Tracking Algorithm Recommendation Committee (TARC) Report II

Tom Diehl, Avto Karchilava, Michiel Sanders,
Rick Van Kooten (chair), Erich Varnes
3 March 2003

1. Charge

In July 2002, the DØ Physics and computing/software management commissioned a Tracking Algorithm Recommendation Committee. That committee led studies over the months of August and September and delivered a report on 2 Oct. 2002 recommending the implementation of the GTR+HTF combination (more details below) in p13 as default for use on the farms. The other recommendations were to encourage all developers to continue code development for p14 and to carry out a similar study for p14, with a greater focus on data.

We assumed that the original charge continues to apply, i.e.,

- Collect information and get data on the performance of the different tracking algorithms. These measures of performance should include efficiencies, fake rates, and mis-reco rates, using the standard procedures developed by the global tracking group. In additions, measurements of efficiencies and fake rates for particle id obtained from standard datasets (both data and Monte Carlo) accumulated by physics/id groups should be made. Finally, average CPU time per event, memory usage, and luminosity dependence of the CPU time should be determined. Input from physics/id/algorithms groups should be solicited.

- Make recommendation(s) on how we should run tracking in p14 on the farm, taking into account the maximum CPU time budget. (out-dated CPU budget numbers omitted here).

The previous committee's last two recommendations were therefore pursued by code developers and with a different TARC committee membership, with setting up of the algorithms and samples in December 2002, with reprocessing and studies starting in earnest in January 2003. Code development and optimization also continued throughout February 2003. This report briefly summarizes the work done by the committee, the global tracking group, physics/ID groups and others over the last three months.

Many details are also documented in the TARC web pages.

2. Tracking Algorithms Considered

Definitions and code improvements since TARC1

- Global Track Finder (gtr): based on TRF++, uses specific paths (roads) during track finding. *Author: GTR group.*
  Improvements:
  - Use of internal magnetic field gradient (speeds up refit by 30%)
  - Make TRF clusters only as needed (speeds up refit by another 20%)
  - Support for tilted dca surface (non-zero beam slope)
  - Energy loss on cylindrical surfaces fully accounted for

- Histogram Track Finding (htf): divides detector in to slices in (□,□) and uses the Hough transform to reduce initial number of combinations. *Author: S. Khanov*
  Improvements:
  - Now possible to use a mixture of axial and 3-dim CFT hits
  - Selection in z-r in SMT templates
  - Optimized options

- Alternative Algorithm (aa): starts from any combination of 3 hits in different SMT super-layers. Track candidates are extended towards CFT. Initially uses 1-dim SMT clusters only. Author: *G. Borissov*
  Improvements:
  - Memory leak fixes
  - Non-uniform *B* field (as used by other algorithms) instead of uniform *B* field
  - Inclusion of CFT-based track search, in addition to SMT-based search
  - Use of 2-dim SMT clusters

Note the absence of the Elastic Algorithm (ELA, by A. Haas) considered in the previous round, but omitted due to the judged shortfall for future support.

Preparatory work included the reprocessing of large samples of hadronic data samples (order 500k) to test the robustness of the improved aar (aa + gtr refit) code.

In collaboration with the Global Tracking Group and the code authors, it was decided to evaluate the following combinations of algorithms:

1. p13 = unmodified p13.05.00 (gtr+htf combined)
2. gtrhtf = same algorithms as p13, but with latest version of htf.
3. aar = latest version (v02.04.01) of aa alone, with gtr refit

4. htf = latest version of htf (v00.02.14) alone
5. htfaa = htf and aa run sequentially, with results merged at the end and followed by gtr refit
6. htfaafast = as above, but optimized for CPU usage and applying aa filters to the merged results (available Feb. 2003; this version replaces htfaa as a candidate for production, but some studies evaluated htfaa only as samples were available).

All of these packages are available already in p14 and the tracking algorithm that is run in `d0reco` can be selected via a RCP switch (in `GtrMetaPkg.rcp`)

3. Reprocessed Data and Monte Carlo Samples

The committee requested algorithm/ID/physics groups to provide their preferred data and Monte Carlo samples to evaluate algorithm performance. The samples selected for processing and the locations of the samples are described in greater detail on the TARC status pages. A brief summary of the samples is:

Data:

Sample	*Approx. Size, each alg.*	*Description*
Run 168382	~100k (371k aa)	Generic hadronic, run with a longer average CPU time/event on farm
Run 168391	~100k (403k aa)	" with shorter average CPU time/event on farm
$J/\psi \to \mu\mu$	~50k ~5k	Selected using track matching with p13; i.e., track bias on J/ψ Selected using local muons only, no track bias on J/ψ
$Z \to \mu\mu$	~500, 2.4k	Two samples; loose-loose local muon selection
μ + jets	~10k	Muon associated to jet with dR < 0.7
$Z \to$ ee	2.4k 1.5-2.4k	At least one cluster with track match Same sample, no ADC cut
EM-id QCD backg.	4k	Di-jet QCD events likely to result in EM-id background.

Monte Carlo:

Sample	*Approx. Size, each alg.*	*Description*
Light quark QCD	~33k	For future, needs harder p_T distribution to be useful for b-id mistag evaluation
$\tau \to$ hadrons	8.5k	Single and 3-prong decays
Top	10k	Electron + jet

The samples were reprocessed on the CAB system, with many thanks to Erich Varnes for his tireless efforts in this task. For run 168382, the full `d0reco` took between 14.3 sec/event (htf) to 25.2 sec/event; (htfaafast); for run 168391, 9.9 sec/event (htf) to 14.4 sec./event (htfaafast).

4. Comparative Performance of Algorithms

Performance results were presented as progress reports at the global tracking and other algorithm/ID/physics group meetings (also see TARC web pages). More detailed presentations can be found linked to the agendas (userid: d0agenda, passwd: D0RunII) of the TARC2 Jamboree on 20 Feb. and a follow-up meeting during the General Computing and Software Meeting. It should be noted that progress and requests for studies were often hampered by (understandably!) higher-priority work of preparation and finalizing analyses for Moriond.

During the studies it was discovered that there is an artificial lower limit of p_T that is stored in the thumbnail. Any p_T values lower than 0.25 GeV are set to 0.25 GeV (resulting in a spike at that value when looking at this thumbnail variable). A fix is being investigated, and tracking improvements may be even more significant once this limit is lowered.

- **Timing:** The timing on data events is given above. On MC events, 0 min. bias, times range from ~2 (aar) to ~4 (gtrhtf) GHz-sec/event and for 2 min. bias, from ~5 to ~10 GHz-sec/event. On MC events, aar is fastest at low luminosity, while htf is fastest at high luminosity (>6 min. bias). For both data and MC, the time per event for htfaafast is slightly below the sum of aar and htf alone. *On MC events, htfaafast is slightly faster than either default p13 or gtrhtf over all luminosities considered.*

- **Tracking performance:** In general terms, aar is better at low p_T and high impact parameter and has a lower fake rate. Htf has higher efficiency at high p_T, has higher efficiency in difficult □ regions like the overlap region, and is less affected by high luminosity. The algorithms together in htfaafast combine the advantages of each while retaining the benefit of the gtr refit (that takes about 10% of the tracking time). Compared to p13, greatest gains of ~10% are made in efficiency at the lowest p_T, and a few percent better at $p_T >$ 0.5. More striking is a fake rate less than 2% across p_T, compared to p13 fake rates of 3 to 5 times higher, while maintaining a comparable misreconstruction rate falling from ~8% at low p_T to ~2% at high p_T.

- **Primary vertices:** aar finds more tracks at lower p_T, and hence has a larger number of tracks assigned per vertex. The primary vertex resolutions (beam spot widths) for all algorithms are comparable, with the resolution being slightly better in aar. Htfaafast does significantly better than the other algorithms in some of the samples, notably the local dimuon sample.

- **High-p_T tracking in data:** From studies of $Z \rightarrow ee$, tracking efficiency in p13 is better than either aar or htf alone, but a few percent of tracking efficiency can be gained over p13 using htfaa or htfaafast. Tracking efficiency ranges from 82–86% in p13, 84–88% in htfaafast, and 85–90% in htfaa. Fake rates appear comparable, but need more statistics for a definitive conclusion. These results are confirmed in $Z \rightarrow \mu\mu$, where signal minus background yields of ~6.5% more signal events in htfaa compared to gtrhtf. Very preliminary studies were made indicating further gains in tracking efficiency by dropping the ADC cut on CFT clusters, but this needs more study.

- **Low-p_T tracking in data:** Improvements in low-p_T tracking were gauged by examining signal minus background yields for various low-mass resonances. A consistent pattern emerged with htfaa(fast) producing the highest yields, with aar a close second, and the worst being htf alone. Comparing htfaa(fast) to p13, 26% more $J/\psi \rightarrow \mu\mu$, 43% more $\Lambda \rightarrow p\pi$, 110% more $K_S \rightarrow \pi\pi$, 63% more $\phi \rightarrow KK$, and 73% more $K^* \rightarrow K\pi$. Higher backgrounds were sometimes observed, but even modest changes in selection criteria easily removed these with very small changes in efficiency. It should be noted that the *B* physics group obtains even further gains of up to a factor of two using a special "extended AA" with a lowered p_T limit on tracking (but which significantly increases CPU time).

- **τ reconstruction:** MC samples of taus decaying hadronically to 1 or 3-prongs were studied, giving some information on tracking performance in a crowded environment. Modest gains in 1-prong tracking efficiency from 95% to 98% were observed comparing p13 and htfaa. The same comparison shows more significant gains in 3-prong tracking efficiency across p_T, with improvement in efficiency from ~75% increasing to ~90% for taus with $p_T > 40$ GeV.

- **B-tagging in MC (top samples):** Jet lifetime impact parameter (JLIP) and secondary vertex (SV) tagging were examined in the top MC samples. The best efficiency was obtained for the JLIP method with the htf algorithm, but it gave the highest fake rate. Htfaafast yields the least fake rate (0.44%) with comparable efficiency (31.6%). For SV tagging, htf yields the best efficiency (32.9%) and smallest fake rate (0.37%), although statistics are marginal to distinguish algorithms at this level. The performance of all algorithms was better than default p13.

- **B-tagging in data (μ+jet samples):** The data sample of muons associated to jets is enriched in b-quark content. P13 is the least efficient algorithm in data for finding secondary vertices and number of tracks per secondary vertex. Htfaafast gives the most number of tagged jets and secondary vertices in each event with aar being a close second. Htfaafast secondary vertex tags 6.8% of jets compared to 5.0% in p13, and 9.9% versus 5.4% following further b-content enrichment after a muon p_T^{rel} cut. Similar results were obtained for impact parameter tags, with htfaafast producing the most double tags in this sample.

Possible problems in htfaafast with respect to tags with large secondary vertex significance are still being investigated.

5. Recommendations

Based on the studies performed on the tracking algorithms considered for p14, the committee recommends to:

- ***implement the htfaafast algorithm as the default algorithm for p14;***
- ***encourage developers to continue code development and optimization for p15, particularly in areas to speed up the code;***
- ***carry out a similar study for the next major release of production track reconstruction code; and***
- ***focus on the remaining loss of high-p_T track efficiency and resolution (indicated by the data/MC Z to lepton mass peak resolution disagreement)***

This recommendation is based on:

- consistent superior performance of the algorithm over a wide range of tracking and physics environments;
- that htfaafast tries to combine intelligently the best merits of each of the alternate track finders and includes the gtr refit;
- still runs slightly faster than the current default p13 track-finding algorithm.

An outstanding issue (related to the last recommendation) is the benefit of removing the ADC cut on CFT clusters. The very preliminary study that was performed is insufficient to come to any conclusion or venture a recommendation concerning this question.

Initial work by Jim Kowalkowski already shows impressive tools with which to identify the CPU "hot spots" that could be addressed to speed up the code as mentioned in the second recommendation.

6. Acknowledgements:

We would like to acknowledge the contributions of colleagues who performed or helped with the studies of the TARC samples and provided feedback: Brad Abbott, Paul Balm, Guennadi Borrisov, Frederic Deliot, Silke Duensing, Harald Fox, Yuri Gerstein, Herb Greenlee, Sasha Khanov, Jim Kowalkowski, Abaz Kryemadhi, Chunhui Luo, Yildirim Mutaf, Flera Rizatdinova, Brian Sabbey, Stefan Soldner-Rembold, Sherry Towers, Boris Tuchming, Gordon Watts, Terry

Wyatt, and Daria Zieminska. Apologies to those whose names we may have inadvertently omitted.

The efforts of Sasha Khanov and Guennadi Borrisov in improving the alternate tracking algorithms and collaborating in their combination is particularly appreciated.