Digitization parameters

07-07-2025

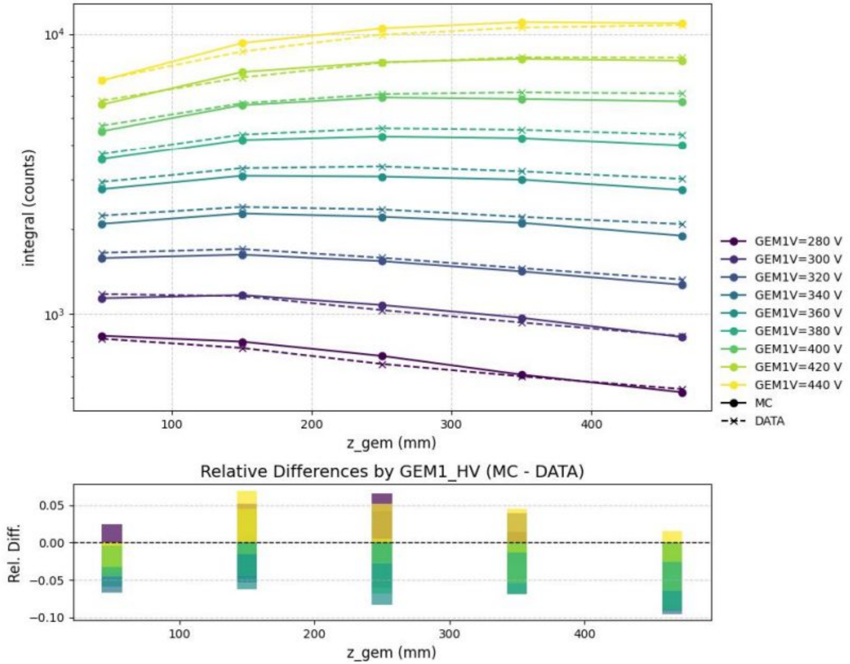
Overview

1. Recap of results shown at 2024 Collaboration Meeting
2. Recap of how to reproduce results shown at 2024 Collaboration Meeting (best parameters for mid-december 2023, RUN 4 iron scans)
3. Pull request with optimal parameters for mid-december 2023 iron scans
4. Procedure to digitize different time period (tuning of single gem gain and absorption length)

Collaboration Meeting 2024

From Collaboration Meeting
2024 we shown how we were able to simulate iron at different votlages:

https://agenda.infn.it/event/43515/contributions/249944/attachments/128808/191057/Detector%20Simulation%20-%20Digitization%20-%202.pdf



- Calibration scans of GEM1V and source position (iron).
- Run Date: December 15, 2023 (Run 4 LNGS). [runs: 42985-43050]

What are the best parameters?

In the following slides I show the best parameters **to reproduce December 15, 2023 (Run 4 LNGS). [runs: 42985-43050] shown at the collaboration meeting 2024**

All parameters except the gain (0.03) and the absorption length (1350 mm) can also be used to simulate other periods of time.

BUT the gain and absorption length should be tuned for each other time period if the data/mc agreement is needed to be at 5% level.

When changing drift field (RUN 5) both abs. length and diffusion coefficients should be adjusted. For the diffusion coefficients we use the Garfield simulation (see slide n. 17)

The digitization code we are referring to is the most updated version in c++:

https://github.com/CYGNUS-RD/digitizationpp/

NOTE: currently the parameters are not the most uptodate ones. See slide 12 for pull request

Best parameters for December 15, 2023 (RUN 4) [runs: 42985-43050]

Optical counts per photon

DigitizationRunner.cxx:
https://github.com/CYGNUS-RD/digitizationpp/blob/main/src/DigitizationRunner.cxx

```
void DigitizationRunner::prepareCameraSettings() {
    std::string camera = config.get("Camera_type");

    y_pix = 2304;
    if (camera == "Fusion") {
        x_pix = 2304;
        optcounts_per_photon = 4.; // (Fusion sheet) counts/photon = QE (0.8) / e-/count (0.21)
        y_sensor_size = 14.976;    // mm
        readout_time  = 184.4;     // ms in in ultra quiet scan (UQS)
```

In the next slide, the reason why it should be 4 and not 2.

Photon Transfer Curve

Conversion Factor (electrons/count)	Ultra quiet scan	0.21
	Standard scan	0.21
	Fast scan	0.23

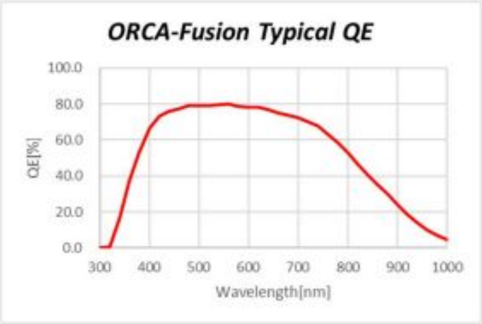
Measurement Conditions:

- Trigger: Internal trigger
- Cooling mode: Air-cooling
- Pixel correction: OFF
- Wavelength: 530 nm

P: Input photon number (photons)
CF: Conversion factor (electrons / count)
Out: Output count (counts)
Off: Dark offset (counts) [1]
Q(λ): Quantum Efficiency [QE] (%) [2]

$$P = \frac{CF \times (Out - Off)}{Q(\lambda)/100}$$

Example: If the Conversion Factor of this sheet is 0.24 (electrons/count), the pixel Intensity of interest at full 16 bit output is 5,850 (counts), Dark offset is 100 (counts) and Q(λ) at 550 nm wave length is 80(%), then the input photon number is calculated as



$$P = \frac{0.24 \times (5{,}850 - 100)}{0.80}$$
$$= 1{,}725\ (photons)$$

[1] The typical value of 100 (counts) or a measured value of output intensity with no input photons on the camera should be used.
[2] The QE (%) at each wave length is given by "Typical QE" graph.

ORCA-FUSION SHEET: https://drive.google.com/file/d/1LG5CoCU-ur-oIl-05HqhC_3vkC30k_HH/view?usp=sharing

Diffusion coefficients

https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

diff_const_sigma0T (mm^2) 0.1225 ---> 0.13475 (**adjusted**)

diff_const_sigma0L (mm^2) 0.0676 ----> 0.0676 (**unchanged**)

diff_coeff_L (mm/sqrt(cm)^2 for 1 kV) 0.00978 ------> 0.0103483 (**now consistent with Garfield**)

diff_coeff_T (mm/sqrt(cm)^2 for 1 kV) 0.013225 ------> 0.0143819 **(now consistent with Garfield)**

To simulate different drift field:

- change manually the two diffusion coefficients. Here a script to compute them according to Garfield simulation: https://github.com/pietro14/digitization/tree/h5/garfield_data
- change manually absorption length (proper value to be fixed as discussed in slides 14-19).

Absorption length

For December 15 2023 (runs 42985-43050) we found an absorption length of 1350 mm is the best value. However, this parameter is highly dependent on impurities in gas, and should be tuned for each period of time (see slide 14-19):

https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

```
'absorption_l'            : 1350.,     # absorption length in [mm]
```

Saturation

Saturation parameters should not depend on the time period

https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

```
# Saturation parameters
'saturation'              : True,      # if 'True' saturation effect is applied on GEM3
'xy_vox_scale'            : 1, # pixel size / xy voxel dimension (MUST be integer AND >= 1)
'z_vox_dim'               : 0.1,       # z voxel size in [mm]
'A'                       : 1,        # free parameter (total scale factor MC/data)
'beta'                    : 0.8e-5,    # saturation parameter
'z_vox_dim'               : 0.1,       # z voxel size in [mm]
'zcloud'          : 20,         # z dimension of the electron cloud in [mm]
```

https://github.com/CYGNUS-RD/digitizationpp/blob/main/src/DigitizationRunner.cxx

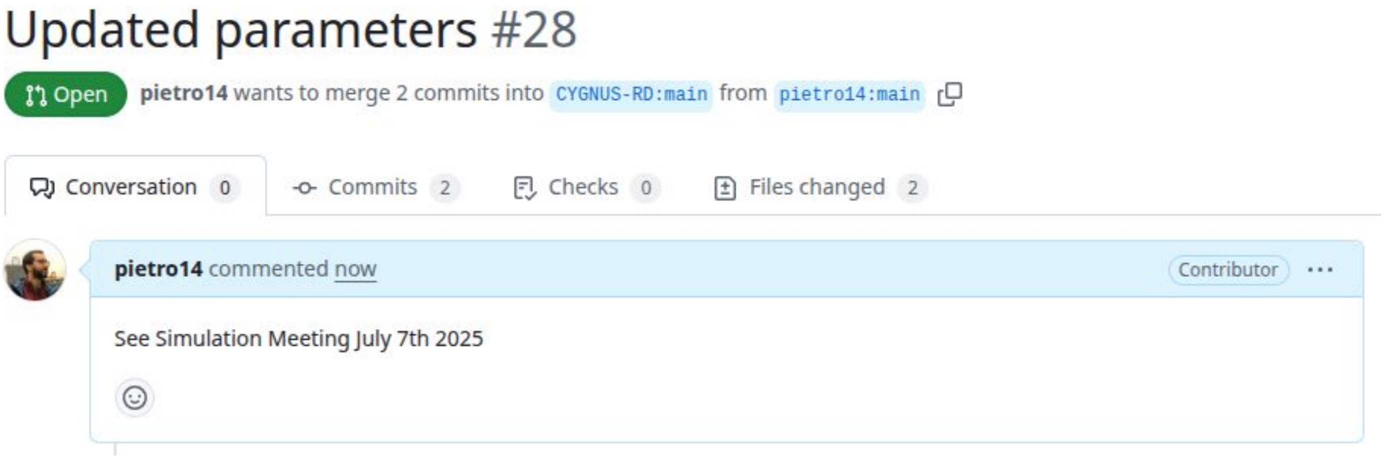
```
void DigitizationRunner::initializeGlobals() {
    GEM1_gain = 0.03 * exp(0.0209 * config.getDouble("GEM1_HV"));
    GEM2_gain = 0.03 * exp(0.0209 * config.getDouble("GEM2_HV"));
    GEM3_gain = 0.03 * exp(0.0209 * config.getDouble("GEM3_HV"));
```

Single gem gain should be adjusted for other time periods (slides 14-19)

Where to find these parameters? Pull Request #28

If you need the most updated version of the digitization with these parameters, you'll find here, once the Pull request is accepted:

https://github.com/CYGNUS-RD/digitizationpp/



15 config/ConfigFile_new.txt

```diff
@@ -1,13 +1,16 @@
 {
 # Detector physics
 ## diffusion parameters from https://arxiv.org/pdf/2007.00608.pdf
- 'diff_const_sigma0T'    : 0.1225,     # diffusion constant [mm]^2
- 'diff_coeff_T'          : 0.013225,   # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV
+ ## for other drift fields see https://github.com/CYGNUS-RD/digitization/tree/master/compute_drift_velocity
+ 'diff_const_sigma0T'    : 0.13475,     # diffusion constant [mm]^2
+ 'diff_coeff_T'          : 0.0143819,   # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV/cm
+ #'diff_coeff_T'          : 0.0198721,   # diffusion parameter [mm/sqrt(cm)]^2 for 500 V/cm
 'diff_const_sigma0L'    : 0.0676,     # diffusion constant [mm]^2
- 'diff_coeff_L'          : 0.00978,    # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV
+ 'diff_coeff_L'          : 0.0103483,    # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV/cm
+ #'diff_coeff_L'          : 0.0152483,    # diffusion parameter [mm/sqrt(cm)]^2 for 500 V/cm
 'ion_pot'               : 0.0350,    # ionization potential for He/CF4 60/40 [keV]
 'photons_per_el'        : 0.07,      # number of photons per electron produced in the avalanche
- 'absorption_l'          : 1400.,     # absorption length in [mm]
+ 'absorption_l'          : 1350.,     # absorption length in [mm]

 # Detector parameters
 'GEM1_HV'               : 420.,      # HV of GEM1
@@ -33,8 +36,8 @@
 'saturation'            : True,      # if 'True' saturation effect is applied on GEM3
 'xy_vox_scale'          : 1, # pixel size / xy voxel dimension (MUST be integer AND >= 1)
 'z_vox_dim'             : 0.1,       # z voxel size in [mm]
- 'A'                     : 1.52,        # free parameter (total scale factor MC/data)
- 'beta'                  : 1.0e-5,    # saturation parameter
+ 'A'                     : 1,         # free parameter (total scale factor MC/data)
+ 'beta'                  : 0.8e-5,    # saturation parameter
```

8 src/DigitizationRunner.cxx

```diff
@@ -75,9 +75,9 @@ void DigitizationRunner::run() {
 }

 void DigitizationRunner::initializeGlobals() {
-    GEM1_gain = 0.0347 * exp(0.0209 * config.getDouble("GEM1_HV"));
-    GEM2_gain = 0.0347 * exp(0.0209 * config.getDouble("GEM2_HV"));
-    GEM3_gain = 0.0347 * exp(0.0209 * config.getDouble("GEM3_HV"));
+    GEM1_gain = 0.03 * exp(0.0209 * config.getDouble("GEM1_HV"));
+    GEM2_gain = 0.03 * exp(0.0209 * config.getDouble("GEM2_HV"));
+    GEM3_gain = 0.03 * exp(0.0209 * config.getDouble("GEM3_HV"));

     extraction_eff_GEM1 = 0.87319885 * exp(-0.002 * config.getDouble("GEM1_HV"));
     extraction_eff_GEM2 = 0.87319885 * exp(-0.002 * config.getDouble("GEM2_HV"));
@@ -90,7 +90,7 @@ void DigitizationRunner::prepareCameraSettings() {
     y_pix = 2304;
     if (camera == "Fusion") {
         x_pix = 2304;
-        optcounts_per_photon = 2.; // apparently calibrated with LEMOn in the past
+        optcounts_per_photon = 4.; // (Fusion sheet) counts/photon = QE (0.8) / e-/count (0.21)
```

And to simulate other periods of time? (e.g. RUN 5 AmBe)

Dotted lines are two simulation

where only GEM_V and

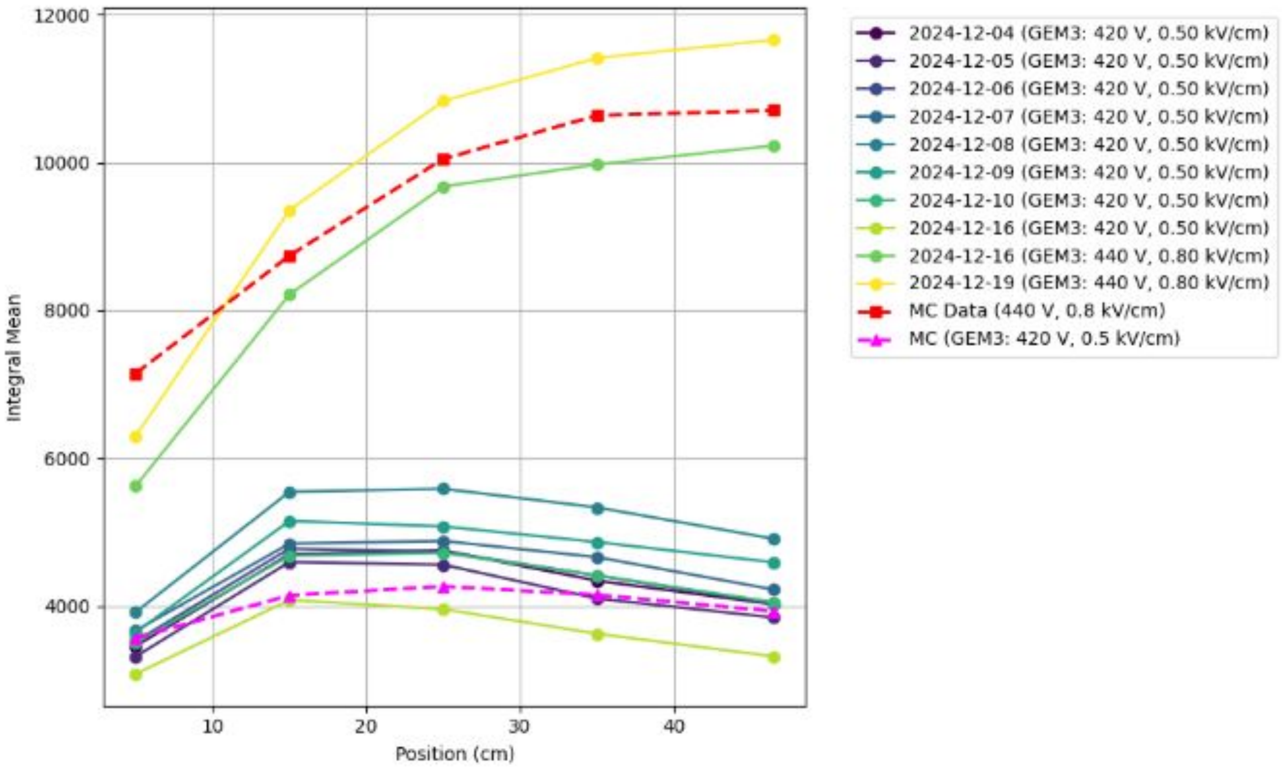
drift field* where changed

The simulation is not

very accurate without

tuning **gain** and

absorption length



Procedure to digitize any data with optimal parameters

Find the closest iron calibration to the data you want to simulate, then do the following:

1. Set GEM_HV in ConfigFile_new.txt
2. Set diff. coeff. in ConfigFile_new.txt (according to drift field)
3. Simulate+digitize+reconstruct iron at step1 and step5
4. Compare sc_integral at STEP1 between data/MC: adjust gain in DigitizationRunner.cxx
5. Compare sc_integral at STEP5 between data/MC: adjust abs. length in DigitizationRunner.cxx

Repeat from 3. to 5. until sc_integral is in data and mc are "reasonably similar" (up to you how you want to quantify this)

1. Set GEM_HV in ConfigFile_new.txt

https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

```
# Detector parameters
'GEM1_HV'                    : 420.,      # HV of GEM1
'GEM2_HV'                    : 420.,      # HV of GEM2
'GEM3_HV'                    : 420.,      # HV of GEM3
```

Consider that we usually operated at 440 V for each GEM in all RUNS, but RUN 5, where the operative voltages are 420 V for each GEM

2. Set diffusion coeff. in ConfigFile_new.txt

https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

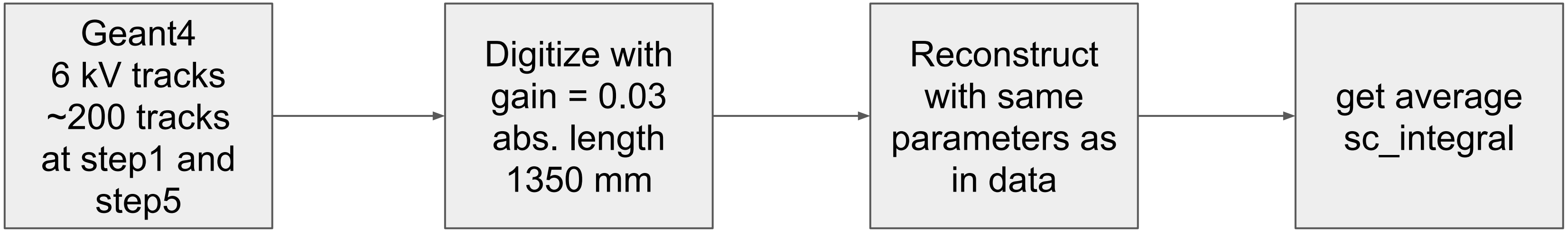
```
# Detector physics
## diffusion parameters from https://arxiv.org/pdf/2007.00608.pdf
## for other drift fields see https://github.com/CYGNUS-RD/digitization/tree/master/compute_drift_velocity
'diff_const_sigma0T'    : 0.13475,     # diffusion constant [mm]^2
'diff_coeff_T'          : 0.0143819,   # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV/cm
#'diff_coeff_T'          : 0.0198721,   # diffusion parameter [mm/sqrt(cm)]^2 for 500 V/cm
'diff_const_sigma0L'    : 0.0676,     # diffusion constant [mm]^2
'diff_coeff_L'          : 0.0103483,    # diffusion parameter [mm/sqrt(cm)]^2 for 1 kV/cm
#'diff_coeff_L'          : 0.0152483,   # diffusion parameter [mm/sqrt(cm)]^2 for 500 V/cm
```

For different drift fields other than 1 kV/cm and 0.5 kV/cm use this script to compute **diff_coeff_T** and **diff_coeff_L**, according to Garfield simulation:
https://github.com/pietro14/digitization/tree/h5/garfield_data

While **diff_const_sigma0T** and **diff_const_sigma0L** should remain constant in theory.

3. Simulate+digitize+reconsctruct iron at step1 and step5

Simulate ~200 tracks of 6 keV electrons in Geant4 at 5 cm (step1) and 46.5 cm (step5) from the GEM plane; **digitize** them **(with gain 0.03 and abs. length 1350 mm)**; **reconstruct** the tracks; and **measure the average** sc_integral.



4. Compare sc_integral at STEP1 between data/MC

Do the same analysis on simulated iron tracks and real tracks. Get average sc_integral for data and simulation for step1

sc_integral in MC is **HIGHER** than in data → **DECREASE** single gem gain: < 0.03

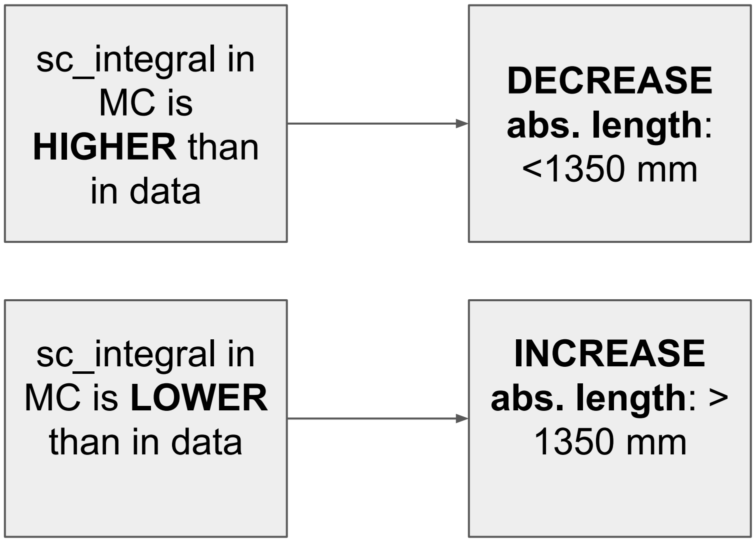
sc_integral in MC is **LOWER** than in data → **INCREASE** single gem gain: > 0.03

https://github.com/CYGNUS-RD/digitizationpp/blob/main/src/DigitizationRunner.cxx

```
void DigitizationRunner::initializeGlobals() {
    GEM1_gain = 0.03 * exp(0.0209 * config.getDouble("GEM1_HV"));
    GEM2_gain = 0.03 * exp(0.0209 * config.getDouble("GEM2_HV"));
    GEM3_gain = 0.03 * exp(0.0209 * config.getDouble("GEM3_HV"));
```

5. Compare sc_integral at STEP5 between data/MC

Do the same analysis on simulated iron tracks and real tracks. Get average sc_integral for data and simulation for step5



https://github.com/CYGNUS-RD/digitizationpp/blob/main/config/ConfigFile_new.txt

```
'absorption_l'          : 1350.,     # absorption length in [mm]
```

Conclusions

To reproduce December 15, 2023 Iron Scans (Run 4, runs 42985–43050)

- **Accept the pull request #28** on GitHub and you are ready to go

For Other Time Periods

- **You have to adjust gain and absorption length** by comparing against iron scans from your target period (slides 14-19)

- **If you don’t adjust gain and absorption length for each period:**
  a. **you won’t achieve better results than with the parameters used so far**
  b. **data/mc comparison will not as good as for December 2023 data (not at 5% level)**