision. Average path length between ionizing collisions in the silicon is a fraction of micron, so it is not difficult to simulate every collision, and then simulate the travel of every generated charge carrier to it's absorption.

How to do it faster

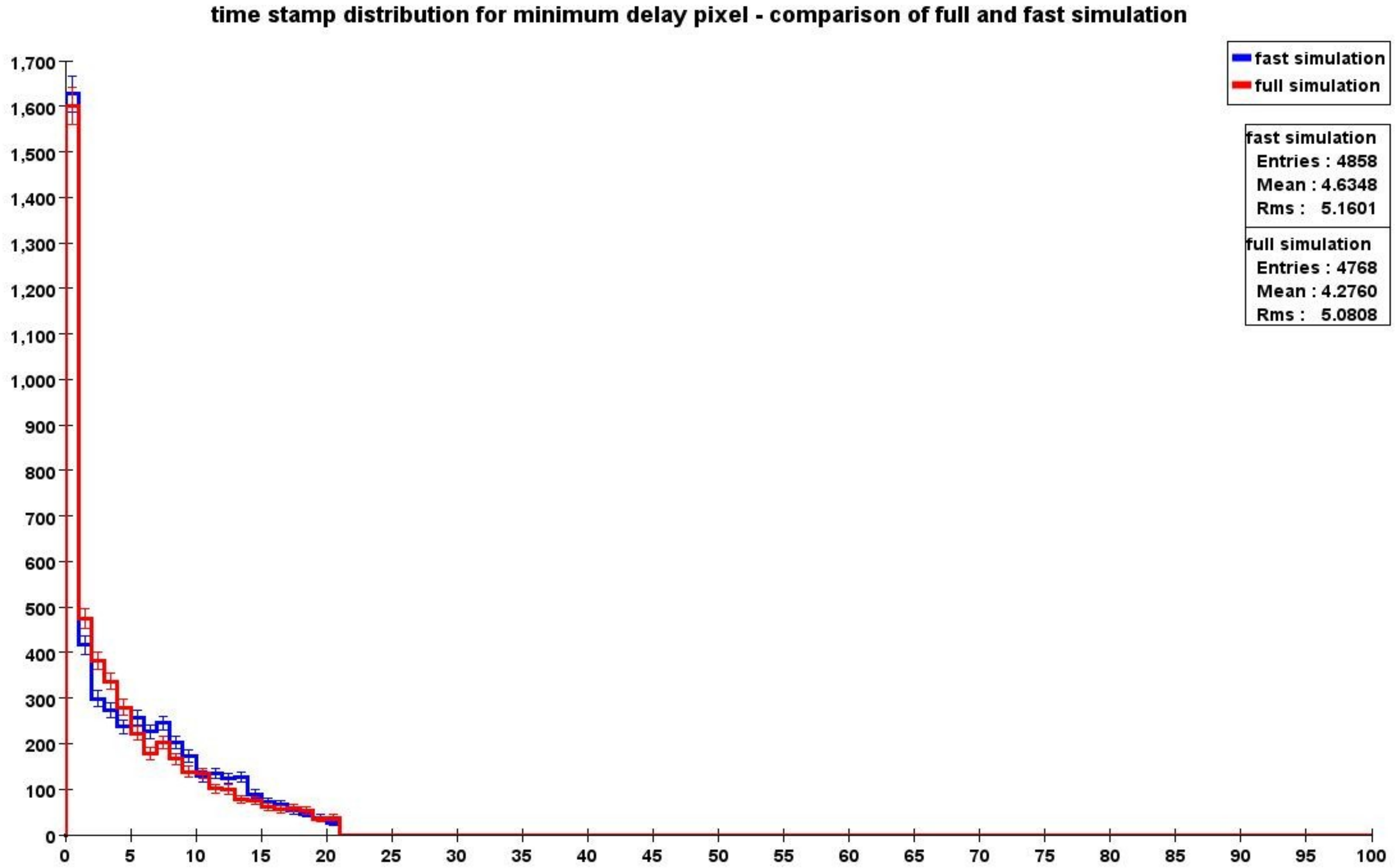
- Described above process is very slow, simulation of one charge particle track takes about 20 seconds.
- To do it faster, we can “pre simulate” fate of charge carrier, generated in the particular point inside sensor. We can find out, what is the probability of such carrier to be collected by charge collection electrode of one of the pixels around that point. And record this probabilities for every point inside pixel into lookup table. More complicated is the process of simulating of time, spent by carriers on that path. Apparently we can’t just use the average time. So, I record 2 parameters of arrival time distribution, and in fast simulation try to generate random numbers having similar distribution. Next slides show how close are results, obtained with full simulation of carrier movement and with simulation using lookup tables. As one could expect, arrival delay of carriers to the peripheral pixels is the most difficult to reproduce using lookup tables, but we don’t care, as we are using only central pixel timestamp to assign cluster to particular bunch crossing.

Comparison of full and fast simulation



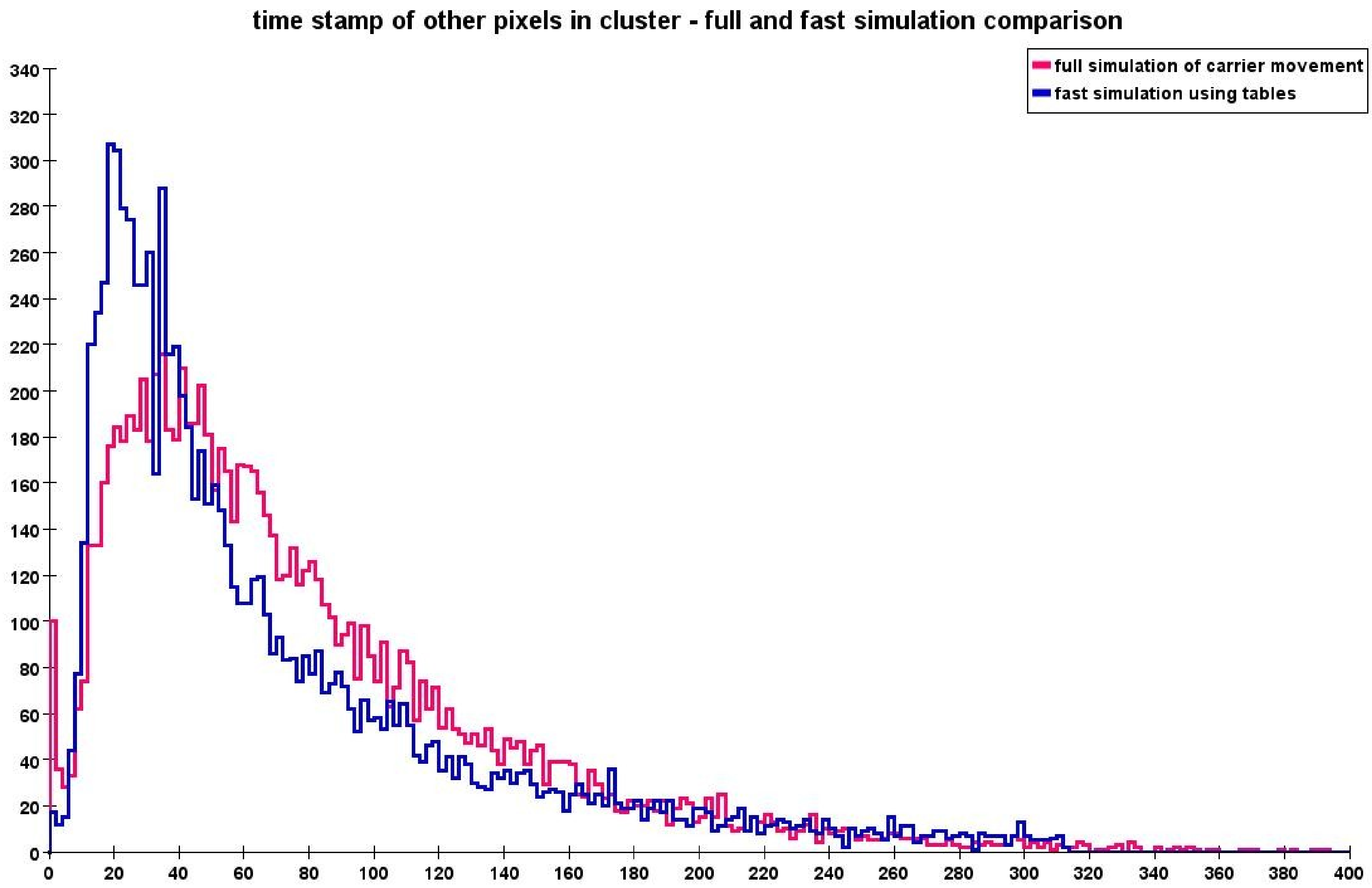
Cluster size (number of pixels in cluster) distribution, obtained with full carriers movement simulation and fast (using tables) simulation.

Comparison of full and fast simulation – 2

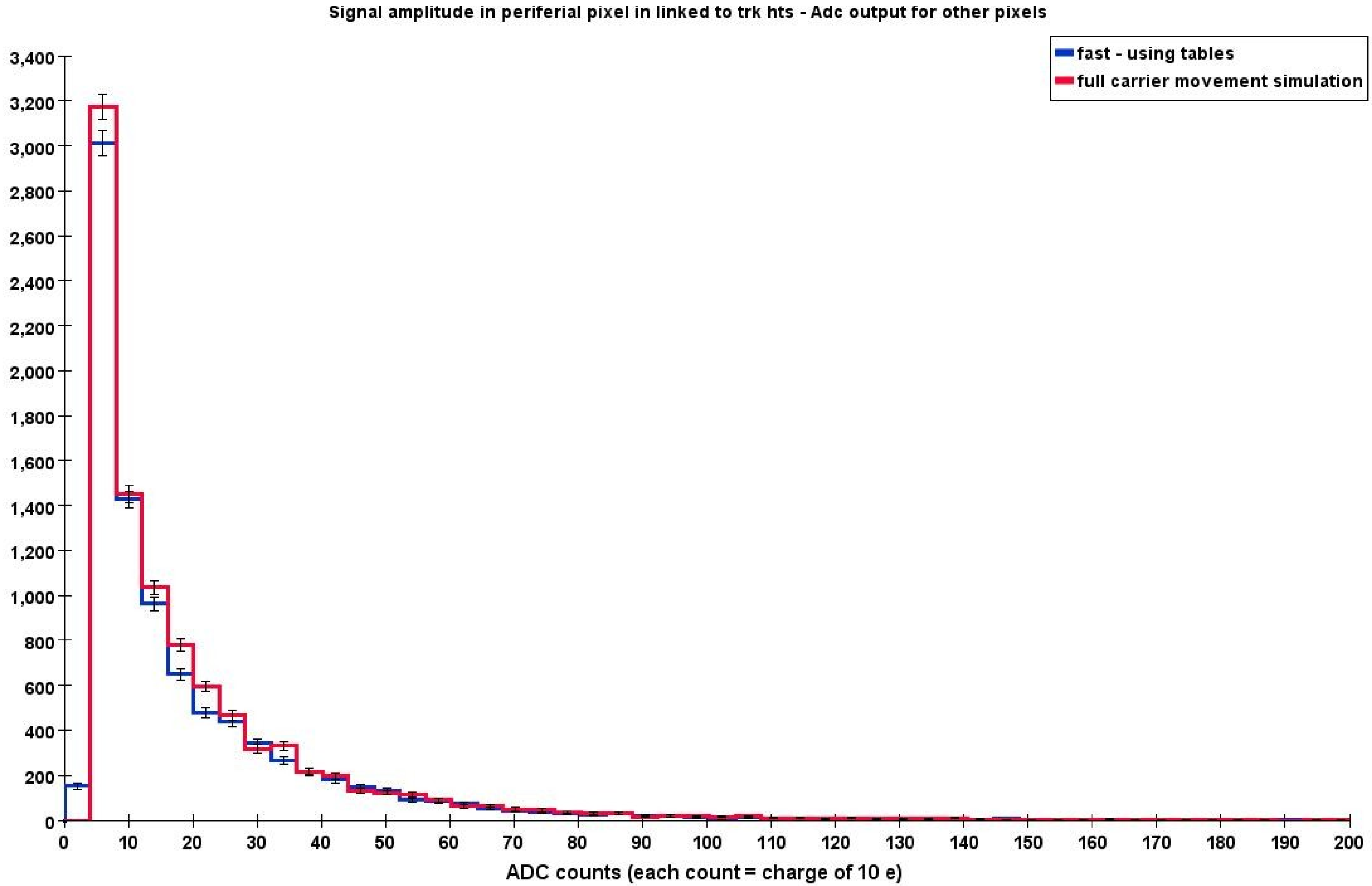


X axis here is in bunch crossing intervals (0.5 ns)

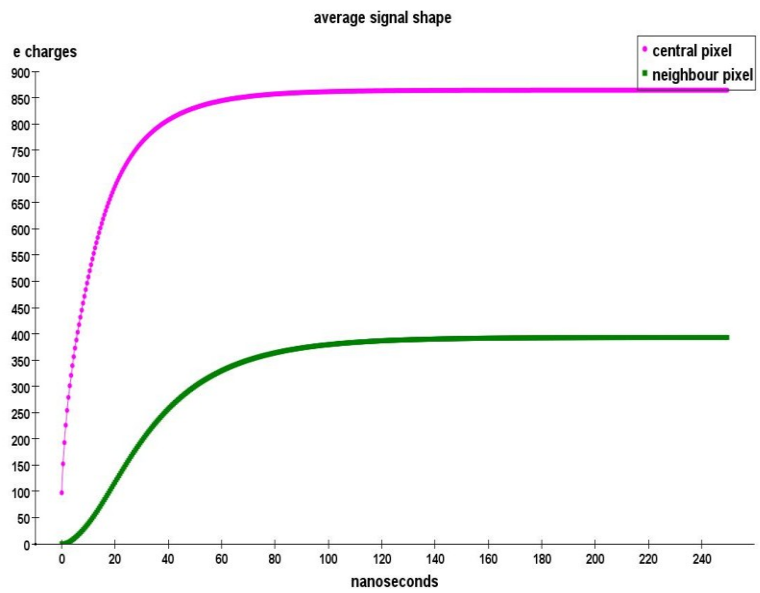
Comparison of full and fast simulation -3

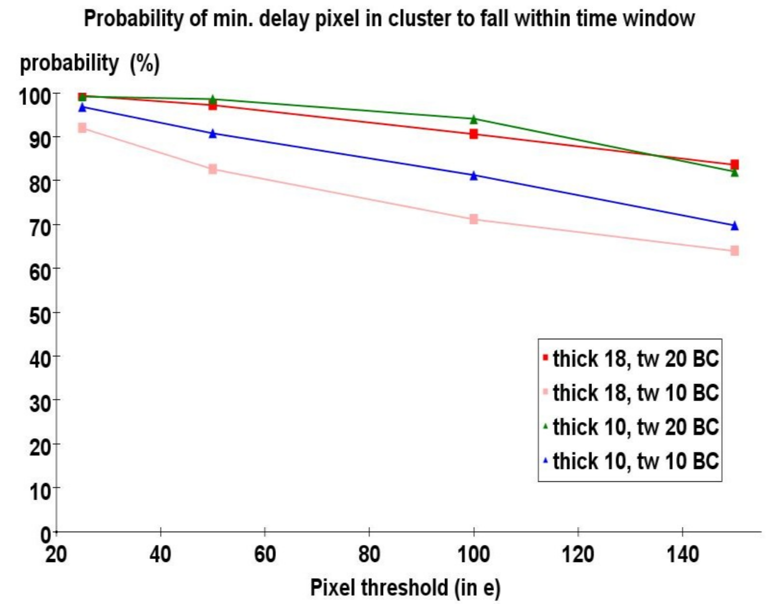


Comparison of full and fast simulation - 4



Is this sensor fast enough for CLIC?

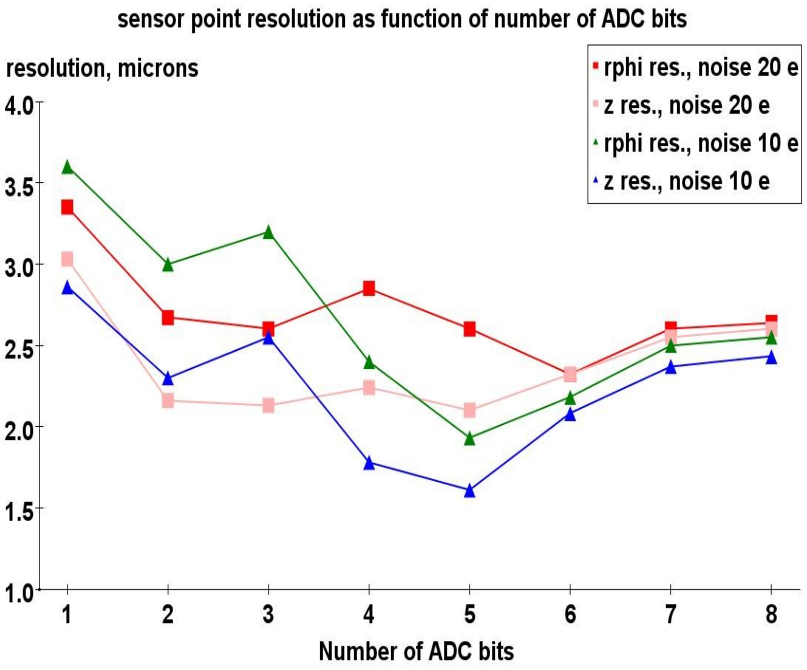




- On the left picture you can see average signal shape in central and peripheral pixels in pixel cluster generated by charged track in 18 μ epi layer.
- On the right plot shown the probability that time stamp of the fastest signal in the cluster falls within time window (expressed in number of CLIC BC intervals) for 18 μ and 10 μ epi layer thickness as function of pixel threshold. Pixel threshold depends on noise level and should be not less than 5 σ of noise for digital readout and 3 σ for analog readout.

Do we need analog readout?

- Plot at the right shows point resolution of pixel detector in perpendicular to B field direction (rphi) and along B field (Z) as function of number of ADC bits. 1 bit means no analog information.
- We can see, that even with very low noise level (10 e) number of bits lager than 5 actually degrade resolution (increased dynamic resolution (increased dynamic range of the signals is the culprit).
- Largest gain in resolution is obtained with transition from 1 to 2 bits.



Impact parameters resolutions

- **Are those noise figures (20e and 10e) realistic?**
  - **We achieved 24e in chronopix with 100 μ^2 collecting electrode area. We can reduce electrode area by factor of 10 easily, and noise scales as squire root of capacitance, which is proportional to area.**
- **Comments to next slide : notice, that resolution in d0 is better than in z0, and d0 resolution in general even better than single point resolution. This is because of an extremely good r-phi resolution of silicon tracker barrels (~7 μ), which provides huge "lever arm" for d0 parameter determination. And this makes it also very little sensitive to the single layer hit loss. Tracker sensor resolution in Z is much worse, because of small stereo angle between strips.**
- **Of course, there is no reason to go to such thin epi layer as 10 μ, but table shows, that if we want to use very narrow time window (like only 10 BC or 5 ns), faster charge collection in thinner epi sensor makes it almost equal to the one with thicker epi.**

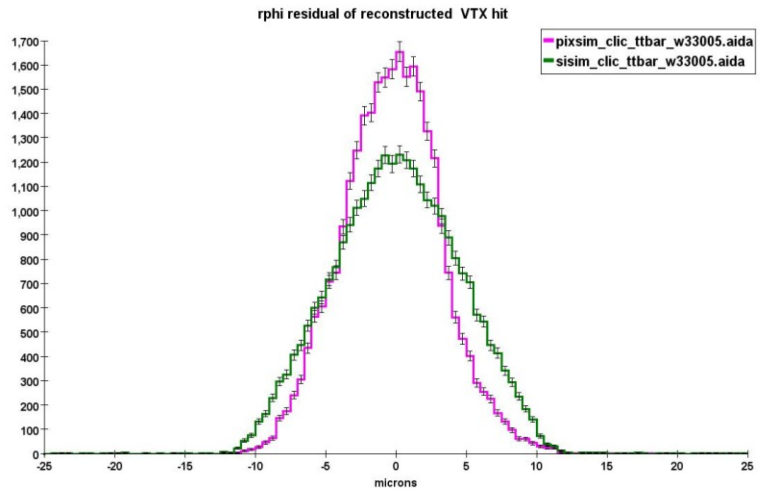
Impact parameters resolution-results

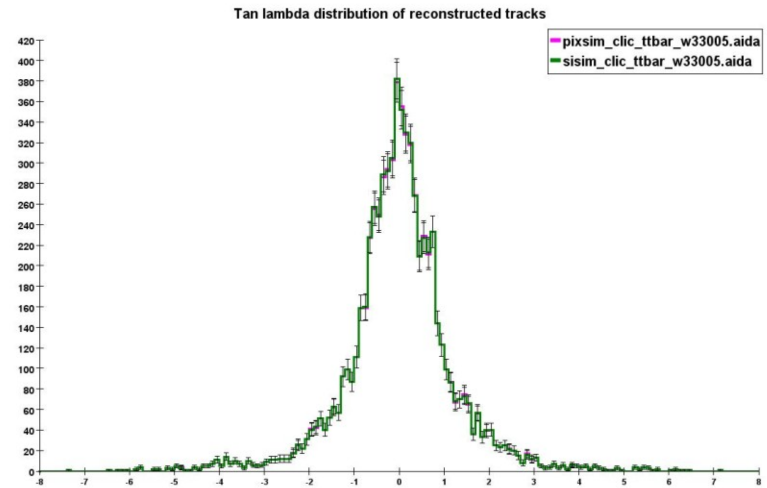
Epi thick.	Noise level	Time window	D0 with 1 bit adc	Z0, 1 bit	D0, 2 bits	Z0, 2 bits	D0, 3 bits	Z0, 3 bits
18 μ	30 e	20 BC	3.03 μ	5.3 μ				
18	20	20	2.84	4.38	2.55	3.04	2.57	2.88
18	20	10	3.22	6.27				
18	10	20	2.86	4.1	2.7	3.2	2.39	2.3
18	10	10	3.21	4.98	2.7	3.3	2.46	2.49
10	20	20	3.21	5.6	2.97	4.6	3.8	7.4
10	20	10	3.32	6.3	2.97	4.74	3.8	7.9
10	10	20	2.93	4.33	2.7	3.4	2.7	3.98
10	10	10	2.95	4.92	2.8	3.6	2.83	4.05

Seed tracking with pixsim

- Detailed instructions about pixsim simulation package can be found at www.slac.stanford.edu/~sinev/pixsim_doc/pixsim_help.html
- The only thing needed to use seed tracking with pixsim simulation instead of sisim simulation (which was a temporary solution) is to put declaration in the header of your reconstruction code (though there is clic_sid in the driver path, this tracking reconstruction code can be used for any detector, not necessarily CLIC.):
  - import org.lcsim.recon.tracking.seedtracker.trackingdrivers.clic_sid.MainTrackingDriver;
  - Instead of: import org.lcsim.recon.tracking.seedtracker.trackingdrivers.sidloi3_digital.MainTrackingDriver

Comparison of two options





- **Figure on the right shows that there is no difference in the track reconstruction efficiency with either option.**
- **Figure on the left shows, that point resolution of the vertex sensors is better with pixsim.**
- **This illustration is obtained with reconstruction of TTbarH events with 1000 GeV CM energy. There was an average of about 70 of reconstructed charge tracks in the event, and pixsim with use of lookup table option added just few seconds to the average reconstruction time about 60 seconds/event**

Conclusions

- Chronopixel sensor (with some modification of time stamping circuit) is capable of the assignment of hits to time window of the order of 10-20 CLIC bunch crossings with efficiency above 90 %.
- To achieve good sensor point resolution (about 3.5 μ) we don't need analog readout.
- Improvement in the point resolution of the order of 20% can be achieved with the use of only 2 bits ADC. If the noise level of the order of 10 e^- can be achieved, 4-5 bits ADC can give additional gain in point resolution, but going to more than 5 bits in the ADC does not help.
- The pixel sensors simulation package is available for use with seed tracking reconstruction package, and use of the lookup tables in pixel simulation gives essentially same results as full simulation of charge carriers movement. Such simulation does not noticeable increase tracking reconstruction time.