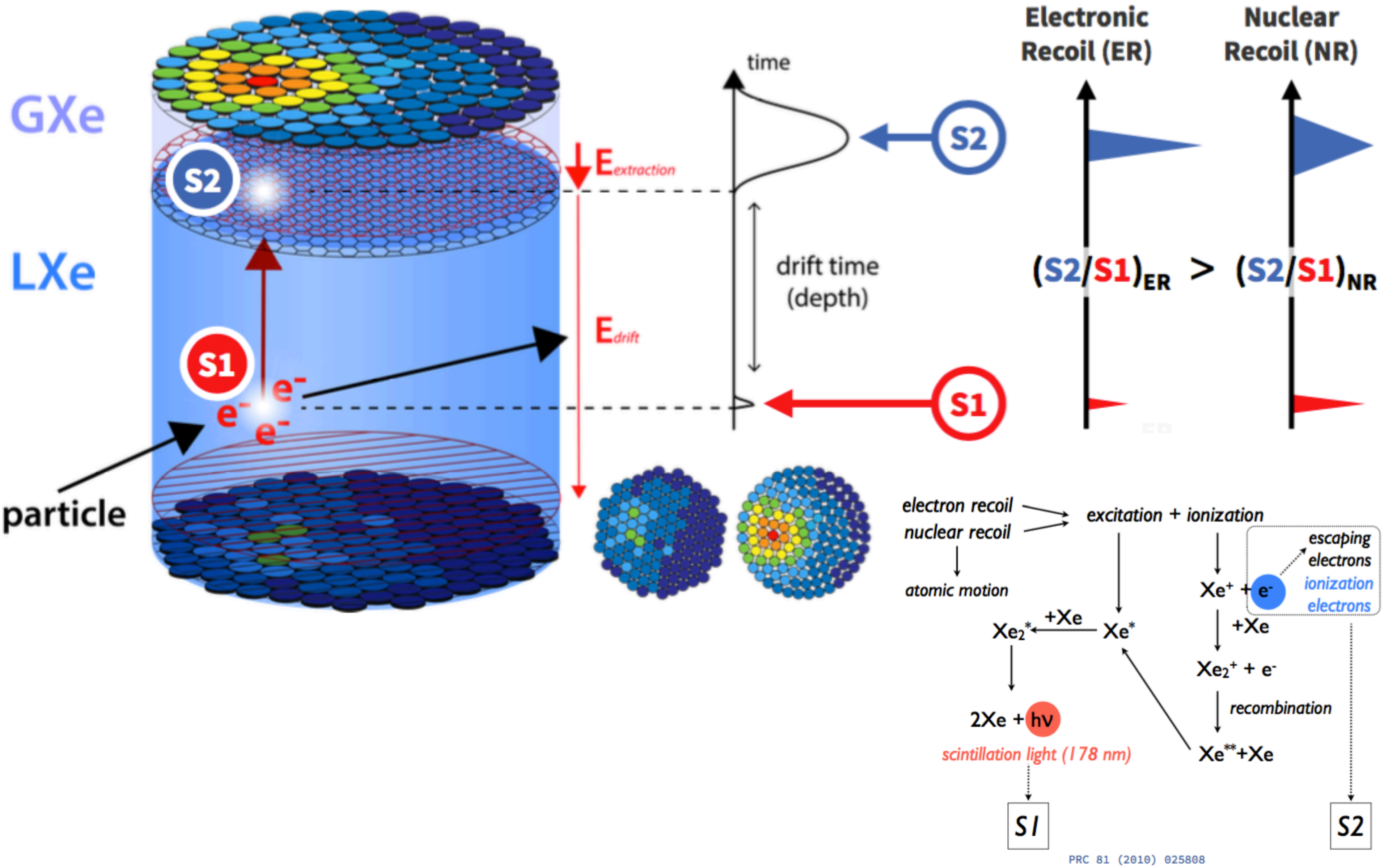
Detailed GEANT4 Simulation of S2 Photons in XENONnT

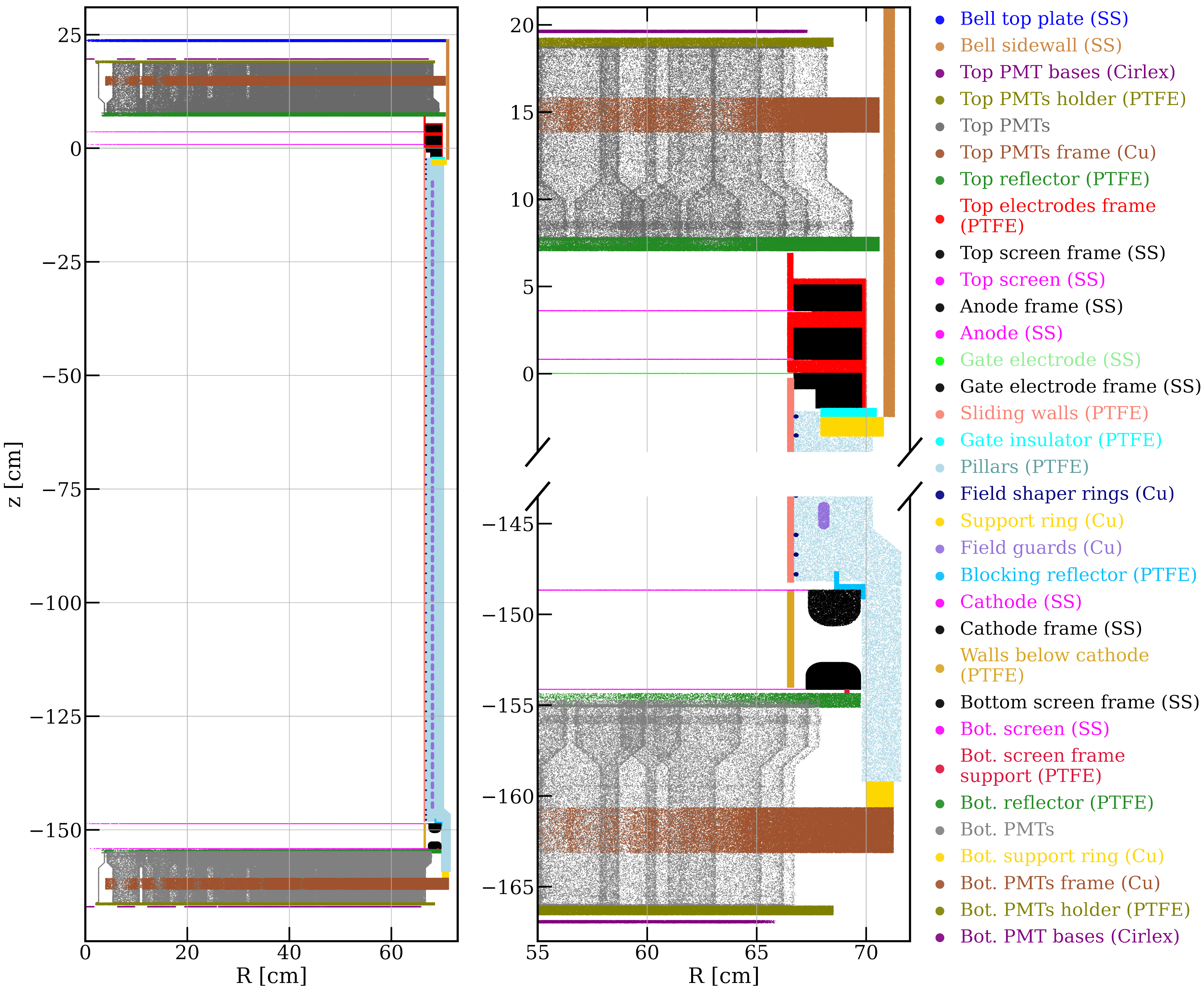
B. Andrieu, O. Dadoun (LPNHE-Paris)

Signal Production and Detection in XENONnT

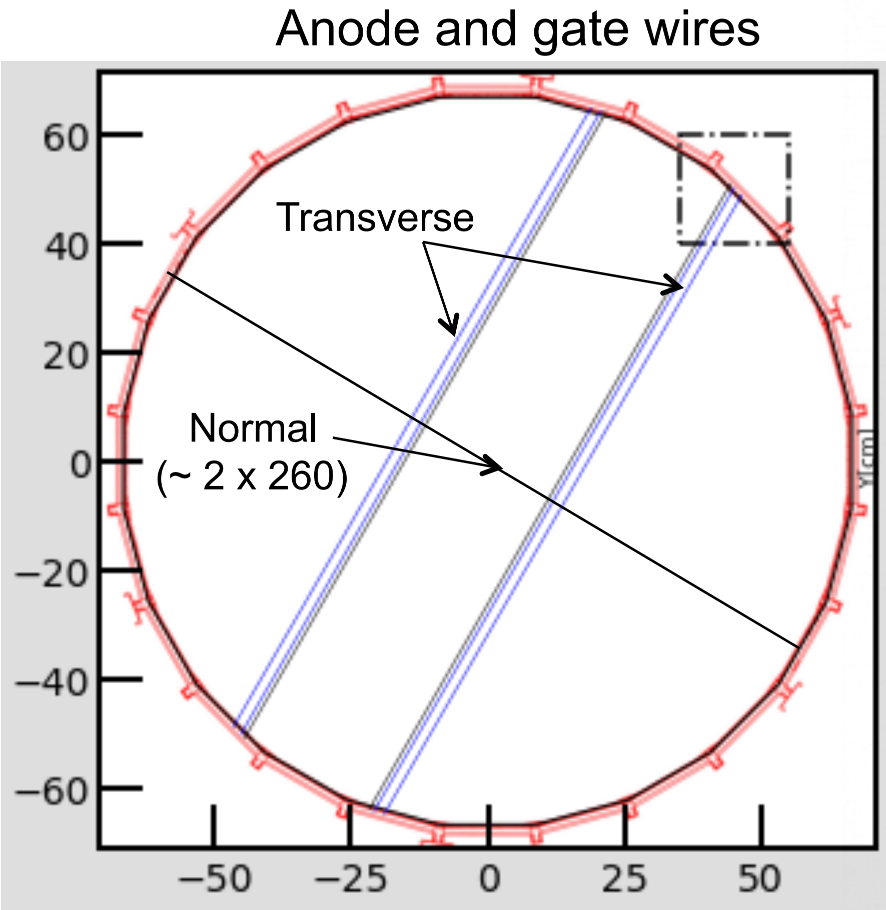
Dual Phase (Liquid-Gas) Xe TPC

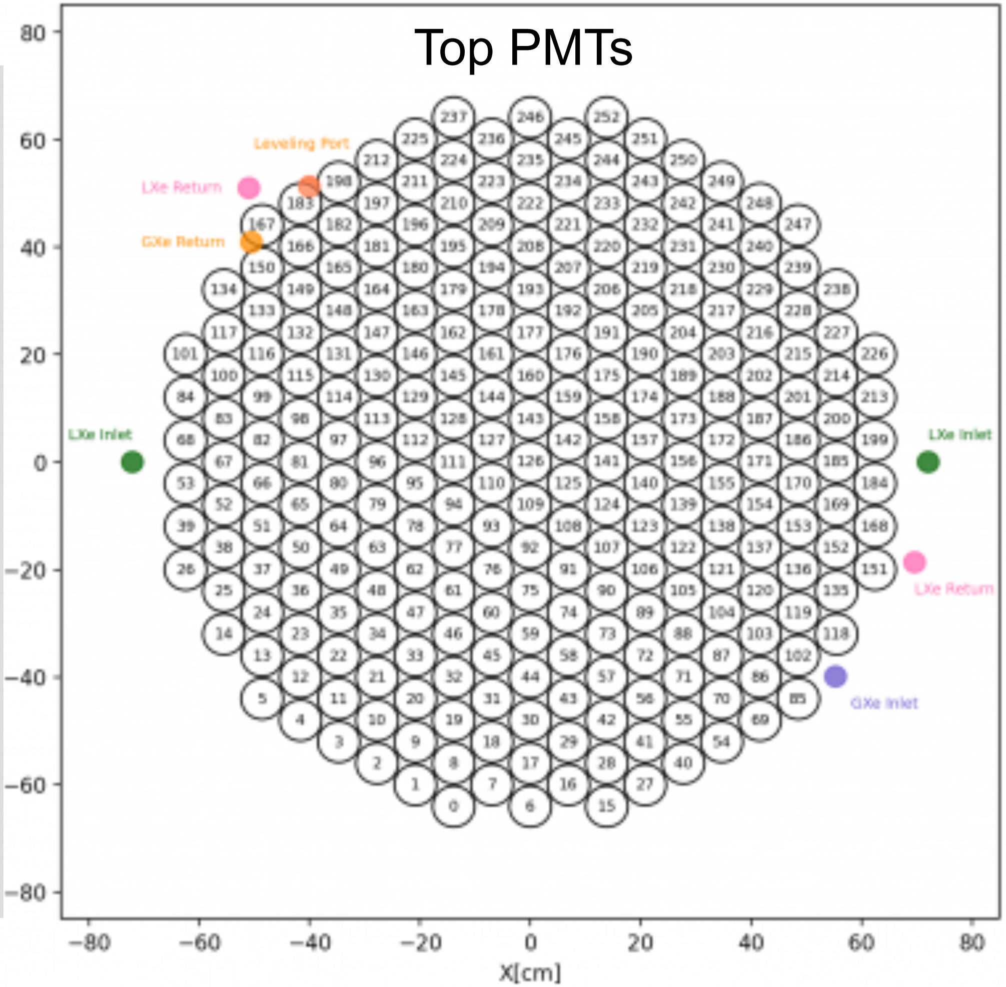


XENONnT Detector Geometry (R-z)



XENONnT Detector Geometry (x-y)



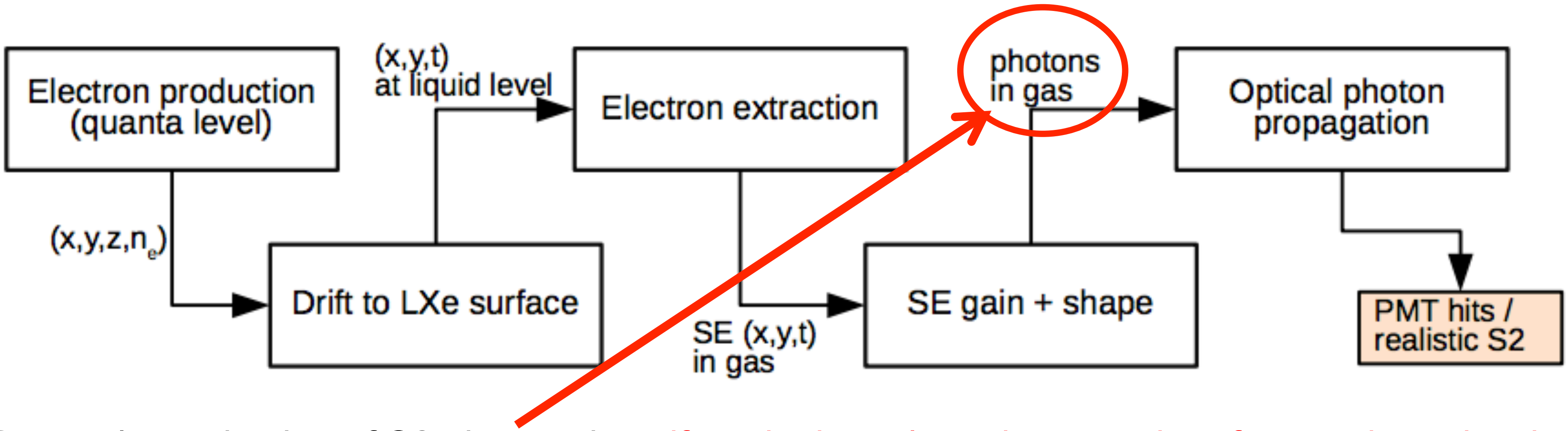


Gate
Wire pitch: 5 mm
Wire diameter: 216 micron
2 perpendicular wires
z height: 0 mm (By definition)
z height cold: 0 mm (By definition)

Anode
Wire pitch: 5 mm
Wire diameter: 216 micron
4 perpendicular wires
z height: 8 mm
z height cold: 7.88 mm

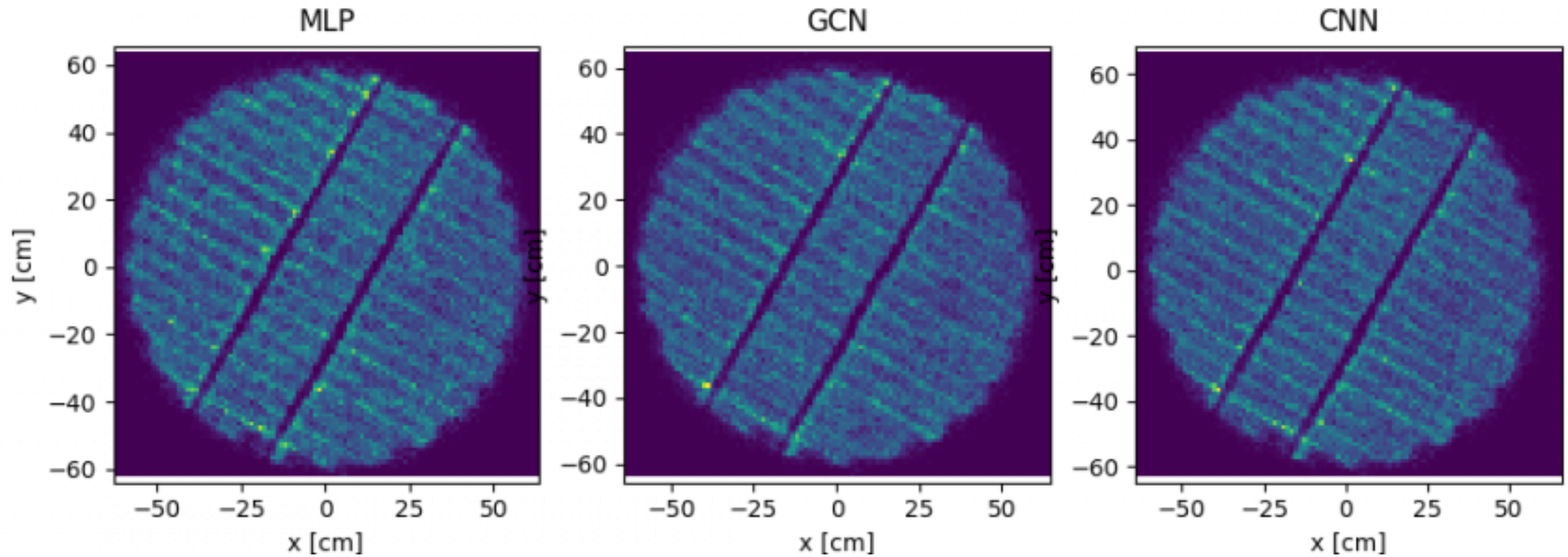
→ work in rotated coordinates (normal wires along x)

S2 Simulation Context



Presently production of S2 photons is uniform in the volume between interface and anode wires

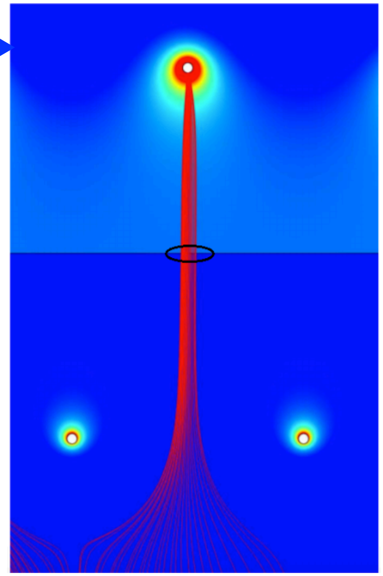
→ Problem: stride patterns in reconstructed position NOT reproduced by simulation



Simulation of Electroluminescence (S2 Photons in Gas)

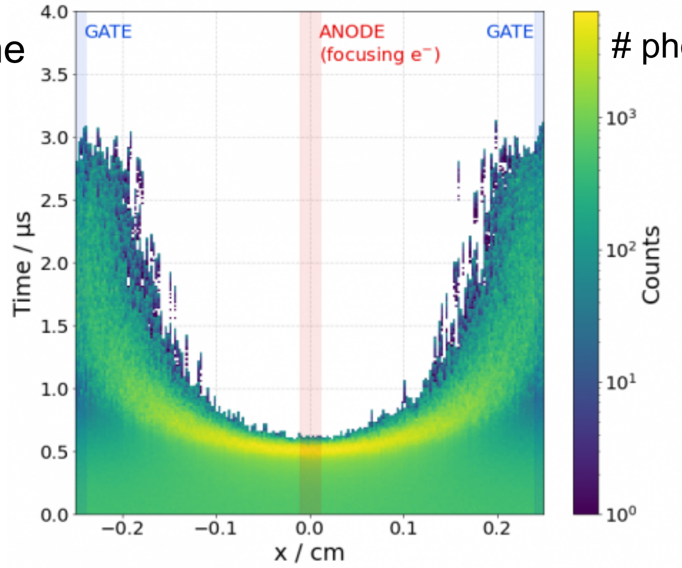
- Electroluminescence path for S2 optical simulation from COMSOL + GARFIELD (F. Toschi)

- COMSOL → computation of electric field lines
  gate focusing of electrons in x (200 µm) right below the anode wires

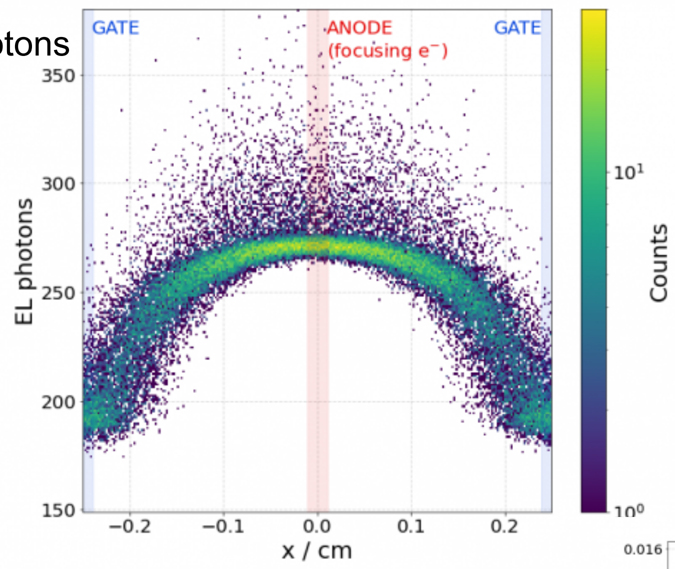


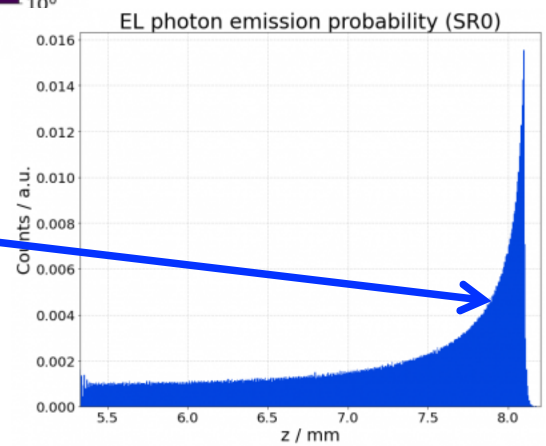
- GARFIELD++ → simulation of EL process for extracted electrons
  (right below the anode → stronger electric field → shorter path, more light)

time



photons



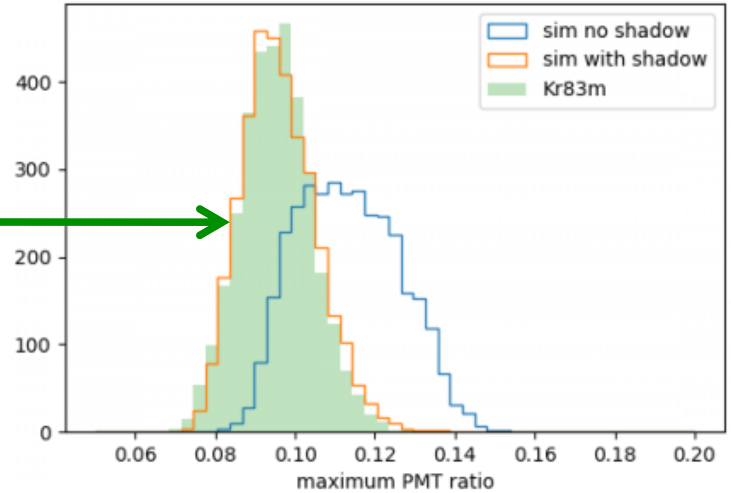


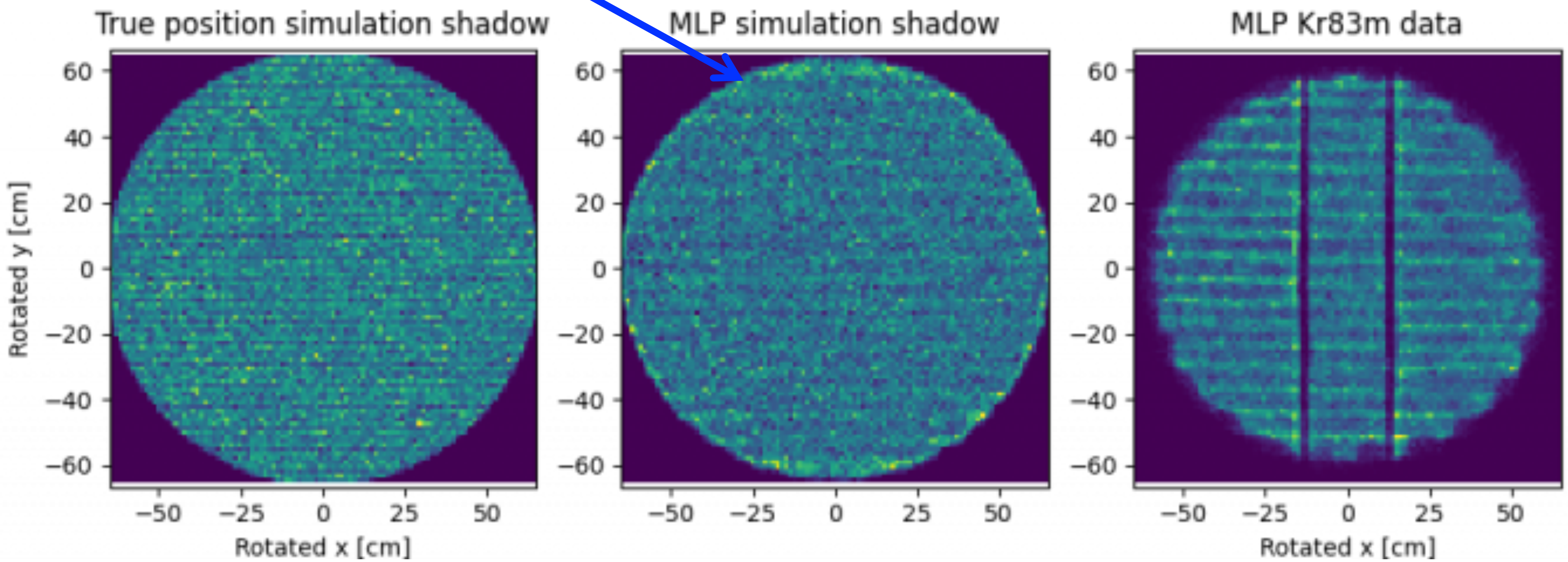
- Also preferential emission of photons close to wires in z

→ shadowing effect important

S2 Anode Shadow First Model

- Toy model to introduce shadowing by hand with empirically adjusted parameters (S. Liang)
- Better modeling of maximum PMT ratio (fraction of S2 signal in PMT with maximum signal)
- Apparition of stride patterns resembling to data

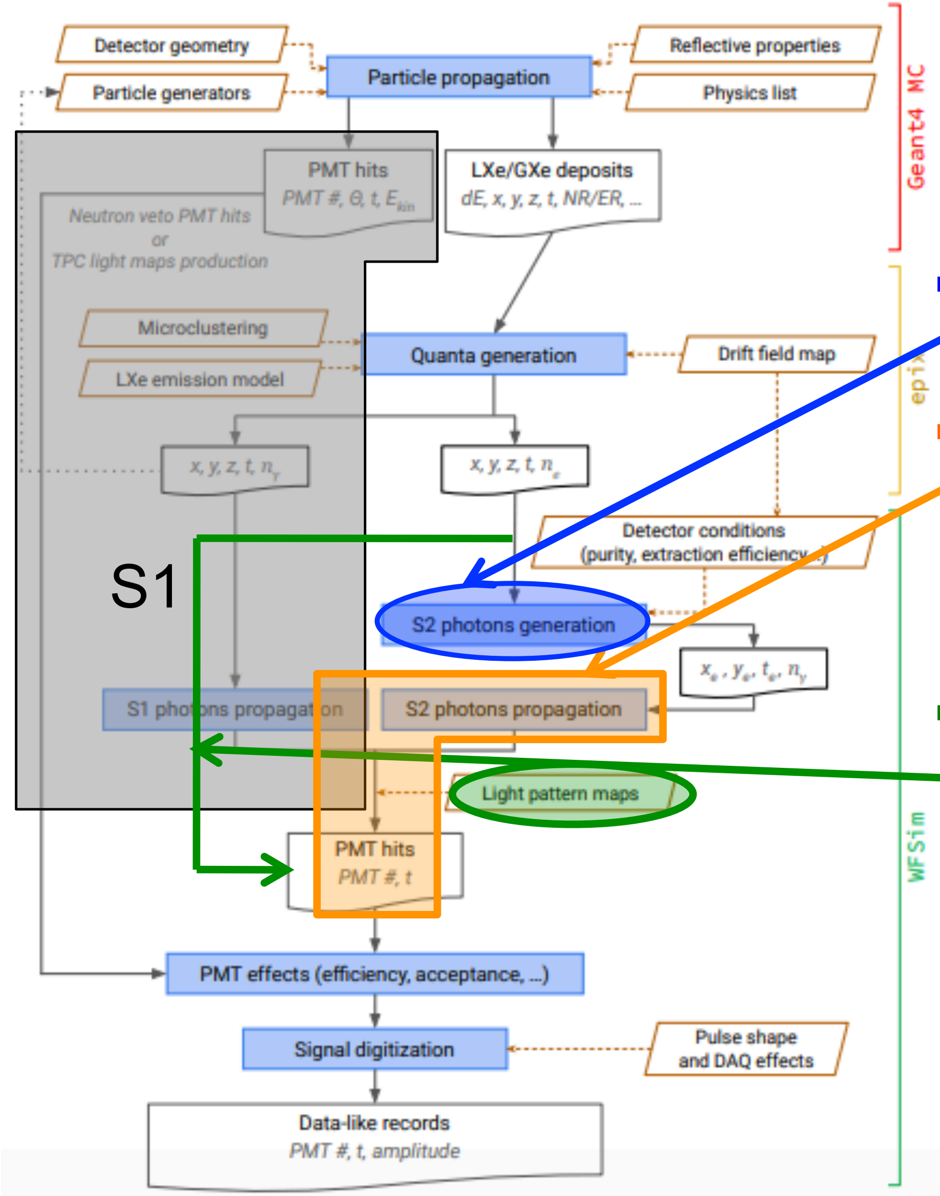




Motivation to implement detailed electroluminescence results in simulation

→ GEANT4 propagation of photons generated according to Garfield parameterization

Electroluminescence Effects in S2 Simulation



XENONnT MC Flow Chart

D. Ramirez, PhD Thesis https://freidok.uni-freiburg.de/data/228338

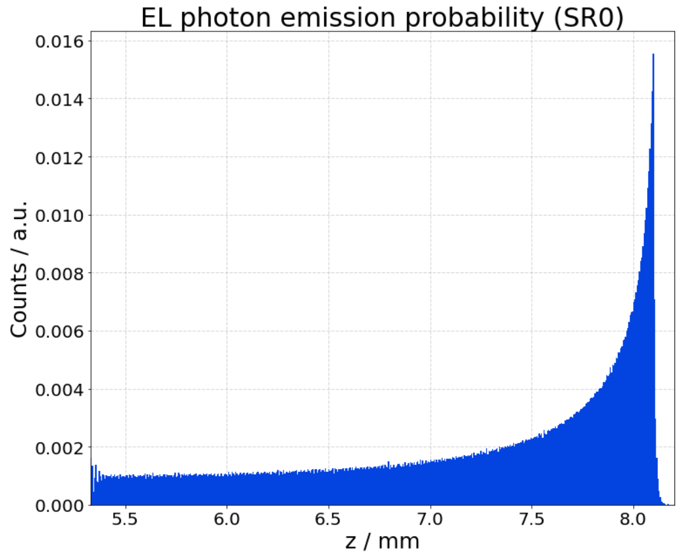
How to simulate shadowing?

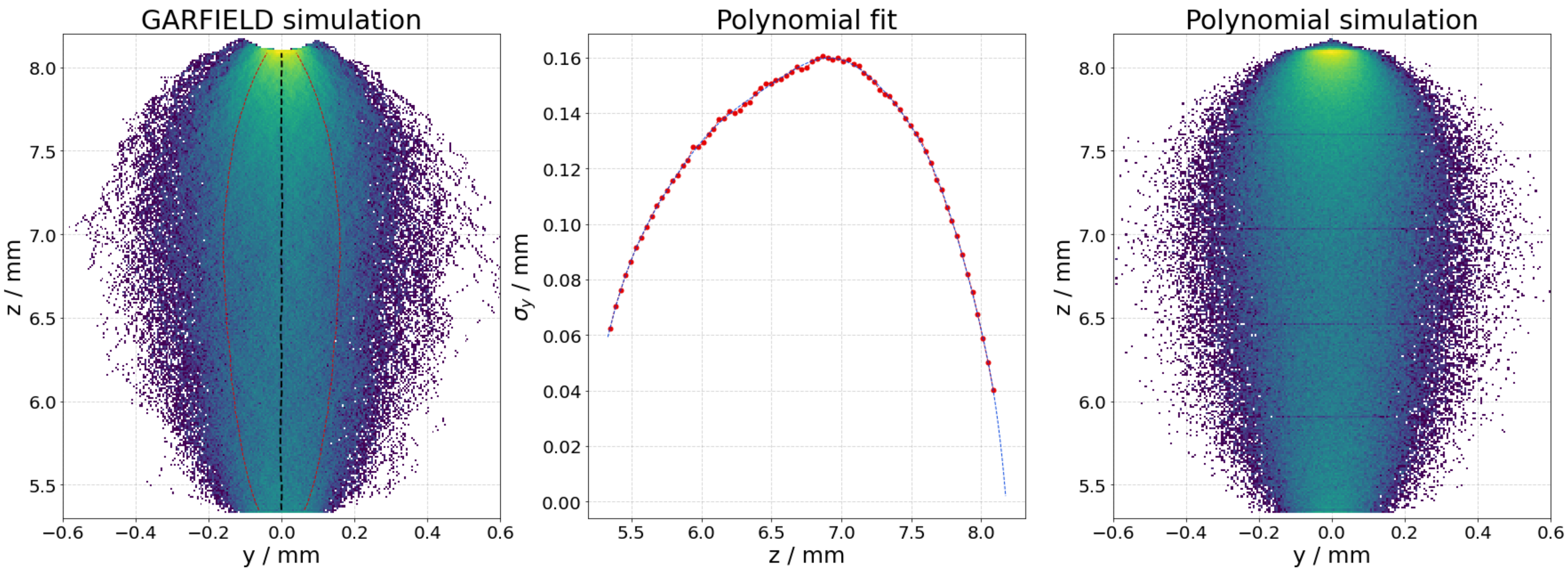
- Modify GEANT4 to generate photons according to GARFIELD results
- LCE (Light Collection Efficiency) maps from massive GEANT4 simulations
  = probability of γ from e^- (x,y) to hit PMT #i
- Use LCE maps in simulation to go directly from extracted e^- (x,y) to PMT hits

S2 Simulation of Electroluminescence: Parameterization

Polynomial fits of GARFIELD results

- Z distribution fitted with polynomial between 0 and 1, then rescaled to actual distance interface ←→ anode
- X and Y distributions (after rotation)
  - (x,z) or (y,z) distributions are divided in 75 slices in z
  - standard deviation is calculated for each slice, assuming a gaussian distribution at fixed z
  - $\sigma_{x,y}(z)$ is fitted using a 8 degrees polynomial

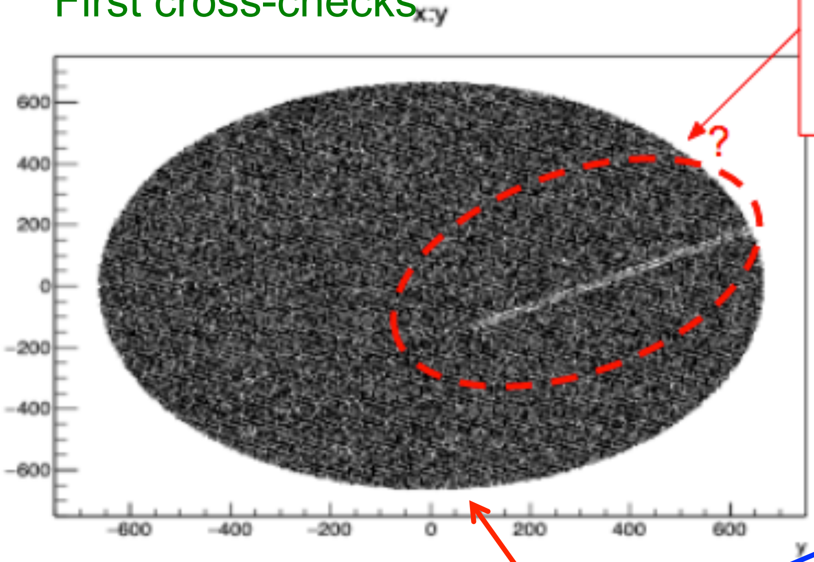




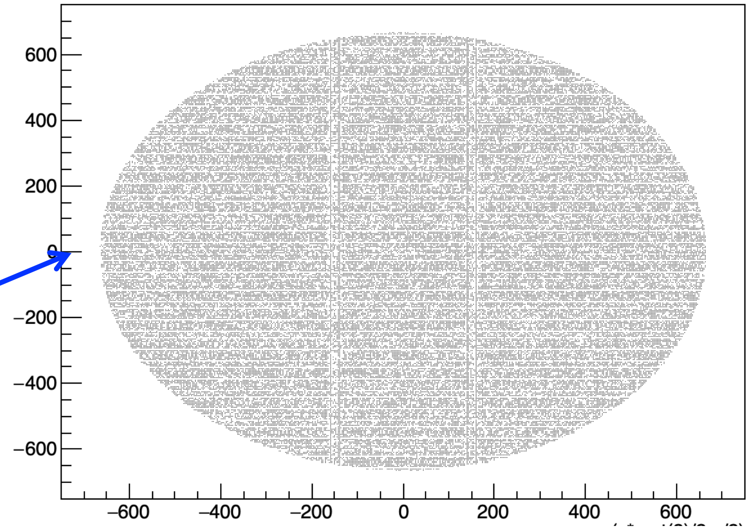
S2 Simulation of Electroluminescence: GEANT4 implementation

- Detailed photon generation from GARFIELD fits in GEANT4 branch (L. Althuser)
- **Main goal:**
  → simulations of O(10^6) (or more?) photons per (x,y) point to produce LCE maps
  = correspondance (x,y) position → probability for one photon to hit a given PMT
  → Maps used to simulate in a realistic way "list of PMT hits" for any single e^- (x,y)
- **First part:**
  cross-check (and correct) this XENONnT GEANT4 branch

First cross-checks



- Is this a wire ?
- Why one ?
- Why off axis ?



x:y distribution of photons before & after correction

S2 LCE maps

S2 Light Collection Efficiency (LCE) maps function of (x,y)

Ideal case: $P_n(x,y)$ = p.d.f. of γ (from e^- (x,y)) of hitting PMT #n. In practice:

- discretize (X,Y) plane in (i,j) bins
  → find mean probability $P_n(i,j)$ of hitting PMT #n for photons from electrons in bin (i,j)

Proposal for bin definition:

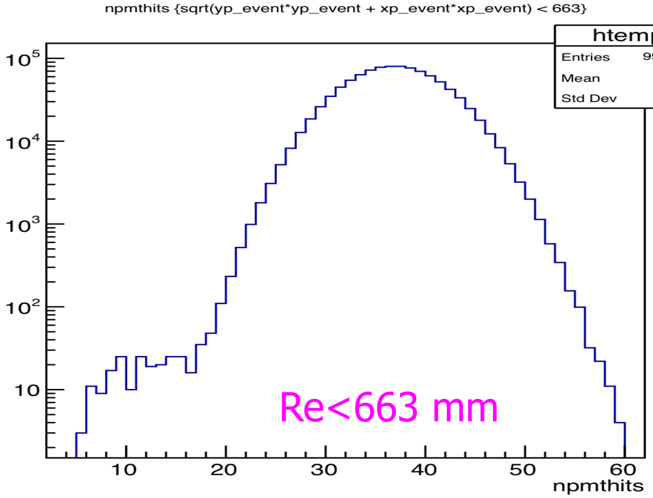
- central anode wire position at Y=0, pitch=5 mm
  → bins centered at Y=0, Y width = 5 mm
- perpendicular wires positions at X=+-13.1 (gate), 14.0 and 15.9 (anode) cm
  → bins centered at X = 0.5 cm (bin limit on gate perp. wires)
- To avoid statistics problems with bins close to TPC border, only keep those whose center lies inside TPC (circle of radius R = 664 mm)

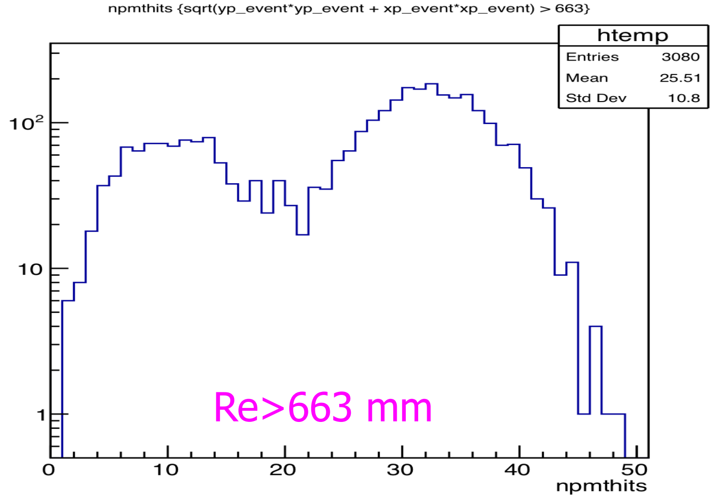
First production tests:

- Simulation of 10 jobs of 1M e-, each e- producing 100 photons, for a total of 1G photons.
  Timing (per job) ~15h (real) or ~5h30 (CPU) - Disk space (per output file) ~5GB
- only a few entries (as low as 1, maybe 0) for some PMTs in certain e- positions
  → multiply statistics by 100 (e.g. 100 jobs of 1M e- x 1000 photons / e-) for final prod
- Reduce number of variables in Geant4 rootuple → Total size (for 100 G photons) ~ 3 TB

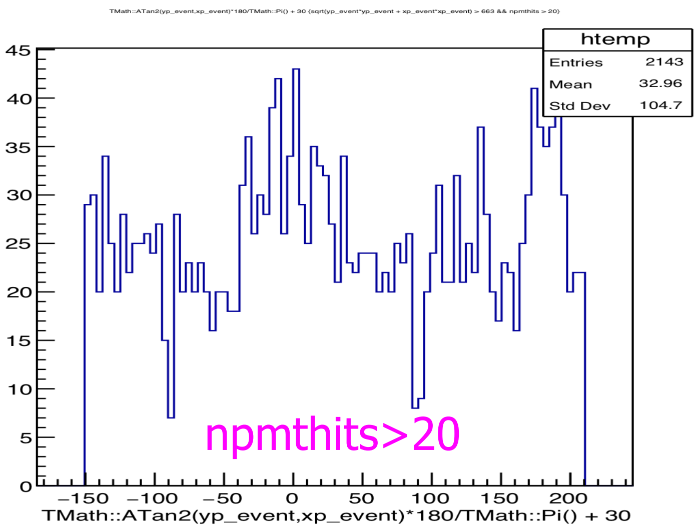
S2 LCE maps : problem at TPC edge?

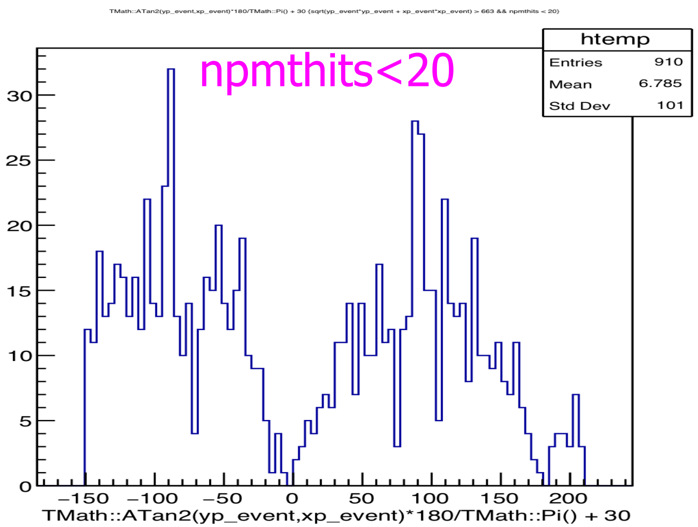
npmthits (# of photons in PMTs per e-) vs Re (radius coordinate of e-)





θ (rotated coordinates) distribution for Re>663 mm





→ Problematic for cells close to limit (mix e- with different PMT response)?

→ Solution (up to now): restrict simulation to Re<662 mm

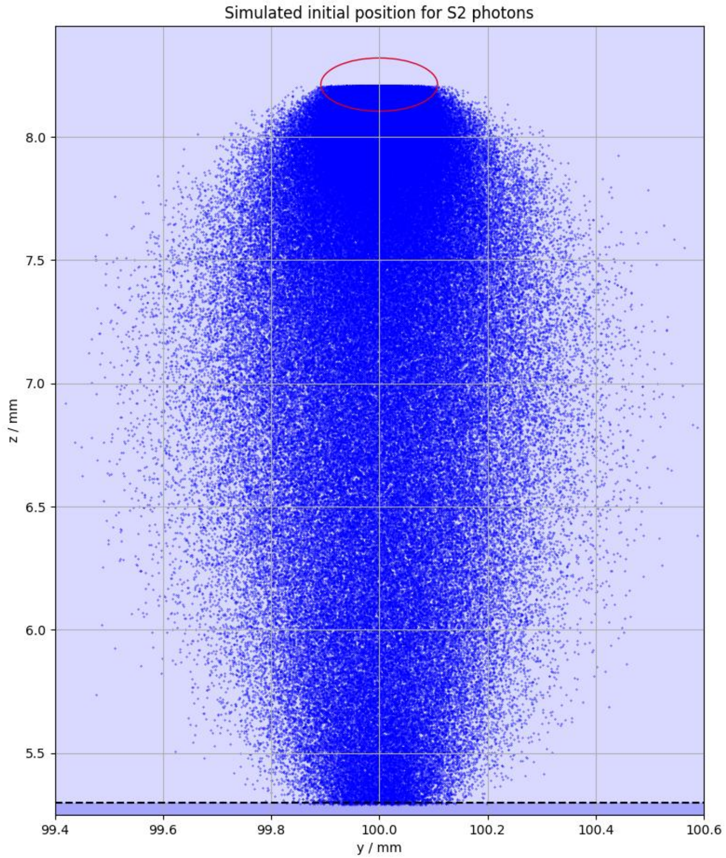
Code checks and corrections I : photons inside anode wires

- Correction to produce photons up to center of anode wire
  → add radius of anode wire in z... but subtract an equivalent amount in z, due to shrinkage
  → detailed geometry check (GEANT4 implementation only, XENONnT geometry untouched)

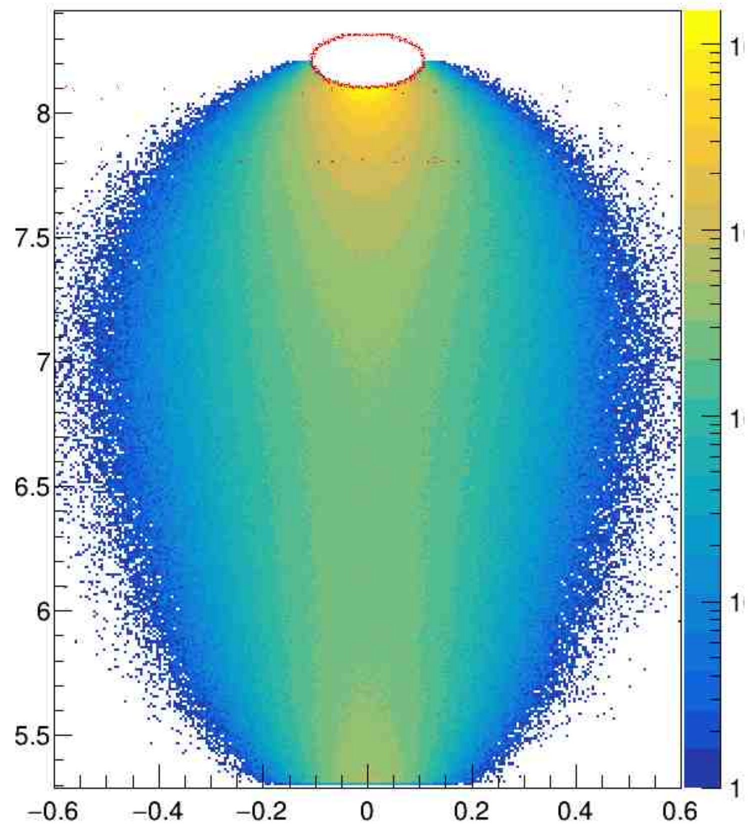
- Correction to avoid generating photons inside anode wires
  → unphysical (no photon produced in wire in real life)
  → slightly rescale z of photons below anode wire (in y) to follow exact wire shape

z vs y of S2 photons

before correction



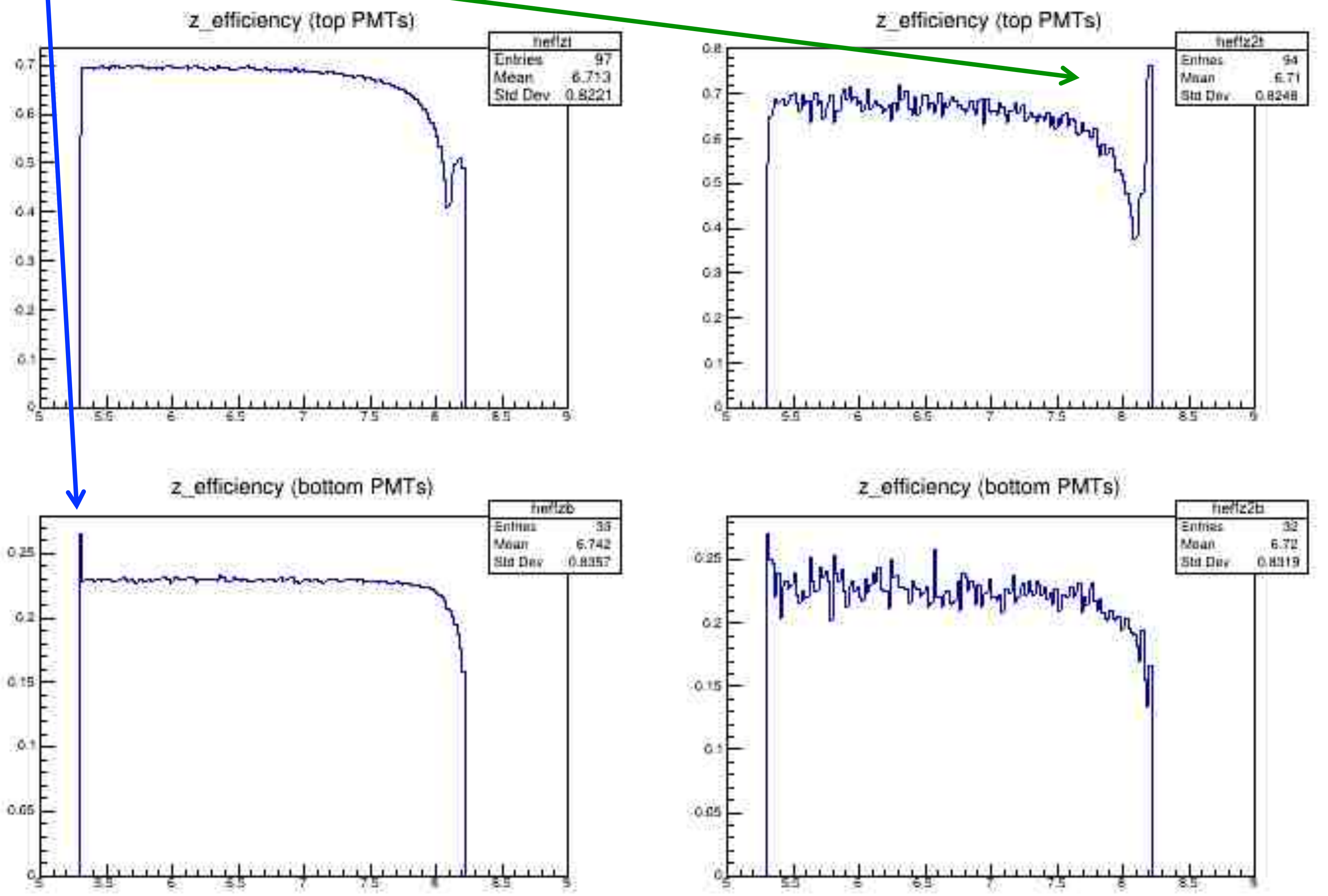
after correction



Production test checks I : efficiency vs z

Production in Lyon: 300 jobs x 100K e- x 1000 photons/e- → 30% of full stat

- Problems identified in efficiency vs z (top & bottom PMTs)...
  - small peak at the liquid-gas interface of bottom PMTs efficiency → photons generated in liquid?
  - big peak at z_{max} of top PMTs efficiency for photons below transverse wires?



top PMTs

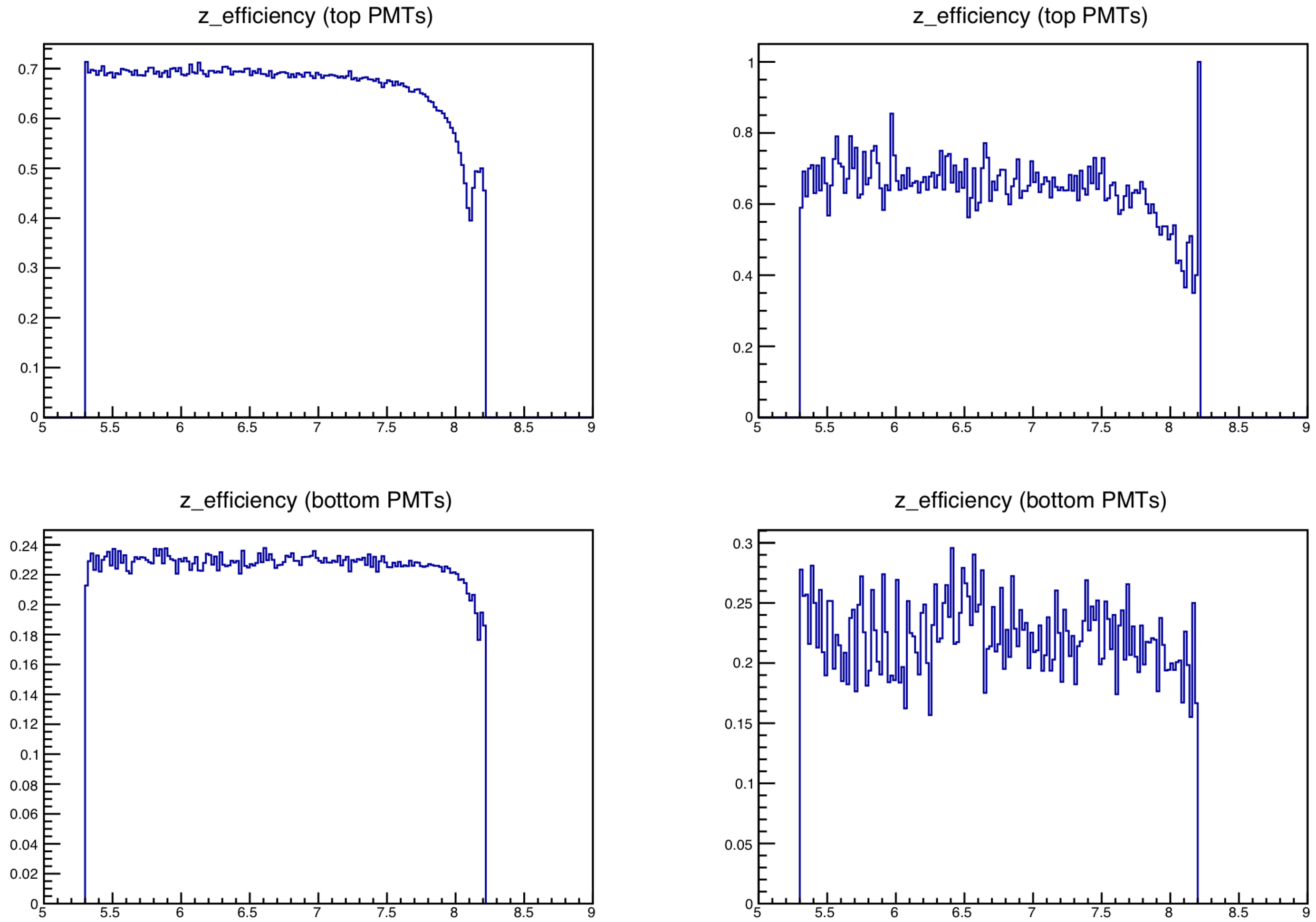
bottom PMTs

far from transverse wires

near transverse wires

Code checks and corrections II : photons in liquid

- Peak in bottom PMTs efficiency due to photons generated in liquid?
  → problem traced back to polynomial parametrization of z distribution of photons
  - z distribution assumed in [0,1] then rescaled to distance between liquid-gas interface and anode
    but z(0)≠0 and z(1)≠1 → force z(0)=0 and z(1)=1 by changing lowest and highest order coefficients



top PMTs

bottom PMTs

far from transverse wires

near transverse wires

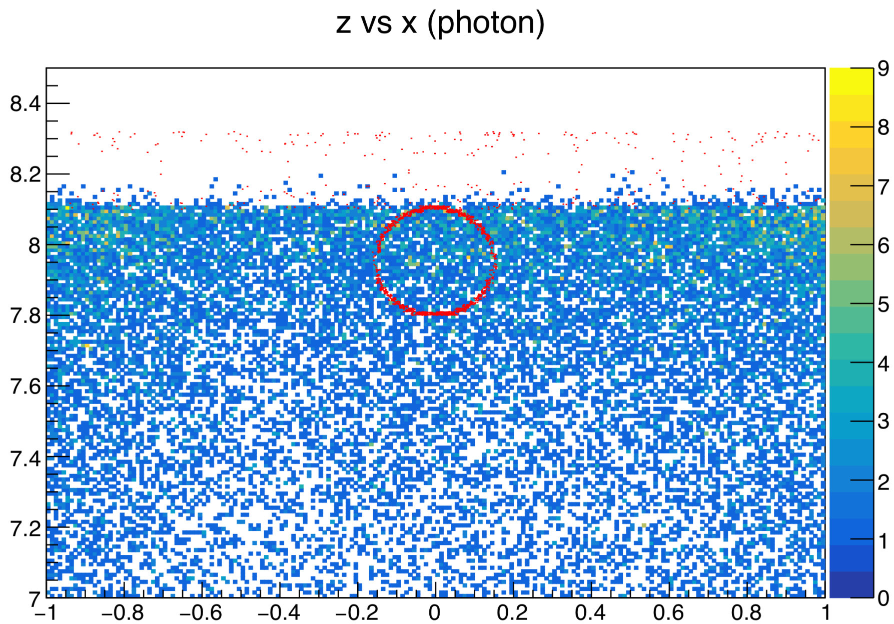
→ Problem of photons in liquid solved, top PMTs efficiency peak at z_{max} still there

Code checks and corrections III : photons inside transverse wires

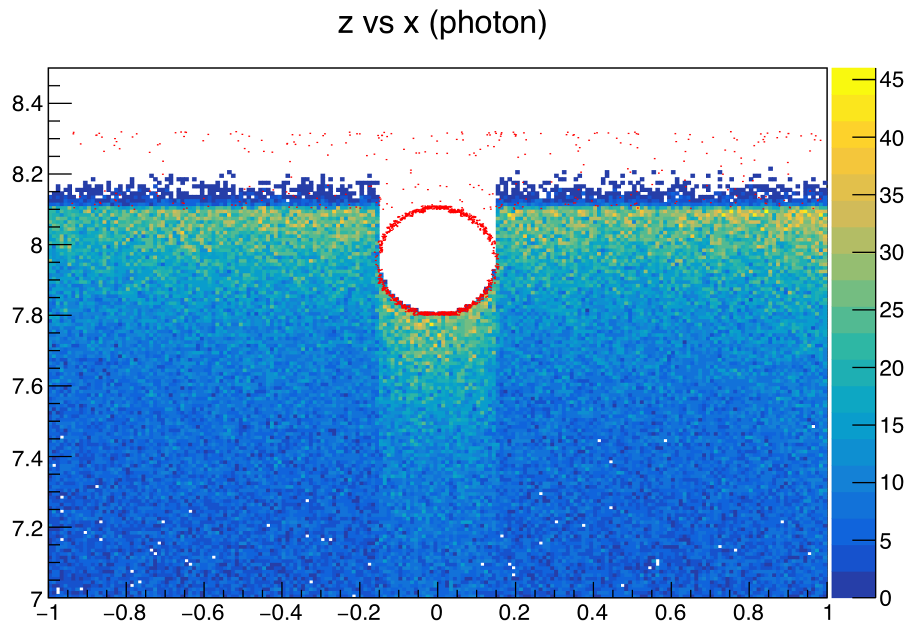
- Peak at zmax of top PMTs efficiency for photons near transverse wires?
  → problem traced back to photon generation inside transverse wires
  - guess: photons generated inside transverse wires are lost so that near transverse wires, photons just below anode wires do not enter in efficiency calculation → artificial increase

  → solution: as for anode wires, slightly rescale z distribution of photons below transverse wires (in x) to follow exact wire shape

z vs x of S2 photons near transverse wires



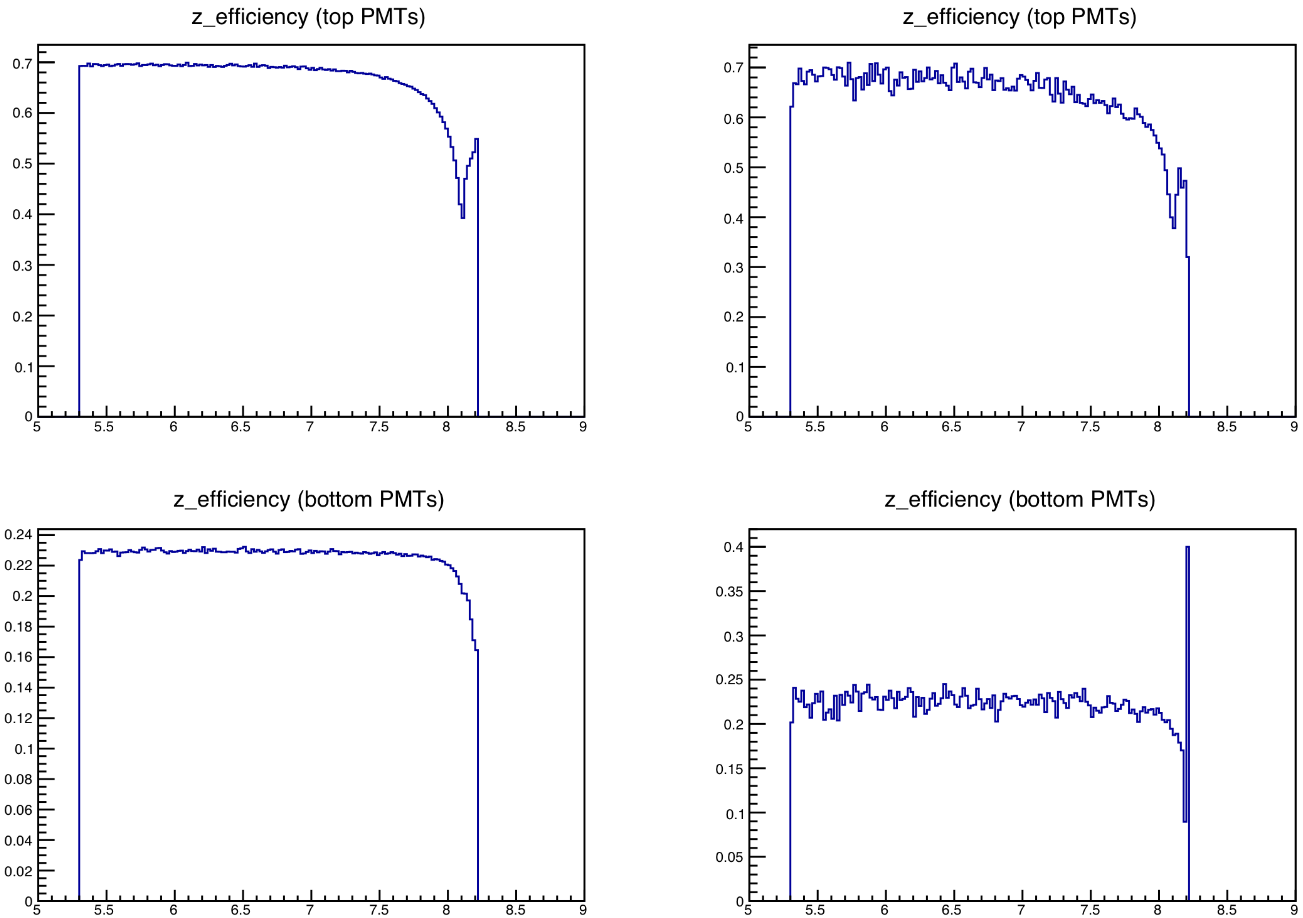
before correction



after correction

Production test checks II : efficiency vs z

- With all corrections included, everything looks consistent now
  - → only problem left at z=zmax for bottom PMTs efficiency near transverse wires?
  - → statistical effect (tbc)



top PMTs

bottom PMTs

far from transverse wires

near transverse wires

→ Code ready for production!

S2 map production at CC-IN2P3

Production at CC-IN2P3 (Lyon)

- Geant4 branch with all modifications installed at CC-IN2P3
- Simulation of 1000 jobs of 100K e-, 1K photons per e- → 100G photons.
  Timing (per job) ~15h (real) or ~5h30 (CPU) – Total Disk space ~3TB

  → Total time for production ~1 day , can be re-done easily

S2 Light Collection Efficiency (LCE) map function of (x,y)

- From Geant4 root files, LCE maps produced (~2-3 h) by root macro as .csv file

  → 2D array of ~28000 lines (1 per x,y position) and 494 columns (1 per PMT)

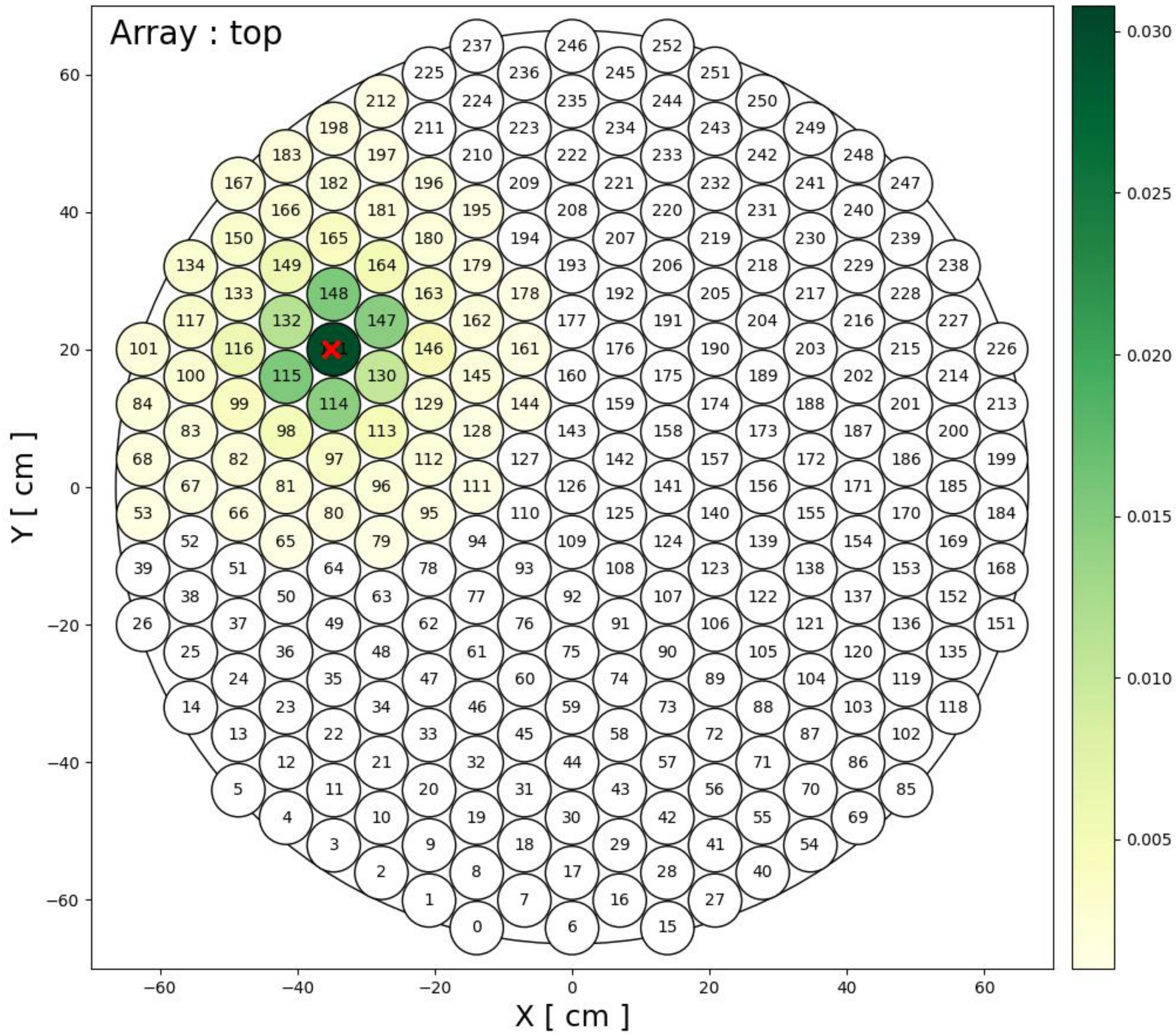
Python function ready to load map and extrapolate between bin centers
→ continuous S2 Light Collection Efficiency (LCE) function of (x,y)

- 1D (along x_{rot}) extrapolation mandatory!
- For the moment, simple linear weighting, but could be easily changed

S2 LCE maps : display

LCE per top PMT for γ from e^- at x= -35 y=20

→ shadowing visible



Summary and outlook

- Geant 4 code implementing electroluminescence processes parameterization for realistic S2 simulation certified for production
- Geant4 massive production (100 G photons) for computing S2 LCE maps done at CC-IN2P3
- S2 LCE maps produced, tool available for visualization
- Implementation in XENONnT full simulation in progress (first version running, still to be fully tested)
- Several experimental effects still to be included once first version validated
  - Sagging
  - Transverse wires
- General lesson:
  → **shadowing effects important for detailed S2 simulation**

Backup

S2 Simulation steps

1. scattering of a particle through the detector is simulated using GEANT4. At each interaction, the energy recoil is extracted, and for each interaction with enough energy recoil, the number n_e of ionization electrons (and n_p of scintillation photons, used for S1 simulation) is estimated using NEST package.
2. electrons drift through the TPC (drift field) towards the top of the detector. The loss of a certain amount of electrons due to the presence of electronegative impurities is simulated according to measured electron lifetime.
3. electrons also diffuse transversally and longitudinally (D_T , D_L), so that their initial position is slightly smeared, as well as their arrival time at the gate mesh.
4. when electrons approach the gate mesh, they are attracted on top, due to the presence of the higher potential of the anode, until they reach the liquid-gas surface, following a well defined path according to the electric field, which is highly non uniform due to the gate and anode shapes (interleaved parallel wires, and 2 * [1 gate + 2 anode] perpendicular wires). The electric field maps used are thus based on COMSOL simulations.
5. electrons are extracted from the liquid to the gas phase according to an extraction yield which depends on the position (extraction yield = f(x,y)) mainly because of the sagging of the two electrodes but also because of the presence of the perpendicular wires.
6. once the electrons are inside the gas phase, they are accelerated up to the anode producing photons (secondary scintillation) along their way according to electroluminescence path information derived from GARFIELD simulations, which use electric field from point #4 and geometry from point #5 as inputs. The outcome is, for each electron e, a certain number n of photons defined by their position (x_i, y_i, z_i) and their emission time t0_i.
7. those photons are then used as an input for Geant4 and propagated (optical simulation) through the TPC so as to extract, for each photon, the corresponding PMT j hit (if any) converted as S2e_i_j signal at arrival time t0_i_j. The signal observed by each PMT j from a given electron e is defined as S2e_j = Sum_{i=1,n} (S2e_i_j)
8. all the S2e_i_j and t0_i_j (signal and arrival time of photon i in PMT j) coming from n_e electrons are given as input to WFSim in order to produce the correct waveform in each PMT j, which can be written formally as S2_j = Sum_{e=1,n_e} S2e_j, while the total S2 signal can be expressed as S2 = Sum_{j=1,nPMT} S2_j.

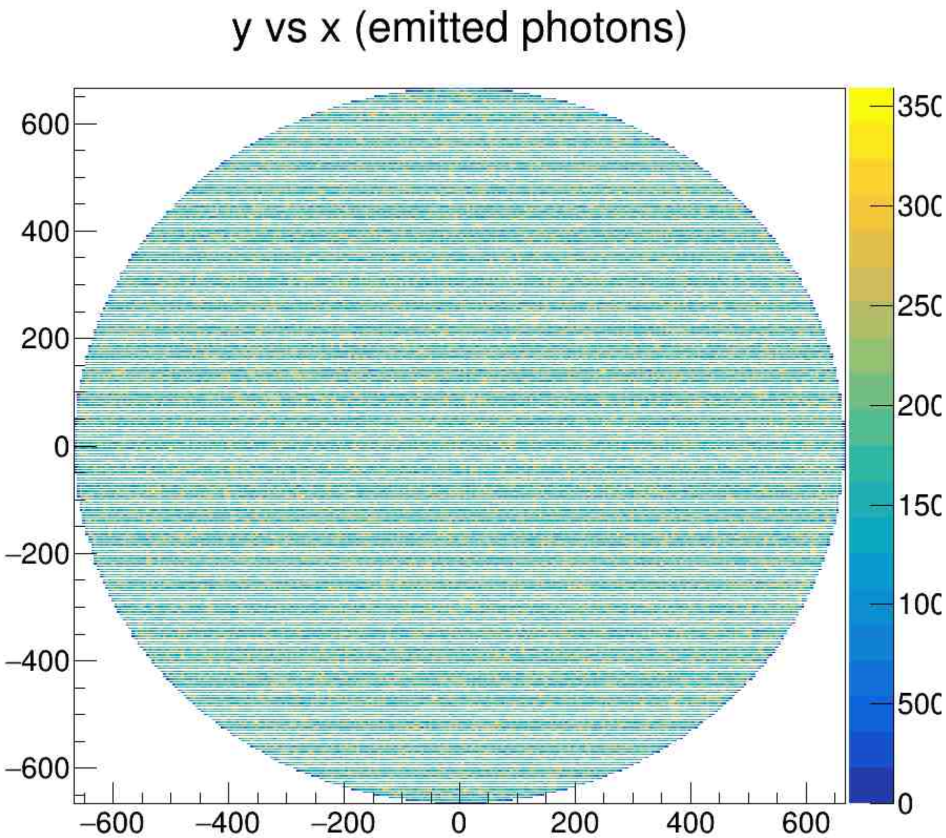
<u>This project</u>

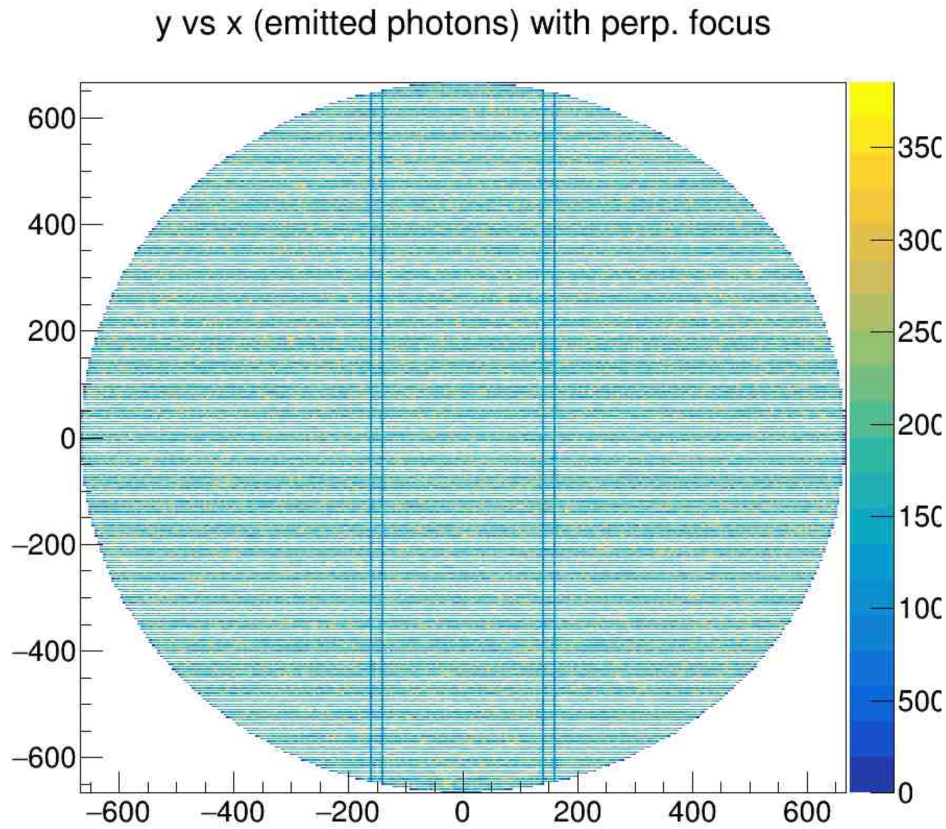
Parameterization of step #6 is used as input in Geant4 massive simulations to produce PMT response maps in (x,y) (step #7)

Detailed checks and code correction

- Small correction for generation with/without transverse wires focusing

  → had to correct some flaw in logic due to C++ "feature"
  → found a small error in distance to wire calculation, also due to C++ vs python "feature"
  → now working fine



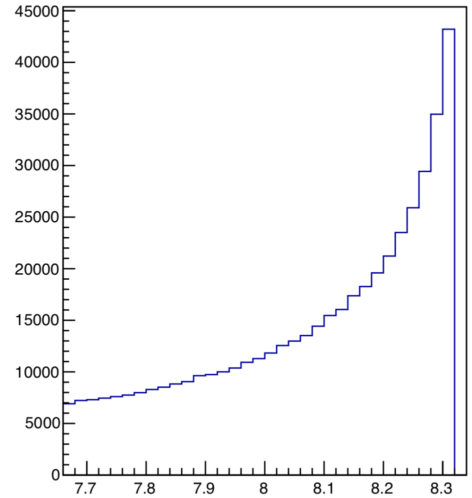


Note: transverse wire focusing is switched OFF for S2 LCE map production

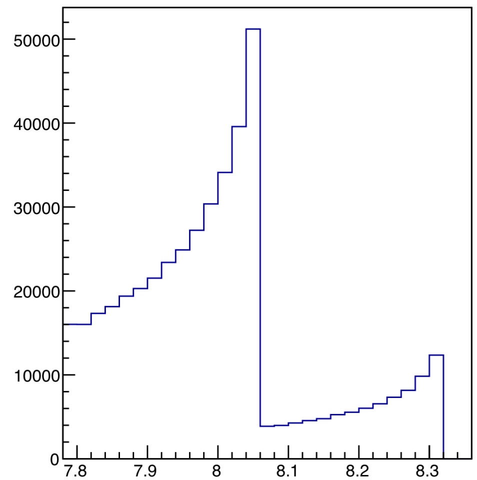
Detailed checks and code correction

- Small correction to produce photons up to center of anode wire (Lutz)
  → had to add radius of anode wire in z...
  → ...but revealed that one had to subtract an equivalent amount in z, due to shrinkage
  → motivation for detailed geometry check, should work fine now (plots not updated yet)

z photon near transverse wires

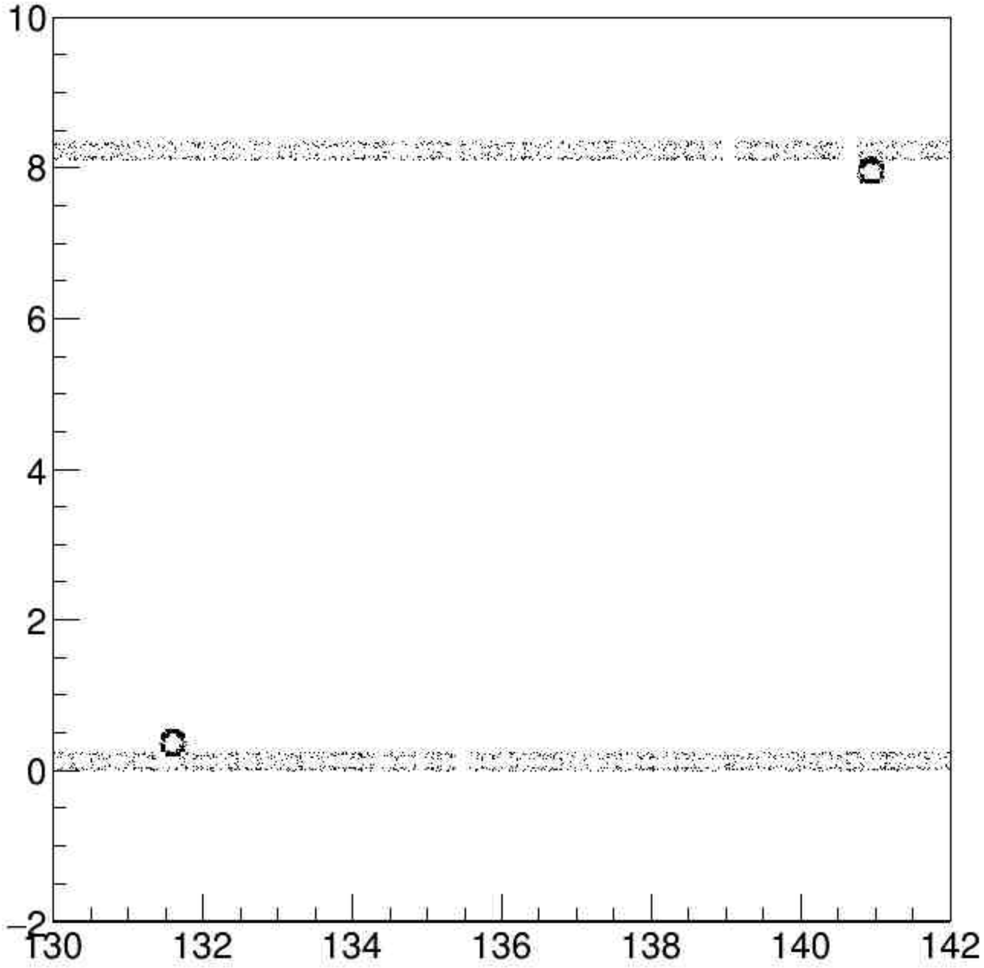


w/o focusing



with focusing

z vs x (anode wires)



Effect of transverse wires

Simulation study of effects of transverse wires based on Garfield + COMSOL (F. Toschi)
main effect: separation of e- path in two branches depending on x position relative to gate perp. wires