# Calculation of global distortions and corrections

## Documentation

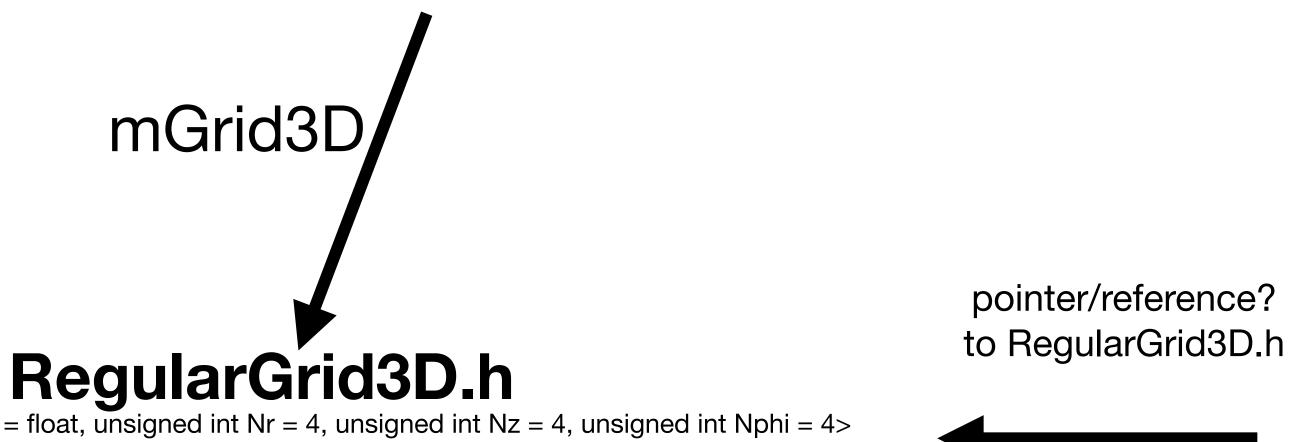
#### O2TPCSpaceCharge3DCalc.h

- The goal of this class is to calculate the global corrections and global distortions on a defined regular grid.
  The number of vertices in r, z, phi and the datatype float/double used for the calculations are template
  parameters. The global corrections and distortions can be calculated by an analytical approach using a
  TFormula (appropriate formulas will be defined in the class) or by using a TH3X as an input for the space
  charge density. The following steps have to be performed for the calculation of the global distortions/
  corrections:
- 0. Set Boundary conditions: fillBoundaryAndChargeDensities();
- 1. Solve Poisson equation:  $\Delta\Phi(r,\phi,z)=\rho(r,\phi,z)$  poissonSolver()
- 2. Electric field:  $\overrightarrow{E} = -\nabla \Phi(r, \phi, z)$  calcEField();
- 3. Calculate local distortions and corrections using Langevin equation: calcLocalDistortionsCorrections()
- 4. Integrate local distortions/corrections to obtain global distortions/corrections calcGlobalDistortionsCorrections();

## overview

### O2TPCSpaceCharge3DCalc.h

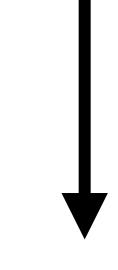
template < typename DataT = floa $\bar{t}$ , unsigned int Nr = 4, unsigned int Nz = 4, unsigned int Nphi = 4>



template <typename DataT = float, unsigned int Nr = 4, unsigned int Nz = 4, unsigned int Nphi = 4>

Container for local/global distortions/corrections Electric field etc.

TriCubic.h



Matrix.h

Vector and matrix multiplication

Storage format in single array (only single scalar values for now):

Small grid: DataT mGridData[] (stack)

Large grid: std::unique\_ptr<DataT[]> mGridData (heap)