

Table of Contents

Using the Millepede II alignment algorithm in a productive manner.....	1
Goal of the page.....	1
Contacts.....	1
Todos.....	1
Using different versions.....	2
CMSSW_7_4_X.....	3
Older releases.....	3
CMSSW_5_3_X.....	3
CMSSW_3_3_X.....	4
CMSSW_2_1_X and CMSSW_2_2_X.....	6
CMSSW_2_0_X and before.....	6
Using the common area for alignment on the CAF.....	6
Using the cron area for automated alignment.....	7
How to use acrontab.....	7
Rationale for some important configuration settings.....	7
Review status.....	9

Using the Millepede II alignment algorithm in a productive manner

Complete: 

Goal of the page

This page explains how






- to set up alignment areas using different versions of CMSSW
- the common alignment area on the CAF should be used for production grade alignment using the MillePedeII alignment algorithm.











Contacts

- Claus Kleinwort claus.kleinwort@desyNOSPAMPLEASE.de
- Matthias Schröder matthias.schroeder@desyNOSPAMPLEASE.de
- the mailing lists cms-millepede@cernNOSPAMPLEASE.ch and hn-cms-tif-alignment@cernNOSPAMPLEASE.ch

Todos

When using Millepede in production mode throughout 2015, certain features turned out to be needed or desired for an improved workflow. The following lists important features that should be developed and implemented until the next data-taking starts.

  - top priority and not yet done;  - important and not yet done;  - good to have and not yet done;  - done

1.  Replace cmsStage, etc. with eos/xrdcp equivalents (to be done before end of the year 2015!)
2.  Streamlined MPS and `setup_align.pl` scripts
 - ◆ Martin Descher (KIT) is working on this, close to final
3.  Automatic plots of fitted parameter values
 - ◆ Various scripts exist by Gero, Matthias, Nazar, need to be streamlined, updated and added to workflow: want to have plots being produced automatically at the end of a pede job --> **needs references!!**
 - ◆ Should parse `alignment_merge.py` to obtain settings
4.  Summary tables automatically produced after workflow
 - ◆ Ntracks, per IOV, per dataset
 - ◆ Used pede parameters, conditions
 - ◆ Partly in new MPS
5.  Inclusion/usage of mille monitoring distributions
6.  Hits per module: tracker maps
7.  B-field filter
8.  APV-mode filter
9.   Tools for easy dataset preparation
 - ◆ DAS interface to create file lists automatically, also 'on-the-fly'?
 - ◆ easy way to apply weights
 - ◆ Print statistics
 - ◆ Create orthogonal samples for alignment and validation
 - ◆ Take into account B-field and APV mode (peak or deco)

10. ✓ Implement 0-APE for Mille
 - ◆ Information collected by Matthias
11. ✓ Write split database
 - ◆ Some work started by Matthias
12. Understand dead-module treatment
 - ◆ Is the SiPixelQualityRcd and analogue for SiStrips used in Mille?
 - ◆ Stat output: list of fixed/not touched modules
13. 'pede DMRs'
 - ◆ need to be added to mps_fetch --> Martin
14. Automatically fire geom-comp (w.r.t. start geometry) after pede job
 - ◆ should wait for the rewritten versions of the mps commands
15. Automated website/presentation upload with all results?
 - ◆ still to be discussed
16. ✓ Multi-IOV constraint of z expansion/shrinkage

In addition, we **should really** get the so-far not fully recommissioned fitting of the calibration parameters done

- Recommissioning of pixel and SiStrip LA fit
- Recommissioning of SiStrip BP correction
- New: TEC BP corrections

Task	Person(s)	Status
1	Gregor	done
2	Martin	done
3	Ola & Patrick & anyone who wants	not-yet started
4	Martin(?)	not-yet started
5	Patrick & (Gregor) & Rostyslav	not-yet started
6	Patrick & (Gregor) & Rostyslav	not-yet started
7	Gregor	done
8	Gregor	done
9		not-yet started
10	Gregor	done
11	Gregor	done
12		not-yet started
13	Martin	not-yet started
14		not-yet started
15		not-yet started
16	Gregor	done

Using different versions

Some hints valid for all versions:

- If you want to use the MPS then don't forget to adjust the paths to your new CMSSW environment in the template files `mps_runMille_template.sh` and `mps_runPede_rfc_template.sh` in `Alignment/MillePedeAlignmentAlgorithm/scripts`. The line to adjust is after the comment line # set up the CMS environment.
- If you plan to use an older version together with a more recent config file and receive a complaint that a certain block from the presigmascenarios is missing then just get a copy of `Alignment/MillePedeAlignmentAlgorithm/python/PresigmaScenarios_cff.py` from a more current version.

- Further useful general information on setting up the Millepede II production environment are provided in these tutorials (however, to obtain the latest CMSSW implementations, please follow the recipes given further below on this page!)
 - ◆ tutorial 2014-11-03 [↗](#) by Matthias Schröder matthias.schroeder@desyNOSPAMPLEASE.de
 - ◆ tutorial 2013-09-17 [↗](#) by Nazar Bartosik nazar.bartosik@cernNOSPAMPLEASE.ch

CMSSW_7_4_X

Millepede alignment in the 7XX release cycle is still in a process of rapid development and commissioning. Starting from CMSSW_7_4_0_pre2, the Alignment package contains all important bug fixes (including the SLC6-bug fix [↗](#) and the averaging constraints [↗](#)). For earlier 7XX releases, individual recipes have to be applied on top of the release version. Please contact the experts for help.

NOTE The Lorentz-angle and backplane correction calibration does not work yet in any of the 7XX releases due to changed/missing interfaces to the RecoLocalTracker package.

Older releases

[▣ show recipes](#) [▣ Hide](#)

CMSSW_5_3_X

Some of the latest features of the Millepede implementation in CMSSW, including capability of Lorentz angle and backplane calibrations, were 'lost' during the transition from CVS to github, and thus, they are not included in the official-release version of the Alignment package because a backport is in general (except for bug fixes) not possible at this point anymore. The modifications are provided via the Alignment-MP-53X-Dev [↗](#) branch of the cms-trackeralignment [↗](#) VO and can be added on top of the official-release version.

The following recipe has been tested for 5_3_23. All later versions **should** also work, but no guarantee.

- Setting up CMSSW_5_3_23

```
scram p CMSSW CMSSW_5_3_23
cd CMSSW_5_3_23/src
cmsenv
```

- Adding the subpackages from the release that will be modified (in order to avoid having to type in your password for each package, you can put the list of packages into a text file `list.txt`, one package per line, and then use `git cms-addpkg -f list.txt`):

```
git cms-addpkg Alignment/CommonAlignment
git cms-addpkg Alignment/CommonAlignmentAlgorithm
git cms-addpkg Alignment/CommonAlignmentMonitor
git cms-addpkg Alignment/CommonAlignmentParametrization
git cms-addpkg Alignment/CommonAlignmentProducer
git cms-addpkg Alignment/MillePedeAlignmentAlgorithm
git cms-addpkg Alignment/OfflineValidation
git cms-addpkg Alignment/ReferenceTrajectories
git cms-addpkg Alignment/TrackerAlignment
git cms-addpkg CalibTracker/Configuration
git cms-addpkg CalibTracker/Records
git cms-addpkg CalibTracker/SiStripESProducers
git cms-addpkg CondCore/SiStripPlugins
git cms-addpkg CondFormats/DataRecord
git cms-addpkg CondFormats/SiStripObjects
git cms-addpkg MuonAnalysis/MomentumScaleCalibration
git cms-addpkg RecoLocalTracker/Records
```

SWGuideMillepedeIIProductionEnvironment < CMSPublic < TWiki

```
git cms-addpkg RecoLocalTracker/SiPixelRecHits
git cms-addpkg RecoLocalTracker/SiStripRecHitConverter
```

- Applying the private modifications. First, you need to add the [cms-trackeralignment/cmssw](#) repository as a remote and then to checkout the [Alignment-MP-53X-Dev](#).

```
git remote add cms-trackeralign git@github.com:cms-trackeralign/cmssw.git
git fetch cms-trackeralign
git checkout --track cms-trackeralign/Alignment-MP-53X-Dev
```

- Then, for the time being, applying a quick fix of the [SLC6 bug](#) until the more sophisticated fix is backported to the release:

```
In Alignment/CommonAlignmentProducer/plugins/AlignmentProducer= in =edm::EDLooper::Status Alignme
const AlignmentAlgorithmBase::EventInfo eventInfo(event.id(), trajTracks, *beamSpot, clusterValue
by
const edm::EventID evtid = event.id();
const AlignmentAlgorithmBase::EventInfo eventInfo(evtid, trajTracks, *beamSpot, clusterValueMapPt
```

- Finally, compile everything

```
scram b -j6
rehash # for csh
hash -r # for sh
```

CMSSW_3_3_X

To use the latest developments concerning Millepede II (Broken lines fit see(<http://indico.cern.ch/getFile.py/access?contribId=7&sessionId=1&resId=0&materialId=slides&confId=63597>)) and the usage of the Runge-Kutta track fit, do:

```
scram p CMSSW CMSSW_3_3_2
cd CMSSW_3_3_2/src

cvs co -r V03-02-00 Alignment/CommonAlignmentAlgorithm
cvs co -r V01-02-01 Alignment/CommonAlignmentProducer
cvs co -r V03-08-02 Alignment/LaserAlignment
cvs co -r V00-21-03 Alignment/MillePedeAlignmentAlgorithm
cvs co -r V02-06-07 Alignment/OfflineValidation
cvs co -r V01-00-01 Alignment/ReferenceTrajectories
cvs co -r V00-01-01 DataFormats/Alignment
```

If you want to use the Runge-Kutta propagator, make sure that the following things are set (a diff output is available further below):

path/file	check
Alignment/ReferenceTrajectories/CMS.BuildFile	should contain <code><use name=TrackPropagation</code>
Alignment/ReferenceTrajectories/src/ReferenceTrajectory.cc	<p><code>#include</code> statement for <code>TrackingTools/GeomPropagators/interface</code> is absent or commented out</p> <p><code>#include "TrackPropagation/RungeKutta/interface</code> present</p> <p>the line <code>AnalyticalPropagator aPropagato</code> commented out and the following lines are pre</p>

```
RKTestPropagator bPropagator(magField,
tolerance = 5.e-5
Propagator &aPropagator = bPropagator;
```

And after all don't forget to issue a

```
scram b
```

=diff= output for the changes needed for using the Runge-Kutta propagator:

```
diff -r CMSSW_3_3_2_withBLnoRK/src/Alignment/ReferenceTrajectories/CMS.BuildFile CMSSW_3_3_2_with
12a13
> <use name=TrackPropagation/RungeKutta>
diff -r CMSSW_3_3_2_withBLnoRK/src/Alignment/ReferenceTrajectories/src/ReferenceTrajectory.cc CMS
24c24,25
< #include "TrackingTools/GeomPropagators/interface/AnalyticalPropagator.h"
---
> // #include "TrackingTools/GeomPropagators/interface/AnalyticalPropagator.h"
> #include "TrackPropagation/RungeKutta/interface/RKTestPropagator.h"
274c275,279
< AnalyticalPropagator aPropagator(magField, propDir);
---
> //AnalyticalPropagator aPropagator(magField, propDir);
> // work around TrackPropagation/RungeKutta/interface/RKTestPropagator.h and
> // http://www.parashift.com/c++-faq-lite/strange-inheritance.html#faq-23.9
> RKTestPropagator bPropagator(magField, propDir); //double tolerance = 5.e-5)
> Propagator &aPropagator = bPropagator;
```

MC alignment: An example configuration file for MC alignment studies can be found here:

```
/afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/hamburg/alignmentCosmMC_GF_cfg.py
```

CRAFT08 alignment: To run on the CRAFT08 data (3rd reprocessing)

```
/Cosmics/Commissioning08_CRAFT0831X_V1_311_AlcaReco_FromTrackerPointing_StreamTkAlCosmics0T_v1
```

you can find an example configuration file here:

```
afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/hamburg/alignment_cfg_EXAMPLE_CRAFT0831X.py
```

NOTE:

- A ready-to-use area is already set up. See this section. A set of template config files for common tasks is available there as well.
- If you want to use the new **CMS.TrackerTrackHitFilter**, make sure the **PixelQualityFlag** is set to **FALSE** if you do **not use the pixel templates**.
- Changes in the tracking code require now explicitly the geometry record:

```
process.load("Configuration.StandardSequences.Geometry_cff")
```

- Furthermore you should change the Navigator School to :

```
NavigationSchool = cms.string('')
```

for the track producer and refitter to avoid long but negligible warnings (see <https://hypernews.cern.ch/HyperNews/CMS/get/recoTracking/740/1.html> for details).

CMSSW_2_1_X and CMSSW_2_2_X

Please use CMSSW_2_2_9 or above. CMSSW_2_2_X can read CMSSW_2_1_X data as well (except particle flow objects that we do not care are all).

To get all the features of Millepede and almost all of the alignment framework as in CMSSW_3_1_X, follow the *Backport of CMSSW_3_1_X features* section at SWGuideTrackAlignmentVersions.

CMSSW_2_0_X and before

Documentation for older CMSSW versions can be found in older versions of Twiki page on the MillePedeII alignment algorithm.

Using the common area for alignment on the CAF

The path to the area is: `/afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/MPproduction`

If you do alignment which is intended to be used later on for validation, presentation or for providing new alignment constants, please observe the following things:

1. Choose a CMSSW environment from within that directory, go there and do `cmsenv` and come back. If you need to change code, please do this VERY CAREFULLY as this might break others alignment. Consider using a subdirectory with your own CMSSW environment.
2. Create a new working directory by invoking `setupNewAlignment.sh` while you are in `MPproduction/`. This does the following:
 - ◆ Creates a new directory `mpXXXX` where the number `XXXX` is created automatically
 - ◆ Adds a new line in the text file `MP_ali_list.txt`. Please edit this file and fill in the relevant information in the newly created line at the bottom. This helps everyone to trace what was done. Don't leave this file open for long time in your editor - another user might want to add a new alignment as well (probably we will need a better solution in the future).
3. If you want to start from some proven configurations, copy appropriate files from `aliConfigTemplates/` to your new directory and modify them according to your needs.
4. If you want to use a common dataset then copy one from the `datasetfiles/` directory to your new directory
5. Start your alignment and enjoy.
6. If your result is nice and you want to refer to it in a presentation or wherever you need a reference, use the directory name `mp00xx`

If you want to do some private tests, create a subdirectory `testYOUR_CERN_USERNAME` and feel free to do within there what you want. But remember: Even when your planned alignment might have flaws, consider running it as described above. No one prevents you from editing `MP_ali_list.txt` again and write there that a certain alignment was not successful - it helps others to see what failed.

If you have a configuration which you want to add to `aliConfigTemplates/` put it there, choose a meaningful name and advertise it by sending an email to the others.

Cleanup/Backup

From time to time it will be necessary to cleanup the common MP alignment area. In addition it might be nice to have a backup of an alignment job in case something is getting lost. For both purposes the script

`MPprodToCastor.sh`

can be used. It is called given the user name as argument. It will automatically look up the user's directories from

MP_ali_list.txt

and in case there hasn't been a backup produced already create a directory.tar and copy it to the user's private castor area.

Using the cron area for automated alignment

So called prompt alignment may be carried out within

/afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/MPcron. The area there is organized as conditions. One condition has one set of config files. **Do not change any existing config without a very good cause!** There may be cron jobs relying on a stable config. Make a new condition instead. The following steps show you how to do this:

1. Go to /afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/MPcron
2. cd into the new directory
3. Copy your desired alignment config file into the new directory
4. Invoke the script =./setupNewCronAlignment.sh"
5. Edit newJobSetup.sh. Adjust the variables at the begin of the file. In most cases it is sufficient to adjust the name of the config file to be used (line with aliconfigsrc)
6. You should now be able to invoke such an alignment by calling newJobSetup.sh from that directory
7. Add a line to acrontab (see below)

How to use acrontab

acrontab is a AFSed version of crontab so that the jobs are able to see the AFS. A simple intro is at <http://dv-zeuthen.desy.de/services/afs/acrontab/>. Here are some points to consider in our context:

- acrontab -l lists your current acrontab
- Because of a known bug, starting jobs in a lxplus5 environment doesn't work. lxplus (i.e. SLC4) needs to be used. In order to see a lxplus5 machine, a trick by Ernest is used (see promptcron.sh in your new directory created using the instructions above)
- A typical line looks as follows:

```
01 02 * * * lxplus /afs/cern.ch/cms/CAF/CMSALCA/ALCA_TRACKERALIGN/MP/MPcron/con0003/pro
```

This line invokes the job specified in con0003 every 24 hours at 1 minute past 2

- **Please** invoke ./putMyAcrontab.sh after you changed your settings. This places a copy of your acrontab in the directory acrontabs/ so that others may see what prompt alignments are running

Rationale for some important configuration settings

The following table gives an idea what the rationale is behind certain settings found in the templates:

Part	Parameter	setting
AlignmentTrackSelector	etaMin	999
	etaMax	999
	nHitMin	8
	nHitMin2D	2

	chi2nMax	999
CMS.TrackerTrackHitFilter	TrackAngleCut	0.35
		0.17
	usePixelQualityFlag	true
	StoNcommands	ALL 18.0
		ALL 5.0
AlignmentProducer.algoConfig	TrajectoryFactory	ReferenceTrajectoryFacto
		BzeroReferenceTrajecto
		CombinedFwdBwdDualTraj
	TrajectoryFactory.MaterialEffects	BrokenLinesCoarse
		BrokenLinesFine
		(not given)
AlignmentProducer.algoConfig.pedeSteerer	options	entries 1
		chisqcut 30
		regularisation

TrackRefitter		TrackRefitter
		TrackRefitterP5
	TTRHBuilder	WithAngleAndTemplate
		WithTrackAngle

Review status

Reviewer/Editor and Date (copy from screen)	Comments
-- FrankMeier - 12-Dec-2009	created page
-- FrankMeier - 8-Dec-2011	added warning about outdated content
-- MatthiasSchroederHH - 2014-12-03	updated recipes

Responsible: ! FrankMeier

Last reviewed by: (not yet reviewed)

This topic: CMSPublic > SWGuideMillepedeIIProductionEnvironment

Topic revision: r31 - 2016-06-09 - GregorHellwig



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
 or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback