Obtaining kinematic offsets for the kaon_lt experiment (the presentation)

Ioana & Gabriel Niculescu
James Madison University

Introduction

As advertised in previous updates...

- (for awhile now) the JMU group worked on developing a set of kinematic **offsets** for this experiment.
- we are happy(-ish) to report that this work is done (at least for the 10.6 GeV settings)
- furthermore, in the process, we developed (or at least we think we did) a new offset–determination **framework/workflow** which can be used in other Hall C (and A) experiments.
- ...and this is what I would like to talk about today.

But why?

- ...sets of kinematic offsets already exist. why not use those?
- and, of course, we have **heepcheck**
- fair enough. Here are my $0.02 – worth of explanation.

Motivation

(also) as explained in previous updates...

- **heepcheck** (and therefore its usage and by–products, like sets of offsets!) has substantial downsides
- AFAIKT these fall into two (distinct but not necessarily disjoint) categories:

I. Theoretical

- **heepcheck** (at least in its basic form) simply provides values for gradients (and expected shifts due to nominal momentum/angle offsets)
- ... and **ONLY** for central kinematics in both the proton and electron angle
- kaon_lt took (quite a bit!) of off–center data
  (trying to access the LT & TT bits of the cross–section)
- no provisions for the (non-negligible!) out–of–plane angle

Motivation (Cont.)

II. Practical

- elastic cross–section drops (dramatically!) w/ Q^2, therefore most experiments (including this one) can only take a token amount of data.
- ...possibly skipping settings if time–pressed
- carrying out systematics checks (which offset–finding ultimately is) w/ statistics–limited is not a recipe for success
- **heepcheck** has very little to say (as in “nothing”) about the agreement (or lack thereof) between data and MC therefore it is
- left to the user/ *ad hoc*/ by eye (you get the idea)

JMU group...

- try to address many/all of the shortcomings listed above
- it won’t fit here (next slide, please!)

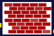
Motivation (Cont. II)

JMU group worked (very hard, of course!) ...

- to provide a different and (we dare to think! **better**) approach to offset finding that...
- it is **data–driven** (uses experiment–specific features)
- leverages small aperture spectrometer **understanding**
- actually **provides offsets**, not just a bunch of #s for DIY offset–assembly
- it is **objective** (within a well–defined set of assumptions)
- built–in mechanism for evaluating associated **uncertainties**
- **transferable**/adaptable to other exp./halls

a few general details

- implemented in python. Tightly integrated in our klt workflow.
- once understood, the method itself can easily be implemented in a language of your choice. Which would be the preferred way to **own** it!

Prerequisites (apologies for the text-. Pictures will follow soon!):

Assume that you have...

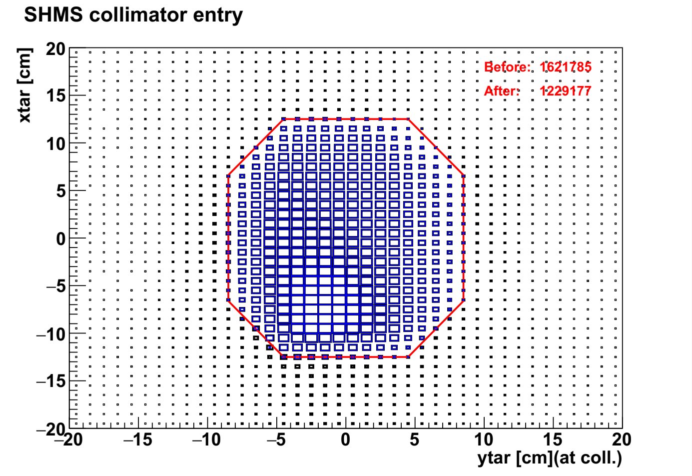
- ROOT (other format) DST files of the experimental data, processed w/ the latest/best set of matrix elements for the exp. *ditto* for SIMC
- (optional) having data and MC use a similar naming convention for variables (as it is done in the JMU workflow) - a definite **+** (simplified code quite a bit)
- **this is IMPORTANT!!** capability of reproducing all kinematic quantities of interest (Q^2, missing mass, etc.) from **basic** spectrometer quantities and central values (i.e. HMS/SHMS δ, in–plane, out–of–plane angles, etc.).
- make sure said code incorporates/uses the (eventual) kin. offsets.
- if you do (Richard's script, its jmu_klt version, whatever) - great! (test it with zero offsets, see if it reproduces the Q^2, mm, etc from the ROOT file!)
- if not, you can still do it, just have to run **hcana** every time (and hide your ignorance behind whining about not enough computing power).

With pre–requisites met...

let's (re)state the problem:

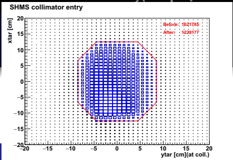
- we would like to find a/some set(s) of spectrometer offsets that make the data and MC relevant distributions **align** (not **agree!!!**)
- these offsets include values for δ as well as the two angles (in–plane and out–of–plane). a total of 6 vars. add 3 more for beam (x, y, z). 9 vars.
- incidentally, we use the **HALLC_p** - variable, which is the beam (z) component, corrected for arc losses. At least according to MJ.
- less than 9 constraints (original **heepcheck** was also an ill–posed problem)
- we would like to do this for each kinematic setting
- ...in an fast(ish) and lightweight way (i.e. less/no dependence on JLab cpu/disk/tape, etc.) - thus the need for it to work on DSTs
- with less constraints than variables, it will not be a fit, it will be *heuristic* approach, aiming to find **plausible** rather than **the best/unique** set of offsets.
- ...and we might as well end up with several viable sets for each kinematic

Divide et impera! Let's talk about angles...



From an E12–10–002 (IN, GN) report (12/03/2020!)

...angles (II)

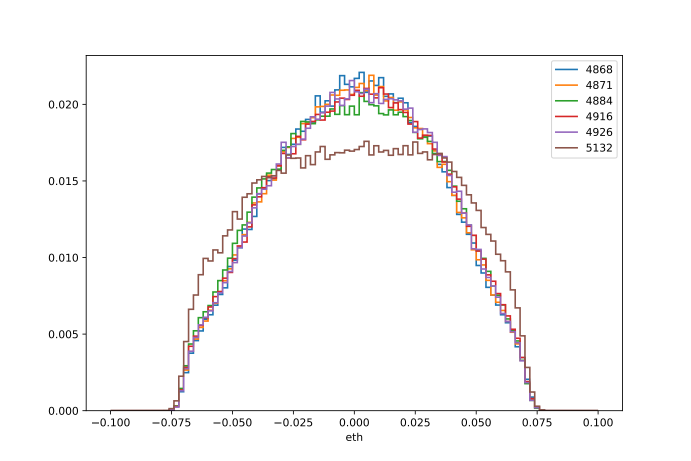


So, what's your angle? Just this:

- For 3–body final states (or if using singles, assuming enough were acquired and processed!) there isn't a strong correlation between the spectrometers
- *xptar* and *yptar* (or *th* and *ph* in hcana speak!) are just the angles wrt the axis going through the middle of the Figure (from last slide)
- the **edges** of these distributions have **way(!)** more to do with the shape of collimator (those clipped edges!) than with the physics at the interaction point
- ...OK, maybe, possibly "aided" by some well–known, understood processes (which would be modeled in **simc**) - Mott
- **IDEA:** Perhaps, instead of blindly charging into the big 9D windmill space, we can do some work on individual angles to start with!!
- Let's take a look!

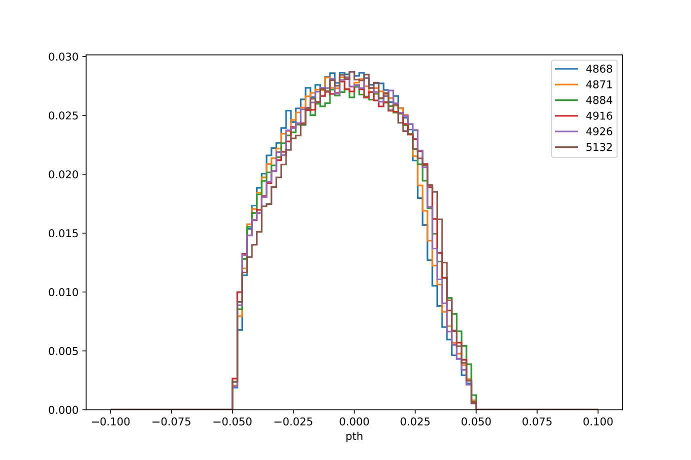
angles (III)

HMS *xptar (th)* angle. For several kinematics. All simc.



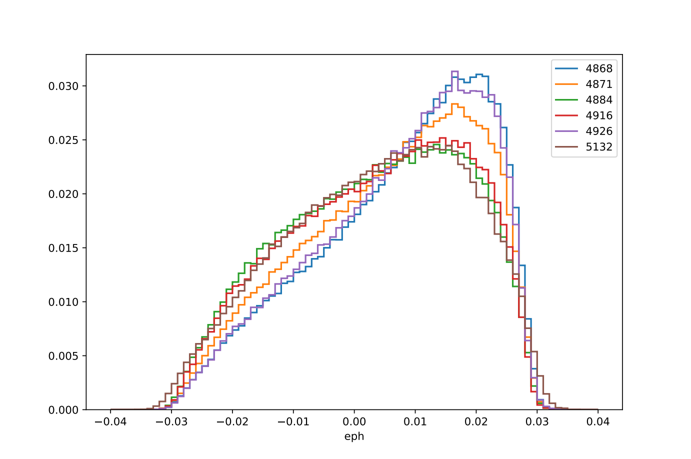
angles (IV)

SHMS *xptar (th)* angle. For several kinematics. All simc.



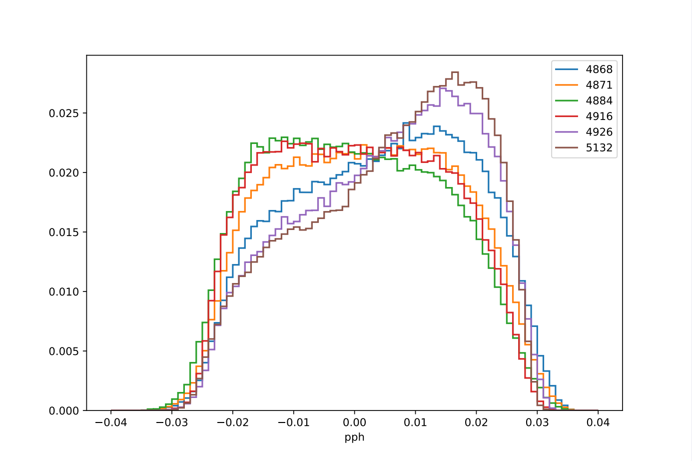
angles (V)

HMS *yptar (ph)* angle. For several kinematics. All simc.



angles (VI)

SHMS *yptar (ph)* angle. For several kinematics. All simc.



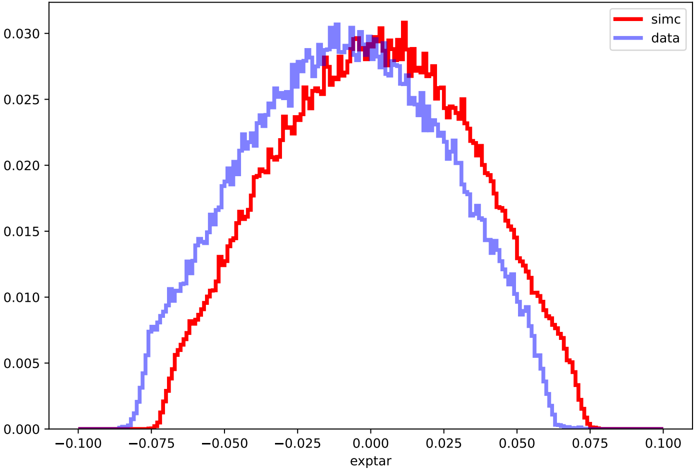
Here (as well as in prev. slides) try to focus on the **edges** of the distribution, not the middle!

angles (VII)

Well...

- the insight about the edges might have been **not** incorrect!
- **Q:** How do we turn this interesting bit into offsets?
- Here is my (GN) $0.02:
    - considering one angle at the time...
    - we “slide” the data distribution wrt its simulated twin
    - ...trying to make the two match (while ignoring the middle of each distribution)

Something like this...



angles (IX)

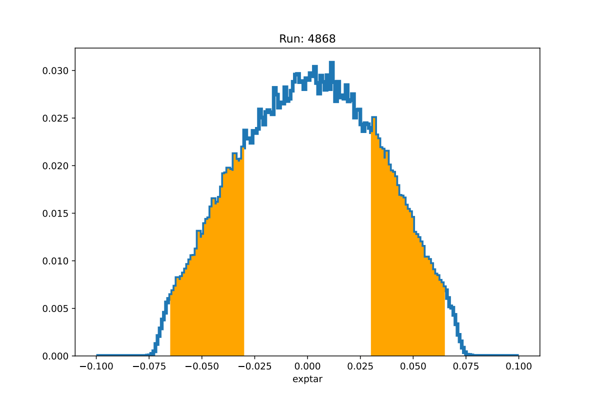
GN's $0.02 (cont.)

- though the animation on the prev. slide seemed *legit*...
- Q: How we decide which offset is "best"?
- to do this we would need to:
  - define the **edges** of the distribution
  - devise a way of matching data a simc edges (of the same angle, of course!)
- implement this in the language of our choice and see what gives...

angles (who's counting?)

Define edge:

- keep away from the very end of the distribution (no statistics there, anyway!)
- keep away (if possible!) from the places where the distribution turns flat-ish.
- something like this, maybe?



angles (XI)

Define "matching":

- aka "cost function", "metric", χ^2, etc. Sorry, this slide contains **math!**
- with the **low (_l)** and **high (_h)** ranges defined
- ...for both data and MC do:
  - Obtain the integrals for these 4 distributions: I_{ij} w/ $i \in [l, h]$ and $j \in$ [data,mc]
  - cross–normalize the two:

$$C = \frac{I_{ld} + I_{hd}}{I_{lmc} + I_{hmc}} \quad (1)$$

  - finally, define the **cost function**:

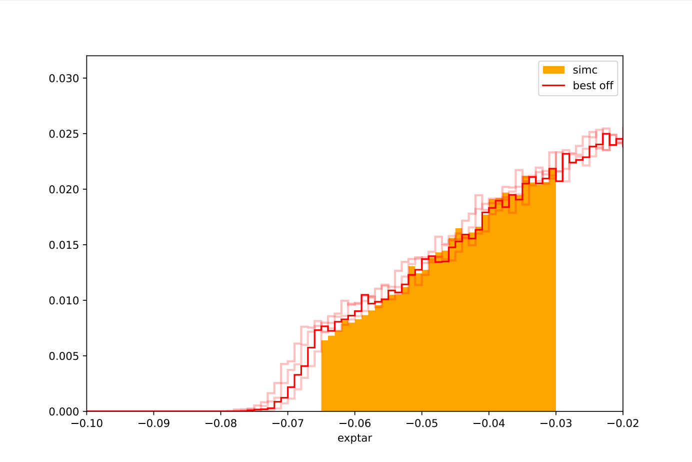
$$cost\_function = \sum_{i=(l,h)} (I_{id} - C \cdot I_{imc})^2 \quad (2)$$

- this leverages the fact that the l/h histograms slide **simultaneously**, not independently. We're trying to match both slopes at the same time!
- this gives us an **objective** way of picking the best offset

Does this work? (angles (XII))

It seems so

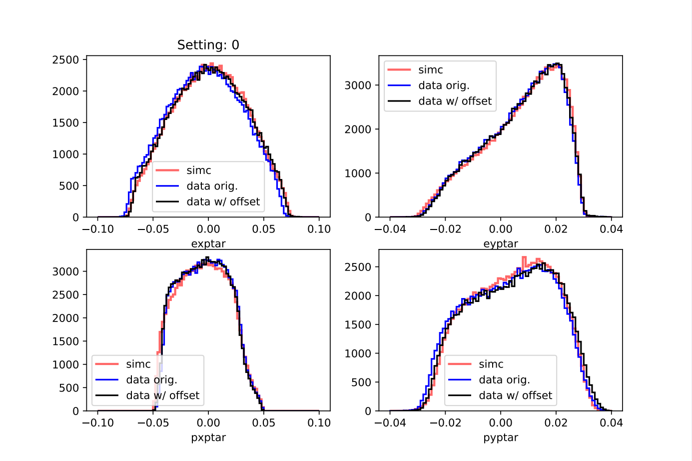
- Here is the **exptar_l** for a few possible offsets.



How about the other angles? (angles (a baker's dozen!))

Glad you asked:

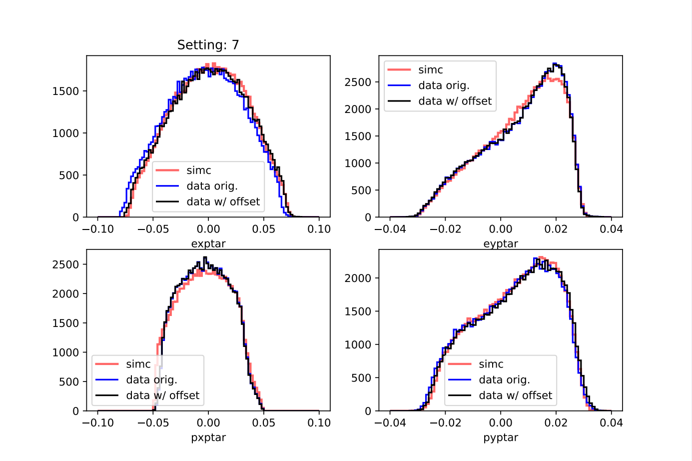
- Here is one 10.6 GeV kinematic setting



How about the other angles? (angles (really?))

Here is another setting:

- (we do have plots for all, of course!)

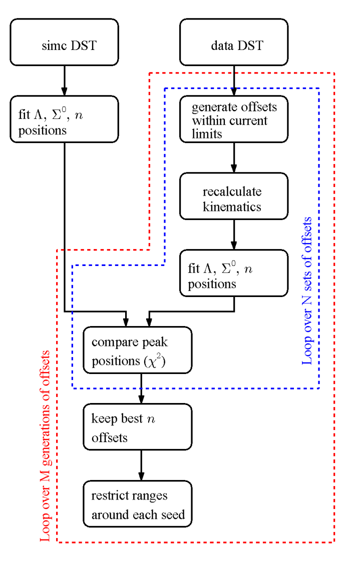


Next steps

The angular slip–and–slide shown above

- should provide us with the best *ansatz* for angular offsets
- we could: a. freeze these and continue or
- b. restrict their range around these best values (say to a small fraction of a mr)
- ...and continue with our 9D search. we opted for b.
- as shown in our 10/08/2025 presentation
- use the known positions of the Λ, Σ^0, and n to get the remaining offsets.
- for the visual learners among us, we'll illustrate this with a flowchart
- (in the 30+ page write–up there is a chart for the angle–finding as well!)

Momentum and beam offsets



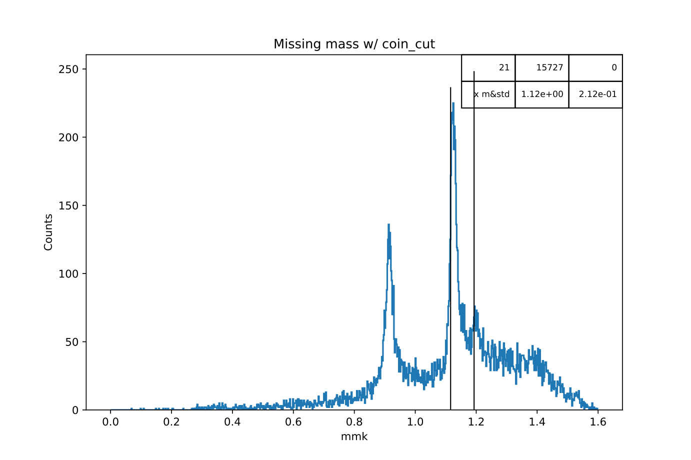
As seen on the chart...

- genetic algorithm
- generate random offsets
- evaluate cost function
- keep the best **n** sets as the **parents** for the next generation (we ran 3 gen.)
- somewhat CPU intensive (though not hdd intensive if using DSTs)
- ~5000 re–evaluations of kinematic quantities for each setting
- minutes–hour–ish on modest hardware (even in python!)
- can (and did!) run in parallel (3.14+)

Does it work?

Here are some samples

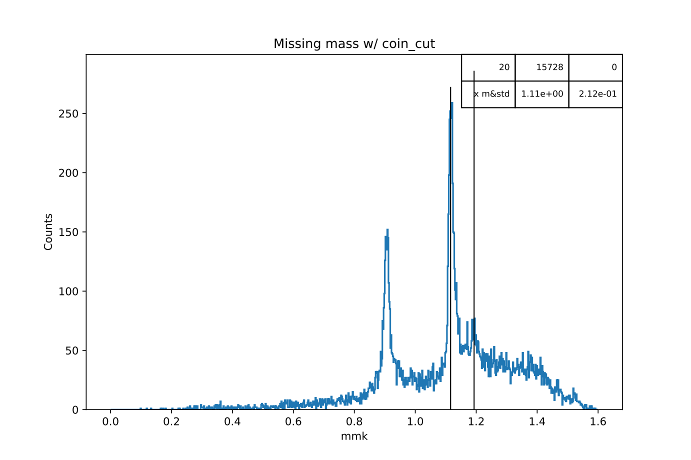
- Before (no offsets)



Yep!

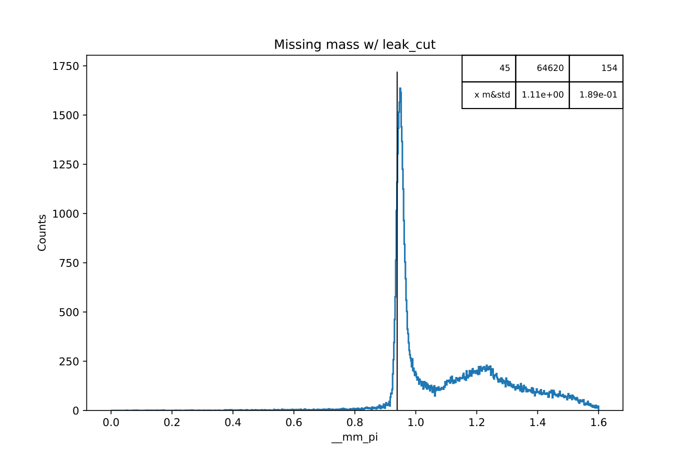
Here are some more samples

- After. And remember, the angles were pre–matched/did not change!



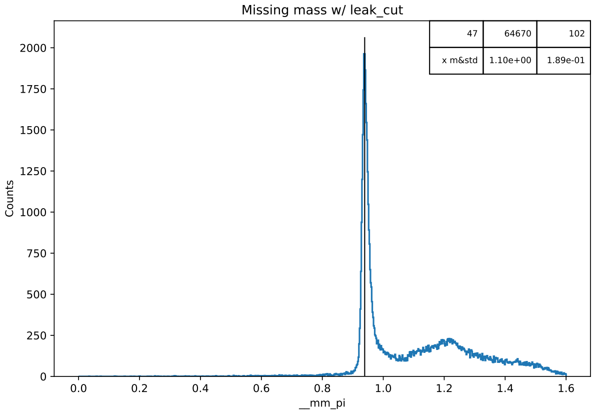
Yep!

- neutron (π^+ missing mass - run this w/ the actual 0.139...). Before



OK

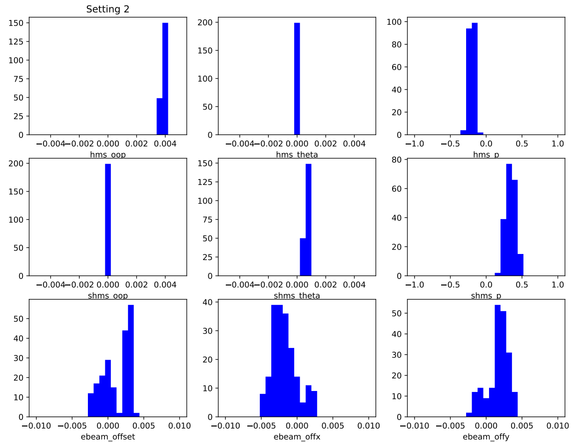
- neutron (π^+ missing mass). After



How large are these offsets?

Let's take a look at a distribution of these offsets

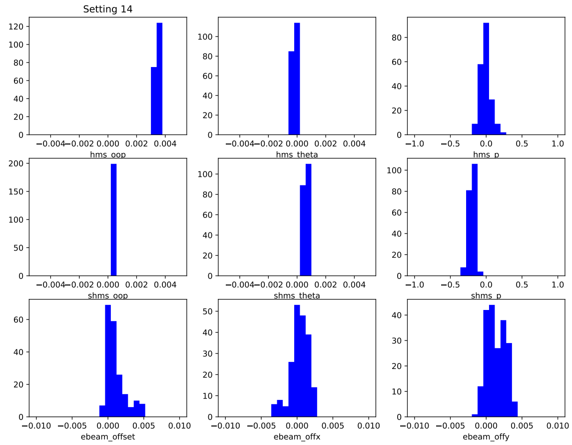
- a random 10.6 GeV setting. ~200 3–rd generation offsets shown.



How large are these offsets? (II)

Let's take a look at a distribution of these offsets

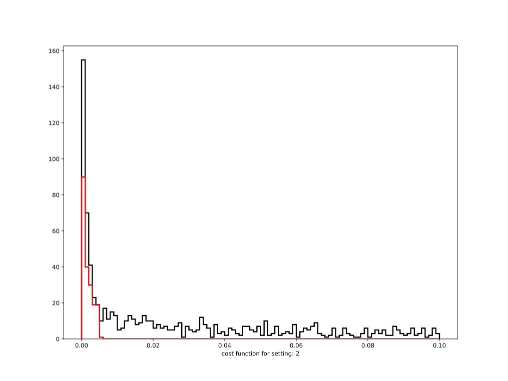
- and another. ~200 3–rd generation offsets shown.



How large are these offsets? (III)

As seen on the prev. slides

- our *(meta)heuristic* algorithm provides series of offsets
- this is expected, as the problem is under–determined
- all of these are **viable** results for our problem
- and χ^2 does not help much! (see below)



Uncertainties

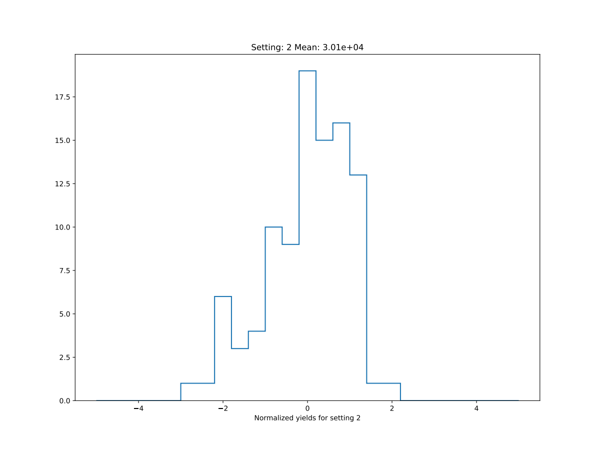
Focusing on the size of the offset might not be the right approach

- ...as the symmetry of the problem can produce versions that look different (as far as offsets go), but are irrelevant for the physics of interest
- see mirror–like image of *beamx* and *beamy* offsets a couple of slides ago
- or, possibly, a positive *out–of–plane* offset in one spectrometer paired w/ a negative oop offset in the other. Unpolarized obs. are immune to this.
- the better (in our opinion) question to ask is this:
- **Q:** how much does the yield change when using "this" or "that" set of offsets?
- luckily, the n peak has plenty of statistics, we cannot think of a mechanism that will invalidate these results for hyperons, and...
- we did keep **all** the offsets from the last generation
- therefore: re–evaluate all kinematic quantities for the best 200 (or so) sets of offsets, and extract the pi^+ missing mass yield
- see how it varies

Uncertainties (II)

π^+ mm yield residuals for various sets of offsets

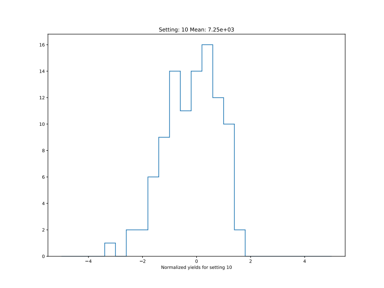
- **NOTE:** in the typical ML/AI fashion, this distribution was **normalized**
- i.e. subtract the mean and divide by std



Uncertainties (III)

Looking at all settings:

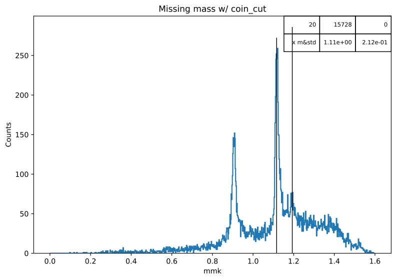
- we find that the yield change due to diff. sets of offsets is at the 0.05% or less for most settings!
- which is not only good.
- it is **justifiable/documentable(?)**



with initial angle fitting
/10 best offsets
pi+ yields

Off #	Yield	Std	%Err
	38718.162	23.389	0.060
	63783.333	21.822	0.034
	30145.465	7.956	0.026
	32296.263	9.999	0.031
	79534.343	28.023	0.035
	41181.535	12.943	0.031
	14529.646	7.376	0.051
	9299.818	1.513	0.016
	14200.384	4.160	0.029
	7252.576	2.458	0.034
	8076.768	2.538	0.031
	3508.566	6.298	0.179
	11482.697	4.493	0.039
	1920.414	0.696	0.036
	3314.333	0.667	0.020
	2069.596	1.490	0.072
	3303.808	1.228	0.037
	1315.101	0.732	0.056
	25609.253	9.022	0.035
	12994.222	5.695	0.044
	8083.848	2.765	0.034

Quo Vadis?



I hope I convinced you that:

- **edge–matching** the angular distributions +
- using **sharp** features (missing mass) in the data itself
- = a method for inferring kinematic **offsets**
- built–in evaluation of uncertainties
- adaptable to other Hall C (A?) exp.

- JMU group's next focus: Λ & Σ^0 PID
- **THANK YOU!**