Deep inelastic scattering cross section extraction

Normalization check in NPS experiment

Content

1. MC ratio method and all related corrections.
2. F1F221 model(...)
3. Mc-single-arm and RC-external(...)
4. Results in previous F2 experiments (E12-10-002)

MC ratio method

$$Y_{data} = \frac{N - BG}{\epsilon}$$

The cross section per bin[1]:

$$\frac{d\sigma}{d\Omega dE'} = \frac{N - BG}{L} * \frac{1}{\epsilon} * \frac{1}{A(\theta, E')} * \frac{1}{\Delta E' \Delta\Omega}$$

Where:

N-BG = Detected electron counts with background subtraction.

L = Integrated Luminosity

ε = Total efficiency

A= Acceptance

1. Hall A/C Software Meeting 2017, M. Eric Christy

$$Y_{data} = L * d\sigma^{data} * (\Delta E' \Delta\Omega) * A(\theta, E')$$

If we use a cross section model to simulate the MC data, we will have

$$Y_{MC} = L * d\sigma^{MC} * (\Delta E' \Delta\Omega) * A_{MC}(\theta, E')$$

Assuming in two yields, A = Amc, by taking the ratio, we will have

$$d\sigma_{data} = d\sigma_{MC} * \frac{Y_{data}}{Y_{MC}}$$

At certain E’ and θ.

Ydata in actual experiment and corrections[2]

$$Y_{data} = \frac{N^{e-} - BG}{\epsilon_{total} E_{LT} C_{LT}} * PS$$

BG = pion contamination + charge symmetric background(target cell wall contribution)
ELT and CLT = electronic and computer livetime

N = # of e- passed all the cuts(PID, Acceptance)

2. F2 Update for Winter Hall C Collaboration Meeting

Code flowchart

mcWt.cpp

Ymc

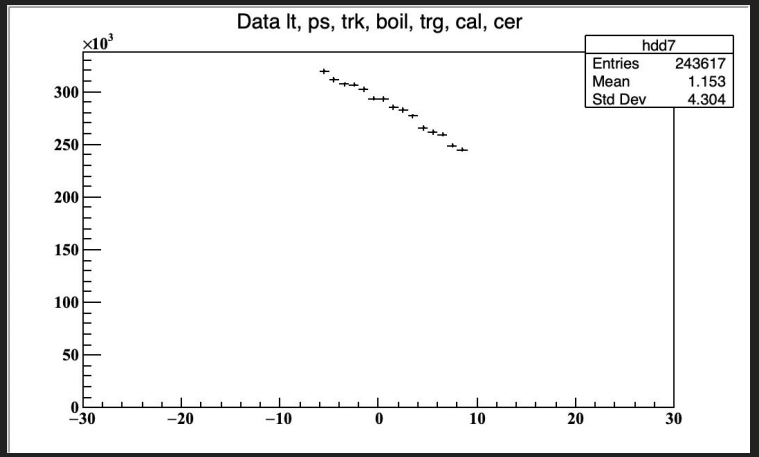
Root file

Production file

dataYield.cpp

Ydata

Ratio.cpp



ExtractCS.cpp

Corrections and retrieve methods

Read from production report generated by coda:
Livetime, trigger efficiency, efficiency, psFactor.

Hard coded: target density correction, Acceptance cuts, Acceptance corrections, pion contamination(based on target type and angle), charge symmetric background.

Problems ongoing

mcWt.cpp: how Ymc is generated and f1f221 model?
Ratio.cpp and dataYield.cpp: seems contain repetitive corrections…need to check how those multiple histograms got filled.

Check under different kinematics, which correction value we need to change.

Select one of the 1D histogram to do the cross section check (dp/p? W2? xb?)

Histogram tracking – dataYield

dataYieldOut/pass70/hms_h21deg3p3_dataYield1639.root

Histogram name: hdd

Fill dp/p with weight

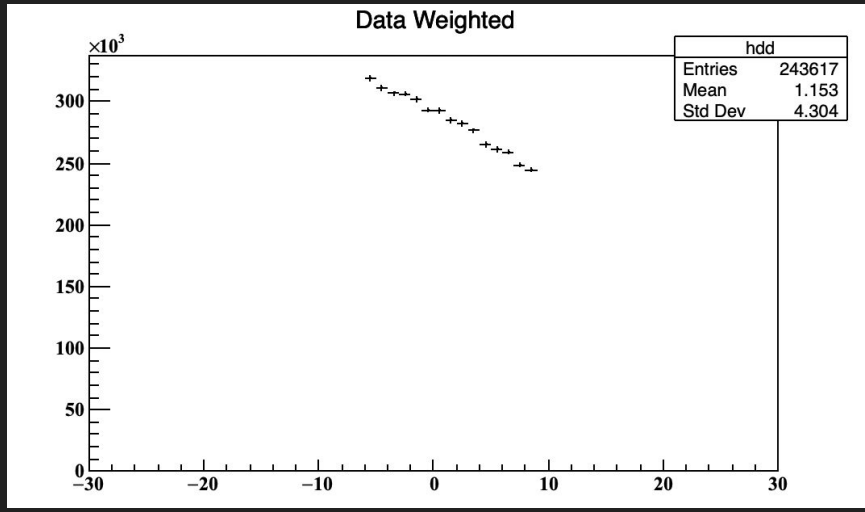
dp/p branch: H.gtr.dp

Remove
shms_acc_corr

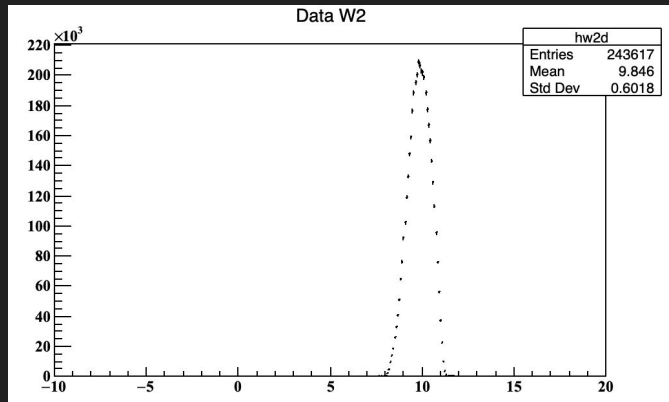
Weight = (1.0 -piC)/calEff/cerEff/shms_acc_corr*scale*dumscale

Here shms_acc_corr = 1 and delta is acceptance corrected for hms (remove).
Scale = livetime + tracking eff + trigger eff + PS
Dumscale = base on target ?

This is without the boiling correction. (implemented in ratio.cpp)



There's also W2 histogram with calculated W2(not from hcana)



Histogram tracking – mcWt – getRadCorrW2

/w/hallc-scshelf2102/etc/inclusive/abels/MC/tables/v0.996t2/hms_h10.6_all.dat_inel_f1f221

Format: eBeam, w2, theta, born, rci. Return: 2D graph with (w2, theta, y) y = Born or born/rad or rad

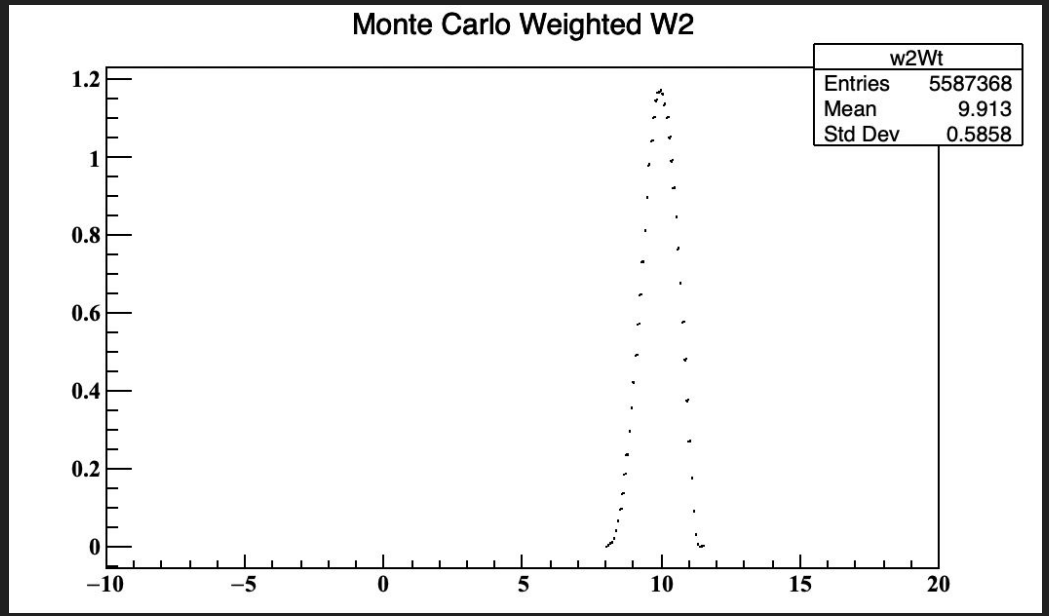
Mc-single-arm output:
/volatile/hallc/spring17/wmhenry/mc-single-arm/worksim/casey/Hms_mc_h21deg3p3.root

```
10.602 1.80775242e+01 16.000 1.58029160e+01  3.15137708e-01
10.602 1.80505447e+01 16.000 1.57318700e+01  3.18300407e-01
10.602 1.80235652e+01 16.000 1.56618470e+01  3.21458116e-01
10.602 1.79965857e+01 16.000 1.55928280e+01  3.24612392e-01
10.602 1.79696062e+01 16.000 1.55247900e+01  3.27749981e-01
10.602 1.79426267e+01 16.000 1.54577180e+01  3.30879590e-01
10.602 1.79156472e+01 16.000 1.53915910e+01  3.34026796e-01
10.602 1.78886677e+01 16.000 1.53263900e+01  3.37137609e-01
10.602 1.78616882e+01 16.000 1.52621000e+01  3.40244117e-01
10.602 1.78347087e+01 16.000 1.51987020e+01  3.43336211e-01
10.602 1.78077293e+01 16.000 1.51361800e+01  3.46420008e-01
10.602 1.77807498e+01 16.000 1.50745160e+01  3.49492662e-01
10.602 1.77537703e+01 16.000 1.50136960e+01  3.52556870e-01
10.602 1.77267908e+01 16.000 1.49537010e+01  3.55612994e-01
10.602 1.76998113e+01 16.000 1.48945180e+01  3.58641903e-01
10.602 1.76728318e+01 16.000 1.48361310e+01  3.61646887e-01
10.602 1.76458523e+01 16.000 1.47785270e+01  3.64675676e-01
10.602 1.76188728e+01 16.000 1.47216900e+01  3.68097787e-01
10.602 1.75918933e+01 16.000 1.46656060e+01  3.70707422e-01
10.602 1.75649138e+01 16.000 1.46102600e+01  3.73710256e-01
10.602 1.75379343e+01 16.000 1.45556410e+01  3.76704253e-01
10.602 1.75109548e+01 16.000 1.45017360e+01  3.79690896e-01
10.602 1.74839753e+01 16.000 1.44485300e+01  3.82665254e-01
10.602 1.74569958e+01 16.000 1.43960120e+01  3.85634035e-01
10.602 1.74300163e+01 16.000 1.43441690e+01  3.88596712e-01
10.602 1.74030368e+01 16.000 1.42929890e+01  3.91542420e-01
10.602 1.73760573e+01 16.000 1.42424600e+01  3.94858543e-01
10.602 1.73490778e+01 16.000 1.41925720e+01  3.97794549e-01
10.602 1.73220983e+01 16.000 1.41433120e+01  4.00721072e-01
10.602 1.72951188e+01 16.000 1.40946700e+01  4.03644206e-01
10.602 1.72681393e+01 16.000 1.40466350e+01  4.06130818e-01
10.602 1.72411598e+01 16.000 1.39991940e+01  4.09034539e-01
10.602 1.72141803e+01 16.000 1.39523410e+01  4.11964726e-01
10.602 1.71872008e+01 16.000 1.39060640e+01  4.14854268e-01
10.602 1.71602213e+01 16.000 1.38603520e+01  4.17739178e-01
10.602 1.71332418e+01 16.000 1.38151950e+01  4.20477824e-01
10.602 1.71062623e+01 16.000 1.37705870e+01  4.23351992e-01
10.602 1.70792829e+01 16.000 1.37265150e+01  4.26222520e-01
10.602 1.70523034e+01 16.000 1.36829710e+01  4.29167109e-01
10.602 1.70253239e+01 16.000 1.36399470e+01  4.32017451e-01
10.602 1.69983444e+01 16.000 1.35974330e+01  4.34884535e-01
10.602 1.69713649e+01 16.000 1.35554220e+01  4.37714645e-01
10.602 1.69443854e+01 16.000 1.35139040e+01  4.40541187e-01
10.602 1.69174059e+01 16.000 1.34728700e+01  4.43352114e-01
10.602 1.68904264e+01 16.000 1.34323140e+01  4.46163237e-01
10.602 1.68634469e+01 16.000 1.33922280e+01  4.48960238e-01
10.602 1.68364674e+01 16.000 1.33526030e+01  4.51755628e-01
10.602 1.68094879e+01 16.000 1.33134330e+01  4.54541460e-01
10.602 1.67825084e+01 16.000 1.32747090e+01  4.57299235e-01
```

```
[root [2] h1->Show(0)
======> EVENT:0
 hsxfp           = 0
 hsyfp           = 0
 hsxpfp          = 0
 hsypfp          = 0
 hsxtari         = -0.0597541
 hsytari         = -1.19418
 hsxptari        = -0.108441
 hsyptari        = -0.0562351
 hsztari         = -2.56507
 hsdeltai        = -4.73498
 hsytar          = 0
 hsxptar         = 0
 hsyptar         = 0
 hsztar          = 0
 hsdelta         = 0
 fry             = 0.0767564
 xsnum           = 0
 ysnum           = 0
 xsieve          = 0
 ysieve          = 0
 stop_id         = 5
 hsvxi           = -0.0125679
```

mcWt

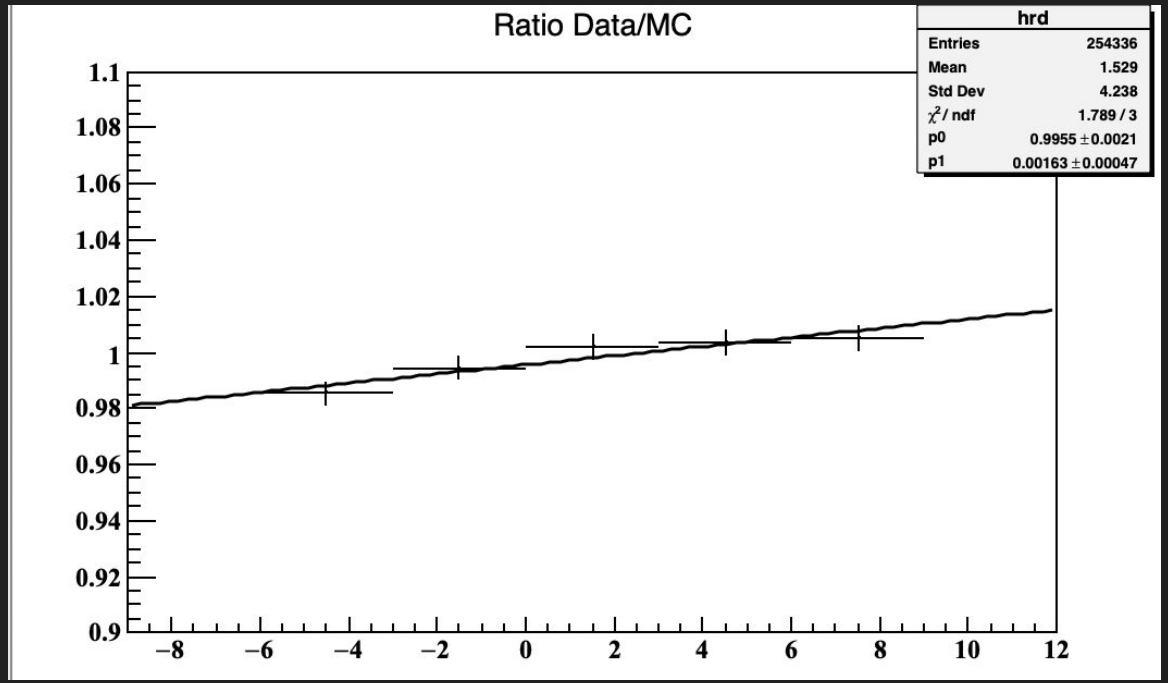
Interpolate the (born/rci ratio(born + higher order) + CSB(?)) * phasespcor to get the weight for mc_single_arm output, scale it with Ldata / Lmc / (charge = 1) , plot weighted w2.
Maybe it’s because Ydata does not subtract CSB?



mcWt

Ratio.cpp

Dummy(or charge?) normalized data yield? Boiling correction, dummy subtraction,



What's dumscale and what's dummy normalized data yield

Ratio.cpp

Charge normalization on hdd, why this normalization applied on all the factors including efficiency, etc.
Clone data histogram
Get dummy data from dummy runs, scale it with charged (why here it used different method other than hdd), I think dummy run contains also all the corrections-> direct subtraction
Subtract dummy runs from hdd -> hsd = hsd(inital value = hdd) - hed
Boiling correction applied to hsd by scaling (why also by scaling)
Hrd = ratio hdd/hmd (why hmd does not need charge normalization)
Scale hsd by hsd->Int() / hmd->Int() hsd = dummy subtracted hmd = MC hdd. (?)
What's the meaning of this step?

What's the integral? For example hmw2->Integral = w2 mc // momentum setting meaning (3p3)

Extract CS