

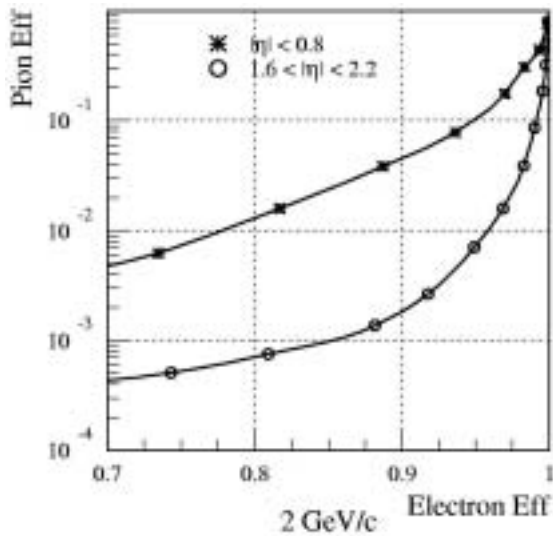
TRT Reconstruction Perspective

F. Luehring
RTF Open Session
March 5, 2003

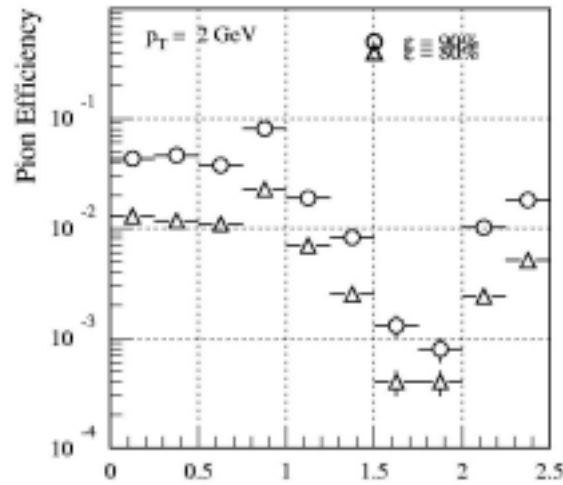
Previous Experience

- Disclaimer:
 - I have not used Athena. All of the reconstruction that I have done used Atlsim to generate a standard ntuple. I analyzed the ntuple using kumac macros and comis routines.
 - I have only worked with the inner tracker.
- What I have done:
 - Used xKalman with single tracks to study the π rejection obtained by using TR.
 - Used xKalman with full luminosity pile-up events to look at track finding efficiencies and fake track rates.
 - Lots and lots of code debugging generating single tracks and reconstructing them.
 - I am an user and not a developer of reconstruction code.

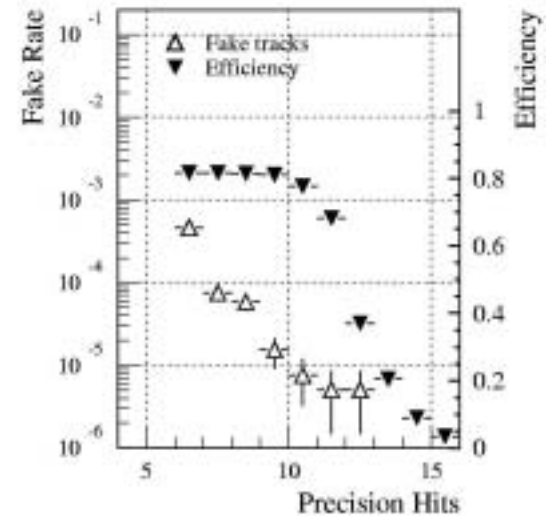
Typical Results (DC1 Code)



TRT Pion Rejection vs. Electron Efficiency



TRT Pion rejection vs. $|\eta|$



ID Track Finding Efficiency and Fake Rate

The TRT Experience

- The results obtained vary from software release to software release.
 - Even results from sequential software releases will vary slightly (changes in system libraries and compilers?).
 - It is very hard to compare results that are separated by a few years (i.e. Physics TDR results to the current results).
 - It is almost impossible to run an old reconstruction code version against a new version events or vice versa.
 - By old I mean Physics TDR or ID TDR.
- The time to reconstruct an event in the inner detector is reasonable for the studies I do with an ~ 2 GHz P4:
 - 200000 π take about 12 hours to generate/reconstruct
 - 500 full L pile-up events take about 2 hours to reconstruct.

Remarks

- Two requests from a user working in a small group at a remote institute:
 - The reconstruction should be easy to install locally. I have access to about 60 local machines but any installation of ATLAS software is done by me.
 - Help (instructions, FAQs, examples, problem reports) should be available via a web page that easily found in an obvious place. There should be only one set of web pages.
 - Access to human consultants would be even better...

Remarks

- A few thoughts about running the reconstruction interactively:
 - Interactive running should be easy to do.
 - The reconstruction should be fast to initialize
 - The user interface should be consistent and simple.
 - It should be possible to generate and reconstruct single tracks interactively. This critical for debugging
 - There should be simple, clear graphics to display the generated tracks/hits, digits, found tracks, hits on tracks etc.
 - It should be possible to quickly modify a few routines and rerun simple test tracks.

Remarks

- Final stray thoughts:
 - Please ensure that the reconstruction code is clean:
 - Keep the code organized so that it is possible to understand it. It should be possible to use the debugger to find problems in the reconstruction code in addition to the user algorithms.
 - Create a good test suite that catches problems before the code is released.
 - Make it possible to switch simply between Monte Carlo and actual data.
 - Reconstructing actual data and Monte Carlo data should use the same algorithmic code. Only the I/O routines should change.
- Interaction with the reconstruction should be light-weight!