
Subject: Implementing Software Alignment into PandaROOT
Posted by [Roman Klasen](#) on Tue, 24 Oct 2017 10:04:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello,

I'm plan to integrate a software alignment framework into PandaRoot. For now, that means that our alignment parameters are stored to transformation matrices that are multiplied to the ideal transformation matrices to simulate a realistic, non-ideally aligned geometry.

As far as I can see, our simulation chain currently works like this:

0. MC tracks are generated by the BoxSim or DPM generator and fired on a geometry that can either be perfectly aligned or realistically misaligned
1. MC hits on the sensors are then digitized in the 1-Digi macro, and digital row, column and sensor ID data are stored to lumi_digi.root
2. The 2-reco macro reads those digi hits and reconstructs the (x,y,z) position in the panda frame of reference assuming an ideal geometry without misalignment. This is done by the PixelClusterTask and more specifically a PndSdsChargeWeightedPixelMapping object
3. Afterwards, our own PndLmdDim class reads this ideal position, and transforms it to a realistically misaligned position using the transformation matrices for the ideal and a real, misaligned geometry (in simulation only, since we obviously don't have those in the actual detector). This could also be where we introduce the alignment parameters found by software alignment.
4. Those hit positions are saved to the reco.root file after they have been transformed from the position they would have been were the geometry perfectly aligned to the position they are in case the geometry suffers from misalignment (the realistic case)

Firstly I'd like to ask if I understood this correctly.

If so, I wanted to ask how other groups are doing this since the PndLmdDim Class is specific to our detector, or if it were better to change this method to actually introduce the misalignment step earlier in the PndSdsChargeWeightedPixelMapping class. That means the reconstructed hits are stored in their realistic, misaligned position by this class directly. Afterwards, a separate aligner class would transform those positions back to the ideal positions using the alignment parameters found by software alignment. It seems the second version is mapping reality more closely and doesn't depend on our PndLmdDim class, so it could be used by other groups as well. But no such aligner class currently exists.

Best wishes,
Roman

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Stefan Pflueger](#) on Fri, 03 Nov 2017 14:37:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello everyone,

Roman and me have done some research and as far as we can understand, the alignment should be handled within the geometry. The difference between data taking of panda and the

simulation world are the steps simulation and digitization. After that all other macros/steps are identical for the simulated data and panda data reconstruction.

IMPORTANT: In order to study misalignment TWO geometries are needed! The simulation and digitization steps use one geometry. And a second geometry is used in the further track reconstruction. This geometry is the only one needed later on, but should not necessarily be the same as the first geometry. And this second geometry should be determined from the design geometry and corrections that come from the alignment procedures ($[\text{design geometry}] * [\text{alignment correction matrices}] = [\text{simulation/data geometry}]$)!

Now we have a question: Did anyone try or plan for this already or which is the correct way to handle all this? Because in the whole reconstruction chain the geometry is usually just generated once in the simulation step, then saved in a parameter root file. The other macros then read in this parameter file and use it to initialize the RTDB of FairRoot. Consequently the geometry used for the simulation is available again, which is exactly the behaviour we do NOT want for alignment studies!

So our idea would be to NOT read in the parameter file in the macros after the digitization step and just initialize the RTDB with another geometry. Is that correct or possible? Has anyone tried this before?

Thanks in advance!

Stefan

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Mohammad Al-Turany](#) on Sat, 04 Nov 2017 17:44:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hallo Stefan,

In the reconstruction macro you can set the geometry manually using:

```
FairRunAna::SetGeomFile(const char* GeoFileName);
```

If a file with valid geometry is delivered (TGeoManager object.) Nothing will be loaded from the parameter files.

best,

Mohammad

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Stefan Pflueger](#) on Sun, 05 Nov 2017 11:58:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Mohammad,

thx for the information! I just checked the sim macro in the fairroot github project template, and

there is a line

```
run->CreateGeometryFile("geofile_full.root");
```

Is that the way we should save such a geometry (TGeoManager object)?

So for the creation of such full geometries, I can just setup the FairRunAna with the detector modules, call Init() and save the geometry for later use with the line above. Basically it's just like simulation, except I skip the actual simulation part. Is that correct?

And just to clarify, this is the way the alignment studies are supposed to be handled right?

Stefan

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Mohammad Al-Turany](#) on Sun, 05 Nov 2017 14:49:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Stefan,

You can do it this way, or you can keep the ideal geometry in the parameter files, save the (miss-)alignment matrices separately and then apply them using the method:

```
FairModule::ModifyGeometry()
```

This will allow you to do simulation with miss-aligned geometry. which you can export and re-use for reconstruction.

See the code in FairRoot tutorial 4 for details.
Some documentation is also available here

best,

Mohammad

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Stefan Pflueger](#) on Tue, 07 Nov 2017 14:06:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thx Mohammad

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Tobias Stockmanns](#) on Tue, 07 Nov 2017 14:17:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Dear all,

in my point of view the really tricky part is the misalignment of the thousands of geometries inside PandaRoot. For sure you do not want to do this by hand but they all might have different placement precisions.

I suggest that we discuss this on our next EVO meeting next Monday.

Cheers,

Tobias

Subject: Re: Implementing Software Alignment into PandaROOT
Posted by [Stefan Pflueger](#) on Mon, 27 Nov 2017 15:10:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

I have been discussing the alignment study issue a bit on the ROOT forum.

<https://root-forum.cern.ch/t/howto-do-alignment-studies-with-tgeo-howto-handle-overlaps/26918>

There were two ways to perform the mis-alignment studies:

Actually mis-align the geometry in the simulation: Pros: easy to handle and realistic. Cons: overlaps in the volumes Mis-alignment of the Digis/Hits/Clusters. Pros: No overlap issue. Cons: Concept of hits vanishing and appearing on different sensors with the mis-alignment cannot be implemented (or its quite complicated). Hence not as realistic

The Alice collaboration has chosen option 1., which also Andrei suggested as a preferred solution.

The biggest task here would be to study the overlaps in detail.

Best regards,
Stefan
