

MpdDst

or

How is it possible?



Grigory Nigmatkulov

National Research Nuclear University MEPhI

Tracks

Three types of tracks are stored in MpdDst [*types (path to the class)*]:

1. GlobalTracks (mpdbase/MpdTrack)
2. MCTracks (mcstack/FairMCTrack)
3. TpcKalmanTrack (kalman/MpdTpcKalmanTrack)

Which one(s) should we use?

Simple (?) start with tracks

Code example:

```
// Activate branches
MpdEvent *mpdEvent = nullptr;
dstTree->SetBranchAddress("MPDEvent.", &mpdEvent);
TClonesArray *mcTracks = nullptr;
dstTree->SetBranchAddress("MCTrack", &mcTracks);
TClonesArray *kalmanTracks = nullptr;
dstTree->SetBranchAddress("TpcKalmanTrack", &kalmanTracks);
for (int iEvent = 0; iEvent < events; iEvent++) {
    // Retrieve event
    dstTree->GetEntry(iEvent);
    // Return number of global tracks in the current event
    int nGlobalTracks = mpdEvent->GetGlobalTracks()->GetEntriesFast();
    // Return number of MC tracks
    int nMcTracks = mcTracks->GetEntries();
    // Return number of TpcKalmanTracks
    int nTpcKalmanTracks = kalmanTracks->GetEntriesFast();
    std::cout << Form("Track number [global/mc/kalman]: [ %d / %d / %d ]\n",
                    nGlobalTracks, nMcTracks, nTpcKalmanTracks);
    ...
}
```

Simple (?) start with tracks

Working on event [300/500]

Track number [global/mc/kalman]: [161 / 58878 / 161]

Track number [global/mc/kalman]: [1103 / 59279 / 1103]

Track number [global/mc/kalman]: [1126 / 62431 / 1126]

Track number [global/mc/kalman]: [1137 / 64512 / 1137]

Track number [global/mc/kalman]: [1120 / 60981 / 1120]

Track number [global/mc/kalman]: [1168 / 63434 / 1168]

Track number [global/mc/kalman]: [1206 / 62568 / 1206]

Track number [global/mc/kalman]: [1229 / 68316 / 1229]

Track number [global/mc/kalman]: [1170 / 62133 / 1170]

Track number [global/mc/kalman]: [1174 / 60244 / 1174]

Track number [global/mc/kalman]: [1118 / 60818 / 1118]

Track number [global/mc/kalman]: [1141 / 60772 / 1141]

Track number [global/mc/kalman]: [1192 / 63392 / 1192]

Track number [global/mc/kalman]: [1201 / 63914 / 1201]

Track number [global/mc/kalman]: [1175 / 62083 / 1175]

Track number [global/mc/kalman]: [1111 / 66221 / 1111]


Need explanation!

What is stored as DCA in global track?

MpdFillDstTask.cxx

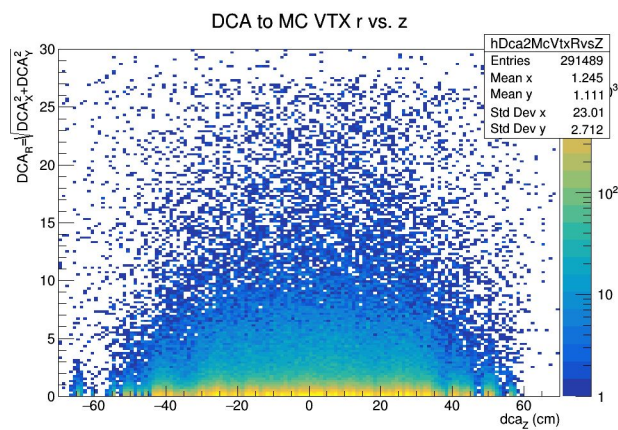
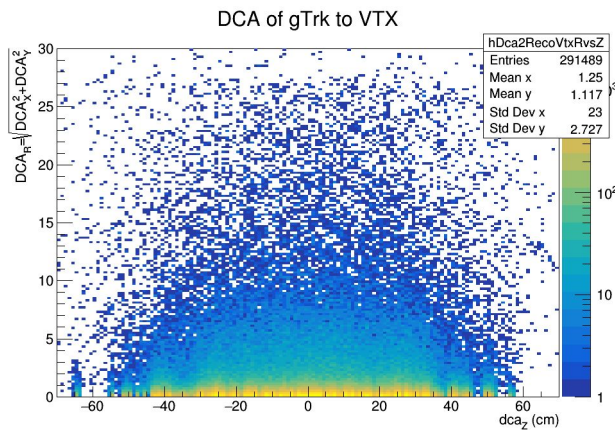
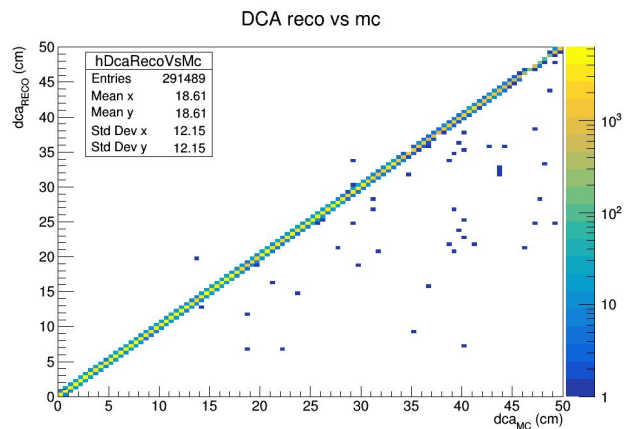
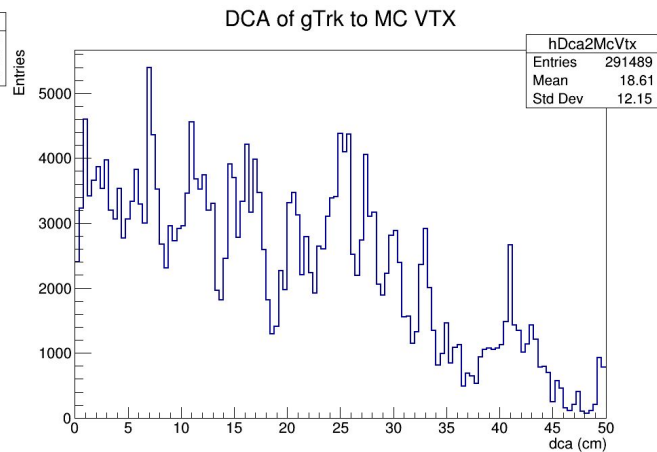
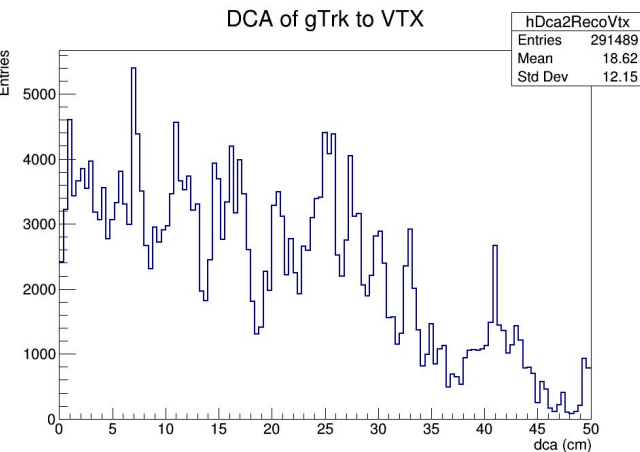
```
Double_t phi = kftrack->GetParam(0) / kftrack->GetPosNew();
track->SetFirstPointX(kftrack->GetPosNew() * TMath::Cos(phi)); // closest to beam line
track->SetFirstPointY(kftrack->GetPosNew() * TMath::Sin(phi));
track->SetFirstPointZ(kftrack->GetParam(1));

track->SetLastPointX(0.); // AZ - currently not available
track->SetLastPointY(0.); // AZ - currently not available
track->SetLastPointZ(0.); // AZ - currently not available
FillTrackDCA(track, &recoVertex, &mcVertex);
FillTrackPID(track);
FillTrackTpcHits(i, track);
```



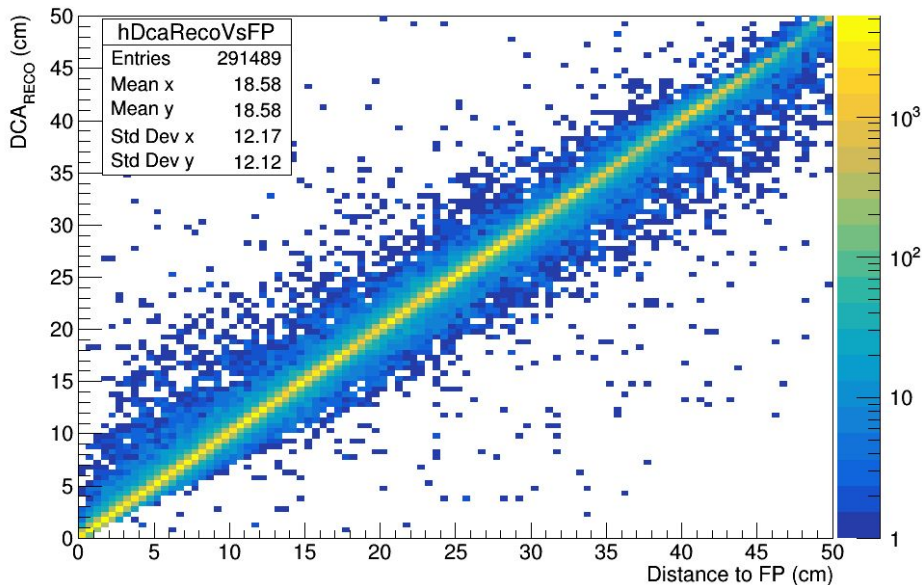
```
void MpdFillDstTask::FillTrackDCA(MpdTrack* track, TVector3 *recoVertex, TVector3 *mcVertex) {
    MpdHelix helix = track->GetHelix();
    Double_t path_at_mcVertex;
    Double_t path_at_recoVertex;
    path_at_mcVertex = helix.pathLength(*mcVertex);
    path_at_recoVertex = helix.pathLength(*recoVertex);
    TVector3 DCA_MC = helix.at(path_at_mcVertex);
    TVector3 DCA_RECO = helix.at(path_at_recoVertex);
    // set dca global as dca to MC vertex DW
    track->SetDCAGlobalX(DCA_MC.X()-mcVertex->X());
    track->SetDCAGlobalY(DCA_MC.Y()-mcVertex->Y());
    track->SetDCAGlobalZ(DCA_MC.Z()-mcVertex->Z());
    // set dca as dca to reconstructed vertex DW
    track->SetDCAX(DCA_RECO.X()-recoVertex->X());
    track->SetDCAY(DCA_RECO.Y()-recoVertex->Y());
    track->SetDCAZ(DCA_RECO.Z()-recoVertex->Z());
}
```

Lets take a closer look

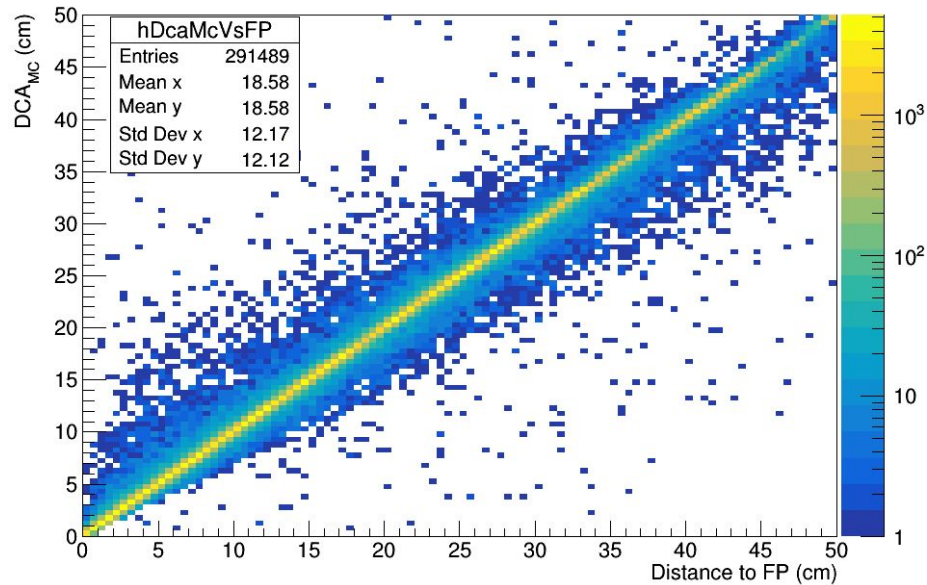


Lets take a closer look

DCA reco vs. distance to the first point



DCA MC vs. distance to the first point



The idea was simple. Take first point and momentum there (fitted point with the fitted momentum), make helix, find DCA of the helix to the primary vertex, take this point for the starting point (i.e. can estimate DCA) and look at the momentum of the track at this point. Momentum components change along the helix.

Helix parameterization

The way Zinchenko does it:

```
MpdMiniPhysicalHelix makeHelix(const MpdKalmanTrack *tr) {  
    const Double_t F_CUR0 = 0.3 * 0.01 * 5 / 10; // 5kG  
    double r = tr->GetPosNew();  
    double phi = tr->GetParam(0) / r;  
    double x = r * TMath::Cos(phi);  
    double y = r * TMath::Sin(phi);  
    double dip = tr->GetParam(3);  
    double cur = F_CUR0 * TMath::Abs(tr->GetParam(4));  
    TVector3 o(x, y, tr->GetParam(1));  
    int h = (Int_t) TMath::Sign(1.1, tr->GetParam(4));  
    MpdMiniPhysicalHelix helix(cur, dip, tr->GetParam(2) - TMath::PiOver2() * h, o, h);  
    return helix;  
}
```

```
MpdMiniPhysicalHelix gHelix = makeHelix(kalmanTrack);  
double pathLength = gHelix.pathLength(recoVtxPos);  
TVector3 gDca;  
gDca = gHelix.at(pathLength);  
gDca -= recoVtxPos;
```

I would do it using momentum.
Solutions **MUST** be equal up to the
double rounding

```
// Obtain helix from Kalman track  
MpdMiniPhysicalHelix oHelix(mpdTrackMom, firstPoint, bField * kilogauss, charge);  
double pathLength1 = oHelix.pathLength(recoVtxPos);  
TVector3 oDca;  
oDca = oHelix.at(pathLength1);  
oDca -= recoVtxPos;
```


Solution comparison (gDca - oDca)

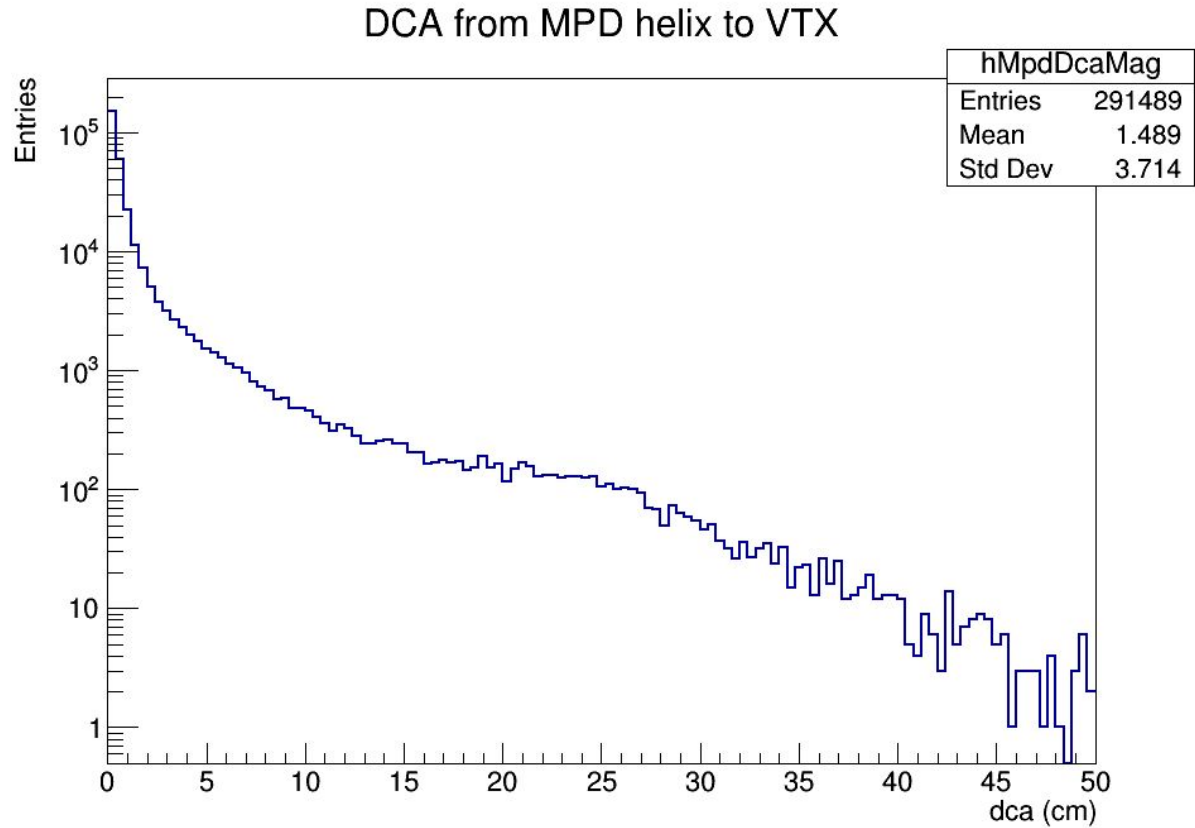
Good!

```
Kalman vertex: 0 0 0
gDca - oDca: -3.08826e-06 -1.45079e-05 -3.15038e-07
gMom - KalMom: 5.47477e-08 -2.13703e-09 -1.87461e-08
gDca - origin: -0.0697422 0.014589 0.260511
gDca - movedHelixOrig: -0.0638918 0.0547657 -21.7783
Kalman vertex: 0 0 0
gDca - oDca: 3.16595e-05 -6.40924e-06 1.45481e-05
gMom - KalMom: 4.08271e-08 -2.91289e-08 8.82274e-08
gDca - origin: 0.0632409 0.313122 0.126744
gDca - movedHelixOrig: -0.0345042 0.0716931 -21.7901
Kalman vertex: 0 0 0
gDca - oDca: -3.27088e-06 2.54226e-06 -5.62467e-07
gMom - KalMom: -7.08488e-08 -4.04386e-08 3.59944e-08
gDca - origin: 0.192278 0.23091 -0.00792605
gDca - movedHelixOrig: 0.0405199 0.0474915 -21.7478
Kalman vertex: 0 0 0
gDca - oDca: 4.1277e-05 -2.23933e-05 -1.69121e-05
```

Good(?)

```
Kalman vertex: 0 0 0
gDca - oDca: -0.00404744 -0.00333947 0.00373371
gMom - KalMom: -1.13485e-08 1.43481e-08 -2.37154e-09
gDca - origin: -1.00324 1.0337 -0.795526
gDca - movedHelixOrig: -0.554288 -0.271239 3.95548
Kalman vertex: 0 0 0
gDca - oDca: 0.000218108 6.05133e-05 -0.000234137
gMom - KalMom: 1.24659e-08 -6.10095e-09 4.83631e-10
gDca - origin: 0.139604 -0.49628 -0.112959
gDca - movedHelixOrig: -0.0679928 -0.12319 3.60594
Kalman vertex: 0 0 0
gDca - oDca: 2.94698e-05 0.000114834 -5.289e-06
gMom - KalMom: -1.67389e-08 1.05657e-09 -6.95942e-09
gDca - origin: 0.462777 -0.152406 -0.223487
gDca - movedHelixOrig: 0.0449517 -0.02399 3.48257
Kalman vertex: 0 0 0
gDca - oDca: 0.401538 -0.0695493 0.196586
```

Anyhow, distance of closest approach



Finally!!!

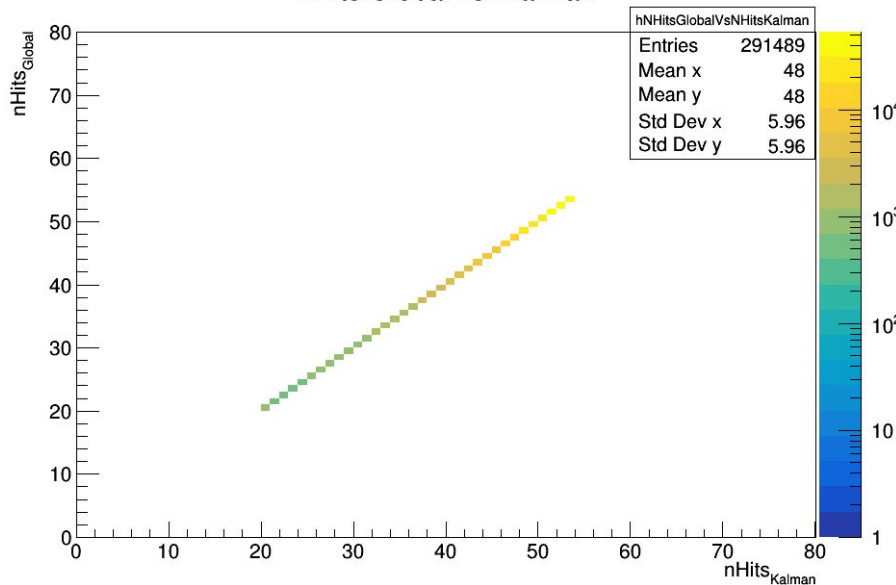
Comparison between Global and Kalman tracks

Used cuts:

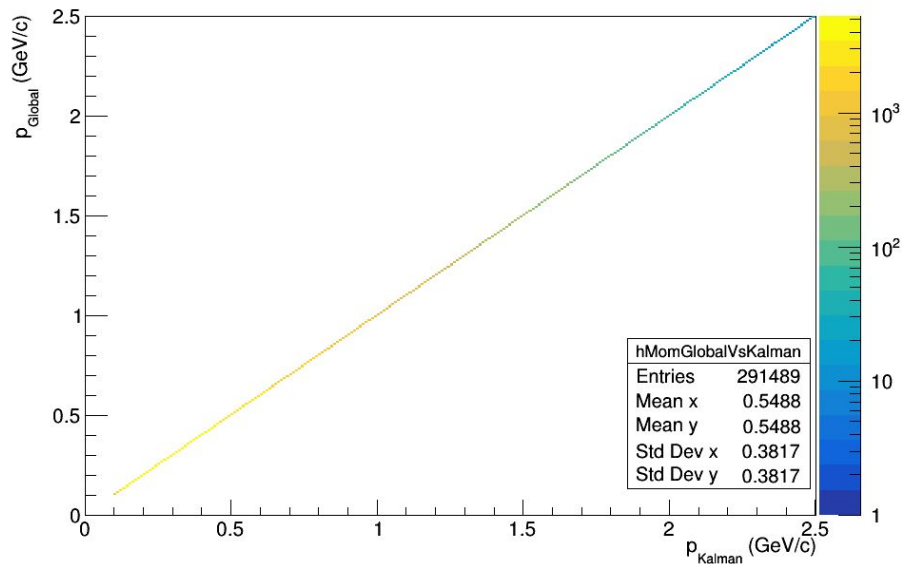
```
// Check that tracks originate from same original track
if ( mpdTrack->GetID() != kalmanTrack->GetTrackID() ) {
    std::cout << "[AHTUNG] Track indices are broken. Dropping event." << std::endl;
    break;
}

//_____
bool isGoodTrack(int nHits, TVector3 mom, int charge) {
    return ( nHits >= 20 &&
            0.1 <= mom.Perp() && mom.Perp() <= 2.1 &&
            TMath::Abs( mom.Eta() ) <= 1. &&
            charge !=0 );
}
```

nHits Global vs. Kalman



Momentum Global vs. Kalman



Looking at tracks used for the vertex reconstruction

```
// MpdVertex position (in case of several vertices)
MpdVertex *mpdVertex = (MpdVertex*)vertices->First();
TVector3 pVtx; mpdVertex->Position( pVtx );

// Return position of the reconstructed primary vertex
recoVtxPos.SetXYZ( pVtx.X(), pVtx.Y(), pVtx.Z() );

// Retrieve number of kalman tracks that were used for the vertex fit
int nVtxTracks = mpdVertex->GetNTracks();

// Loop over pointers to Kalman tracks
for ( int iTrk=0; iTrk<nVtxTracks; iTrk++ ) {

    //std::cout << "Kalman vtx track index: " << mpdVertex->GetIndices()->At(iTrk) << std::endl;

    // Retrieve Kalman track
    MpdKalmanTrack *kalmanTrack = (MpdKalmanTrack*)kalmanTracks->UncheckedAt( mpdVertex->GetIndices()->At(iTrk) );

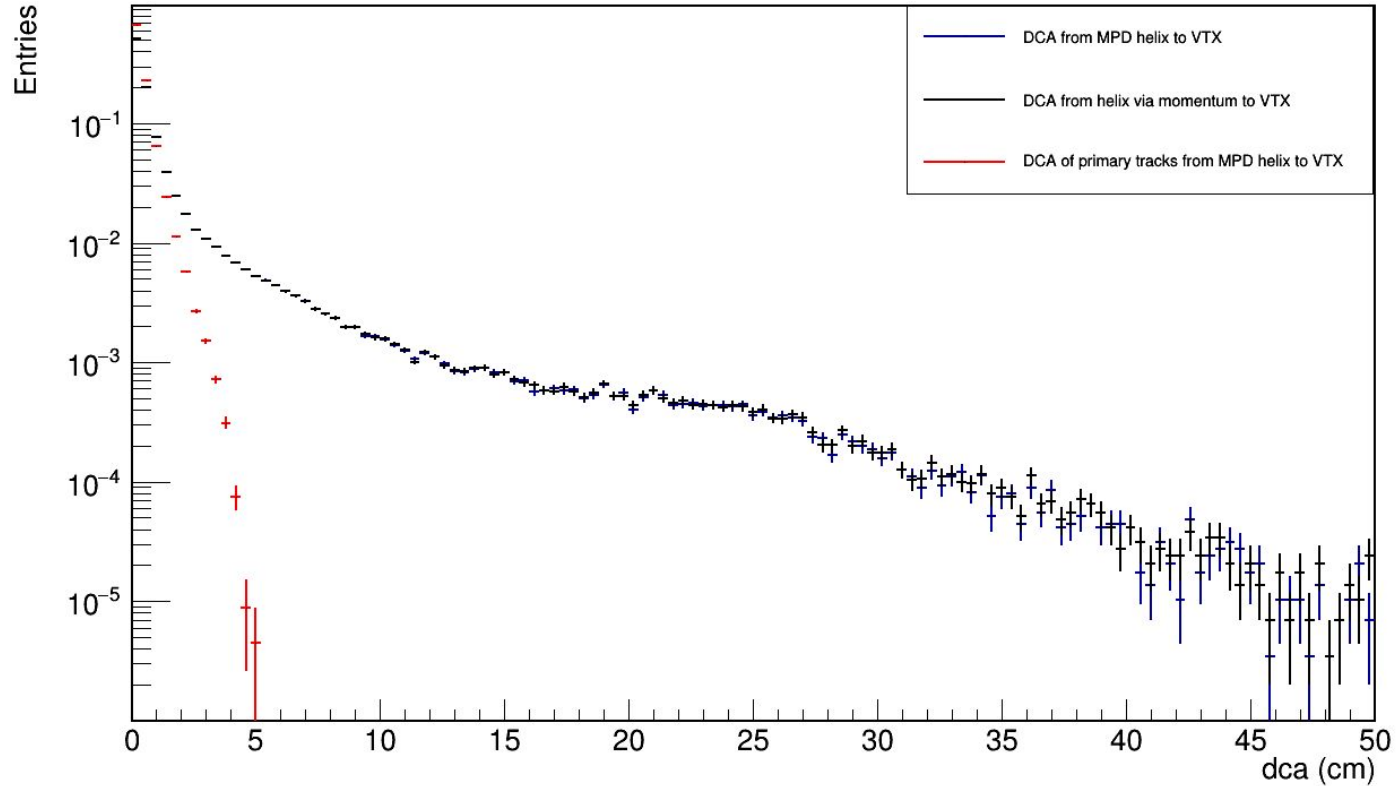
    // Check that track exists
    if ( !kalmanTrack ) continue;

    // Check good track (?)
    if ( !isGoodTrack( kalmanTrack->GetNofHits(), kalmanTrack->Momentum3(), kalmanTrack->Charge() ) ) continue;

    MpdMiniPhysicalHelix gHelix = makeHelix(kalmanTrack);
    double pathLength = gHelix.pathLength(recoVtxPos);
    TVector3 gDca;
    gDca = gHelix.at(pathLength);
    gDca -= recoVtxPos;
    // DCA to VTX from MPD helix
    hMpdPrimDcaMag->Fill( gDca.Mag() );
} // for ( int iTrk=0; iTrk<nVtxTracks; iTrk++ )
```

```
Track number [global/mc/kalman/vtxtrack]: [ 1124 / 59001 / 1124 / 804]
Track number [global/mc/kalman/vtxtrack]: [ 279 / 58446 / 279 / 22]
Track number [global/mc/kalman/vtxtrack]: [ 1179 / 62250 / 1179 / 867]
Track number [global/mc/kalman/vtxtrack]: [ 154 / 60929 / 154 / 13]
Track number [global/mc/kalman/vtxtrack]: [ 1148 / 60905 / 1148 / 842]
Track number [global/mc/kalman/vtxtrack]: [ 1012 / 58294 / 1012 / 743]
Track number [global/mc/kalman/vtxtrack]: [ 1124 / 59140 / 1124 / 827]
Track number [global/mc/kalman/vtxtrack]: [ 1053 / 56285 / 1053 / 795]
Track number [global/mc/kalman/vtxtrack]: [ 1052 / 57745 / 1052 / 757]
```

DCA comparisons (for helices)



Momentum comparison

