Charge sharing in pixelated semiconductor sensors

Ivan Kotov, Jonathan Pellicari, Jiemin Li, Valentina Bisogni
Brookhaven National Laboratory

Presentation at CPAD 2022 Workshop, Nov.30, 2022

Introduction

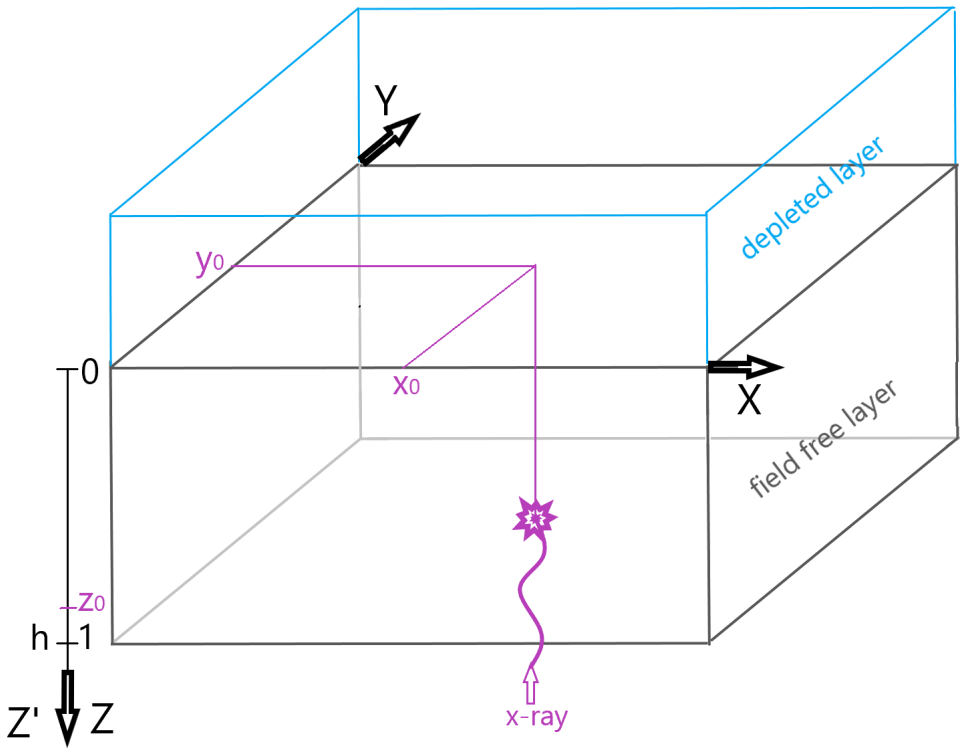
- The accurate model of the **charge distribution shape**
  - ❖ is essential to achieve ultimate coordinate accuracy.
- The charge sharing is caused by charge carriers diffusion on the path from the generation point to pixels.
- The focus is on the diffusion in the **field free region**
- The **diffusion equation solution** is obtained using the separation of variable and Fourier synthesis method
- resulting solution is integrated over pixel areas
  - ❖ We use obtained solution to calculate the fraction of the charge collected in a pixel directly **without intermediate step** of calculating the charge distribution.
- pixel charge **numerical** calculations → numerical function

The problem geometry

the field free layer of a semiconductor with the height **h** and constrained by two parallel planes.

- the entrance window surface at z'=**h**
- the depletion zone beginning above z'=0 plane

It is convenient to use dimensionless variable
x=x'/**h**, y=y'/**h**, z=z'/**h**, sigma = sigma'/**h**, Dt=D't/**h**^2
where x',y',z' are Cartesian coordinates,
t is diffusion time,
sigma is the charge carriers initial spread,
D' is the diffusivity.
The initial interaction point is at (x_0, y_0, z_0) and the generated charge amount is Q_0.



The diffusion equation solution

- The diffusion equation for the carriers charge density ρ(**r**,z,t) created by a particle or an x-ray is

$$D\nabla\rho(\mathbf{r}, z, t) = \partial\rho(\mathbf{r}, z, t) / \partial t \quad (1)$$

where **r** is a vector in XY plane

- The diffusion equation is separable ρ(**r**,z,t)=X(x)·Y(y)·Z(z)·T(t)
- There are no boundaries in XY-plane and ρ(**r**,t) solution can be obtained applying spatial Fourier transform, then solving differential equation for individual modes and applying initial conditions
- The initial conditions we considered are ρ(**r**, z, 0) = Qo δ(**r**−**r**0)δ(z−z0) and Gaussian
- the full solution is

$$\rho(\boldsymbol{r}, z, t) = = \frac{Q_0}{2\pi\sigma^2} \cdot 2 \cdot \sum_{n=0}^{\infty} \sin(\alpha_n z_0) \cdot \sin(\alpha_n z) \cdot \exp\left(-\frac{\alpha_n^2\sigma^2}{2} - \frac{(r-r_0)^2}{2\sigma^2}\right)$$

$$\sigma^2 = 2Dt + \sigma_0^2$$

$$\alpha n = (n + 1/2) \cdot \pi$$

Charge flux and pixel charge

- The charge flux j(r,t)) through the boundary between field free and depletion zones is

$$\mathrm{j(r,t)} = \mathrm{D}\frac{\partial\rho(r,z,t)}{\partial z}|z=0 = D\,\frac{Q_0}{2\pi\sigma^2}\cdot 2\cdot\sum_{n=0}^{\infty}\alpha_n\cdot\sin(\alpha_n z_0)\cdot\exp\left(-\frac{\alpha_n^2\sigma^2}{2}-\frac{(r-r_0)^2}{2\sigma^2}\right) \quad (4)$$

- The quantity of interest is the fraction of the charge collected in a pixel.
  - The charge fraction qij in pixel (i,j), can be obtained integrating flux j(r,t) at the depletion zone boundary over pixel area and over time. The rectangular pixel boarders are at x = [bi, bi+1] and y = [cj , cj+1].

$$q_{ij} = 2\cdot\sum_{n=0}^{\infty}\frac{\cdot\sin(\alpha_n z_0)}{\alpha_n}\int_0^{\infty}\frac{d(\alpha_n^2\sigma^2)}{2}\exp\left(-\frac{\alpha_n^2\sigma^2}{2}\right)\cdot\frac{1}{\pi}\int_{b_i}^{b_{i+1}}\frac{dx}{\sqrt{2}\sigma}\int_{c_j}^{c_{j+1}}\frac{dy}{\sqrt{2}\sigma}\cdot\exp\left(-\frac{(r-r_0)^2}{2\sigma^2}\right) \quad (5)$$

- The spatial integral over pixel ij area is

$$p_{ij}=\frac{1}{4}\cdot\left(\mathrm{erf}\left(\frac{(b_{i+1}-x_0)}{\sqrt{2}\sigma}\right)-\mathrm{erf}\left(\frac{(b_i-x_0)}{\sqrt{2}\sigma}\right)\right)\cdot\left(\mathrm{erf}\left(\frac{(c_{j+1}-y_0)}{\sqrt{2}\sigma}\right)-\mathrm{erf}\left(\frac{(c_j-y_0)}{\sqrt{2}\sigma}\right)\right) \quad (6)$$

- Using substitution $\eta = \exp(-\alpha_n^2\sigma^2/2)$ the remaining integral over time becomes

$$I_{ij}(\alpha_n,x_0,y_0) = \int_0^1\frac{1}{4}\cdot\left(\mathrm{erf}\left(\frac{\alpha_n(b_{i+1}-x_0)}{2\sqrt{-ln\eta}}\right)-\mathrm{erf}\left(\frac{\alpha_n(b_i-x_0)}{2\sqrt{-ln\eta}}\right)\right)\cdot\left(\mathrm{erf}\left(\frac{\alpha_n(c_{j+1}-y_0)}{2\sqrt{-ln\eta}}\right)-\mathrm{erf}\left(\frac{\alpha_n(c_j-y_0)}{2\sqrt{-ln\eta}}\right)\right)\mathrm{d}\eta \quad (7)$$

- And the qij can be expressed as

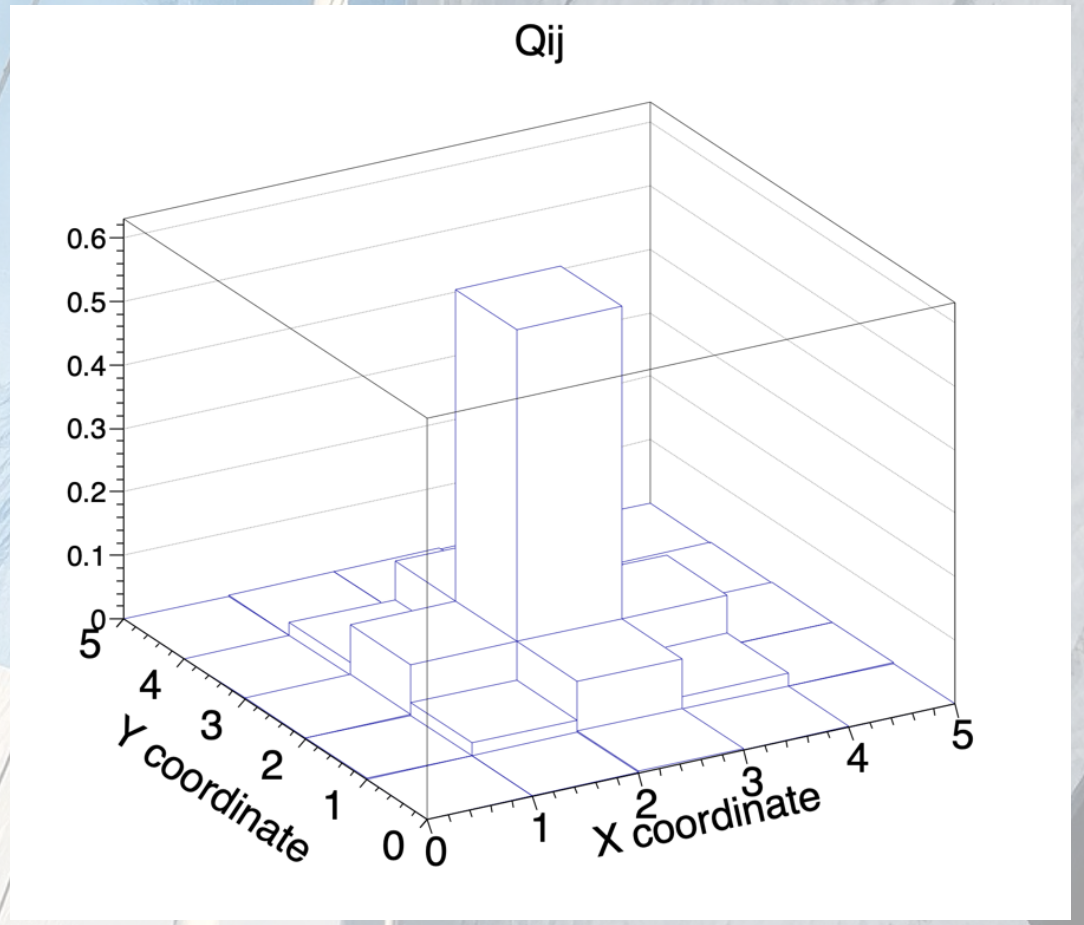
$$q_{ij} = 2\cdot\sum_{n=0}^{\infty}\frac{\sin(\alpha_n z_0)}{\alpha_n}\cdot I_{ij}(\alpha_n,x_0,y_0) \quad (8)$$

Charge spread example

- The charge distribution example calculated using Eq.(8) for x-ray hitting the pixel center and conversion point at the entrance window, z0 = 1, is shown →

- Let's use EMCCD sensors from [1] as the test case.

In these sensors the field free region depth is 8µm,
the depletion zone is 6µm,
pixels are square and the pixel size is 16µm.
The x-ray energy range 0.2 − 2.0keV and
this correspond to 50 − 550e− signals.
The system noise is ~ 1e−.
The computation accuracy should be better than
error due to system noise and to satisfy this requirement
10−3 accuracy is sufficient.

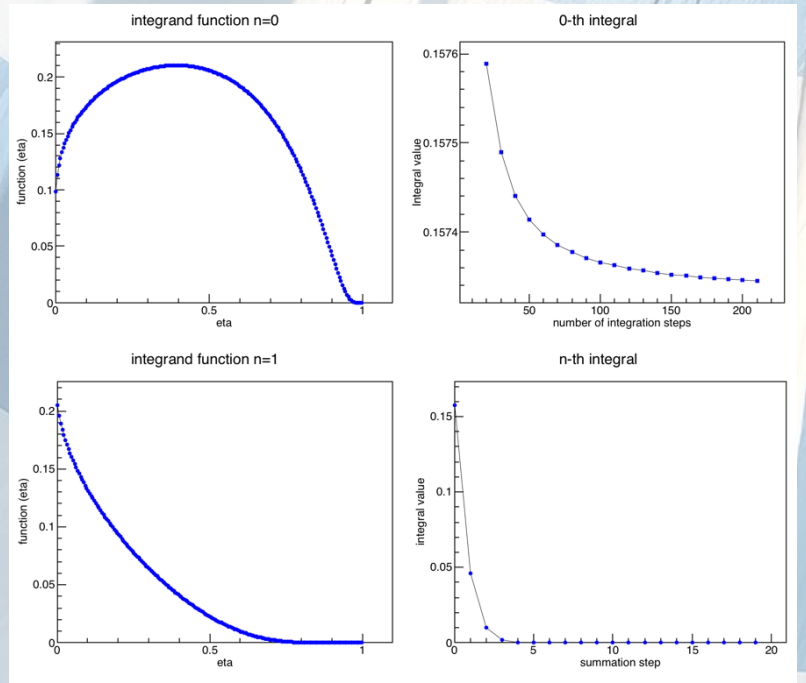


[1] Analysis of the EMCCD point-source response using x-rays
I.V. Kotov, S. Hall, D. Gopinath, A. Barbour, ... V. Bisogni
Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 985, 164706 (2021)

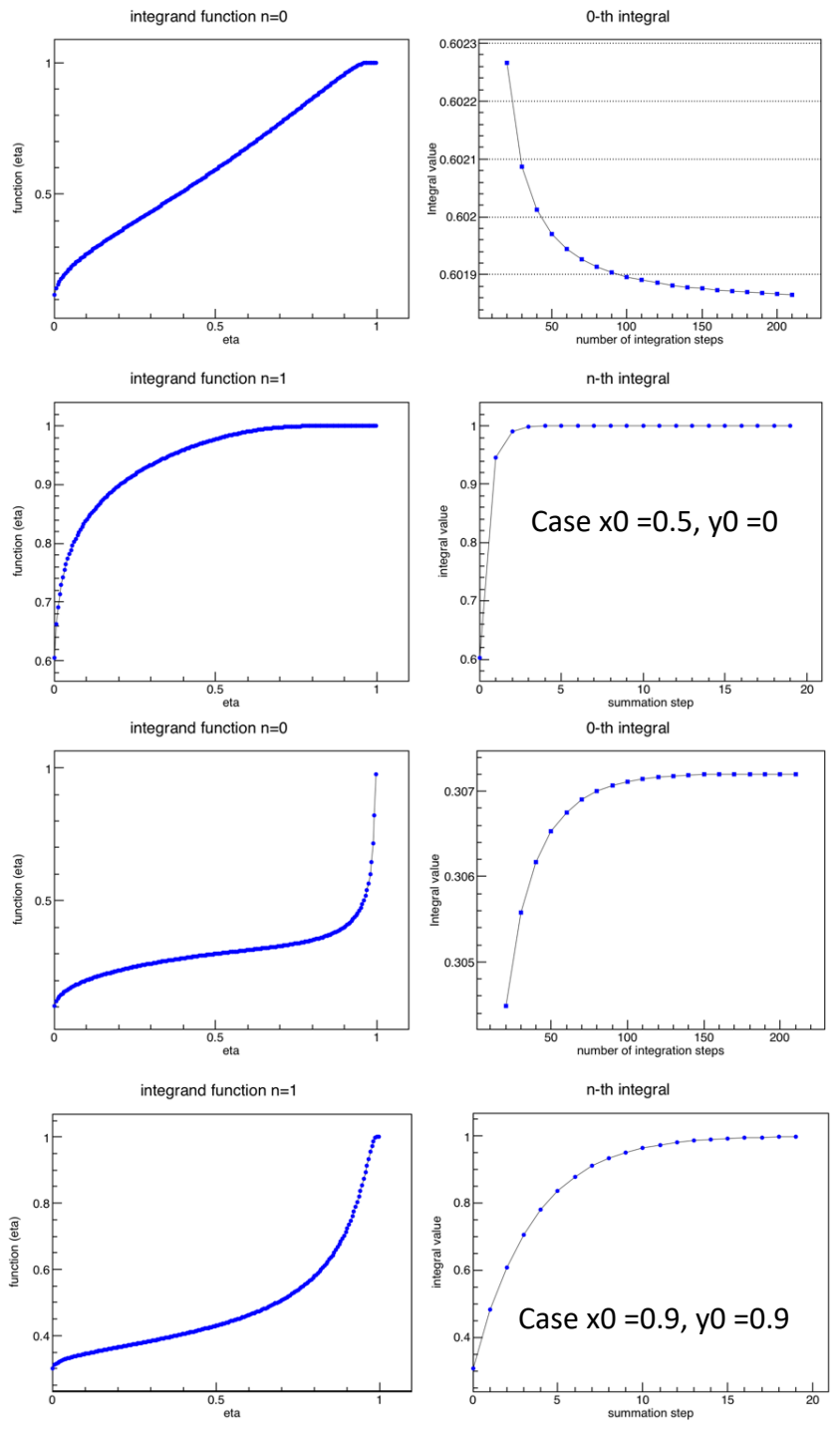
Numerical analysis, integrals

- The integrand in Eq.(7) is the smooth function of η and required accuracy can be achieved with modest number of integration steps.
- Examples of integrand function forms and numerical integration convergence speed are shown.
  - Left panels show integrand functions for α0 and α1.
  - Top right panels show how integral value depends on the number of integration steps.
  - ~ 60 steps are enough to reach accuracy better than 10-3.

$$q_{ij} = 2 \cdot \sum_{n=0}^{\infty} \frac{\sin(\alpha_n z_0)}{\alpha_n} \cdot I_{ij}(\alpha_n, x_0, y_0)$$



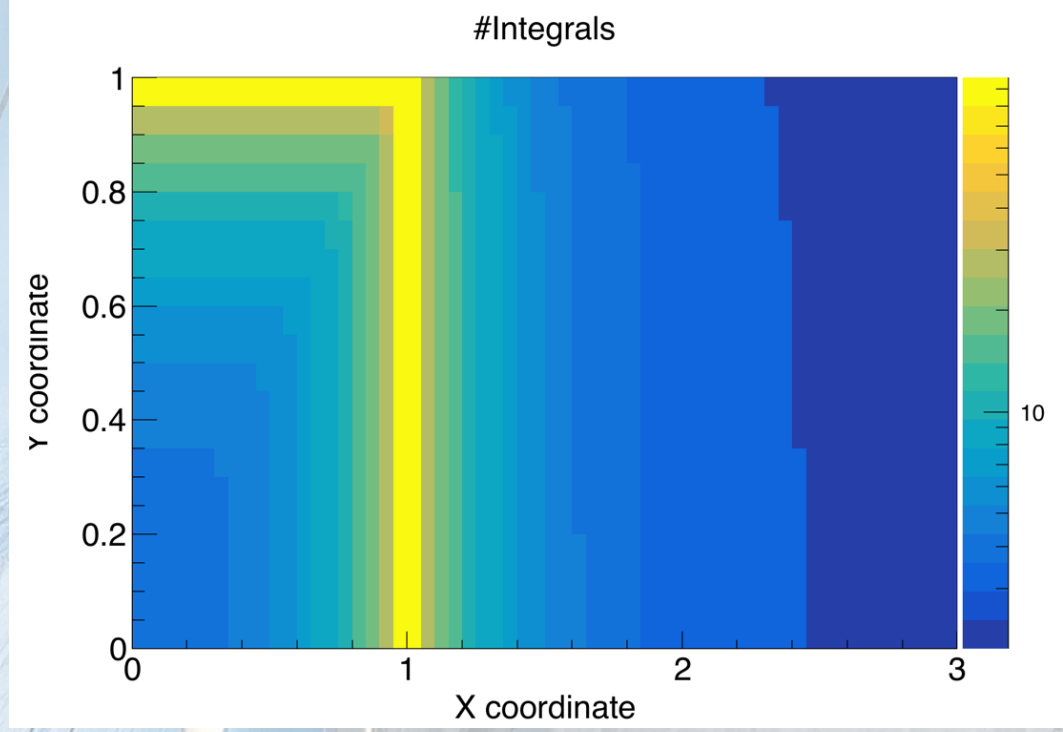
x_0=1.5, y_0=0 case



Numerical analysis, summations

The number of integrals needed to achieve the required accuracy in the pixel charge is shown.
Only few integrals are required when the origin point is in the central area of the pixel.
This number grows with (x0,y0) approaching the pixel boundaries.
The sum $2 \cdot \sum_{n=0}^{\infty} \frac{\sin(\alpha_n z_0)}{\alpha_n}$ = 1 for any z0 > 0.
The z0 parameter is the scaling factor.
The z0 value affect both term max value and decay rate.
The max(sin(az0)/a) = z0 at a = 0.
The sin(αnz0) function oscillate with period N = 2/z0.
Partial sums with summation over the period converge as O(1/n^2).
These numerical calculations even in the optimized form are still time consuming.
They are suitable for cluster shapes generation when only limited amount is required.

$$q_{ij} = 2 \cdot \sum_{n=0}^{\infty} \frac{\sin(\alpha_n z_0)}{\alpha_n} \cdot I_{ij}(\alpha_n, x_0, y_0)$$



Because of the charge distribution central symmetry only one quadrant of the central pixel is shown.
Pixel boundaries are at x = 1,3 and y = 1 coordinates.
The pixel center is at (0,0) and the center of adjacent pixel is at (2, 0).

Numerical function

- The drastic reduction in computation time can be achieved by computing pixel charge fractions ahead of function calls.
  - The look up table approach.
    - At the function initialization, pixel charge fractions for discrete values of (x, y, z) are calculated and the array is filled up.
    - The indexing operation on this array is used instead of runtime computation.
    - this approach trades computing time for memory size. The lookup table is simply used to retrieve values from the memory for a discrete point closest to desired (x,y,z) point. Interpolation between closest discrete points can be performed.
- The array size depends on desired coordinate accuracy.
  - In our test case the required coordinate accuracy is ~ 2µm.
  - To assure this accuracy the pixel surface is covered with 16 × 16 equidistant grid points.
  - the charge distribution central symmetry → only 64 (8 × 8) points covering the quarter of the pixel area are needed.
  - z_0 coordinate → 8 grid points along this coordinate are reasonable.
  - This creates 3D grid with 1µm spacing and 512 realizations of pixel charge fraction sets are needed. X-ray clusters are contained in 5 × 5 zone around the cluster central pixel. Then for each 3D grid point a set of 25 pixel charge fractions have to be stored.
- In total the storage size is 12.8 k values.
  - All values are positive and less or equal to 1. With a proper scaling factor (up to 65535) this values can be stored as two-byte unsigned integers without affecting the accuracy.
- required memory size is 25 kB.

Outlook

- The function to fill up the look up table has been developed in C/C++.
  - The function accuracy and calculations speed were tested.
  - The required accuracy better than 10^{-3} is achieved for all grid points.

- minimization procedure → (x_0, y_0, z_0)
  - first and second derivatives are required → the retrieval function needs smoothly interpolate between grid points to facilitate derivative calculations
  - The minimization could be performed on the grid and no derivatives are needed.
  - We explore both options.

- The preliminary timing measurements (time spent by a process was measured as wall time) demonstrated great increase in computational speed for look up table approach.
  - On the data set from NSLS-II SIX beam line, EMCCD sensors, the computational time spend on x-ray cluster analysis decreased from ~1.5 hours for direct numerical computation to ~4 sec for look up table calculations.
- The look up table and data retrieval speed is under optimization with minimization procedure in mind.