8.882 LHC Physics

Experimental Methods and Measurements

B Physics Triggers and Lifetime Fitting

[Lecture 19, April 15, 2009]

Organization

Project 2

- status: nothing yet received

Project 3

- *B* lifetime measurement is due May 2

Final Conference project

- starts: 12:00 Kolker room on May 19
- each student gives
  - 17 minutes presentation
  - +3 minutes discussion
- agenda to be published

Final Conference Project

LHC Physics: “Experimental Methods and Measurements”

- C.Paus: Welcome and LHC Overview
- ?: Search for Higgs in $H \to ZZ^*$
- ?: Search for Higgs in $H \to WW^*$
- ?: Search for Higgs in $qqH \to qqWW^*$

MIT

Physics Colloquium Series

'09 SPRING

The Physics Colloquium Series

Wednesday, April 15 at ***4:15 pm*** *in room* ***4-349***

Sidney Drell

SLAC National Accelerator Laboratory

Steps Toward a World Free of Nuclear Weapons: Rekindling the Vision of Reagan and Gorbachev at Reykjavik

For a full listing of this semester's colloquia, please visit our website at **web.mit.edu/physics**

Lecture Outline

B Trigger Strategies

- CDF trigger system overview
- high momentum displaced track triggers
  - experimental implications

A bit of probabilities and fitting

- probability density functions and expectation value
- least square method
- likelihood method
- lifetime example
- some technicalities for our application

Review: Hadron Collider Triggers

Trigger Systems at Hadron Colliders come in levels

- per level: time to analyse and data to analyse increase
  - Level 1 – implemented in hardware, custom electronic boards
    - rates at CDF: 3 MHz → 30 kHz or CMS: 25 MHz → 300 kHz??
  - Level 2 – implemented in programmable hardware (FPGAs*)
    - rates at CDF: 30 kHz → 1 kHz (CMS has no Level 2, merged with Level 3)
  - Level 3 – implemented as PC farm, runs reconstruction program
    - rates at CDF: 1 kHz → 100 Hz (CMS has no Level 3)
  - CMS High Level Trigger (HLT) merges Level 2 and Level 3
    - rate: 100k → 100 Hz
- dead timeless design by long trigger pipelines: parallelized

* Field Programmable Gate Array

Trigger System at CDF

Speed

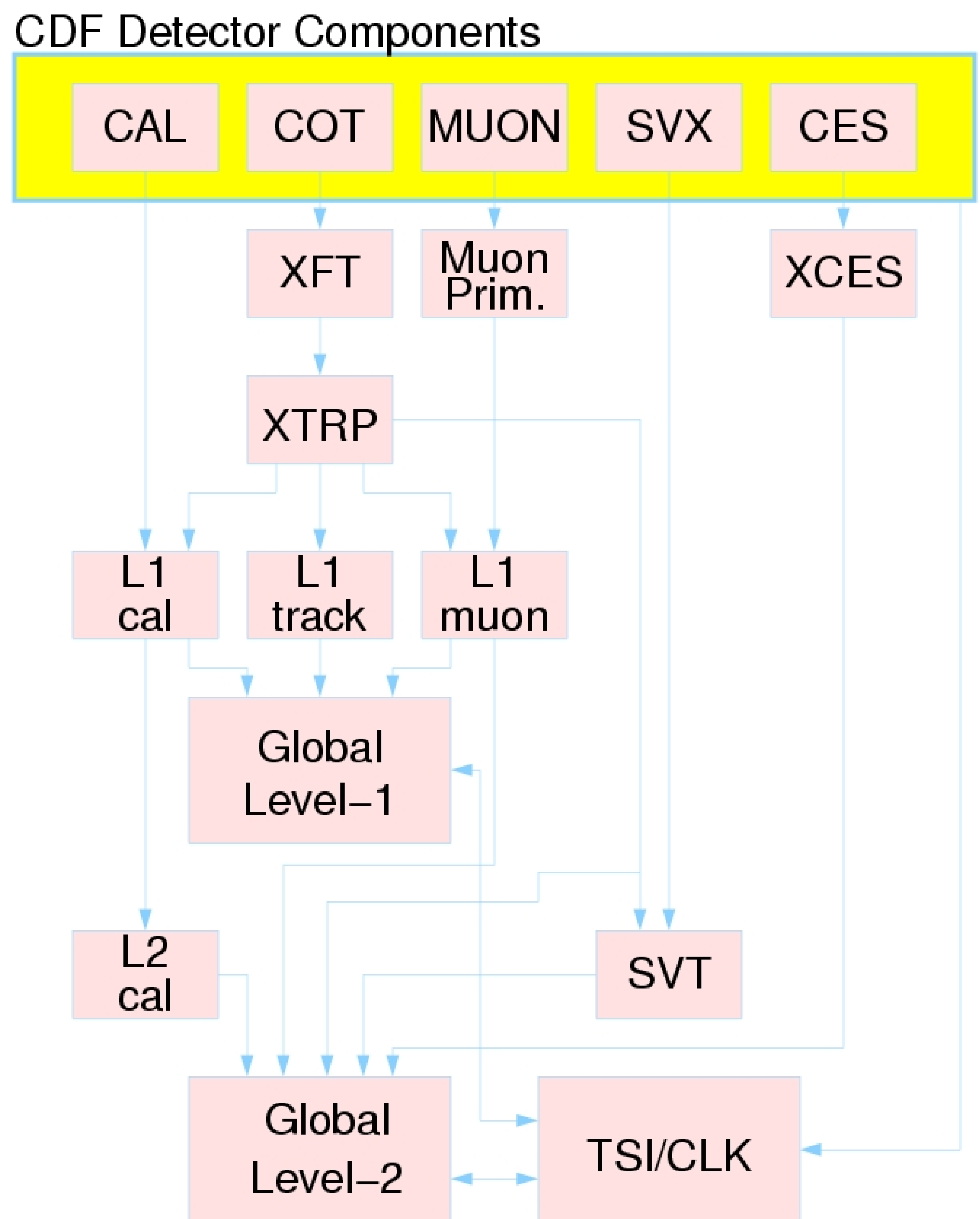
- trigger primitives are fast digitized objects
- less precise but fast

Trigger Primitives

- muon stub
- electron cluster
- track

Triggers

- combination of trigger primitives and application of certain cuts
- *ex.* muon is already combined with track at level1



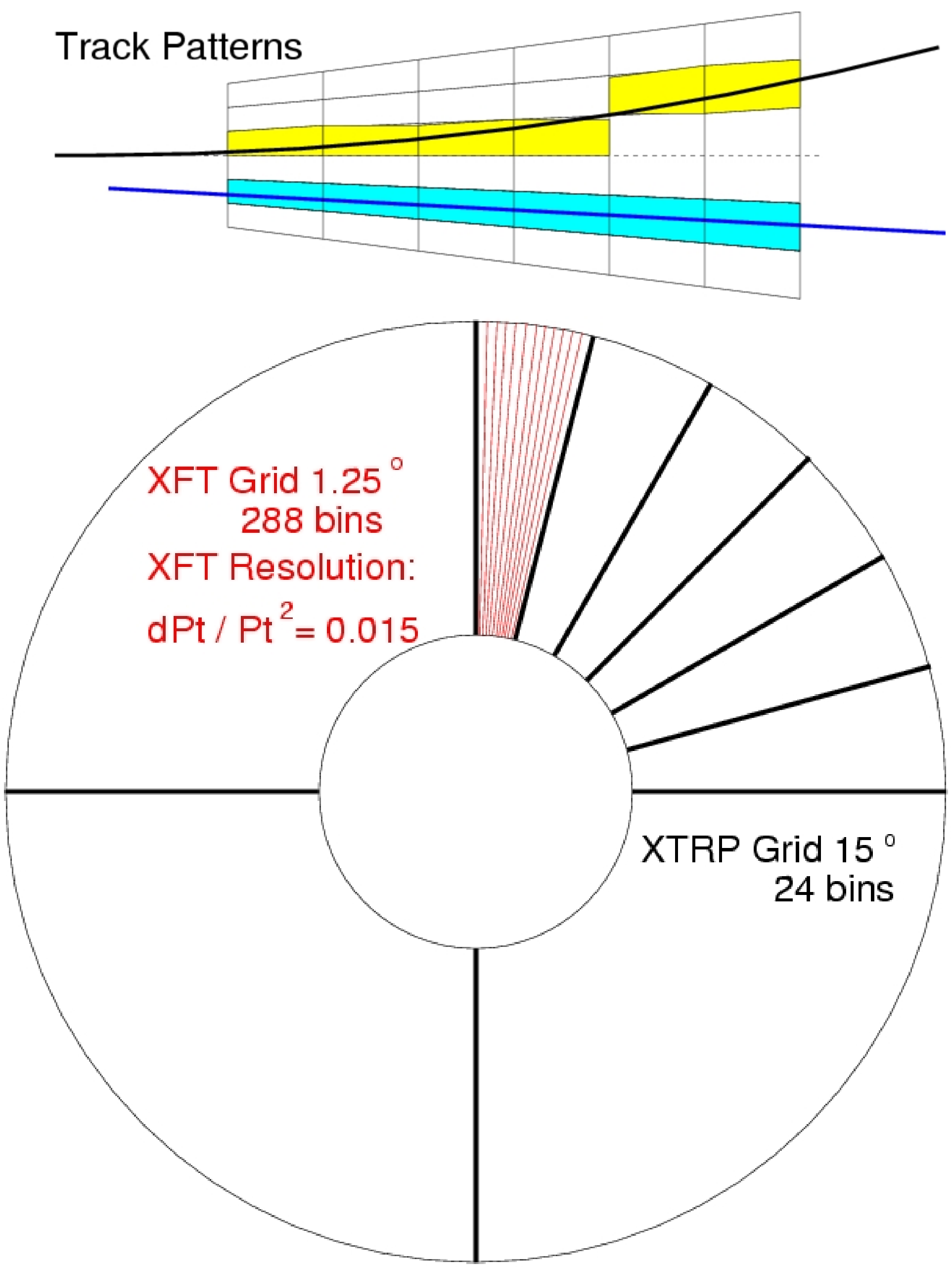
Outline of Track Trigger

Tracking is 2 dimensional

- look for high momentum

Main idea: patterns/roads

- granularity gives resolution



Hardware used

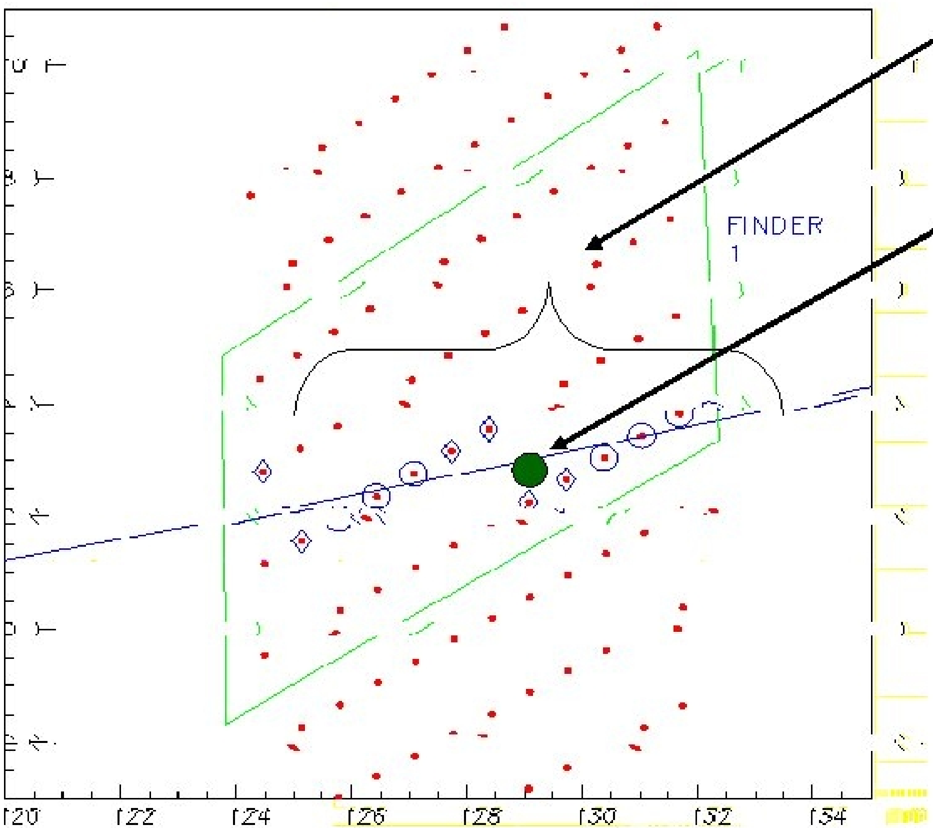
- four axial layers of COT
- XFT/XTRP trigger boards

Three step process

- hit finding
- segment finding
- track finding

Hit and Segment Finder

Track segments are found by comparing hit patterns in a given layer to a list of valid patterns or "**masks**".



○ "Prompt" hit

◊ "Delayed" hit

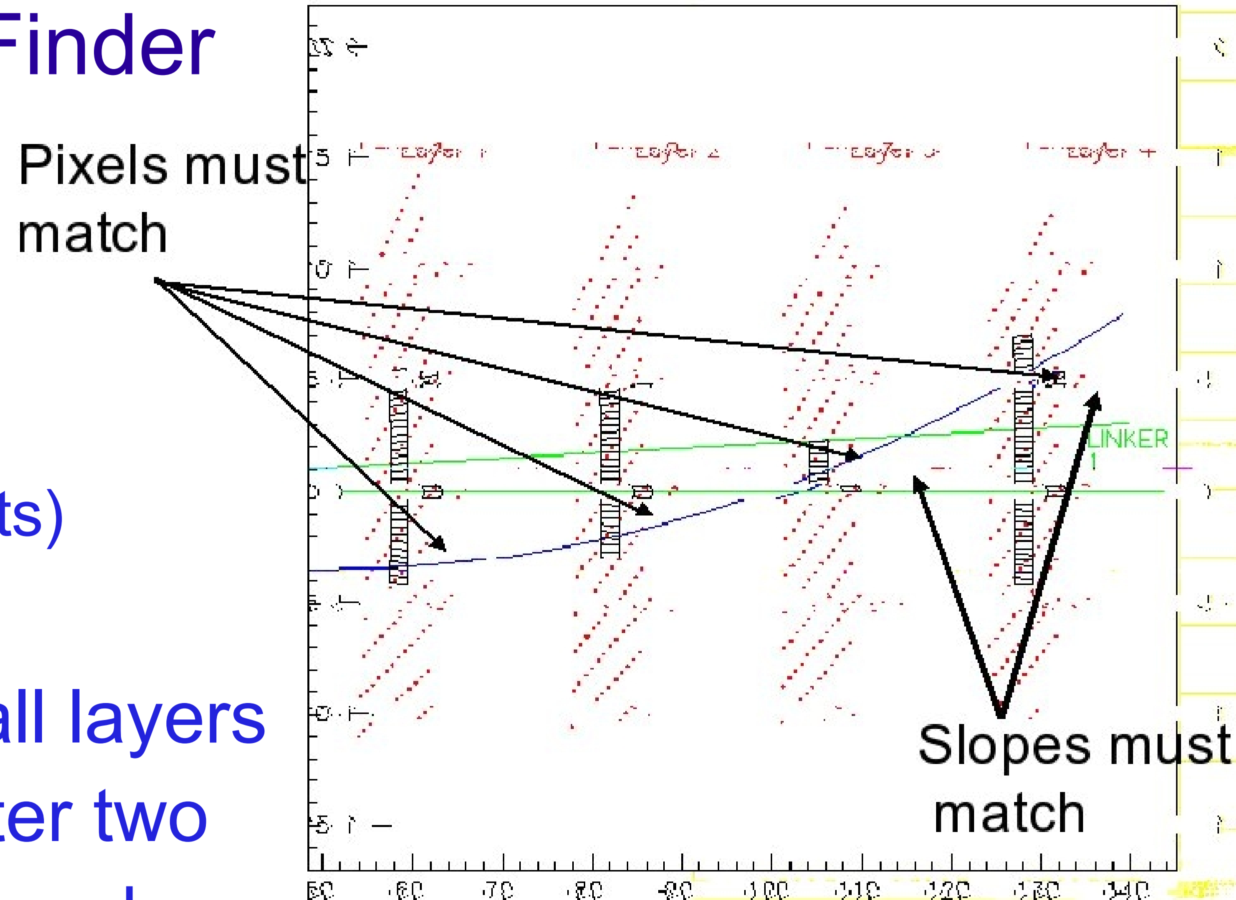
Hits : 2 bins - Prompt or delayed
Mask : A specific pattern of prompt and delayed hits on the 12 wires of an axial COT layer generated by simulating tracks
Pixel: represents the **phi** position of the track at the midpoint of the cell.

Layer	Cells	Masks	Pixels
1	192	2304	166
2	288	3456	227
3	384	2304	292
4	480	2880	345

Segment Finder and Linker

Output of Segment Finder

- inner two layers
  - phi position only
- outer two layers
  - phi position
  - slope of segment (3 bits)



Segment Linker

- uses phi position in all layers
- and slopes in the outer two
- combines them to a road
- 288 linkers (per 1.25°) each linker uses a table of 1200 predefined roads

XTRP Board

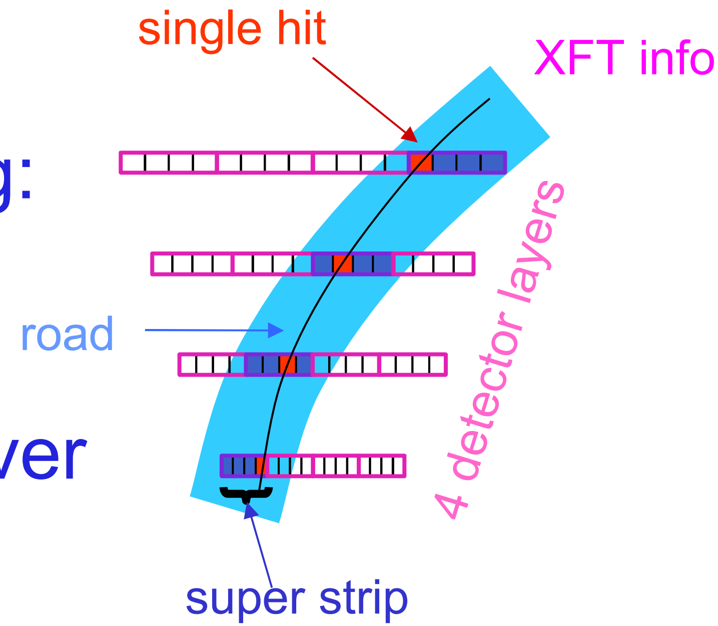
Responsibility

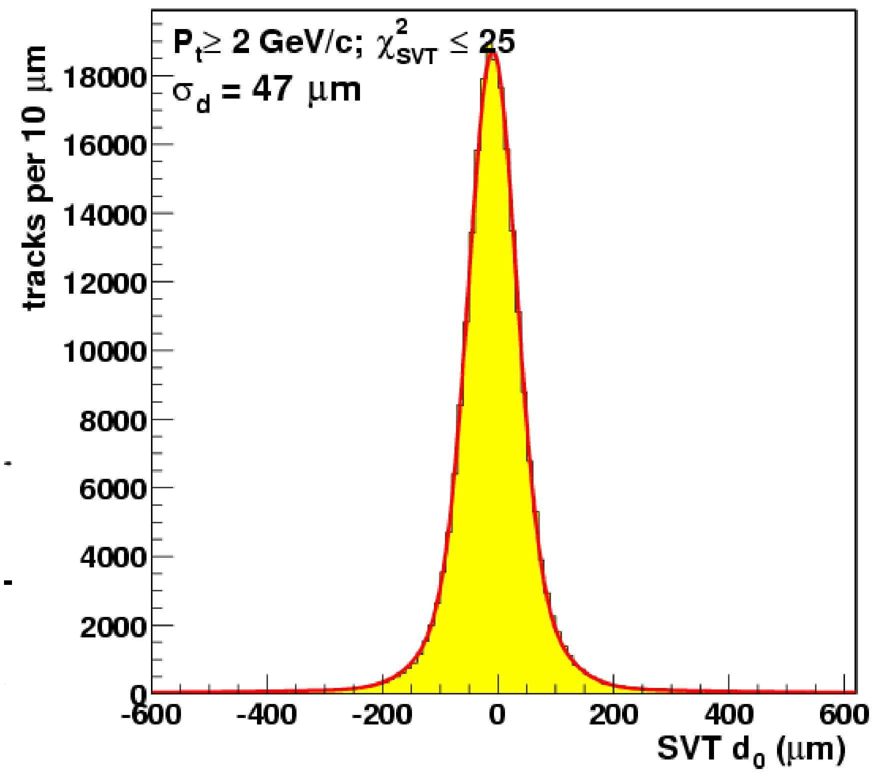
- select the best tracks (highest p_T)
- distribute XFT tracks to other systems
  - level1 muons trigger
  - level1 calorimeter trigger
  - level1 track trigger (pass info)
  - level2 track trigger, SVT (pass info)
  - level1 and level2 global triggers
- extrapolate position to corresponding detector
  - extrapolate to muons chambers
  - extrapolate the calorimeter

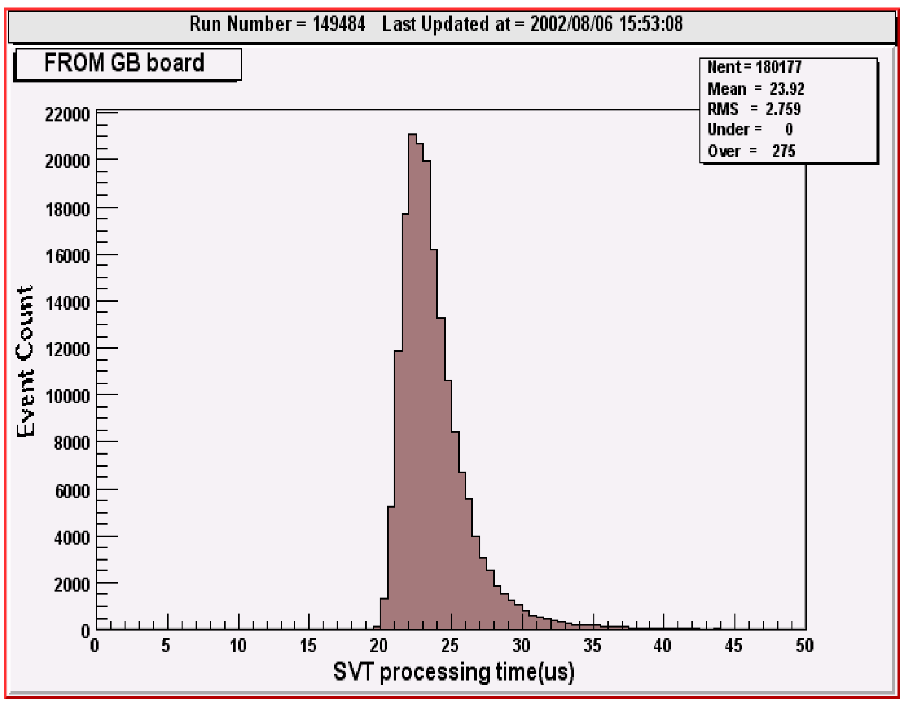
Secondary Vertex Tracker (Trigger)

Another trigger based on patterns

- resolution and speed are astonishing: 47μm at 15 μs
- 32,000 roads to be searched
- first such trigger at hadron collider ever







Secondary Vertex Tracker (Trigger)

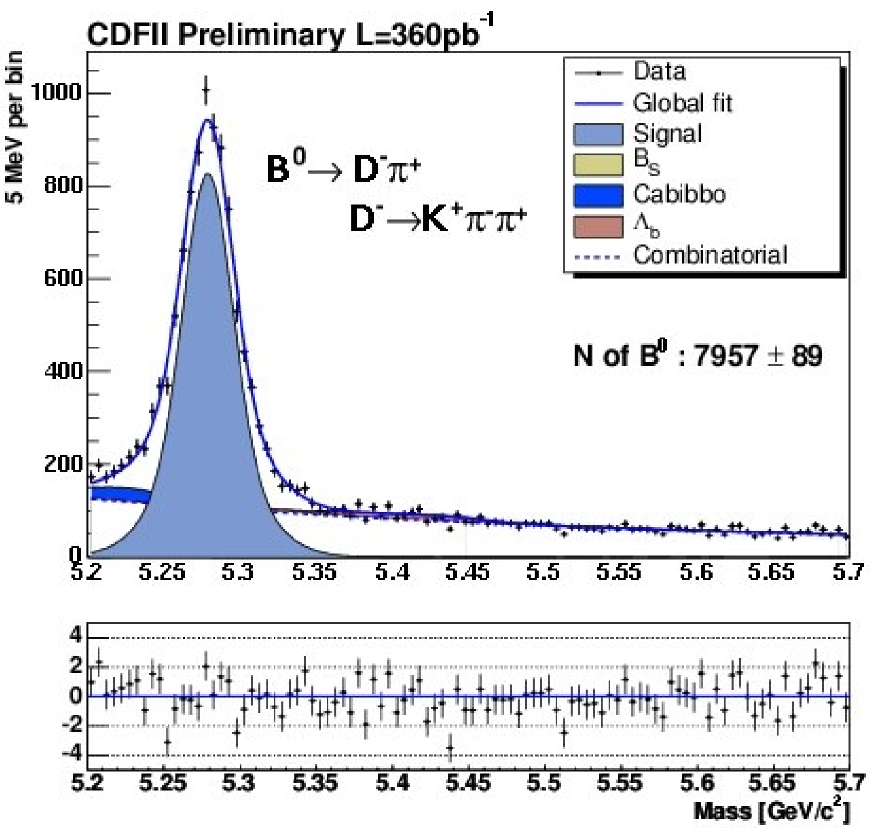
Typical trigger requirements

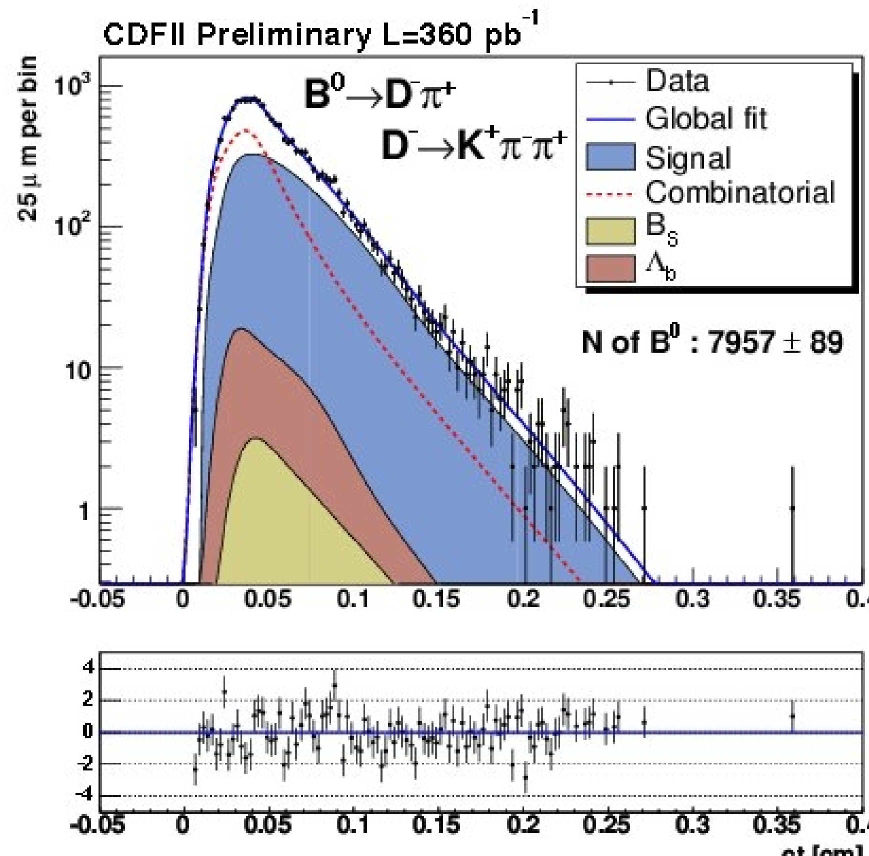
- two tracks with $p_T > 2$ GeV
- $p_{T1}+p_{T2} > 5.5$ GeV
- opening angle: $0° < \Delta\varphi < 135°$ (level 2: $12° < \Delta\varphi < 90°$)
- impact parameter: $120\ \mu m < |d_0| < 1$ mm
- upper cut due to pattern limitations (not enough memory/time to search through more pattern)
- increased sensitivity to purely hadronic *B* decays by 4-5 orders of magnitude: this was the biggest upgrade for B physics in Run II of the Tevatron
  - $B \rightarrow D\pi$ (B_s mixing), *etc.*
  - $B \rightarrow \pi\pi$ (direct *CP* violation, maybe α)

Secondary Vertex Tracker (Trigger)

Problems you have to deal with

- intrinsic bias of proper time distribution
- sharp prompt peak disappears, reducing the background
- also gets ride of some signal (bias)

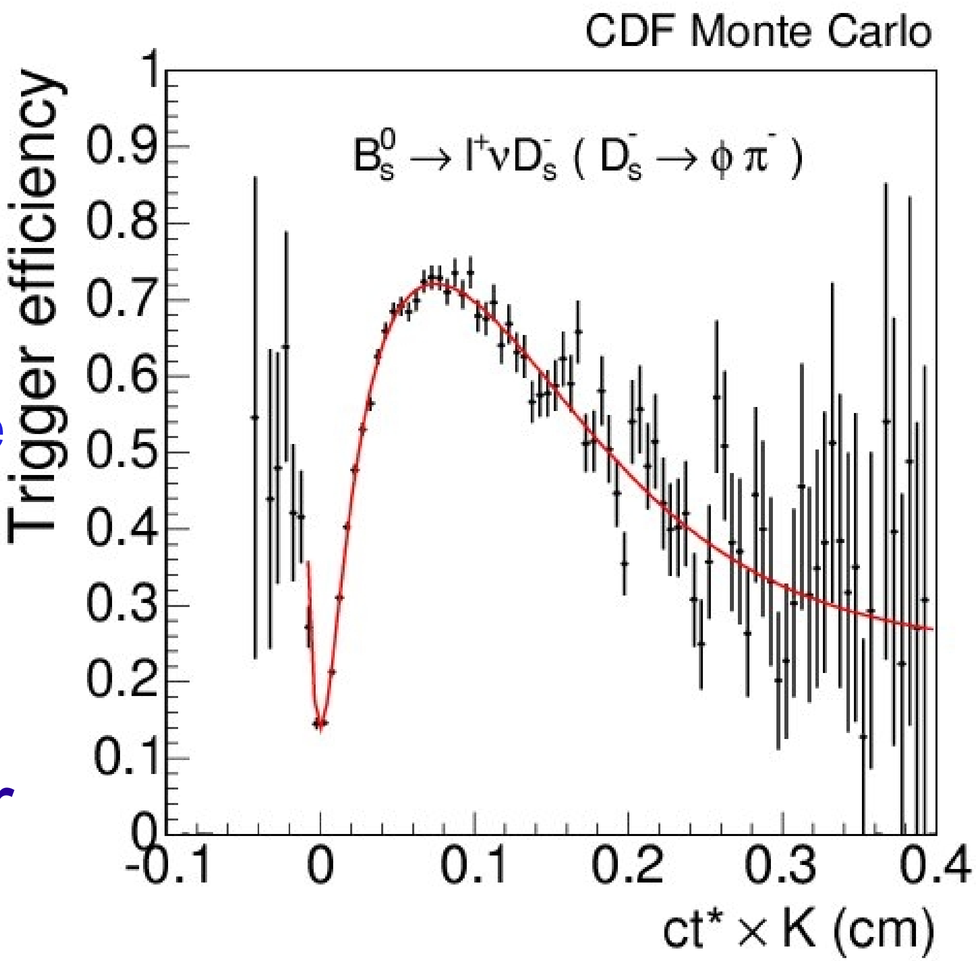




Trigger Bias Curve

Simulation of trigger bias

- events we need to study are not in the data, to first order
- needs careful Monte Carlo study: generate events and keep them
  - silicon hit occupancy needs to be reasonably well simulated
  - check hit resolution function carefully, too
  - Monte Carlo must also have the right lifetime ....



The plot is for semileptonic events but looks similar for purely track based trigger

Likelihood Fits

Have found a spectacularly nice talk from Colin Gay given at a Particle Physics students retreat. Please check it out, I am stealing a lot from there:

- http://physics.bu.edu/NEPPSR/TALKS-2006/Likelihood_Gay.pdf

You also might want to check the statistics talk on the same site:

- http://physics.bu.edu/NEPPSR/TALKS-2006/Statistics_Blocker.pdf

Probability

Assume *X* is a random variable, like for example our *ct* or *m* of an event

- the probability to throw *X* into window *[x,x+dx] is P(x)dx*
- *P(x)* is the probability density function (PDF) for *X*
- if we look at the entire range of possible values for *X* like for example: $[-\infty, +\infty]$ we can be sure that X at any throw will take one of those values and this means the integral of the PDF should be one

$$\int_{-\infty}^{+\infty} P(x)dx = 1$$

The expectation value of function *f(x)* over *P(x)*

$$E(f) = \int_{-\infty}^{+\infty} f(x)P(x)dx$$

Probability

Expectation values we are using all the time

- normalization: *f(x) = 1*

$$E(1) = \int_{-\infty}^{+\infty} 1P(x)dx = 1$$

- *mean value: f(x) = x*

$$E(x) = \int_{-\infty}^{+\infty} xP(x)dx = \langle x \rangle$$

- *variance: f(x) = (x - <x>)²*

$$E((x - \langle x \rangle)^2) = \int_{-\infty}^{+\infty} (x - \langle x \rangle)^2 P(x)dx = \sigma^2$$

Conditional Probability

Conditional probability *P(x|a)*

- is the PDF for *X* given that *a* is true
- $P(m|m_0)$ would be the probability that we measure the mass, *m*, for example for the Upsilon candidate if the true mass of the Upsilon was m_0

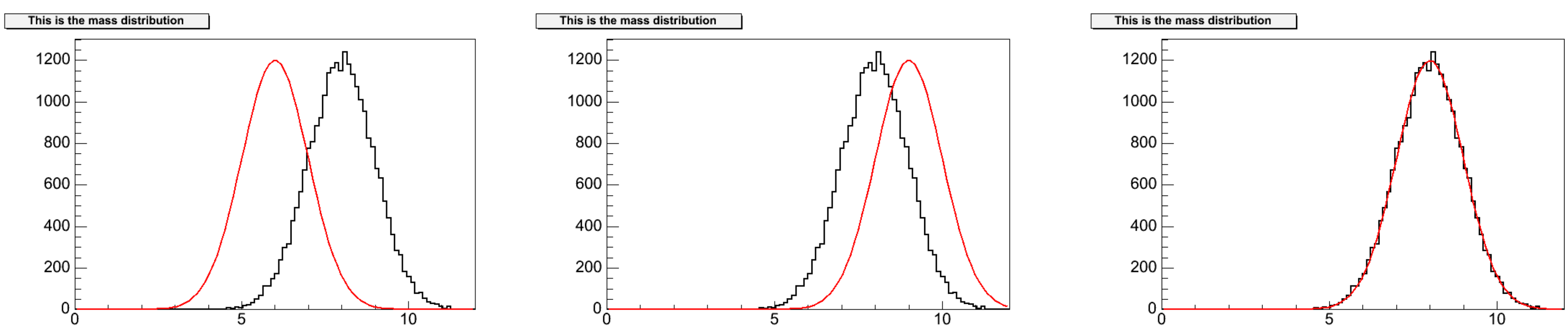
Application of conditional probability

- in Monte Carlo simulation for the Upsilon for example
  - we know the mass and the detector resolution: PDF defined!
  - now we can through the values for each specific event
- fitting the data, again the Upsilon data
  - we do not know the mass, so we have to solve the inverse problem
  - infer the mass given a limited number of measurements
  - might have to also determine the PDF ....

Samples

With our random variable *X*, which follows the PDF *P(x)*, we can generate a set of throws, lets say *N*:

- set of throws called a sample, represented by tuple $\{x_i\}$, for the example we have a set of mass measurements $\{m_i\}$
- visualize the result in a histogram and put on top the conditional probability density function $P(m|m_0)$
- chosen m_0 very likely not the correct one, shift it around until the curve “fits” the data, hmmm...
- we need to find a good estimator for m_0



Estimators

What are the requirements to an estimator?

- consistency: the value of the estimator should converge to the real value as our sample size goes to infinity
- efficiency: theory limits the variance about the true value of an estimator for a given sample size N (Minimum Variance Bound – MVB), if the variance of the estimator is equal to the MVB the estimator is called efficient
- unbiased: an estimator whose value is equal to the true value is unbiased

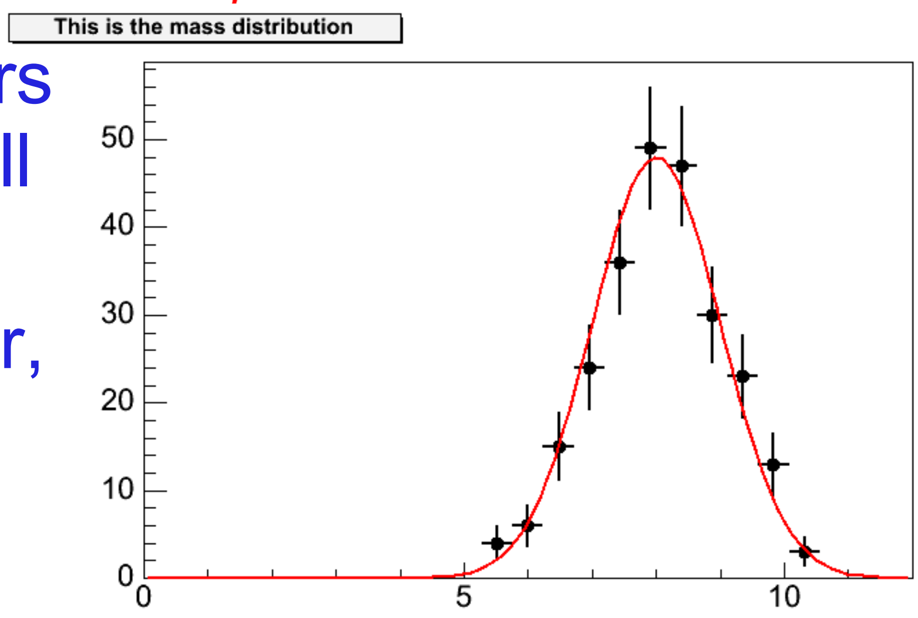
Examples of good estimators

- minimum of the $\chi 2$
- maximum of the likelihood

Least Squares Fit (χ2 Fit)

Given a set of measurements

- x_i and y_i, σ_i (histogram, bins, mean values, uncertainties)
- choose a function: $f(x_i|\vec{p})$
- function predicts y_i in dependence of parameter values: $\vec{p}$
- use the χ2 given by: $\chi^2 = \sum_i \frac{(f(x_i|\vec{p}) - y_i)^2}{\sigma_i^2}$
- estimator for set of parameters is given by the ones which will minimize the χ2
- likelihood is another estimator, which allows more complex formulations



Binned versus Unbinned

The least square fit, a binned fit

- provides reliable unique quality criterion for the fit parameter to evaluate whether the data comply with the general fitting hypothesis
- has problems if there are very few entries per bin
  - assuming Gaussian statistic: $\Delta n = \sqrt{n}$
  - is approximately correct for *n>10 ....*
  - for $n = 0$ uncertainty is zero, crashes least square
  - root is smart enough to exclude those bins... but parameter do not come out correctly (biased)

Likelihood formulation

- in general is unbinned (can be also binned)
- ideal for small statistics and sparse samples
- allows formulation of complex multidimensional problems

Maximum Likelihood Estimators

Similar to the Least Square ($\chi 2$)... but more powerful

- use our setup: *n* measurements of the mass $\{m_i\}$ and a given PDF of $P(m|m_0)$
- likelihood is given by $$L(m_0) = \prod_{i=0}^{N} P(m_i|m_0)$$
- for a given event it is likely that $P(m_i|m_0)$ is large if the parameters are chosen to be close to the correct ones
- the estimator is given by the maximum of the likelihood
- they are consistent and efficient
- extracting the maximum likelihood from the data by adjusting the parameters is called fitting
- but some care please:
  - maximum likelihood estimators are not always unbiased

Lifetime Likelihood

Probability density function for exponential decay given as:

$$P(t|\tau) = \frac{exp(-t/\tau)}{\tau}$$ with proper intrinsic normalization

The Likelihood:

$$L(\tau) = \prod_{i=0}^{N} P(t_i|\tau)$$ with measurements t_i from $1..N$

Technically, logarithms make more sense (use general minimization package minuit = TMinuit)

$$\log L(\tau) = \sum_{i=0}^{N} \log P(t_i|\tau) = N \log \tau + \frac{1}{\tau} \sum_{i=1}^{N} t_i$$

Lifetime Fit Solution

Analytical solution is here possible

- maximize: find derivative, set to zero extract parameter(s)

$$\frac{\partial(-logL)}{\partial\tau} = 0 = \frac{N}{\tau_{\text{fit}}} - \frac{1}{\tau_{\text{fit}}^2}\sum_{i=1}^{N} t_i$$

$$N\tau_{\text{fit}} = \sum_{i=1}^{N} t_i \quad \rightarrow \quad \tau_{\text{fit}} = \frac{1}{N}\sum_{i=1}^{N} t_i \quad (= E(t) = \langle t \rangle = \text{ mean})$$

the maximum likelihood estimator for the true lifetime

TMinuit provides numerical algorithm to find the maximum even if not analytically solvable

- fitting is an art form though and requires some serious experience: but it is used everywhere today!

Fitter for B Lifetime Analysis

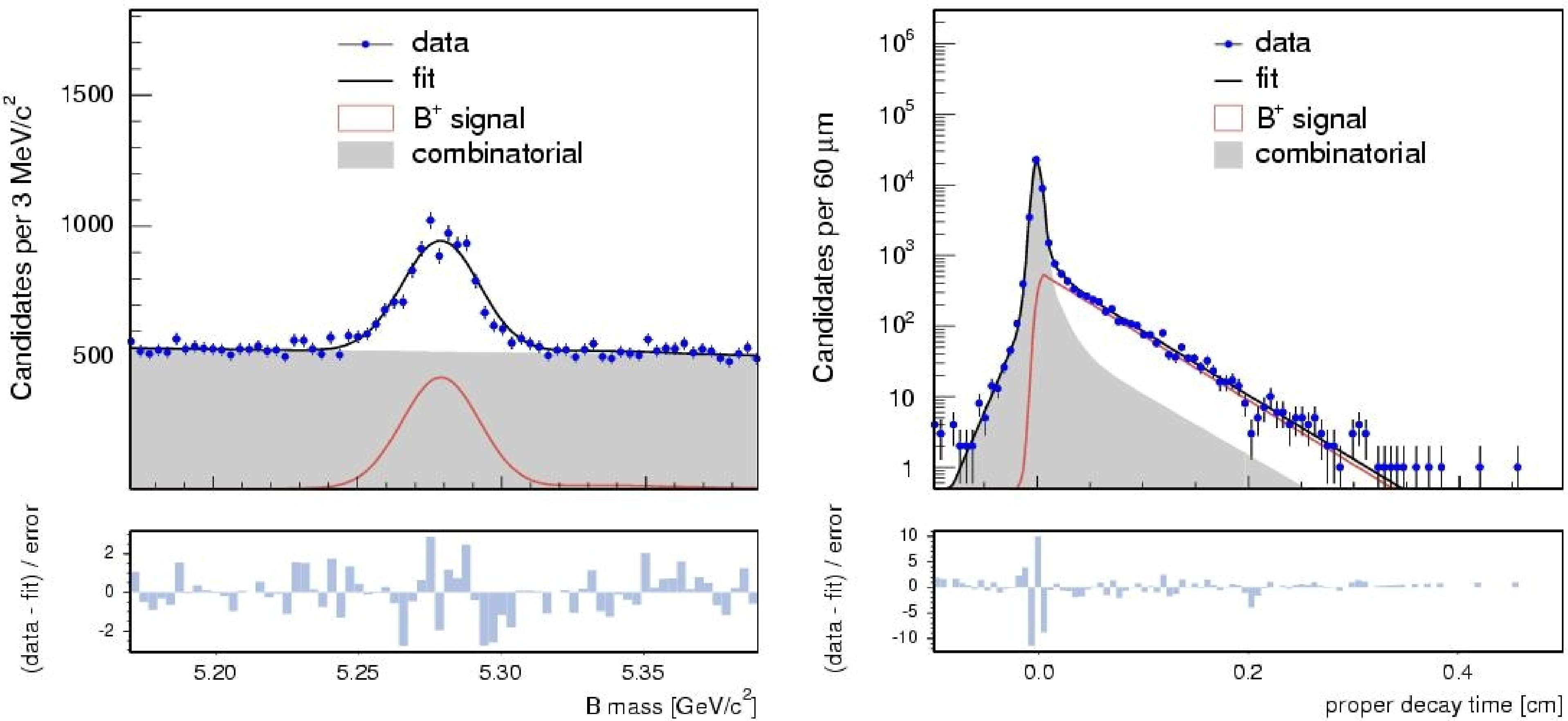
Three packages for the unbinned likelihood fitter

- Fitter – base package
- RemoteFit – facility to run the fitter on a large number of machines parallelizing the calculations (not needed for us, but creates a dependence)
- MixFit – the core package where you change things and implement your ideas
- packages are available in ~paus/8.882/614/Fitter.tgz
- compilation and shared library loading is explained
- something like
  - cd ~/8.882/614/results
  - root -l ../614/MixFit/scripts/fitCTauBuJpsiK.C
  - root -l ../MixFit/scripts/draw.C
  - and check the output :-) and the input ....

Fit Output Could Look Like

Remarks

- selection not optimal (you can do much better)
- fits have some details (do not worry too much)
- another lecture on this will follow.... TWiki update soon



Conclusion

Organization

- please sign up for your presentation

B Trigger Strategies

- displaced track trigger was a revolution for *B* physics
- but it brings additional problems: biased proper time

Fitting the data

- use probability densities to evaluate a likelihood
- likelihood are ideal to describe small/sparse data samples
- likelihood allows to implement very complex contexts
- likelihoods are more computing intensive then least square fits usually
- invest in fitting, it needs experience and is very useful

Next Lecture

Fitting and sanity checking

- likelihoods and fits
  - statistical uncertainties
- checking whether it makes sense
  - goodness of fits
  - projections

Sophisticate Selections

- likelihoods
- neural networks