Refitting Tracks from DST

E. Rodrigues, NIKHEF

LHCb Tracking and Alignment Workshop, Lausanne, 8-9th November 2006

- Motivations
- Step-by-step ...
- Current Performance
  - Tests with Brunel v30r12
- Outlook
Motivations

Reconstruction and Physics

- Refitting tracks is a necessity in real world analyses
  - e.g.: needed after improvement in knowledge of alignment

Event Model

- One of the reasons for the track event model review in 2004-5
  - Nobody knew how to refit tracks from DST with the old event model

- Ability to refit was a request to us

... WAS STILL AN UNTACKLED ISSUE ...
In Brunel …

- Whole of the tracking sequence:
  - Pattern recognition
  - Track fitting
  - Clone killing
  - tracks stripped of non-persistent information
    - left basically with LHCbIDs and flags
  → job done with the “TrackToDST” algorithm

… saved on the DST

- Container of “BEST” tracks
Two “kinds” of refitting

I.  Refit a single track
   - probably mostly useful for tests, special studies, etc.
   - Note: can also be done in Python … can show interesting things …

II. Refit all tracks in a container
    - the common situation, I guess

☐ Solving situation II implies solving I
    - options to refit a track simply applied on the set of container tracks
Refitting tracks from DST

- Read the cocktail of tracks in the “best” container
- Separate them back accordingly, in their original containers
  - Rec/Track/Forward, Rec/Track/Match, etc.
- Refit per container with appropriate options
  - options depend on the track type!

→ Need for a new algorithm …

✓ New TrackFromDST algorithm in Tr/TrackUtils

- classifies the tracks given as input according to their History / pattern recognition algorithms, i.e. remakes the PR algorithms produced in Brune!
- BUT: all clones are gone! Not quite the same containers …
Home-made DST

- Minimalistic Brunel job – only tracking
  - still done in Brunel v30r10

Refit tracks from DST

- Can do study in (at least) 2 ways:
  - in DaVinci
  - with a standalone / simple Gaudi job
  - opted for a standalone Python script

- New DaVinciRefitting.opts: options file with “refit” process phase (details given later)

- own code to match “Brunel original’s” to refitted tracks

- and then compare the outcome …
Looking first at Long tracks from PatForward

- Same fitting options as in Brunel!
- Got many messages of the kind:

  FitForward.Fitter.RefInfoTool WARNING LongTrackReferenceCreator:: No Velo State Status Code=FAILURE

- At the end of the job:

  FitForward INFO ======================= TrackEventFitter Summary =======================
  FitForward INFO Fitting performance : 99.60 %
  FitForward INFO ( 9604 / 9643 tracks )

Conclusions:

- All these tracks had been successfully fitted in Brunel
  - by construction, since they were taken from the “best” container
- This first result is encouraging, but not good enough …
- Let’s have a look at the reference information tools …
Tr/TrackTools

- LongTrackReferenceCreator:
  - Adapted so that it can be used for tracks from DST, i.e. without “EndVelo” and AtT” states – these are only available in Brunel
  - ... now it takes closest states
    - improvements do not influence behaviour in Brunel

Reference information tools for refitting:

- In Brunel we have for now tools for only Long and seed (Tsa) tracks
- Tool for Long tracks successfully adapted/generalised
  - LongTrackReferenceCreator
- Tool for Tsa seed tracks cannot be used for now outside Brunel
  - Accesses pattern recognition intermediate information
  - We could/should make the tool flexible for refitting applications
  - Or make a new dedicated tool
- Tools for VeloTT and KShort fitting would be desirable ...
The special case of Velo tracks:

- **In Brunel they are “prepared for fitting”**
  - Get a $P_T$ of 400 MeV
  - $P$ set accordingly, depending also on slope

- **Are prepared only those Velo tracks that are not ancestors of tracks in the “best” container**

- **For other purposes/tests it can be handy not to take into account the “best” container …**

- **Tr/TrackUtils**
  - **TrackPrepareVelo:**
    - modified so that Velo tracks ancestors of tracks in the “best” container can also be “prepared”
Refitting Options

- For now all in a file: *DaVinciRefitting.opts*
  - “Refit” process phase
  - Loading of all necessary libraries
    - tracking-related libraries
    - libraries for general needed tools (e.g.: cluster position tools)
  - Remake all original containers from the “best” container
  - Decoding of all tracking detectors – needed for track fit
  - Refitting of all containers
    - with dedicated options according to track type
    - same options as in Brunel (except for ref. tools – see comments before)
Second Round of Tests (1/2)

**Some words on technical details**

- Reference information set only for Long tracks
  - tracks from PatForward and TrackMatching

```
FitForward   INFO ====================== TrackEventFitter Summary ======================
FitForward   INFO   Fitting performance : 100.00 %
FitForward   INFO                             ( 9643 / 9643 tracks )
FitForward   INFO ======================= ===============================================
FitMatch     INFO ====================== TrackEventFitter Summary ======================
FitMatch     INFO   Fitting performance : 100.00 %
FitMatch     INFO                             ( 6461 / 6461 tracks )
FitMatch     INFO ======================= ===============================================
```

- Testing on other track types as well …

Much better!
| FitVeloTT | INFO ================================================================ TrackEventFitter Summary ============================================================== |
| FitVeloTT | INFO  Fitting performance : 99.86 % ( 2138 / 2141 tracks )            |
| FitVeloTT | INFO ================================================================ |
| FitTsaSeed| INFO ================================================================ TrackEventFitter Summary ============================================================== |
| FitTsaSeed| INFO  Fitting performance : 100.00 % ( 11000 / 11000 tracks )          |
| FitTsaSeed| INFO ================================================================ |
| FitKShort| INFO ================================================================ TrackEventFitter Summary ============================================================== |
| FitKShort| INFO  Fitting performance : 99.91 % ( 6483 / 6489 tracks )             |
| FitKShort| INFO ================================================================ |

- Also encouraging
- some fit failures left to be investigated later …
**Home-made DST**

- Minimalistic Brunel job – only tracking
- Now done in Brunel v30r12
- Extended DST:
  - has linker tables to be able to re-associate tracks

**Refit tracks from DST**

- Extended standalone Python script
- same *DaVinciRefitting.opts*
- own code to match “Brunel original’s” to refitted tracks
- Tracks re-associated to MC truth
- comparisons to same Brunel original track and MC truth
**Brunel**
- Track’s original value in Brunel

**DV**
- “DaVinci” value, i.e. value after track refitting

**X (DV – Brunel)**
- Resolution in variable “X” – comparison refitting/fitting

**Plots for x, y, t_x, t_y:**
- Variable comparison looping over all states on the track
Comparisons - Long Tracks (1/6)

- Very good agreement in slopes

Tails to be understood
Very good Agreement in positions ~1-2 μm ... looping over Velo/TT/OT/... hits!

Forward X (DV-Brunel)

<table>
<thead>
<tr>
<th>Entries</th>
<th>Mean</th>
<th>RMS</th>
<th>Underflow</th>
<th>Overflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>57474</td>
<td>5.545e-06</td>
<td>0.002734</td>
<td>9720</td>
<td>9595</td>
</tr>
<tr>
<td>$\chi^2$/ndf</td>
<td>1.333e+04</td>
<td>4.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prob</td>
<td>0</td>
<td>Constant</td>
<td>240.2 ± 7.0</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>-6.202e-06 ± 9.36e-06</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sigma</td>
<td>0.001319 ± 0.000037</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Forward Y (DV-Brunel)

<table>
<thead>
<tr>
<th>Entries</th>
<th>Mean</th>
<th>RMS</th>
<th>Underflow</th>
<th>Overflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>57474</td>
<td>-1.284e-06</td>
<td>0.002859</td>
<td>1.282e+04</td>
<td>1.257e+04</td>
</tr>
<tr>
<td>$\chi^2$/ndf</td>
<td>1.011e+04 / 497</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prob</td>
<td>0</td>
<td>Constant</td>
<td>90.9 ± 2.4</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>1.352e-05 ± 2.227e-05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sigma</td>
<td>0.0002805 ± 0.000072</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Forward Z (DV-Brunel)

<table>
<thead>
<tr>
<th>Entries</th>
<th>Mean</th>
<th>RMS</th>
<th>Underflow</th>
<th>Overflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>57474</td>
<td>5.667e-08</td>
<td>9.552e-05</td>
<td>3821</td>
<td>3791</td>
</tr>
<tr>
<td>$\chi^2$/ndf</td>
<td>4.841e+04 / 470</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prob</td>
<td>0</td>
<td>Constant</td>
<td>4.305 ± 9.168</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>-2.078e-06 ± 2.219e-05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sigma</td>
<td>0.006353 ± 0.0000316</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Very good agreement in momentum
Very good agreement in momentum
Comparisons - Long Tracks (6/6)

**Momentum resolutions as in Brunel!**

\[ \frac{\delta p}{p} \sim 4.5 \text{ per mille} \]
Double Gaussian
Core resolutions
$\delta p / p \sim 4.2$ per mille
Comparisons - KShort Tracks (1/3)

Very good agreement in positions.

KsTrack Y (DV-Brunel)

- Entries: 27084
- Mean: 1.279e-05
- RMS: 0.003637
- Underflow: 8456
- Overflow: 8657
- $\chi^2$/ndf: 2080 / 497
- Prob: 0
- Constant: 31.65 ± 0.67
- Mean: -4.3836e-05 ± 4.395e-05
- Sigma: 0.003605 ± 0.000067

KsTrack Z (DV-Brunel)

- Entries: 27084
- Mean: 2e-005
- RMS: 0
- Underflow: 216
- Overflow: 191
- $\chi^2$/ndf: 2.669e+004 / -2
- Prob: 0
- Constant: 5.986e+011 ± 1
- Mean: 2.6e-005 ± 2.3e-015
- Sigma: 3.076e-013 ± 1.861e-012

KsTrack: X (DV-Brunel)

- Entries: 27084
- Mean: 1.695e-05
- RMS: 0.00287
- Underflow: 5553
- Overflow: 5490
- $\chi^2$/ndf: 7269 / 497
- Prob: 0
- Constant: 120.9 ± 5.4
- Mean: 7.824e-06 ± 1.235e-05
- Sigma: 0.001154 ± 0.000050
Comparisons - KShort Tracks (2/3)
Comparisons - KShort Tracks (3/3)

Fair agreement in momentum

More fit iterations needed?
Too many outliers?
First proof that we can refit tracks from DST

→ Refitting is now a reality!

First tests are rather encouraging

→ Differences between “Brunel-fitted” and “DST-refitted” tracks are in general well within errors
→ Momentum resolutions of refitted tracks very much the same as in Brunel

BUT one needs to understand:

• Tails in the various distributions
• Fit failures for tracks that had been successfully fitted in Brunel
• Refitting of all track types in detail