CPT week - PRS Muon meeting

CERN 23 April 2001

Progress on Common muon-tracker navigation

Stefano Lacaprara, *INFN, Padova.*

- ◇ SmartPropagator;
- ◇ muon tracker navigation school
- ◇ trajectory building;
- ◇ muon navigation school

SmartPropagator:

`SmartPropagator` *is a* `Propagator`, and it acts as a a wrapper to two concrete Propagator to propagate FreeTrajectoryState inside or outside tracker volume.

Available in HEAD of `MuonReco/CommonNavigation`.

◇ The default algorithm are: **GtfPropagator** for Tracker, and **Geane** outside,

◇ other choices can be defined in the constructor,

◇ the tracker volume is defined as a bound cylinder with tracker dimension $(+5cm)$,

◇ the user interface is that of Propagator:

- `TrajectoryStateOnSurface propagate(const FreeTrajectoryState& fts, const Surface& surface) const;`
- `TrajectoryStateOnSurface propagate(const FreeTrajectoryState& fts, const BoundPlane& plane) const;`
- `TrajectoryStateOnSurface propagate(const FreeTrajectoryState& fts, const BoundCylinder& cylinder) const;`
- `FreeTrajectoryState propagate(const FreeTrajectoryState& fts, const string& volumeName) const;`

◇ whenever a propagate method is invoked, the SmartPropagator check if initial and final position are inside tracker and uses *tracker propagator* otherwise uses *outer propagator*,

◇ SmartPropagator is **lazy**, i.e. a concrete propagator is instantiated *only if* is used.

Muon tracker navigation:

The navigation is defined in the class `MuonTkNavigationSchool` which *is a* `NavigationSchool` and builds the links between the DetLayers of Tracker and Muon systems.

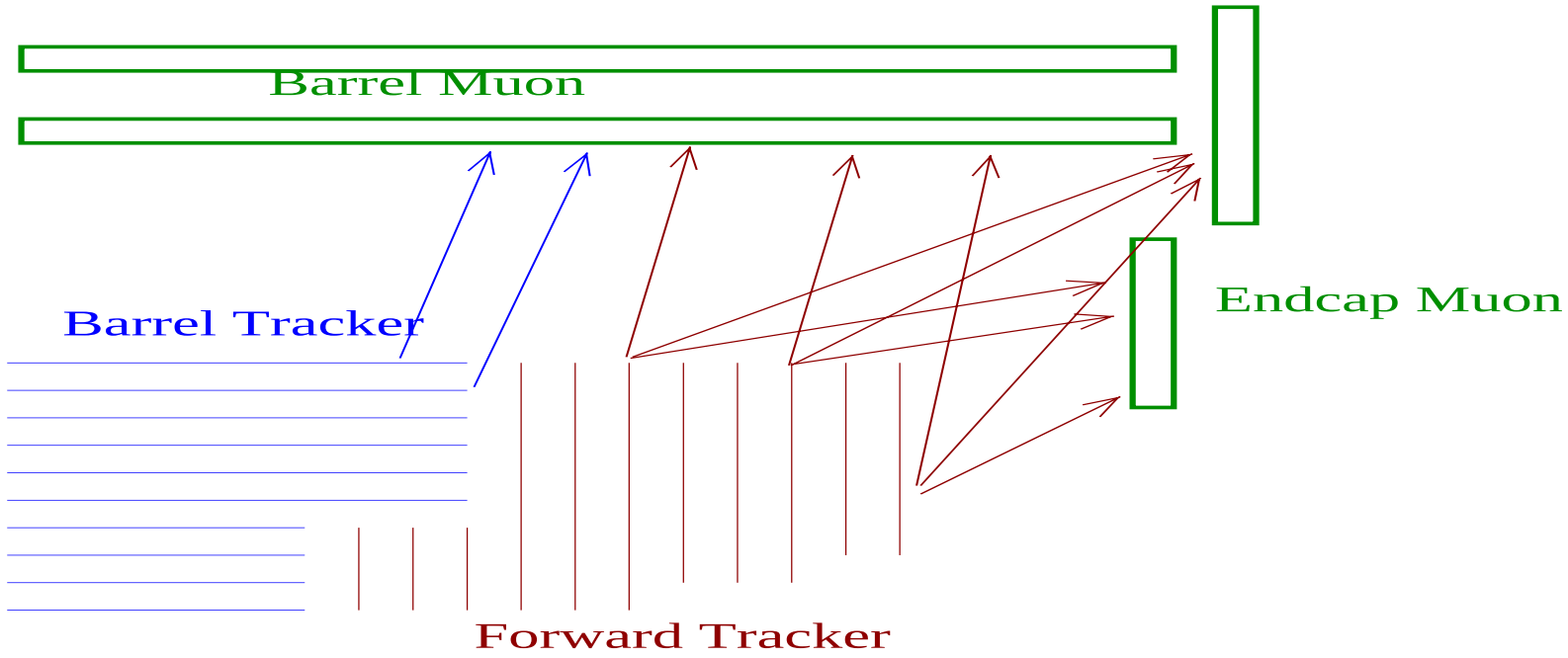
Available in HEAD of `MuonReco/CommonNavigation`.

◇ It's based on tracker SimpleNavigationSchool, but in addition the first muon DetLayer is linked to the outer tracker layers;

◇ Tracker navigation is the same (some code duplication here, better solution would be to get the results from a Tk navigation school and *add* the muon links);

◇ Tracker outermost barrel layer is linked to Muon innermost barrel one;

◇ Tracker outer forward layers are linked to barrel and forward Muon layers;



◇ During trajectory propagation a further selection on reachable layers is done via a propagation w/o error (so far GtfPropagator is used, in future SmartPropagator ?);

TrajectoryBuilder:

The trajectory builder is the one used in the tracker, so far I've tested only `CombinatorialTrajectoryBuilder`.

- ◇ Small modifications are needed to account for different Muon RecHits: an Endcap RecHit is a segment, but we want to add to the trajectory the components of the segments, i.e. 3-D points, which are RecHits as well;
- ◇ So we add the components of the compatible recHits, instead of the recHits itself: already foreseen in CommonDet, trivial mod's!;
- ◇ Major modification for Muon navigation, the one used for muon tracking is not suited for this need.

Muon Navigation:

When a trajectory started in the tracker reach the muon system, the Muon DetLayers take care of further propagation,

but

the navigation used so far for muon tracking is only partially compatible with CommonDet:

CommonDet way to build trajectory:

- ◇ A trajectory is defined in some DetLayer (e.g. by a seed generator, or by previous tracking);
- ◇ the Trajectory builder ask to this DetLayer which is the next DL according to the trajectory parameters and direction;
- ◇ the DL asks the answer to its NavigableLayer and return to the builder the result, which is the immediate next layer (could be more than one, e.g. in overlap region);
- ◇ the trajectory it propagated to the next DetLayer and eventually updated with compatible measurement;
- ◇ the iteration go on until no next layers are available;

Muon way is different:

◇ You get a seed (local or global), and ask to a navigation class which are **all** the layers compatible with that trajectory;

◇ the navigation class returns in one shot all next layers, not only the immediate one;

◇ then you iterate over these DetLayers to grow the trajectory;

In particular the muon DetLayers don’t have a NavigableLayer, so the DL can’s answer to question:“who’s next?”, so when a tracker trajectory reach muon, can’t be propagated any more.

Muon Navigation School:

The solution (following Tracker approach) is to build a class (`MuonNavigationSchool`), which “teaches” to muon DetLayers the navigation, i.e. which defines for each muon DL a NavigableLayer who is responsible of navigation.

A new muon subpackage (**Muon/MuonNavigation** in the HEAD) contains all relevant classes.

◇ The `MuonNavigationSchool` *is a* `NavigationSchool` and defines all the links, and builds `Muon(Barrel|Forward)NavigableLayers` for each muon DetLayers;

- `Muon(Barrel|Forward)NavigableLayer` *is a* `NavigableLayer` which are fed into DetLayer by `MuonNavigationSetter`;
- when a Muon(Barrel|Forward)NavigableLayer is asked for next layers according to a FreeTrajectoryState, a further selection based on a η compatibility criteria is done;
- So far only DT and CSC are used, but RPC addition is foreseen in the framework.

Conclusion and plan:

A SmartPropagator class is available.

A first common muon-tracker navigation has been build and committed: it has been debugged using CombinatorialTrajectoryBuilder on a small sample of single muon $|\eta| < 2.4$ and works (no plots yet, sorry!).

A Muon navigation school has been built.

Problems:

◇ the trajectory builder must be slightly modified to use composed recHits;

◇ the number of candidates trajectory during the steering become large when muon are reached, even if all (but one) are removed at the end;

◇ maybe a better solution is to build the Tracker tracks, clean them, and then extend the survivors to muon system: studies are needed.

Plan:

◇ optimise the Tk-Muon navigation (e.g. the last forward tracker DL's can't reach barrel Muon, maybe neither ME2-3/1);

◇ test the tracker-muon tracking on larger and realistic samples;

◇ modify the muon trajectory builder to use the new navigation;