BABAR Analysis Document #1698, Version 2
January 19, 2007

BaBar DCH $\frac{dE}{dx}$ calibration and a new technique of energy loss calculation.

Korneliy Todyshev

BINP, Novosibirsk

Abstract

This work gives a survey of the current and new features of the DCH $\frac{dE}{dx}$ calibration. A new ionization energy loss calculation strategy is presented. The procedures of the getting calibration parameters are described.

Contents

1 Introduction

To calculate energy loss of the charged particle passed through sensitive volume, it is necessary to apply corrections where hit and track level information are used. The energy-loss distributions of different particles are compared using the truncated mean or the maximum likelihood method. The truncated mean (TM) method is simpler and easy to realize.

This work is intended as description of correction procedures for systematics effects of ionization measurements in the BABAR drift chamber. The new approach of the energy loss calculation are presented. This technique occurred from optimization of the TM method and gives significant improvement of the $\frac{dE}{dx}$ resolution.

2 Amplitude systematic corrections

2.1 Fit functions.

If we deal with dependence of the $\frac{dE}{dx}$ versus some variable we have to fit $\frac{dE}{dx}$ spectrum, which have a form like Landau distribution. Usually the following parameterizations are used:
1) Simple Landau form:

$$f(q) = N * e^{-0.5*(v - e^{-v})}, \tag{1}$$

where $v = (\frac{dE}{dx} - \frac{dE}{dx}_{m.p.})/\xi$,
$\frac{dE}{dx}_{m.p.}$—the most probably $\frac{dE}{dx}$,
ξ—the width of distribution, N is a normalization factor
2) Unsymmetrical Gaussian parametrization [1, 2]:

$$f(q) = N * e^{-0.5*\left(\frac{\left(1+\frac{sinh(A*\sqrt{\log 4})}{\sqrt{\log 4}}\frac{(\frac{dE}{dx}-\frac{dE}{dx}_{m.p.})}{\sigma}\right)^2}{A^2}+A^2\right)} \tag{2}$$

where $\frac{dE}{dx}_{m.p.}$—the most probably $\frac{dE}{dx}$, A–asymmetry parameter.

One should note, that $\frac{dE}{dx}$ is normalized according to the Bethe-Bloch parameterization (the normalization factor is $\frac{BetheBlochFunction_{minimal}}{BetheBlochFunction}$) for given values of momentum and particle hypothesis.

The $\frac{dE}{dx}$ spectra are accumulated in normalized arbitrary units[n.a.u.], it allows to consider all particles types together if it is necessary. [1] The second parametrization (2) gives a little bit better result than first, but more slowly. The examples of the $\frac{dE}{dx}$ spectra are shown on Fig.1,2

2.2 Layer by Layer and Wire by Wire correction.

Different layers of DCH have various gas gains it takes place also for different wires in the same layer To avoid this discrepancy, individual spectra of $\frac{dE}{dx}$ are accumulated for every layer and every wire. After getting the most probably values of energy loss — $\frac{dE}{dx}_{m.p.}$ for each histogram the correction coefficients are obtained. The figures 3,4 show dependences $\frac{dE}{dx}_{m.p.}$ versus wire and layer number.

[1] To calculate correction coefficients at each considered dependence we have to divide $\frac{dE}{dx}$ spectra on common most probably value, in that case the getting values are about 1.

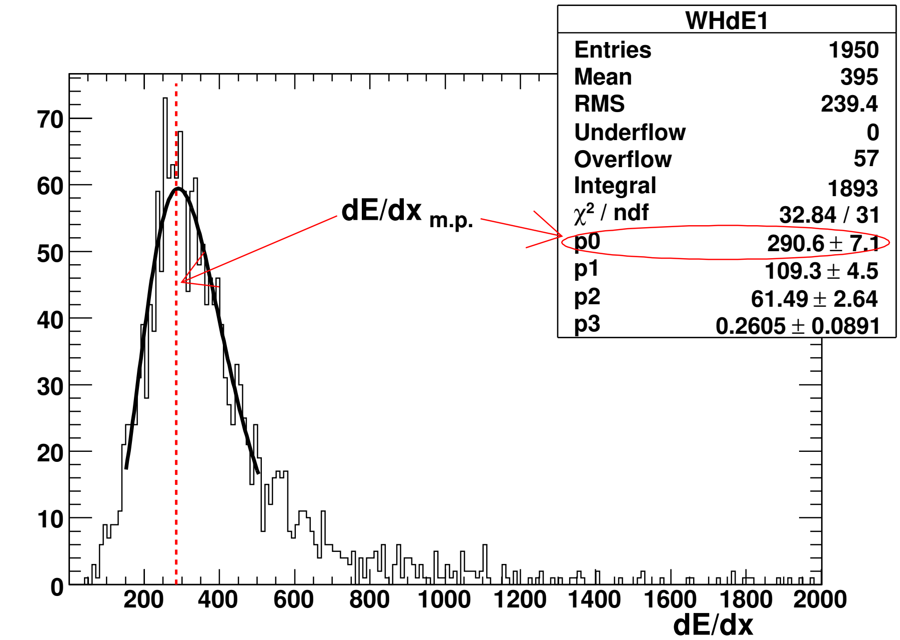


Figure 1: The typical distribution of energy loss (wire number 1). Unsymmetrical Gaussian fit is presented. $\frac{dE}{dx}$ is shown in normalized units.

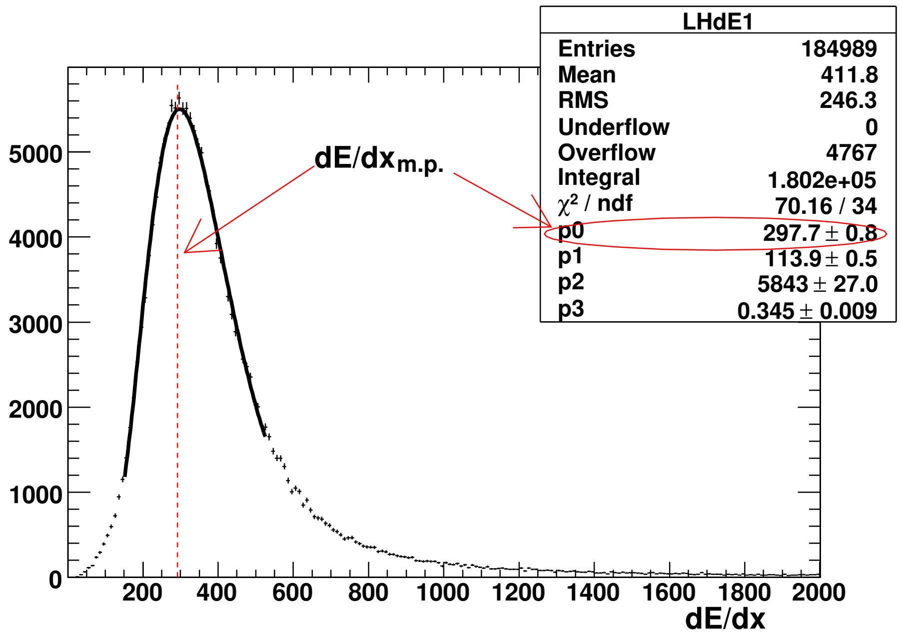


Figure 2: The typical distribution of energy loss (layer number 1). Unsymmetrical Gaussian fit is presented. $\frac{dE}{dx}$ is shown in normalized units.

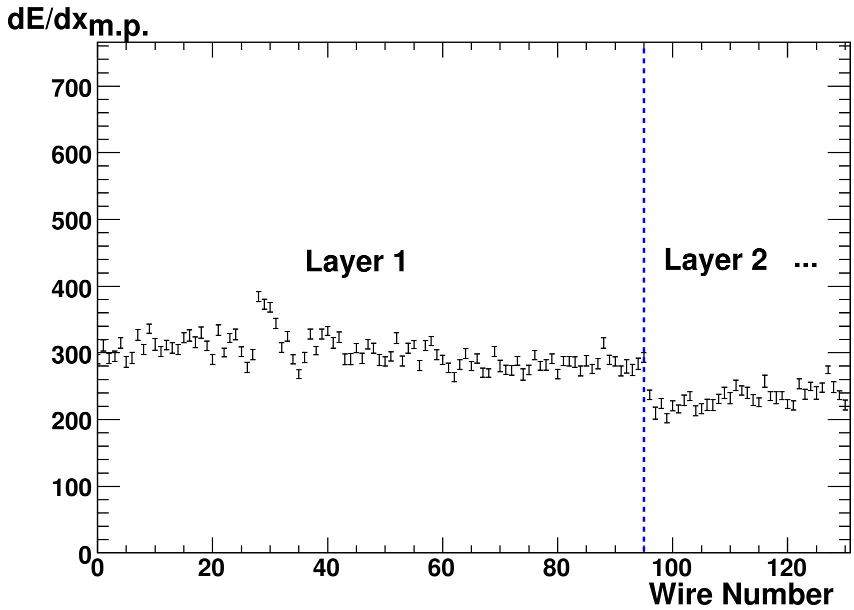


Figure 3: $\frac{dE}{dx}_{m.p.}$ vs number of wire ($\frac{dE}{dx}$ is shown in normalized units)

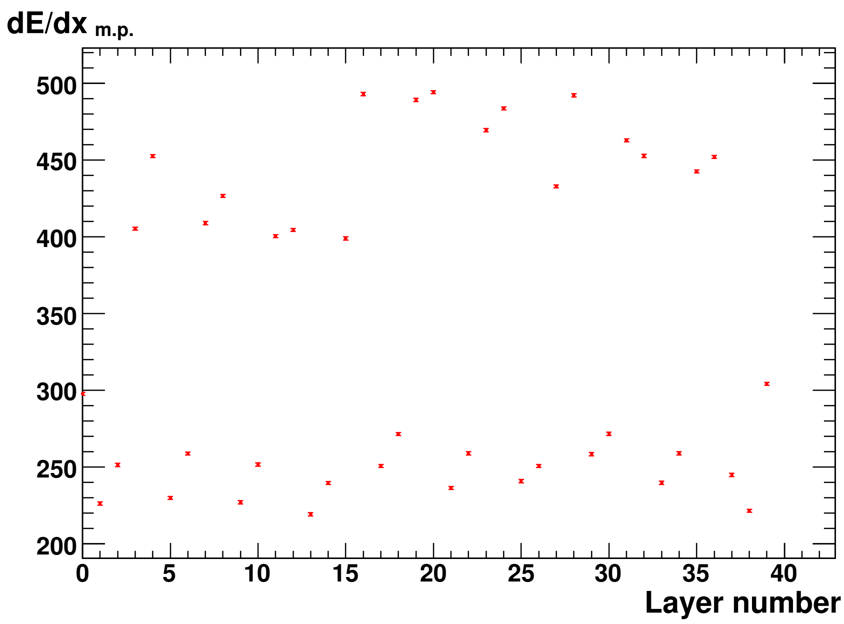


Figure 4: $\frac{dE}{dx}_{m.p.}$ vs number of layer ($\frac{dE}{dx}$ is shown in normalized units)

2.3 Saturation correction.

Saturation effect can be described by following simple equation:

$$Q_{meas} = Q_{ioniz} * e^{-\frac{k*Q_{ioniz}}{\sqrt{\sin\lambda^2+d^2}}} \quad , \tag{3}$$

where Q_{meas} is the measured charge, Q_{ioniz} is the charge which appeared from ionization. It is better to use another equation:

$$\rho_{meas} = \rho_{ioniz} * e^{-\frac{k*\rho_{ioniz}}{\sqrt{\sin\lambda^2+d^2}}} \quad , \tag{4}$$

where $\rho_{ioniz} = \frac{Q_{ioniz}}{dx*GlobalGasGain}$, dx is the path length of particle in cell. It is easy to find from this equation next Taylor expansion:

$$\rho_{ioniz} \simeq \rho_{meas} * \left(1 + \frac{k*\rho_{meas}}{\sqrt{\sin\lambda^2+d^2}} + \frac{1}{2} * \left(\frac{k*\rho_{meas}}{\sqrt{\sin\lambda^2+d^2}}\right)^2 - \frac{1}{3} * \left(\frac{k*\rho_{meas}}{\sqrt{\sin\lambda^2+d^2}}\right)^3 + ...\right) \tag{5}$$

Since third component is negligible we get

$$\rho_{ioniz} \simeq \rho_{meas}/A * \left(1 + \frac{k*\rho_{meas}/A}{\sqrt{\sin\lambda^2+d^2}} + \frac{1}{2} * \left(\frac{k*\rho_{meas}/A}{\sqrt{\sin\lambda^2+d^2}}\right)^2 - \frac{1}{3} * \left(\frac{k*\rho_{meas}/A}{\sqrt{\sin\lambda^2+d^2}}\right)^3 + ...\right) \tag{6}$$

To find these constants the surface of ρ_{meas} vs ρ_{ioniz} should be consider, but due to historical reason [3] the following expression is used:

$$Q_{raw} = Q_{meas} \cdot \left(1 + p_0 \frac{Q_{meas}}{\sqrt{\sin\lambda^2+d^2}} + p_1 \Big(\frac{Q_{meas}}{\sqrt{\sin\lambda^2+d^2}}\Big)^2\right) \tag{7}$$

To determine charge saturation parameters we have to minimize χ^2 defined as:

$$\chi^2 = \sum_{tracks} \left(\frac{(\frac{dE}{dx})_{track} - (\frac{dE}{dx})_{exp}}{\sigma_{track}}\right)^2 \tag{8}$$

where $\frac{dE}{dx}_{track}$ is measured value $\frac{dE}{dx}$ for given p_1, p_2, p_3,
$\frac{dE}{dx}_{exp}$ is expected value $\frac{dE}{dx}$ calculated according to Bethe-Bloch parametrization,
σ_{track} is expected $\frac{dE}{dx}$ resolution.

Actually we have to fit additional parameter for scaling(to keep constant the value $\frac{dE}{dx}$ for Bhabha events). Also we can to use a wide number of parameters if add dip. angle parameters and parameters of Bethe-Bloch function. Existing code has such possibilities.

2.4 Dip-angle correction.

The number of the electron clusters which are collected on the sense wire depends from dip-angle. It means that measured value of the $\frac{dE}{dx}$ depends from dip-angle due to changing $\frac{dE}{dx}$ distribution form. This behavior can be fitted by Chebyshev's polynomial of the 4th degree

$$F_{Ch}(\sin\lambda) = \sum_{n=0}^{4} c_n * \Big(\cos(n \arccos(2|\sin\lambda| - 1))\Big) \tag{9}$$

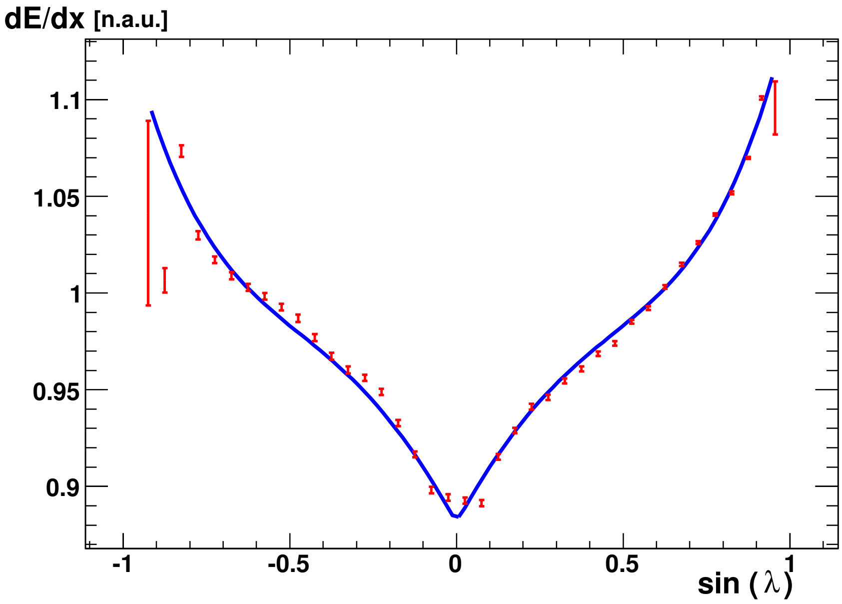


Figure 5: $\frac{dE}{dx}$ versus dip. angle.

2.5 Doca & entrance angle correction.

The $\frac{dE}{dx}$ distribution on the ($doca$,ϕ/π)($doca$ is normalized variable) is considered and the region $-1 \leq \phi/\pi \leq 1$ and $-1 \leq doca \leq 1$ is selected and divided into rectangular cells. Currently a table with 40 slices in $doca$ and 40 in ϕ/π is used with not regular step. The color $\frac{dE}{dx}$ plots for different layers types are shown on figures 6,7,8.

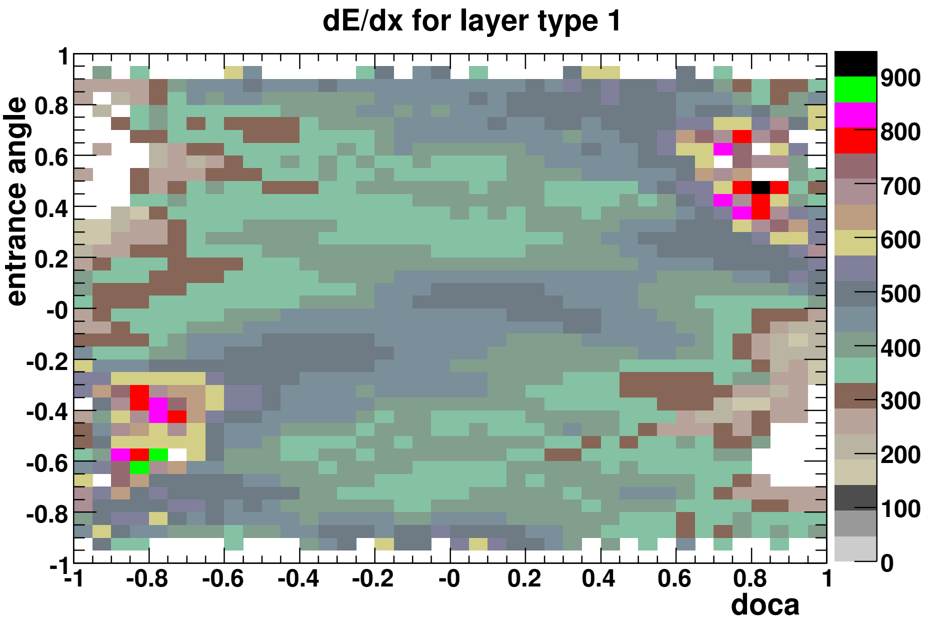


Figure 6: $\frac{dE}{dx}$ on doca & entrance angle surface(layer type 1).

2.5.1 Doca correction

The remaining $\frac{dE}{dx}$ dependence versus **doca** can be decreased using analogous procedure as for dip. angle correction (Fig.9).

The corresponding fitting function is

$$f = p_0 - p_1 * e^{-0.5*(\frac{doca}{p_2})^2} - p_3 * e^{p_4*|doca|} + p_5 * e^{-0.5*(\frac{doca-p_6}{p_7})^2} \tag{10}$$

also there is option using of the Chebyshev's polynomial fit

$$f = \sum_{n=0}^{10} p_n * c_n(doca) \tag{11}$$

Note that the $\frac{dE}{dx}$ dependence vs doca still exists if we consider proton sample (Fig.10).

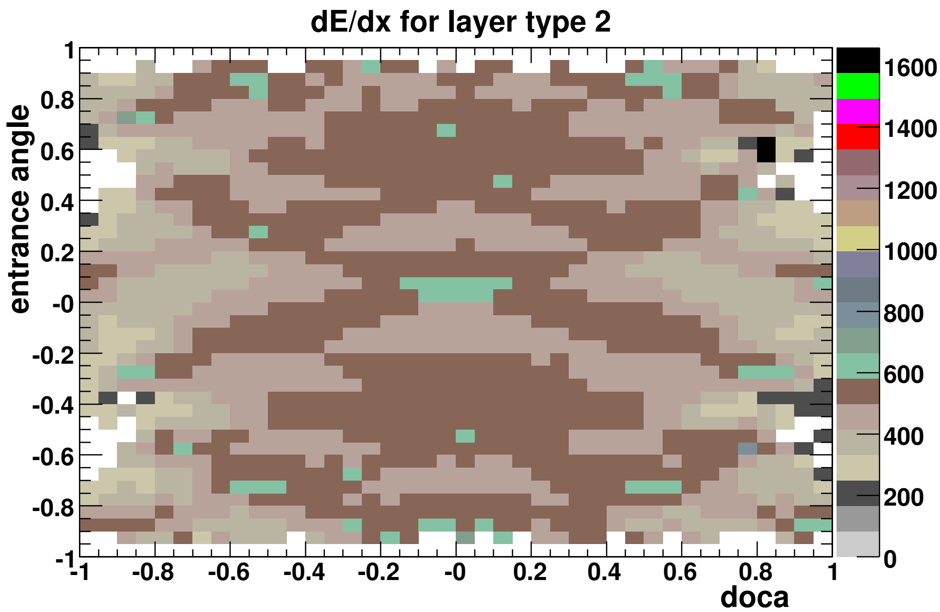


Figure 7: $\frac{dE}{dx}$ on doca & entrance angle surface(layer type 2).

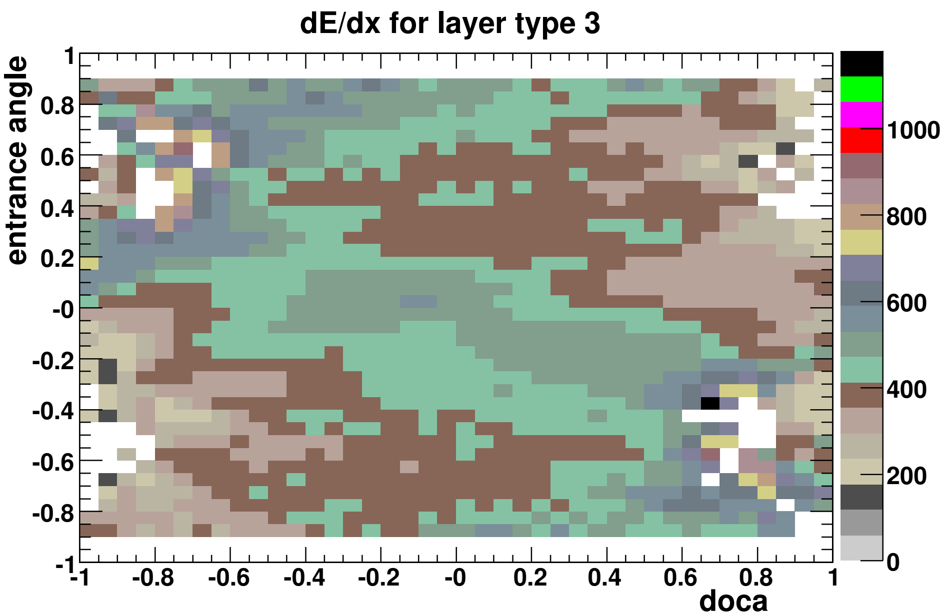


Figure 8: $\frac{dE}{dx}$ on doca & entrance angle surface(layer type 3).

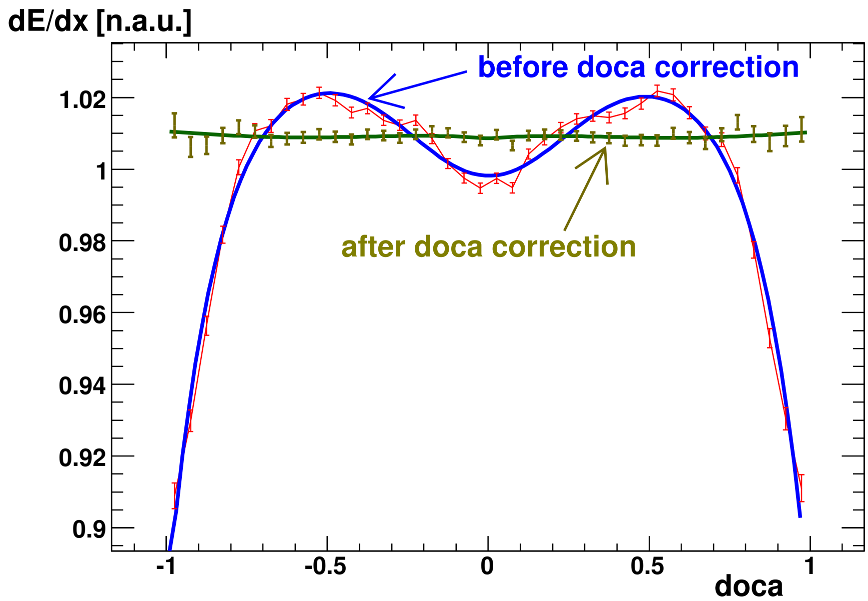


Figure 9: $\frac{dE}{dx}$ vs doca for pion sample.

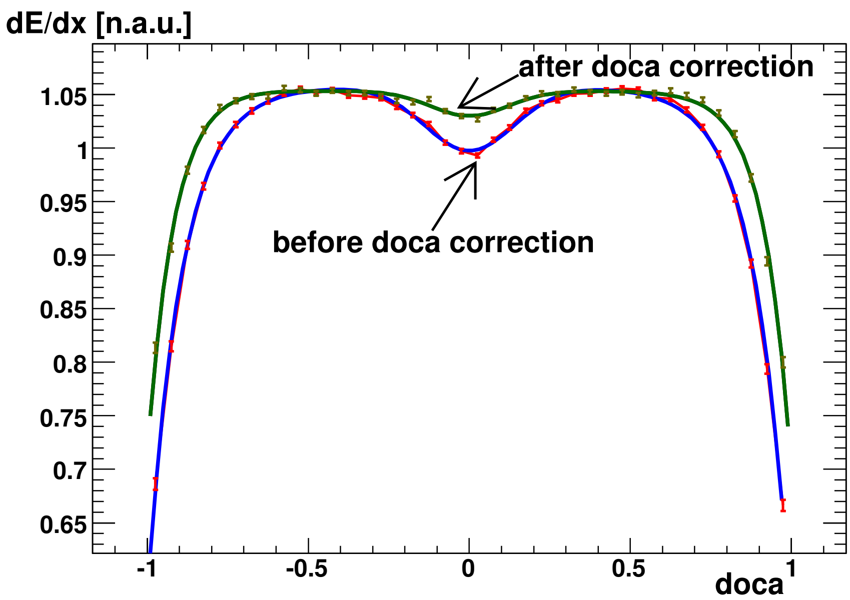


Figure 10: $\frac{dE}{dx}$ vs doca for protons sample.

3 $\frac{dE}{dx}$ analysis

3.1 Optimization of the truncated mean algorithm

The simple estimator of the energy loss for charged particle is mean value of all amplitude samples along the track. In standard truncated mean method a fixed fraction of high-side (and or or sometimes low-side) tail is rejected in calculating the $\frac{dE}{dx}$ of a track, prior to taking the truncated mean of the sample the values are sorted. The current upper truncation parameter for BaBar drift chamber is U equals 0.8 and the lower cut parameter is zero what corresponds small fraction of noise fluctuations.

It is obviously that we can get the optimal value U which depends on N_{hot} [number hits on track]. Suppose we have taken tracks with N_{hot} equals 30. Define the ionization density $\rho_{ionization}$ of the track according to equation:

$$\rho_{ionization} = \frac{\sum\limits_{N_{hot}} (\frac{dE}{dx})\text{hot}}{L_{track}} \tag{12}$$

Consider the family of $\frac{dE}{dx}$ resolution curves for different ionization density.

Build similar dependences for different values of N_{hot} the following expression for optimal U_{opt} can be found:

$$U_{opt}(\rho_{ionization}) = p_0 + (1 - p_0) * e^{-\frac{\rho_{ionization}}{p_1}} \tag{13}$$

This form $U_{opt}(\rho_{ionization})$ of outcome from physical behavior at infinity and zero point of $\rho_{ionization}$. The second parameter is scaled like as minimum Bethe-Bloch curve. The fit of the optimal points U_{opt} getting from the data is shown on Fig. 12.

Define number hits on the track which have to take to account as integer value N_{opt}

$$N_{opt} = int(U_{opt}(\rho_{ionization}) \cdot N_{hot}) \tag{14}$$

$\frac{dE}{dx}$ resolution depends from length of the track therefore it is necessary to introduce the weight for each of amplitude corresponds to path length. The common expression for energy loss on the track is:

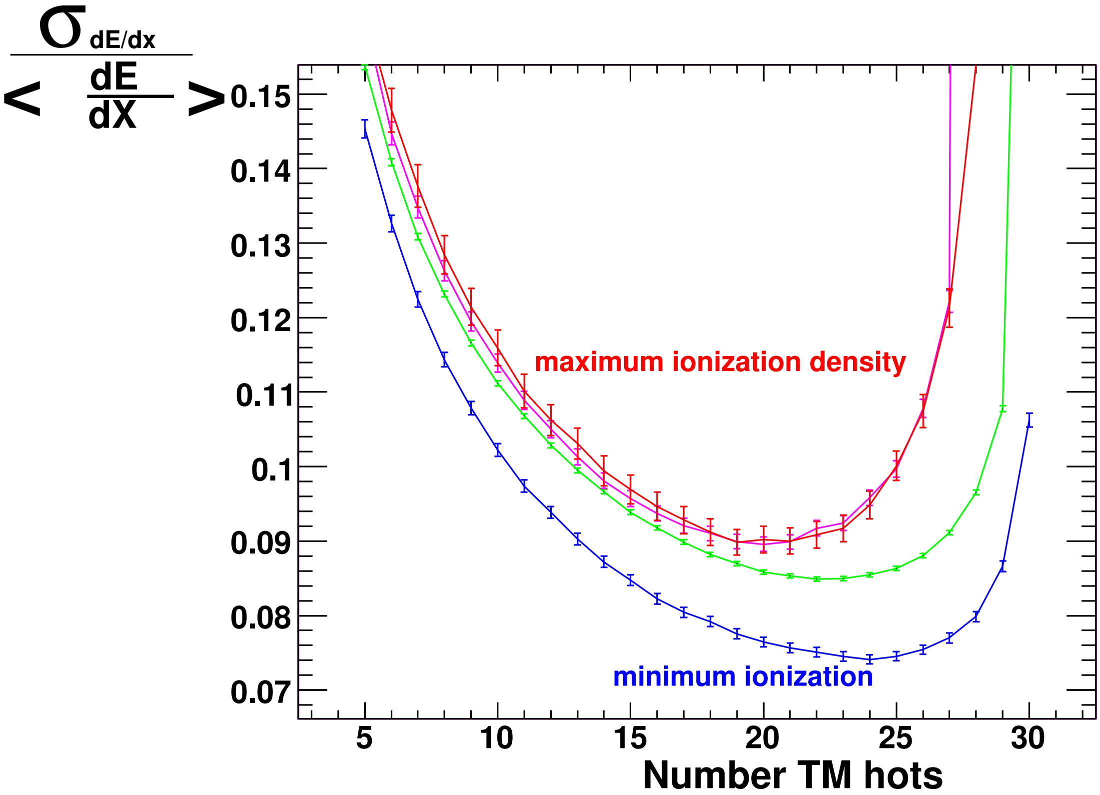


Figure 11: $\frac{dE}{dx}$ resolution vs number of the hots have been taken to account. For tracks which only contain $N_{hot} = 30$. Different colors correspond to different values of the $\rho_{ionization}$.

$$\langle\frac{dE}{dx}\rangle_{\text{track}} = \frac{\sum_{1}^{N_{opt}} (\frac{dE}{dx})^{sort}_{\text{hot}} * W_{\text{hot}}}{\sum_{1}^{N_{opt}} W_{\text{hot}}} \qquad (15)$$

$W_{\text{hot}} = dX^{\alpha}$, dX is *hot* length, α parameter is about ~ 0.62, $(\frac{dE}{dx})^{sort}$ is sorted in ascended order array of amplitudes. One should note, the main improvement of $\frac{dE}{dx}$ resolution appears from optimization number of hits taken into account and weights gives small effect.

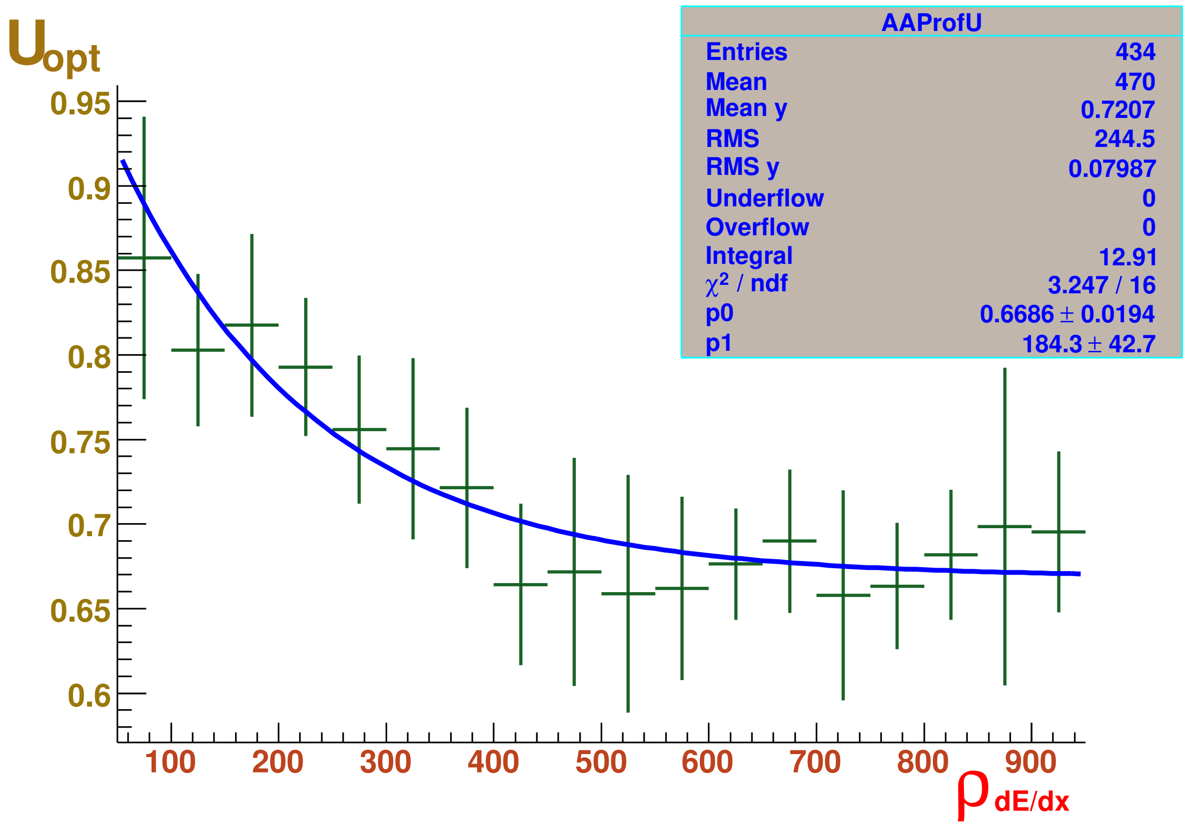


Figure 12: The rejected high-side fraction versus ionization density

3.2 The parameterization of the Bethe-Bloch

To get successful particle species separation, the energy loss on momentum should be approximated with good accuracy. Two types of function have been tested to fit experimental data: current Bethe-Bloch parametrization [1]

$$BB(\beta,\gamma) = \frac{p_0}{\beta^{p_4}}\Big(p_1 - \beta^{p_4} - \log\left(p_2 + (\beta\gamma)^{-p_3}\right)\Big) \tag{16}$$

and modified Bethe-Bloch formula:

$$BB(\beta,\gamma) = \frac{p_0}{\beta^{p_4}}\left(p_1 - \beta^{p_4} - p_6 \log\left(\frac{1 + p_2\cdot(\beta\gamma)^{p_3}}{1 + p_5\cdot(\beta\gamma)^{p_3}}\right)\right) \tag{17}$$

Alternative function describes the behavior of the energy loss much better than current and gives significant improvement in χ^2 for fitting same data (Fig.13) This form of the Bethe-Bloch curve is used to calibrate $\frac{dE}{dx}$.

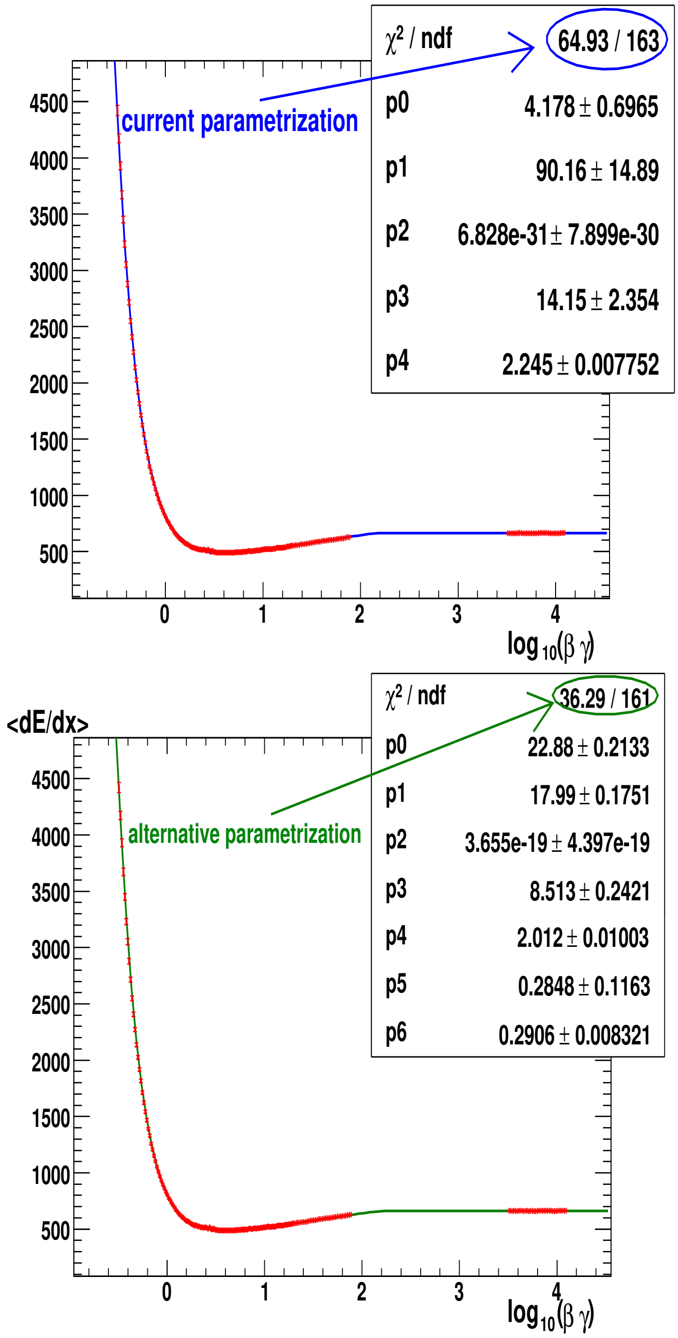


Figure 13: Fitted energy loss for different Bethe-Bloch parameterizations

3.3 Parameterization of the $\frac{dE}{dx}$ resolution

The following empirical function to parametrize $\frac{dE}{dx}$ resolution is used :

$$\frac{\sigma_{\frac{dE}{dx}}}{\langle\frac{dE}{dx}\rangle} = p_0\Big(\frac{N_{hot}}{N_0}\Big)^{p_1}\Big(\frac{L_{track}}{L_0}\Big)^{p_2}\Big(1 + p_3 * (x - p_4)^2\Big) \cdot (1 + p_5 * (p_t - p_6)^2) \qquad (18)$$

$N_0 \equiv 40.$, $L_0 \equiv 100.$,
N_{hot} — number active hit on track
L_{track} — track's length [cm]
p_t — transverse momentum [GeV/c]
$x = \frac{BB(\beta\gamma) - BB_{min}(\beta\gamma)}{BB_{min}(\beta\gamma)}$ (BB — Bethe-Bloch function)
Rewrite the equation (18) to separate the dependences from different parameters:

$$\frac{\sigma_{\frac{dE}{dx}}}{\langle\frac{dE}{dx}\rangle} \equiv p_0 \cdot f_N(N) \cdot f_L(L) \cdot f_{bb}(x) \cdot f_{p_t}(p_t) \qquad (19)$$

The functions $f_N(N)$, $f_L(L)$, $f_{bb}(x)$ and $f_{p_t}(p_t)$ must have the similar form as in (18)($p_0 \sim 0.05$ by using new calibration) The pictures 14 illustrate that this form of the function is correct.

4 Results

4.1 $\frac{dE}{dx}$ resolution comparison

Current and new $\frac{dE}{dx}$ resolution versus $log_{10}(\beta\gamma)$ on track is shown on Fig. 15. The improvement of absolutely $\frac{dE}{dx}$ resolution is about 0.7%

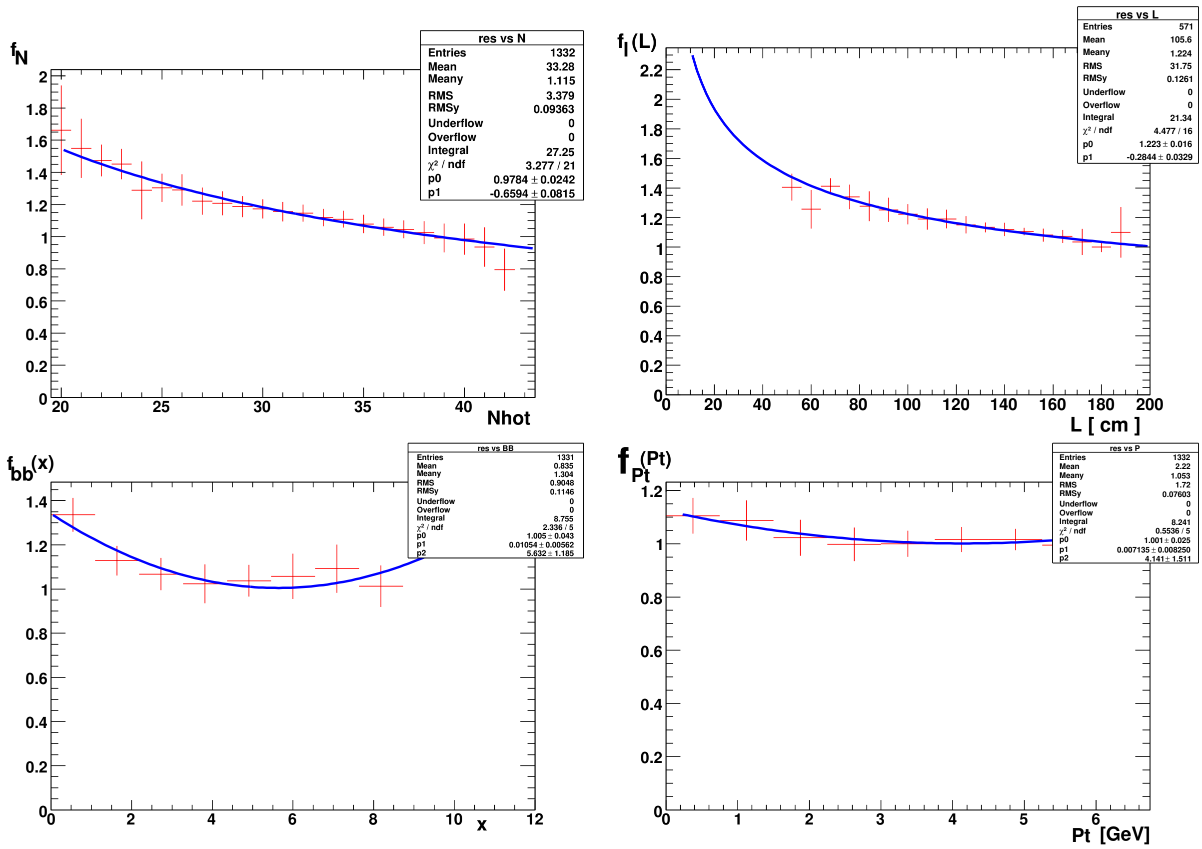


Figure 14: Multiplies of the $\frac{dE}{dx}$ resolution function.

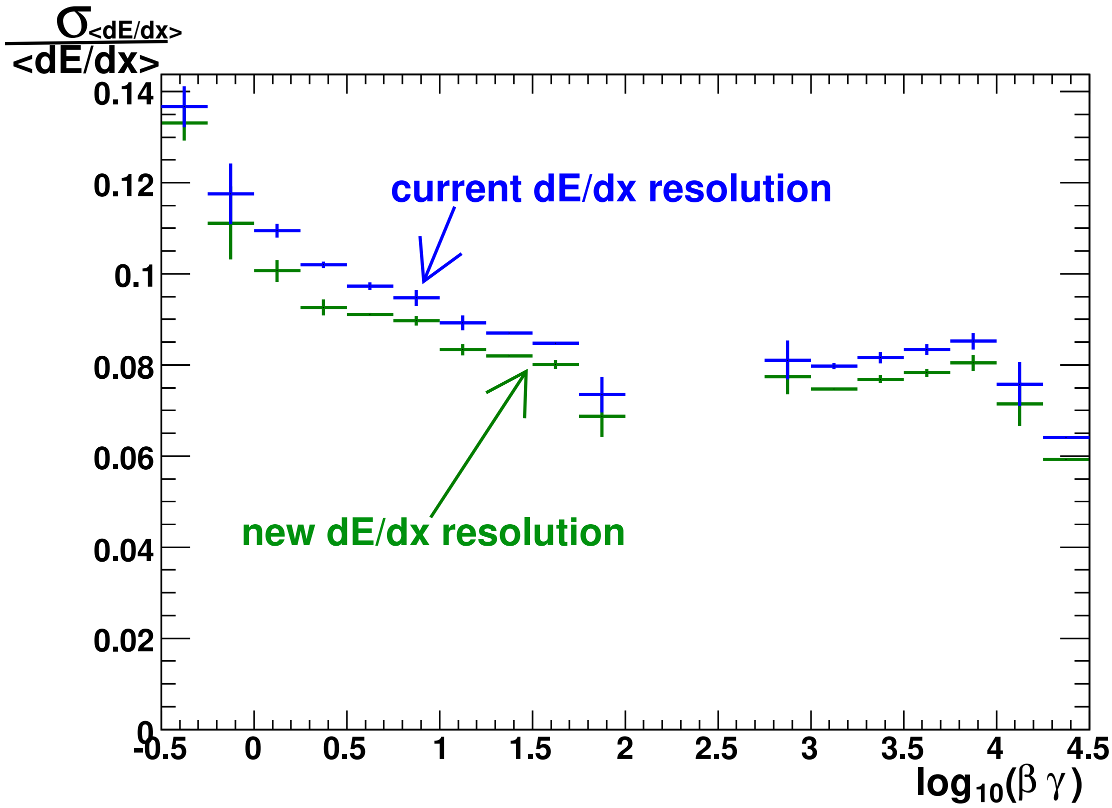


Figure 15: Current and new $\frac{dE}{dx}$ resolution vs $log_{10}(\beta\gamma)$

5 The new $\frac{dE}{dx}$ revealed particularities

Some peculiarities of behavior $\frac{dE}{dx}$ unknown before have been recognized. Unfortunately, there are many reasons why we can not to introduce these corrections in current BaBar DCH software.

5.1 Corrections on the hit level

5.1.1 The $\frac{dE}{dx}$ dependence vs TDC

It seems obviously that amplitude of the signal could changing due to the slewing effect. That means there is dependence $\frac{dE}{dx}$ versus TDC, this behavior is shown on Fig.16

$$f = \begin{cases} p_0 + p_1 \cdot (t - p_2)^4 + p_3 + p_8 & t < p_2 \\ p_0 + p_3 * e^{-0.5 \cdot \frac{(t-p_2)^2}{p_4^2}} + p_8 & p_2 \geq t < p_5 \\ p_0 + p_3 * e^{-0.5 \cdot \frac{(p_5-p_2)^2}{p_4^2}} + p_7 \cdot (t - p_5)^2 + p_8 & p_5 \geq t < p_6 \\ (p_0 + p_3 * e^{-0.5 \cdot \frac{(p_5-p_2)^2}{p_4^2}} + p_7 \cdot (p_6 - p_5)^2 + p_8) * e^{-0.5 * \frac{(t-p_6)^2}{p_9^2}} + p_8 & t \geq p_6 \end{cases} \quad (20)$$

5.1.2 The $\frac{dE}{dx}$ dependence vs path length

The dependence of the most probably value $\frac{dE}{dx}$ versus dX — hit length has been observed (Fig.17) .

$$f = \begin{cases} \frac{p_0}{dx^2} + p_1 \cdot \log dx - p_2 \cdot e^{-0.5 \cdot \frac{(dx-p_7)^2}{p_3}} & \\ +p_4 \cdot \sqrt{dx} + p_5 \cdot (dx - p_6) & dx < p_6 \\ \frac{p_0}{p_6^2} + p_1 \cdot \log p_6 - p_2 \cdot e^{-0.5 \cdot \frac{(p_6-p_7)^2}{p_3}} & \\ +p_4 \cdot \sqrt{p_6} + p_8 \cdot (dx - p_6) + p_9 \cdot \sqrt{dx - p_6} & dx \geq p_6 \end{cases} \quad (21)$$

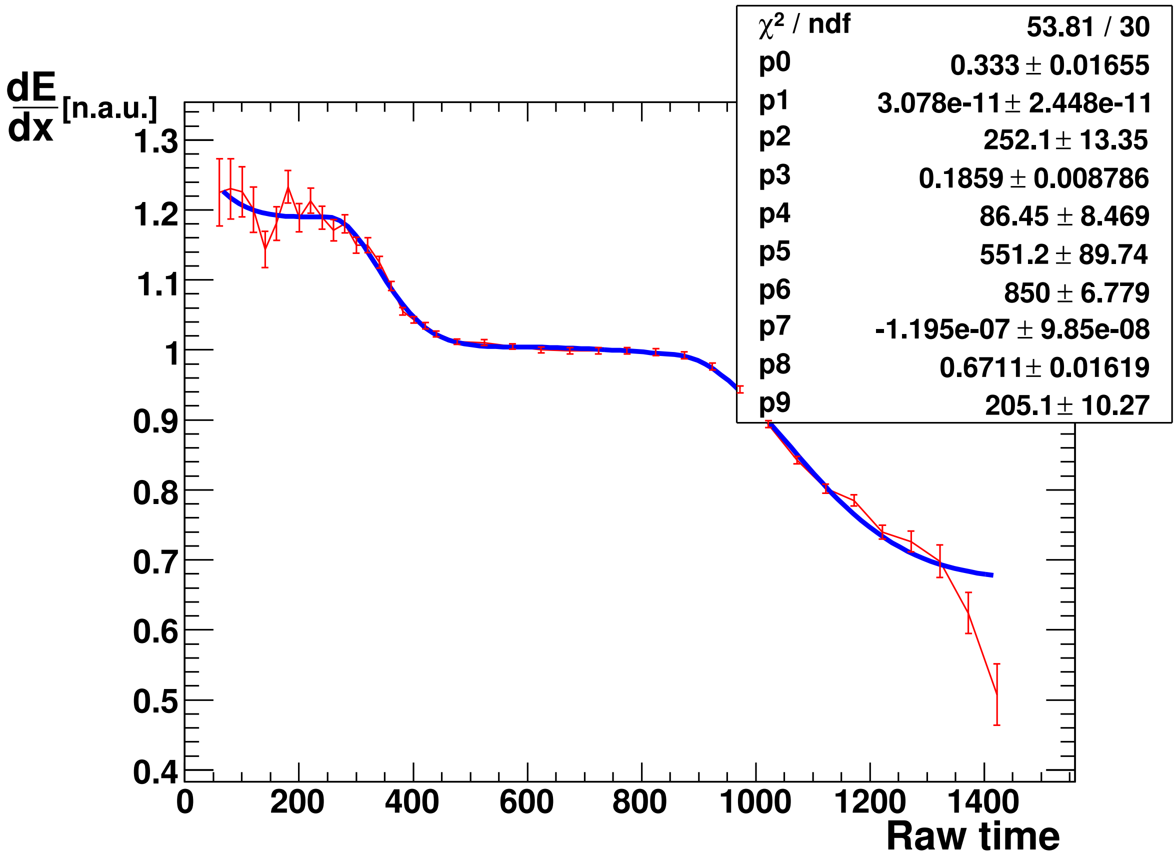


Figure 16: Most probably value $\frac{dE}{dx}$ vs raw time

5.2 Corrections on the track level

The dependence of $\frac{dE}{dx}$ versus N_{hot} have been observed on the track level (Fig. 18). This function had a similar form for both cases using standard fixed TM method a new optimal $\frac{dE}{dx}$ calculation strategy.

$$f = \begin{cases} p_0 - p_2 \cdot (p_3 - p_4)^2 + p_1 \cdot (N - p_3) & dx < p_3 \\ p_0 - p_2 \cdot (N - p_4)^2 & p_3 \leq dx < p_5 \\ p_0 - p_2 \cdot (p_5 - p_4)^2 + p_6 * (N - p_7)^2 - p_6 * (p_5 - p_7)^2 & p_5 \leq dx < p_9 \\ p_0 - p_2 \cdot (p_5 - p_4)^2 + p_6 \cdot (p_9 - p_7)^2 - p_6 \cdot (p_5 - p_7)^2 + p_8 * (N - p_9) & p_9 \leq dx \end{cases} \quad (22)$$

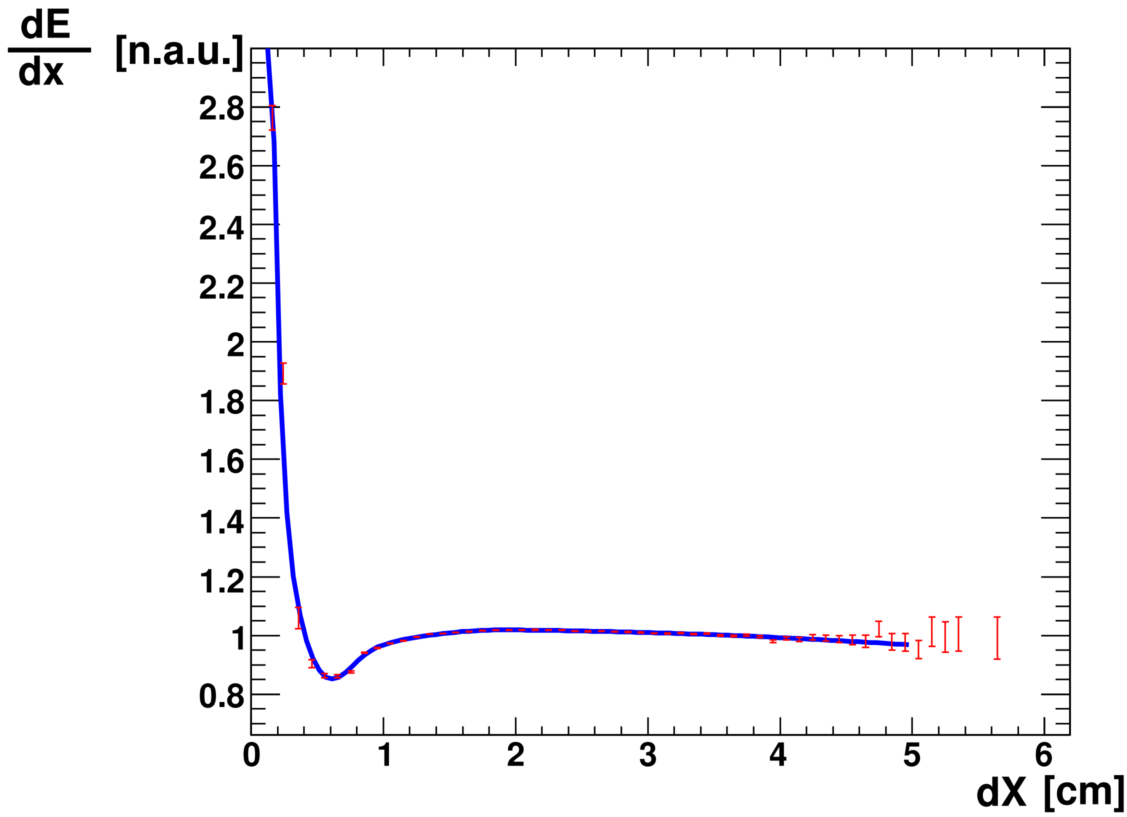


Figure 17: Most probably value $\frac{dE}{dx}$ vs dx

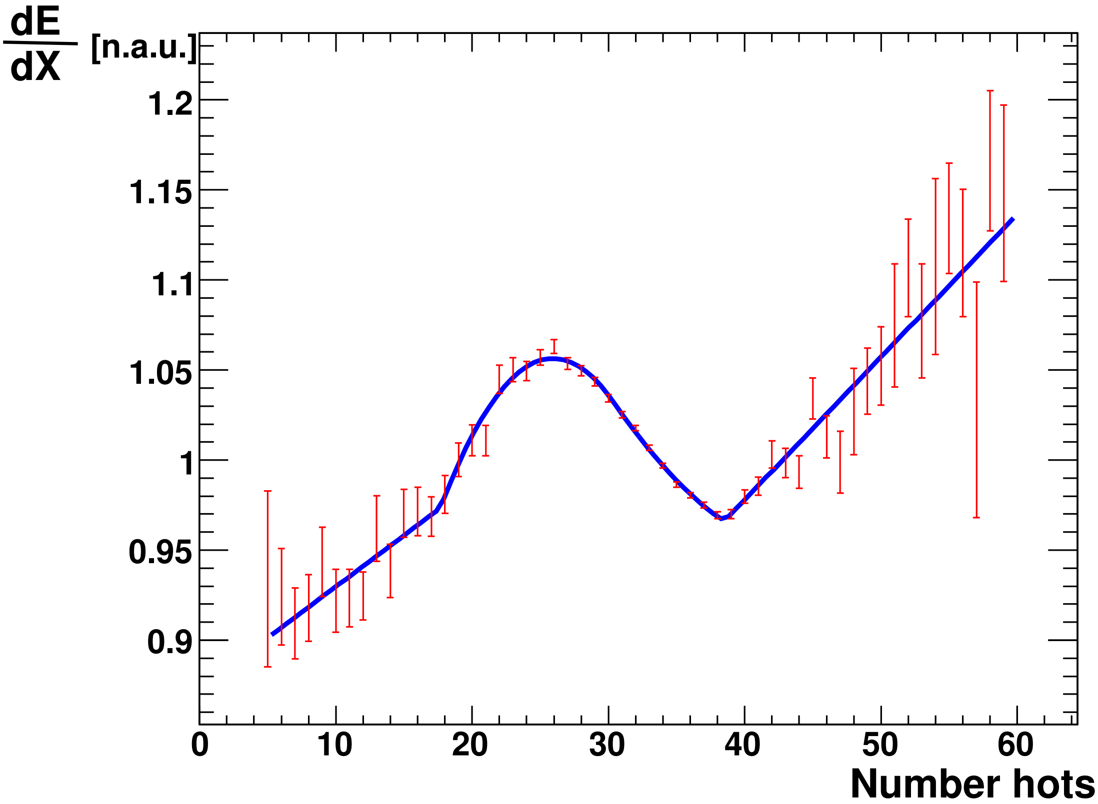


Figure 18: Most probably value $\frac{dE}{dx}$ vs number of the hit on track

6 Conclusions

The software to do full calibration by $\frac{dE}{dx}$ DCH has been written. Some "unknown" before particularities of $\frac{dE}{dx}$ behavior have been described and the new technique of the $\frac{dE}{dx}$ calculation has been considered. One can concludes the new $\frac{dE}{dx}$ strategy gives significant improvement in the $\frac{dE}{dx}$ resolution. Some existing systematics can be decreased by approach described in [4].

Appendix

A DCH $\frac{dE}{dx}$ calibration software

The following packages are used to calibrate DCH $\frac{dE}{dx}$ (Fig. A).

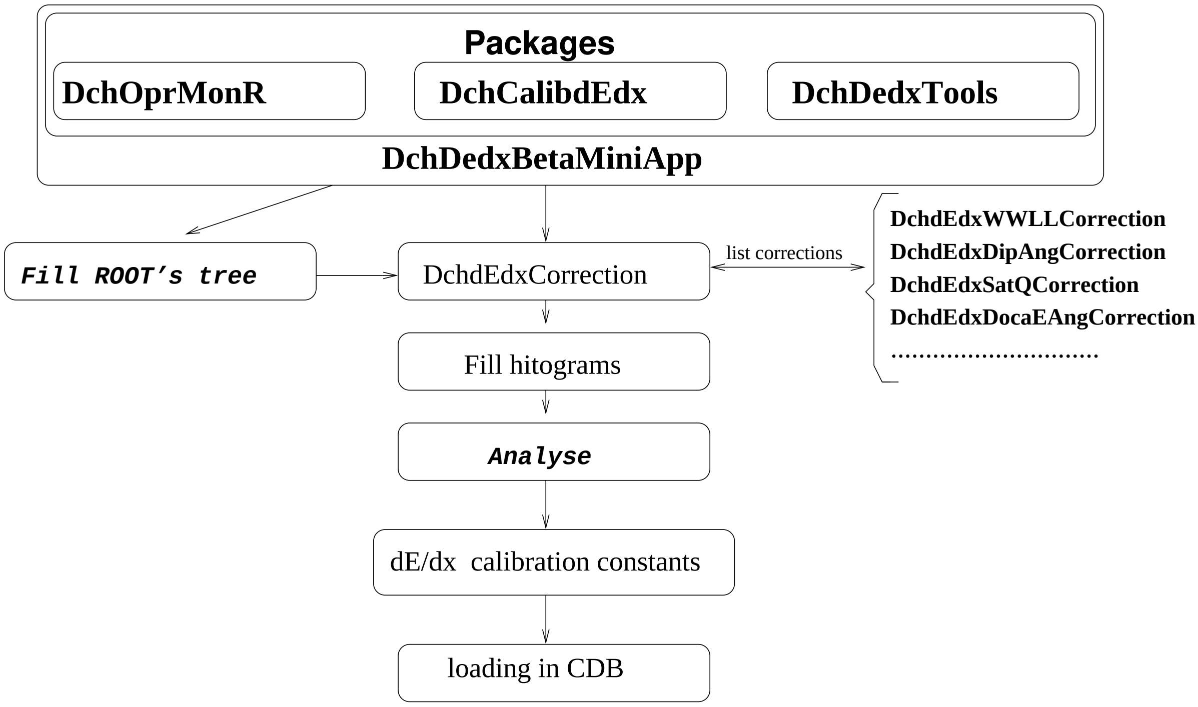
- *DchDedxTools – contains main sequences of the particle selection*
- *DchCalibdEdx – library for producing different correction parameters*
- *DchOprMonR – ROOT's tree library*

Add these packages in your *workdir*
$YourRelease >$ addpkg DchDedxTools
$YourRelease >$ addpkg DchCalibdEdx
then build the libraries and executable file
$YourRelease >$ *gmake DchCalibdEdx.lib*
$YourRelease >$ *gmake DchDedxTools.lib*
$YourRelease >$ *gmake DchDedxTools.DchDedxBetaMiniApp*

The main executable file DchDedxBetaMiniApp is built from *DchDedxTools/DchDedxBetaMini.cc* The Framework sequences are constructed in this file. The main parameters are defined in *DchDedxBetaMiniApp.tcl* called according to sequence
$runBetaMini.csh \rightarrow runBetaMini.tcl \rightarrow DchDedxBetaMiniApp.tcl$

A.1 Particle sample selection

The conditions of the particle selection can be found in *DchDedxTools* package:



- bhabha's events — DchDedx_electron.tcl
- μ sample — DchDedx_muon.tcl
- π sample — DchDedx_pion.tcl
- p sample — DchDedx_proton.tcl

A.2 How to get particle sample.

You will probably want to produce *Bhabha* sample. Create directory *collections* in your *workdir* and generate a list of files which you want to use as input runs.
YourRelease > cd workdir
workdir > mkdir collections
workdir > cd collections
collections > BbkExpertTcl –dbname bbkr18 -ds prelim-AllEvents-Run5-OnPeak-NoRqmValidation -t 250000 –splitruns –basename Test-Bhabha –run=65000-65050

As result you will get a list of files
collections >ls
Test-Bhabha-1.tcl
Test-Bhabha-2.tcl
......
Test-Bhabha-59.tcl

Go to back to *workdir* and put scripts *jobarray* and *runarray* from *DchDedxTools* in *workdir*. These scripts [2] allow to run an array of jobs and produce root files for given particles sample (the executable sequence is *runarray* $\rightarrow$ *jobarray* $\rightarrow$ *DchDedxTools/runBetaMini.csh*)
collections > cd ../
workdir > cp ../DchDedxTools/runarray runarray
workdir > cp ../DchDedxTools/jobarray jobarray
Create the output directory *Run5root* (if you want to change the name you should edit the script *runarray*).

Run batch array:
workdir > runarray Test Bhabha 1 59 250000

This will produce a files :
$Run5root/logs\_YourLogin/Test - Bhabha - [1-59].log$
$Run5root/root\_YourLogin/Test - Bhabha - [1-59].root$

The example how to compose all files together is here:
DchCalibdEdx/tclfiles/DchMakeTChain.tcl
####### ...
mod disable all
mod input DummyInput
mod enable DchdEdxTChainComposition
mod talk DchdEdxTChainComposition
outFileRoot set "Run5root/Test-Bhabha.root"

[2] author Aleksei Buzykaev

name output file
MakeTChain set true
SkipModification set false
modify(false) or not (true) particle type
UseListOfFiles set true
to use list of files
verbose set true
NameListOfFiles set "listTest.dat"
file name list of runs
inpFileRootFirst set "FirstRootFile.root"
first input file name is used in case # UseListOfFiles equals false
inpFileRootSecond set "SecondRootFile.root"
second input file name is used in case # UseListOfFiles equals false
FirstIdent set 0
particle type [0-4 corresponds to e, μ, π, K, p]
in case of -1 modification of the particle type is skipped
SecondIdent set 0
particle type for inpFileRootSecond
exit
quit
ev begin -nev 1
####### ...
Where *listTest.dat* contains list of files
$Run5root/root\_YourLogin/Test - Bhabha - 1.log$
$Run5root/root\_YourLogin/Test - Bhabha - 2.log$
...
$Run5root/root\_YourLogin/Test - Bhabha - 59.log$
Run *DchDedxBetaMiniApp* to produce bhabha's events file.
workdir DchDedxBetaMiniApp DchMakeTChain.tcl
The way of producing others samples of particles is similar. It is necessary to set *FirstIdent* according to particle type.

When you get all samples of particles you can compose all them in one file. Set in *DchMakeTChain.tcl*

outFileRoot set "Run5root/Test-All.root"
SkipModification set true
NameListOfFiles set "listAll.dat"

Where *listAll.dat* contains files like listed below:
Run5root/Test-Bhabha.root
Run5root/Test-Muon.root
Run5root/Test-Pion.root
Run5root/Test-Proton.root
The particle selection is achieved using *BetaPidCalib* package according to table below.

Particle selection		
Partycle type	particle sample	.tcl file in DchDedxTools
e	bhabha(EMC based)	DchDedx_electron.tcl
μ	$\mu\mu$,$\mu\mu\gamma$,$\mu\mu\gamma\gamma$	DchDedx_muon.tcl
π	Ks	DchDedx_pion.tcl
p	Λ	DchDedx_LambdaProton.tcl

A.3 Getting calibrations constants

When you get root trees for different particles types you can start calibration procedure. The examples of the *tcl* files for $\frac{dE}{dx}$ calibration are placed in package *DchCalibdEdx/tclfiles* . These files contains a short descriptions of the commands. The examples tcl files are listened in the table.

The list of tcl files	
	.tcl file
WWLL correction	*DchCalibdEdx/tclfiles/DchWWLL.tcl*
Rescaling WWLL constant Dip. angle correction	*DchCalibdEdx/tclfiles/DchRescaleWWLL.tcl* *DchCalibdEdx/tclfiles/DchDipAng.tcl*
Charge saturation correction	*DchCalibdEdx/tclfiles/DchSatQ.tcl*
Doca & entrance angle correction	*DchCalibdEdx/tclfiles/DchDocaEAng.tcl*
Doca correction	*DchCalibdEdx/tclfiles/DchDoca.tcl*
getting Bethe-Bloch parameters	*DchCalibdEdx/tclfiles/DchBetheBloch.tcl*
getting $\frac{dE}{dx}$ resolution function	*DchCalibdEdx/tclfiles/DchResolution.tcl*

To produce the file of output parameters run:
workdir > bsub -J Dch[Correction] -q kanga -G babarDch
-N -o OutputDir/Dch[Correction].log DchDedxBetaMiniApp Dch[Correction].tcl

A.4 Loading and fetching $\frac{dE}{dx}$ calibration constants in test federation

In first of all you should create test federation. The detailed description look here [5]. To load CDB conditions you have to use *DchProxy* package.
$YourRelease >$ addpkg DchProxy
$YourRelease >$ gmake DchProxy.lib
Build the following binary file:
$YourRelease >$ gmake DchProxy.DchCondLoad
Examples of tcl files are placed in *DchProxy*
The main tcl file is *DchProxy/DchCondLoad_NewStrategy_2006_R5_HV1930_Toff.tcl*
Files which are called in that file can be found in *DchProxy/dat*

To complete loading new $\frac{dE}{dx}$ constants in CDB run:
workdir > DchCondLoad YourDchCondLoad.tcl
To check loaded constants you have to use DchCondModules package.
$YourRelease >$ addpkg DchCondModules
$YourRelease >$ gmake DchCondModules.lib
$YourRelease >$ gmake DchCondModules.lib

$YourRelease >$ gmake DchCondModules.fetchDchBB
$YourRelease >$ gmake DchCondModules.fetchDchDedx

Follow prompt after starting *fetchDchBB* to get Bethe-Bloch and $\frac{dE}{dx}$ resolution parameters and *fetchDchDedx* to get $\frac{dE}{dx}$ corrections parameters.
$workdir >$ fetchDchBB
$workdir >$ fetchDchDedx

References

[1] *D. Coupal, M. Dubrovin* $\frac{dE}{dx}$ Systematics in the BaBar Drift Chamber. BaBar Note 492

[2] *V.Blinov The BaBar* Drift Chamber $\frac{dE}{dx}$ study. Note December 23,1998.

[3] *F. Coleccia ,P. Faccin* The BaBar Drift Chamber $\frac{dE}{dx}$ calibration procedure. BaBar Note 539

[4] *A. Telnov* Track-level $\frac{dE}{dx}$ Calibration for the BaBar Drift chamber and Silicon Vertex Tracker. BAD 1500

[5] *http://www.slac.stanford.edu/BFROOT/www/Public/Computing/Databases*