

A brief guide to MPDROOT

Luis Valenzuela-Cázares for MExNICA Collaboration
luisvc3737@gmail.com

March 2020

MPDROOT is a detector simulation and reconstruction, and data analysis framework developed for the MPD-NICA experiment and based on FairRoot. This software can be installed by following the instructions at the MPD website [2]. MPDROOT simulates particles and interactions, detectors geometries, interactions of particles with matter and electromagnetic fields. Detailed information is available at the MPD website [13] [15]. Updates to this document, macros, detector classes, and other files can be consulted at the GitLab of the MeXNICA Physics Analysis and Offline group [3].

Installing BE-BE

After MPDROOT installation is finished, we can proceed to add BE-BE, which is called *bmd* (beam monitoring detector) in the MPDROOT context. First, the BE-BE files should be downloaded from the GitLab repositories [3] [4], and we have to add them in two different directories. The geometry root file *bbc_hex_5cm_NDetScin_v1.root* must be added to the directory *.../mpdroot/geometry/*, and the rest of the files (*BmdDetector.cxx*, *BmdDetector.h*, *BmdPoint.cxx*, *BmdPoint.h*) will replace their old versions at */mpdroot/bmd/*.

Libraries are loaded in the macro *mpdloadlibs.C*, which is located at *...mpdroot/macro/mpd/*. We have to add *bmd* libraries with next code line:

```
gSystem->Load("libbmd");
```

Comments: Due that MPDROOT is constantly being upgraded, sometimes it is necessary to modify files according to the software changes. Usually, you will know what to correct given the errors after compiling or running the transporting macro. On the other hand, even if the compilation does not have errors, occasionally some libraries are not created correctly, and it could work to delete the *build* directory and to compile MPDROOT again. It is also important to check out the commits in the NICA GitLab [8] to see the new changes.

MPDROOT simulation

In order to transport the generated particles through the detectors, we have to run the macro *runMC.C* located at *...mpdroot/macro/mpd/*. Before running this macro it is necessary to adjust some code lines, and to choose the detectors in the *geometry_stage1.C* macro, located at the same directory. All the detectors are activated as default, but we only need BE-BE or MBB. The first thing to do is to include BE-BE with next lines in *geometry_stage1.C*:

```

29
30 // Detectors
31 gSystem->Load("libtpc");
32 gSystem->Load("libTof");
33 gSystem->Load("libEtof");
34 gSystem->Load("libEmc");
35 gSystem->Load("libZdc");
36 gSystem->Load("libSts");
37 gSystem->Load("libCpc");
38 gSystem->Load("libStrawECT");
39 gSystem->Load("libStrawendcap");
40 gSystem->Load("libFfd");
41 //gSystem->Load("libFsa");
42 gSystem->Load("libBbc");
43 gSystem->Load("libbmd");
44 //gSystem->Load("libNDet");
45 //gSystem->Load("libStt");
46 //gSystem->Load("libSft");
47

```

Figure 1: MPDROOT libraries.

```

34
35 R_ADD_INCLUDE_PATH($VMCWORKDIR)
36 #include "macro/mpd/mpdloadlibs.C"
37 #include "macro/mpd/geometry_stage1.C"
38 // #include "macro/mpd/geometry_v2.C"
39
40 #define URQMD // Choose generator: URQMD VHLLE FLUID PART ION BOX HSD LAQGSMD HADGEN
41 #define GEANT3 // Choose: GEANT3 GEANT4
42

```

Figure 2: It is necessary to specify both generator and transport package for simulation.

```

FairDetector *Bmd = new BmdDetector("BMD", kTRUE );
Bmd->SetGeometryFileName("bbc_hex_5cm_NDetScin_v1.root");
fRun->AddModule(Bmd);

```

The rest of the MPD must be turned off by commenting the *fRun->AddModule()*; line for each detector, but basic MPD elements should remain activated for the simulation. Specifically the cave, pipe, and magnet, as we can see in figure (3).

Note: For a more realistic simulation of the MPD the rest of the detectors can be activated.

The next step is to prepare the transport macro by specifying the inputfile path of generated particles, the particle generator, and the transport package. The supported generators are UrQMD, LQGSMD and BOX. The supported transport packages are geant3 and geant4. In the figure (2) we can see an example for choosing geant3 and UrQMD.

```

1 void geometry_stagel(FairRunSim *fRun)
2 {
3     // Set Material file Name
4     fRun->SetMaterials("media.geo");
5
6     // Create and add detectors
7     //-----
8     FairModule *Cave= new FairCave("CAVE");
9     Cave->SetGeometryFileName("cave.geo");
10    fRun->AddModule(Cave);
11
12    FairModule *Pipe= new FairPipe("PIPE");
13    Pipe->SetGeometryFileName("pipe.geo");
14    fRun->AddModule(Pipe);
15
16    FairModule *Magnet= new FairMagnet("MAGNET");
17    Magnet->SetGeometryFileName("magnet_v5.root");
18    fRun->AddModule(Magnet);
19
20    FairDetector *Bmd = new BmdDetector("BMD", kTRUE );
21    Bmd->SetGeometryFileName("bbc_hex_5cm_NDetScin_v1.root");
22    fRun->AddModule(Bmd);
23
24    //FairDetector *Ffd = new MpdFfd("FFD", kTRUE );
25    //Ffd->SetGeometryFileName("ffd.geo");
26    //fRun->AddModule(Ffd);
27
28    FairDetector *Tpc = new TpcDetector("TPC", kTRUE);
29    Tpc->SetGeometryFileName("tpc_v8.root");
30    // fRun->AddModule(Tpc);
31
32    FairDetector *Tof= new MpdTof("TOF", kTRUE );
33    Tof->SetGeometryFileName("tof_v7.root");
34    // fRun->AddModule(Tof);
35

```

Figure 3: MPD elements.

BBC version

An independent version of BEBE can be installed in the same way that we did before. The files can be downloaded from the GitLab repository [9] [10]. Now, instead of copying the files at */mpdroot/bmd*, we will put them at the */mpdroot/bbc* directory. The main differences between these versions are the hits selection considerations. While for the BMD, a hit is any charged particle that goes through a BEBE-cell completely, for the BBC, a hit is any nonzero energy charged particle that arrived in any cell.

- The bmd flaw is that we will lose some particles that deposite all their energy at the BEBE, but they are detected.
- The bmd advantage is that we are sure that selected hits have a high probability of being detected.
- The bbc flaw is that we will select some particles that have a low probability of being detected.
- The bbc advantage is that we will recover some particles.

We can see and change these considerations in *BmdDetector.cxx* and *MpdBbc.cxx* files.
For bbc

```
if(gMC->IsTrackEntering())
```

For bmd

```
if( fELoss > 0. && (gMC->IsTrackExiting() || gMC->IsTrackStop() ||  
gMC->IsTrackDisappeared()) && gMC->TrackCharge() ) {  
  
    BmdPoint* p=AddHit(fTrackID, fVolumeID, fPos.Vect(), fMom.Vect(), fTime,  
fLength, fELoss);  
p->SetStep(gMC->TrackStep());  
((FairStack*)gMC->GetStack())->AddPoint(kBMD);  
ResetParameters();  
  
}
```

Hits can be recovered if we only consider next lines:

```
fELoss    = 0.;  
fTime     = gMC->TrackTime() * 1.0e09;
```

Mini BE-BE

We can proceed to add mini BE-BE, which is called MBB(inner beam-beam monitoring detector) in the MPDROOT context. First, the MBB files should be downloaded from the GitLab repository [11] [12], and then add them to two different directories. The root file *mbbv1.root* must be added to the directory *.../mpdroot/geometry/*, and the rest of the files (*MbbDetector.cxx*, *MbbDetector.h*, *MbbPoint.cxx*, *MbbPoint.h*, *MbbGeo.cxx*, *MbbGeo.h*, *MbbGeoPar.cxx*, *MbbGeoPar.h*, *mbbLinkDef.h*, and *CMakeLists.txt*) should be located at */mpdroot/mbb/*.

Libraries are loaded in macro *mpdloadlibs.C*, which is located at *...mpdroot/macro/mpd/*. We have to add mbb libraries with next code line:

```
gSystem->Load("libmbb");
```

Next thing to do is to specify mbb directory address by adding the line

```
add_subdirectory (mbb)
```

in the file *CMakeLists.txt* located at *.../mpdroot/*.

MBB should be added on *MpdDetectorList* file located at *.../mpdroot/mcstack* as we see in the next line:

```
enum DetectorId {kSTS, kTPC, kTOF, kETOF, kFFD, kECT, kECAL,  
kNDET, kCPC, kBBC, kZDC, kFSA, kSFT, kBMD, kMBB, kRECOIL};
```

Before running the transport macro, we need to include MBB in *geometry_stage1.C*, by adding the next lines:

```
FairDetector *Mbb = new MbbDetector("MBB", kTRUE);  
Mbb->SetGeometryFileName("mbb_v1.root");  
fRun->AddModule(Mbb);
```

Analysis macros

Analysis macros can be downloaded from the GitLab repositories [5] [6] [7]. With them, we can produce a characterization of BE-BE and MBB detectors through the study of observables such as multiplicity, p_T , η , *Energy*, and time of flight. We can also produce centrality determination studies.

- *readOutput.C*: η , T , multiplicity, energy, and time of flight distributions and efficiency.
- *centrality.C*: impact parameter centrality classes, multiplicity vs impact parameter correlation, XY distribution, multiplicity distribution.
- *centralityclasses.C*: NBD multiplicity fit, centrality classes.

These macros can be used with both BMD and BBC, and they can be adapted for MBB as well. We can study several observables of these detectors through the classes *BmdPoint*, *MpdBbcPoint*, and *MbbPoint*. We can reproduce Monte Carlo studies with *MCTrack*, and study each event condition through *MCEventHeader* too.

More macros can be found at */mpdroot/macro/*. Additional information is available at the quick guide *README* located at *mpdroot/macro/physical_analysis/Flow*. Finally, more macros related to the MPD experiment can be found at the MPD website [14].

NOTE: On recent MPDROOT updates (January 2020), some changes are needed on the analysis macros: it is necessary to change the tree name from *cbmsim* to *mpdsim*.

References

- [1] MPDROOT manual <https://gitlab.com/mexnica-physics-analysis-and-offline-group/mpdrootnotes>
- [2] MPDROOT <http://mpd.jinr.ru/howto-install/>
- [3] BEBE installation files <https://gitlab.com/mexnica-physics-analysis-and-offline-group/be-be-classes>
- [4] BEBE root file geometry <https://gitlab.com/mexnica-physics-analysis-and-offline-group/bebe-geometries>
- [5] BEBE Analysis macros <https://gitlab.com/mexnica-physics-analysis-and-offline-group/bebe-macros-analysis>
- [6] Centrality macros <https://gitlab.com/mexnica-physics-analysis-and-offline-group/centrality-studies>
- [7] MBB Analysis macros <https://gitlab.com/mexnica-physics-analysis-and-offline-group/mbb-macros-analysis>
- [8] MPDROOT commits <https://git.jinr.ru/nica/mpdroot/commits/>
- [9] BBC installation files <https://gitlab.com/mexnica-physics-analysis-and-offline-group/bbc-classes>
- [10] BBC geometry root files <https://gitlab.com/mexnica-physics-analysis-and-offline-group/bbc-geometries>
- [11] MBB installation files <https://gitlab.com/mexnica-physics-analysis-and-offline-group/mbb-classes>
- [12] MBB geometry root files <https://gitlab.com/mexnica-physics-analysis-and-offline-group/mbb-geometries>
- [13] Simulation of the MPD experiment <http://mpd.jinr.ru/simulation-of-the-mpd-experiment/>
- [14] MPDROOT macros <http://mpd.jinr.ru/mpdroot-macros/>
- [15] MPDROOT Start Guide <http://mpd.jinr.ru/mpdroot-start-guide/>
- [16] GitLab MeXNICA Physics Analysis and Offline group <https://gitlab.com/mexnica-physics-analysis-and-offline-group>