Work towards Realistic B0

Sasha Bylinkin
The University of Kansas

September 9th 2021

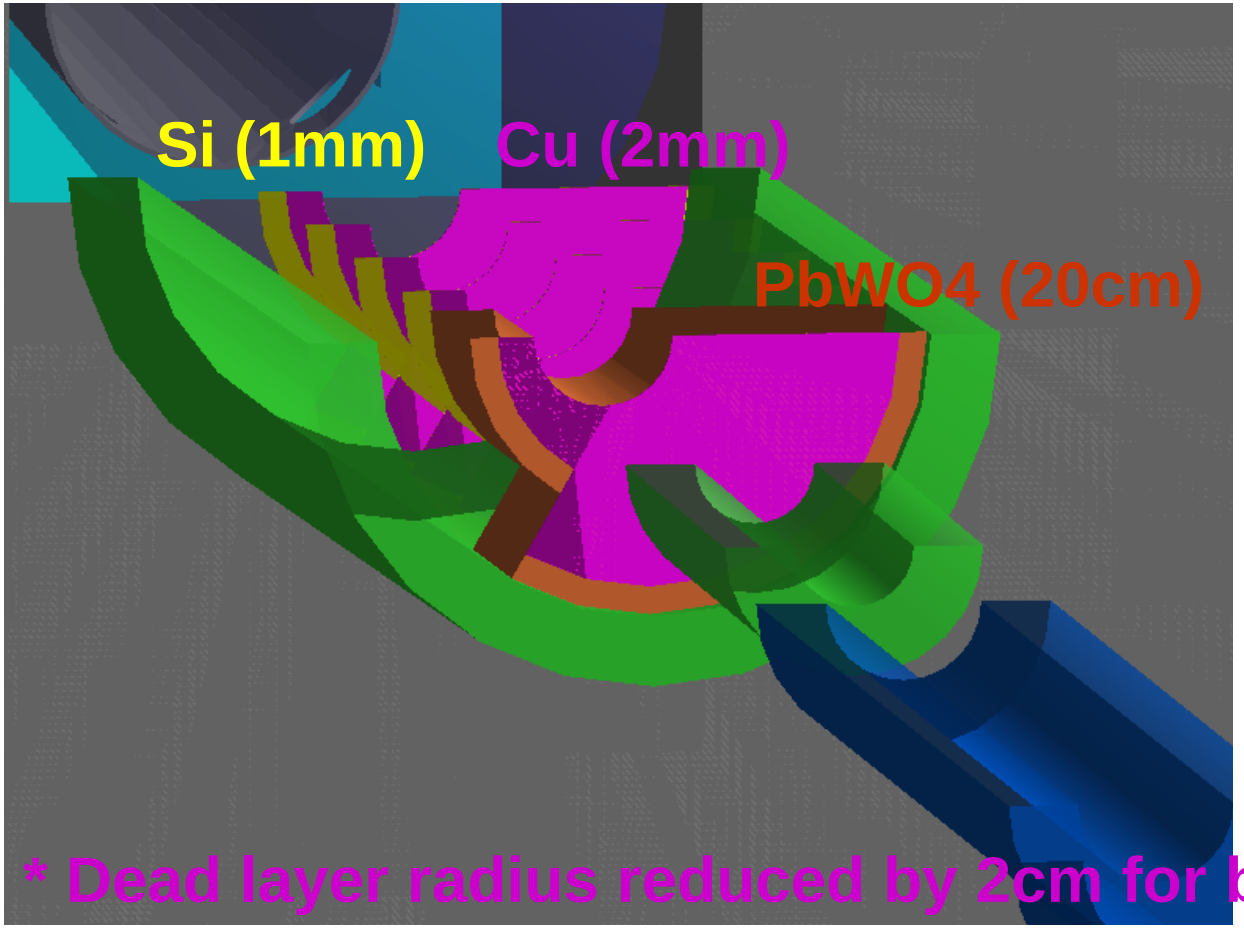
Realistic B0

- Dead layers (cooling, boards, cables) – 2mm copper after Silicon
- Realistic B0 shape (Packman)
- Beampipes placement

- Code available at github

Realistic B0

- I modified the *EICG4B0ZDC* class written by Quan for PbWO4 to include layers of Si and Cu with the same geometry into one SuperDetector with 10 layers
- I merged this class with *diff_tagg_ana* by Bill to produce one library during compilation
- All shape parameters are controlled from the modified *G4_hFarFwdBeamLine_EIC.C* so no recompilation is needed to change them



Controlled shape parameters (with current values):

Layer Length (0.1, 0.2, 20cm)
Layer Material (Si, Cu, PbWO4)
Radius (20cm)
ΔRadius (5cm)
Spanning Angle (240°)
Beampipe position (-3.4 cm)

Github instructions

- Git clone https://github.com/ECCE-EIC/macros
- Git clone https://github.com/abylinkin/EIC_B0
- cd EIC_B0/source/

  chmod +x autogen.sh

  mkdir build

  cd build

  ../autogen.sh –prefix=$MYINSTALL

  make install

Github instructions

- cd ../../../macros/detector/EICDetector/
- cp ../../../EIC_B0/macros/*.C .
- Root -l

  .x Fun4All_G4_EICDetector1.C

Github instructions

- ls EIC_B0/macros/
  - Fun4All_G4_EICDetector1.C

    same macro with USER and DISPLAY enabled
  - G4Setup_EICDetector.C

    includes new version:
  - G4_hFarFwdBeamLine_EIC_Realistic.C

• G4_hFarFwdBeamLine_EIC_Realistic.C

```
1  #ifndef MACRO_G4HFARFWDBEAMLINE_EIC_C
2  #define MACRO_G4HFARFWDBEAMLINE_EIC_C
3
4  #include <GlobalVariables.C>
5
6  #include <g4detectors/BeamLineMagnetSubsystem.h>
7  #include <g4detectors/PHG4BlockSubsystem.h>
8  #include <g4detectors/PHG4ConeSubsystem.h>
9  #include <g4detectors/PHG4CylinderSubsystem.h>
10
11 #include <eicg4zdc/EICG4ZDCHitTree.h>
12 #include <eicg4zdc/EICG4ZDCNtuple.h>
13 #include <eicg4zdc/EICG4ZDCSubsystem.h>
14
15 #include <difftagganab0/EICG4B0ZDCSubsystem.h>
16
17 #include <g4main/PHG4Reco.h>
18
19 #include <TSystem.h>
20
21 R__LOAD_LIBRARY(libg4detectors.so)
22
23 R__LOAD_LIBRARY(libdifftagganab0.so)
24
25 float PosFlip(float pos);
26 float AngleFlip(float angle);
27 float MagFieldFlip(float Bfield);
```

New Class Reference

New Library

•G4_hFarFwdBeamLine_EIC_Realistic.C

```
366
367     auto *detRP = new PHG4CylinderSubsystem(Form("rpTruth_%d", 2*i), 2*i);
368     detRP->SuperDetector("rpTruth");
369     detRP->set_double_param("place_x", PosFlip(rp_xCent[i]));
370     detRP->set_double_param("place_y", 0);
371     detRP->set_double_param("place_z", rp_zCent[i] - hFarFwdBeamLine::enclosure_center);
372     detRP->set_double_param("rot_y", AngleFlip(0.047 * TMath::RadToDeg()));
373     detRP->set_double_param("radius", 0);
374     detRP->set_double_param("thickness", 25);  // This is intentionally made large 25cm radius
375     detRP->set_double_param("length", 0.03);
376     detRP->set_string_param("material", "G4_Si");
377     detRP->OverlapCheck(overlapCheck);
378     detRP->SetMotherSubsystem(hFarFwdBeamLine::hFarFwdBeamLineEnclosure);
379
380     detRP->SetActive();
381     if (verbosity)
382       detRP->Verbosity(verbosity);
383     g4Reco->registerSubsystem(detRP);
384 /* *** Section for Realistic Roman Pots *** */
385     auto *detRPe = new PHG4CylinderSubsystem(Form("rpTruth_%d", 2*i+1), 2*i+1);
386     detRPe->SuperDetector("rpTruth");
387     detRPe->set_double_param("place_x", PosFlip(rp_xCent[i]));
388     detRPe->set_double_param("place_y", 0);
389     detRPe->set_double_param("place_z", rp_zCent[i] - hFarFwdBeamLine::enclosure_center + .1015);
390     detRPe->set_double_param("rot_y", AngleFlip(0.047 * TMath::RadToDeg()));
391     detRPe->set_double_param("radius", 0);
392     detRPe->set_double_param("thickness", 25);  // This is intentionally made large 25cm radius
393     detRPe->set_double_param("length", 0.2);
394     detRPe->set_string_param("material", "G4_Cu");
395     detRPe->set_color(1,0,1,.8);
396     detRPe->OverlapCheck(overlapCheck);
397     detRPe->SetMotherSubsystem(hFarFwdBeamLine::hFarFwdBeamLineEnclosure);
398
399     detRPe->SetActive();
400     if (verbosity)
401       detRPe->Verbosity(verbosity);
402     g4Reco->registerSubsystem(detRPe);
403 /* End */
```

Section for dead material in RPs

•G4_hFarFwdBeamLine_EIC_Realistic.C

Section for Realistic B0

```
406     const int b0DetNr = 4;
407     const double b0Mag_zCent = 590;
408     const double b0Mag_zLen = 120;
409     const double b0Cu_zLen = .2; //B0 dead material length
410     const double b0Si_zLen = .1; //B0 Si length
411     const double b0Ecal_zLen = 20.; //B0 Ecal length
412     const double pipe_hole = 5.0; //detector cut off for beam pipe
413     const double pipe_x = -3.4; //pipe hole position
414     const double d_radius = 7.0; //detector cut off Packman
415     const double b0_radius = 20.0; //outer radius of B0-detector
416     const double spanning_angle = 240; //spanning angle Packman
417     const double b0Ecal_z = 48;
418     double start_angle = spanning_angle - 360; //start angle Packman
419     for (int i = 0; i < b0DetNr; i++)
420     {
421       auto *detB0 = new EICG4B0ZDCSubsystem(Form("b0Truth_%d", 2*i), 2*i);
422       detB0->SuperDetector("b0Truth");
423       detB0->set_double_param("place_x", 0);
424       detB0->set_double_param("place_y", 0);
425   //  detB0->set_int_param("ispipe", 0); //for future pipe implementation
426       detB0->set_double_param("pipe_hole", pipe_hole);
427       detB0->set_double_param("outer_radius", b0_radius);
428       detB0->set_double_param("d_radius", d_radius);
429       detB0->set_double_param("length", b0Si_zLen);
430       detB0->set_string_param("material", "G4_Si");
431       detB0->set_double_param("detid",2*i);
432       detB0->set_double_param("startAngle",start_angle);
433       detB0->set_double_param("spanningAngle",spanning_angle);
434       detB0->set_double_param("pipe_x", pipe_x);
435       detB0->set_double_param("pipe_y", 0);
436       detB0->set_double_param("pipe_z", 0);
437       detB0->set_double_param("place_z", b0Mag_zLen / (b0DetNr + 1) * (i - b0DetNr / 2));  // rel
438       detB0->SetActive(true);
439       if (verbosity)
440         detB0->Verbosity(verbosity);
441       detB0->OverlapCheck(overlapCheck);
442       detB0->SetMotherSubsystem(hFarFwdBeamLine::B0Magnet);
443       g4Reco->registerSubsystem(detB0);
444
445       auto *detB0e = new EICG4B0ZDCSubsystem(Form("b0Truth_%d", 2*i+1), 2*i+1);
446       detB0e->SuperDetector("b0Truth");
447   //  detB0e->set_int_param("ispipe", 0); //for future pipe implementation
448       detB0e->set_double_param("pipe_hole", pipe_hole);
449       detB0e->set_double_param("place_x", 0);
```

```
458       detB0e->set_double_param("detid",2*i+1);
459       detB0e->set_double_param("startAngle",start_angle);
460       detB0e->set_double_param("spanningAngle",spanning_angle);
461       detB0e->set_double_param("place_z", (b0Mag_zLen / (b0DetNr + 1) * (i - b0DetNr / 2
462       detB0e->SetActive(true);
463       if (verbosity)
464         detB0e->Verbosity(verbosity);
465       detB0e->OverlapCheck(overlapCheck);
466       detB0e->SetMotherSubsystem(hFarFwdBeamLine::B0Magnet);
467       g4Reco->registerSubsystem(detB0e);
468     }
469
470     auto *B0Ecal = new EICG4B0ZDCSubsystem("B0Ecal", 2*b0DetNr);
471     B0Ecal->SuperDetector("b0Truth");
472   //  B0Ecal->set_int_param("ispipe", 0); //for future pipe implementation
473     B0Ecal->set_double_param("pipe_hole", pipe_hole);
474     B0Ecal->set_double_param("place_x", 0);
475     B0Ecal->set_double_param("place_y", 0);
476     B0Ecal->set_double_param("place_z", b0Ecal_z);
477     B0Ecal->set_double_param("pipe_x", pipe_x);
478     B0Ecal->set_double_param("pipe_y", 0);
479     B0Ecal->set_double_param("pipe_z", 0);
480     B0Ecal->set_double_param("length", b0Ecal_zLen);
481     B0Ecal->set_double_param("outer_radius", b0_radius);
482     B0Ecal->set_double_param("d_radius", d_radius);
483     B0Ecal->set_string_param("material", "G4_PbWO4");
484     B0Ecal->set_double_param("startAngle",start_angle);
485     B0Ecal->set_double_param("spanningAngle",spanning_angle);
486     B0Ecal->set_double_param("detid",2*b0DetNr);
487     B0Ecal->SetActive(true);
488     if (verbosity)
489       B0Ecal->Verbosity(verbosity);
490     B0Ecal->OverlapCheck(overlapCheck);
491     B0Ecal->SetMotherSubsystem(hFarFwdBeamLine::B0Magnet);
492     g4Reco->registerSubsystem(B0Ecal);
493
494     auto *B0Ecale = new EICG4B0ZDCSubsystem("B0Ecale", 2*b0DetNr + 1);
495     B0Ecal->SuperDetector("b0Truth");
496   //  B0Ecale->set_int_param("ispipe", 0); //for future pipe implementation
497     B0Ecale->set_double_param("pipe_hole", pipe_hole);
498     B0Ecale->set_double_param("place_x", 0);
499     B0Ecale->set_double_param("place_y", 0);
500     B0Ecale->set_double_param("place_z", b0Ecal_z + (b0Ecal_zLen + b0Cu_zLen)/2);
501     B0Ecale->set_double_param("pipe_x", pipe_x);
502     B0Ecale->set_double_param("pipe_y", 0);
```

Conclusion

- The code for the Realistic B0 is available at github and instructions for running are provided
- Tests and comments are welcome

Thank you very much for your attention!