Methods for non-iterative muon sagitta bias calculations

Christopher Lester

10th August 2020

Contents

1 Preamble

This document contains a description of a method of non-iterative muon sagitta bias calculation that was put together in the first week of August 2020 to assist with a charge-flavour asymmetry analysis (colloquially known as the 'emu' search).

This write up is preceded by a description of numerous problems we encountered when trying to understand one source's description of the existing iterative baseline method(s) used by ATLAS for past muon sagitta bias corrections. It is important to make clear that statements we have made about the inadequacies of that particular description of the baseline method:

- are presented purely to provide context which might explain/motivate why we found it simpler to create a new method by ourselves,
- are not intended to suggest that the existing sagitta bias corrections are described badly everywhere (there are probably many better write-ups which we simply not seen), and
- are not intended to suggest that the existing iterative method is itself 'bad' in some way, or that our alternative method is in some way 'better'. On the contrary, the existing methods for estimating sagitta biases have had the attention of numerous members of the Muon Combined Performance (MCP) group for many years, and so is presumably very well understood and trusted. In contrast, ours is the work of only a fortnight of study from persons without any significant experience of muon calibration work.

We present our work here only with the hope that discussing different approaches could be useful in some way, perhaps by provide pointers towards directions in which further developments could be made, if desired. **We therefore make no claim that our method in its current form is any better than those you already have, or solves any particular problems you may be facing!**

2 Problems with a description of the baseline method for sagitta bias determination

Section 6.1.1 of [Aad+20] describes what we will refer to (for the purposes of this document) as the **baseline method** for determining ATLAS muon sagitta biases. The core of that **baseline method** are the statements copied into the following box for ease of reference:

Displacements of the reconstructed hits in the bending plane orthogonal to the track path result in a charge-antisymmetric alteration of the track curvature, which is parameterised as

$$p' = p(1 + qp_T\delta)^{-1} \tag{1}$$

where the un-primed quantities correspond to the true values, the primed quantities correspond to the reconstructed values, q refers to the sign of the electric charge of the particle and δ_{sagitta} is a bias parameter common to all measured momenta and uniquely defines the detector geometry deformation.

[... snip ...]

In general, geometrical distortions that bias sagitta measurements can be localised in specific regions of the detector. As a result, the sagitta bias parameter explicitly depends on the path of the track, which can be approximated by the direction of the track at the pp interaction point, given by η and ϕ: $\delta_{\text{sagitta}} \to \delta_{\text{sagitta}}(\eta, \phi)$. The difference at leading order in $\delta_{\text{sagitta}}(\eta, \phi)$ between the reconstructed dimuon invariant mass using the uncorrected geometry ($m_{\mu\mu}$) and the expected mass (m_Z) for each event is given by:

$$m^2_{\mu\mu} - m^2_Z \approx m^2_Z \left(p'^+_T \delta(\eta^+, \phi^+) - p'^-_T \delta(\eta^-, \phi^-) \right). \tag{2}$$

An iterative procedure is used to determine $\delta_{\text{sagitta}}(\eta, \phi)$. For the i-th iteration, $\delta_{\text{sagitta},i}(\eta, \phi)$ is computed for every muon in the $Z \to \mu^+\mu^-$ sample with:

$$\delta_{\text{sagitta},i}(\eta, \phi) = -q \frac{m^2_{\mu\mu} - m^2_Z}{2m^2_Z} \frac{(1 + qp'_T \langle \delta_{\text{sagitta},i-1}(\eta, \phi) \rangle)}{p'_T} + \langle \delta_{\text{sagitta},i-1}(\eta, \phi) \rangle \tag{3}$$

where $\langle \delta_{\text{sagitta},i-1}(\eta, \phi) \rangle$ is the mean of the previous iteration for all muons in that (η, ϕ) region. The value of $m^2_{\mu\mu}$ is computed as in Eq. (2) also using the mean of δ_{sagitta} from the previous iteration. The iterations are repeated until convergence is reached.

We tried to implement the **baseline method** above but found ourselves unable to do so on on account of the following sources of confusion present in the text reproduced above:

1. It was not clear what quantity m_Z was intended to represent. The text says that it is 'the expected mass for each event', but the text is not clear whether the word 'expected' was intended to mean:
   - the statistically expected mass of the Z-boson from the PDG book, namely 91.2 GeV, or
   - the mean mass one would expect to see in a sagitta bias-free detector after averaging over all the dimuon events inside the mass window used to select events close to the Z (this is different to 91.2 GeV and

depends on the window end positions), or

- the mass one would 'expect' to see for each individual event if sagitta biases had been removed (this differs from event to event).

.

2. It was not clear what quantity $m_{\mu\mu}$ was intended to represent as it appears (at least to us) to be defined in two mutually incompatible ways in the text:

   - One part of the text describes $m_{\mu\mu}$ as 'the reconstructed dimuon invariant mass using the uncorrected geometry'. [We take here 'uncorrected' to mean 'no sagitta bias corrections have been applied'.]
   - But elsewhere the text also says that: 'The value of $m^2_{\mu\mu}$ is computed as in Eq. (2) also using the mean of δ_{sagitta} from the previous iteration.'

   The above statements read as if they are incompatible as the former seems to say $m_{\mu\mu}$ is constant while the latter appears to say that it is progressively corrected and so depends on iteration!

3. It was not clear what the primed quantities are. The text says that 'the primed quantities correspond to the reconstructed values' but it does not say whether these are the values reconstructed before or after sagitta corrections. We cannot infer the intent by process-of-elimination using the definition of the un-primed quantities, as it is itself also broken! [See next issue.]

4. It was not clear what the unprimed quantities were either! The text says that 'un-primed quantities correspond to the true values'. This text seems clear enough: 'true' values should be ones which nature chose for our event. The unprimed quantities should therefore be things which we (as experimenters) may never know the actual values of, although we might approach them closely. As such they ought not to change from iteration to iteration of an external algorithm. They are constants within any event. Alas, in the LHS of (2) and in the RHS of (3) there appears the unprimed quantity $m_{\mu\mu}$ together with an instruction that 'The value of $m^2_{\mu\mu}$ is computed as in Eq. (2) also using the mean of δ_{sagitta} from the previous iteration'. This instruction tells us that $m^2_{\mu\mu}$ (despite its lack of a prime) **does** depend on iteration after all, and so cannot represent a constant truth quantity! It must represent something else which has not been described unambiguously.[1]

[1] The counter argument could be made that perhaps the lack of a prime on the $m_{\mu\mu}$ is of no consequence since perhaps it only intended that the presence or absence of primes has meaning when applied to the quantities in (1). But that argument does not wash as there are primed quantities in (2) (and elsewhere) which are not found in (1) – so they would then themselves be undefined if the counter argument were applied.

5. The initial values of the $\delta_{\text{sagitta}}, i(\eta, \phi)$ for the $i = 0$ iteration are not defined despite appearing (probably unintentionally) to be critical for the method as described. The lack of initial values would not matter if the initial values were ultimately not relevant (e.g. if the δ corrections could be initialised with almost any small value, perhaps zero, as we had originally assumed was likely). However, closer inspection of the text shows that it is not possible to set $\delta_{\text{sagitta},0}(\eta, \phi) = 0$. The reason is that if one does that, then the RHS of (2) becomes identically zero at iteration $i = 0$. This would in turn mean that when calculating $\delta_{\text{sagitta},1}(\eta, \phi)$ using (3) one would find (3) would reduce itself to

$$\delta_{\text{sagitta},1}(\eta, \phi) = -0 + \langle \delta_{\text{sagitta},0}(\eta, \phi) \rangle = 0$$

since the quantity $(m^2_{\mu\mu} - m^2_Z)$ in (3) would be zero when $i = 1$ given the instruction to use "the value of $m^2_{\mu\mu}$ using the mean of δ_{sagitta} from the previous iteration". The text, therefore, (if taken literally) leads to the values $\delta_{\text{sagitta},i}(\eta, \phi)$ being zeros in ALL iterations, unless a non-zero initial value is taken! This is clearly crazy, so we assume that there is some other typo somewhere in the text that would resolve this issue.[2]

6. The text tries to explain (in (3)) how each iteration of bias approximations would obtained from the last, however this step does not state what *precisely* the update step attempts to achieve in total or each time it is used. In more detail:

   - While it is clear that it is in the mind of the writer that the progressive updates are intending to 'get better bias approximations by the repeated use of earlier bias approximations', what is never specified is what actually constitutes the intended convergence goal that an idealised 'best' bias approximation would have. In other words, if this convergence method were to converge, what property is the converged solution intended to have? Would an eventual converged solution minimise some a function of the data? Or is nothing being minimised and it's just a heuristic method? If something is being minimised, then what is it? Would the reconstructed Z-width be minimised? Or is something else targetted? If it were to be the Z-width that is being minimised, then under what conditions is it being minimised? E.g. does the method aim to leave the mean mass fixed or is that free to shift too? [with consequences for the overall mass scale!]
   - It is not stated whether it is known whether (or under what circumstances) this iteration step it is guaranteed to lead to convergence, or whether convergence depends on any (unstated) initial conditions on δ, etc.

[2] That is to say: we presume that actually it is *intended* that one *can* take zeros as initial values for the $\delta_{\text{sagitta},1}(\eta, \phi)$, and that there is a typo somewhere in the box that (if fixed) would stop all subsequent iterations from remaining at zero.

- It is not stated whether the (unstated) goal can have multiple solutions (local optima as well as global optima) and if so how the iteration process copes with that.[3]

In short, there were so many sources of confusion or potential confusion in the text of Section 6.1.1 of [Aad+20] that all our attempts to implement the **baseline method** failed in one way or another. Furthermore, the lack of any explicitly stated goal for the method (i.e. what property optimal biases were intended to satisfy) made it impossible to re-derive the method from a firm starting point. Attempting to reverse-engineer the design goals from the stated text were not possible on account of the missing details and ambiguities (or typos) already mentioned above.

For the above reasons, it was decided to put the whole **baseline method** of [Aad+20] to one side and approach the problem afresh by making our own *ab initio* determination of sagitta biases.

3 Determining sagitta biases *ab initio*

3.1 Important caveat regarding proof-of-principle method presented here

When constructing the first proof-of-principle implementation of the method below, no proper distinction was made between **momenta** and **transverse-momenta** for the purposes of the sagitta definition.[4]

This simplification is in no way intended to stay long term, and there is nothing in the mathematics that prevents the proper distinction between p and p_T being included. The algorithm presented here will not run more slowly if so corrected. Nor will it become more complex. The only reason this fix has not yet been implemented is that it was not (yet) needed for the work we were doing on the 'emu' analysis, and it would take a short period of time to validate it. Until that change is implemented, though, the bias corrections estimated by our method will appear consistently slightly too big (or maybe it's too small?) at large $|\eta|$.

We repeat, however, that this inadequacy is (in principle) trivial to remove from the proof-of-principle method presented here.

[3] Aside: perhaps this omission us because the iterative method was not originally seen as "a thing aiming to calculate *the* solution to a particular well defined but unstated problem", but was instead seen or conceived as the goal in itself – that the method's relevance is somehow supposed to be obvious to the reader, and so the outcome of the iteration sort of **defined itself to be** the desired outcome? This is just guesswork from an outsider!

[4] In other words: although in the inner detector it is only the p_T (not the p) which is constrained by the measured muon sagitta, we have ignored that distinction and have defined sagittas to be inverse p not inverse p_T.

3.2 Notation used here

- A reconstructed quantity which is not yet sagitta corrected (or which perhaps never needs or will never get a sagitta correction) is represented by a symbol **without** a dot ($\bullet$). For example: m_{ll} would mean a reconstructed invariant mass of two leptons prior to any sagitta corrections to the lepton momenta.

- A symbol with a dot ($\bullet$) is used to represent a reconstructed quantity that **has** been fully sagitta corrected. For example: $m_{ll\bullet}$ might indicate m_{ll} after it has received sagitta corrections.

- Various subscritpts are used, but the most common are:

  - The subscript i labels which **event** the quantity comes from. The number of events is taken to be N so $1 \leq i \leq N$.
  - The subscript s (and sometimes also t) is used to indicate which **sagitta bias bin** a quantity comes from. A single bin label s is used regardless of whether the binning structure is one-dimensional (e.g. binning only in eta) or two-dimensional (e.g. binning in both eta and phi). The number of bins is taken to be B, so $1 \leq s \leq B$ (and $1 \leq t \leq B$). See examples of usage in the text around and following equation (22).

- In certain places it is necessary for us to write down means, variances or covariances of reconstructed quantities (either with or without sagitta bias correction). In all cases we intend these quantities to be obtained by averaging over the N events ($i = 1, 2, \ldots, N$) in the relevant sample. For example by $\mathrm{Var}\left[m_{lli}\right]$ we would mean:

$$\mathrm{Var}\left[m_{lli}\right] \equiv \left(\frac{1}{N}\sum_{i=1}^{N} m_{lli}^2\right) - \left(\frac{1}{N}\sum_{i=1}^{N} m_{lli}\right)^2 \tag{4}$$

  while for $\mathrm{Cov}\left[m_{lli}, p_i^+\right]$ we would mean

$$\mathrm{Cov}\left[m_{lli}, p_i^+\right] \equiv \left(\frac{1}{N}\sum_{i=1}^{N} m_{lli} \cdot p_i^+\right) - \left(\frac{1}{N}\sum_{i=1}^{N} m_{lli}\right) \cdot \left(\frac{1}{N}\sum_{i=1}^{N} p_i^+\right). \tag{5}$$

  However, because the i indices are not explicitly used on the left hand sides of (4) or (5) we will sometimes omit the i for brevity, writing $\mathrm{Var}\left[m_{ll}\right]$ for $\mathrm{Var}\left[m_{lli}\right]$ or writing $\mathrm{Cov}\left[m_{ll}, p^+\right]$ for $\mathrm{Cov}\left[m_{lli}, p_i^+\right]$, and so on. We hope that this simplification (where used) will not confuse readers.[5] Likewise, $\sum_{i=1}^{N}$ will often be notated as just $\sum_i$ since the range of event indices i is implicit (as already noted).

[5] Perhaps the shorter form is actually closer to mainstream notation anyway. Many probability text books use notations similar to $E(X) = \frac{1}{n}\sum_{i=1}^{n} x_n P(X = x_n)$ which also omit indices on the left while using some other signifier (in this case capitalisation) to distinguish a random variable from values which it might take.

3.3 Definitions used here

Ignoring the masses of the individual muons, we define the invariant mass of the dilepton system to be:

$$m_{ll}^2 = 2p^+p^-(1-\cos\theta_{ll}) \tag{6}$$

We define sagittas as $s^\pm$ inverse **momenta** not inverse **transverse momenta**:[6]

$$p^+ = \frac{1}{s^+} \tag{7}$$

$$p^- = \frac{1}{s^-} \tag{8}$$

We define the sagitta corrections $\delta^\pm$ in terms of the corrected and uncorrected sagittas (respectively $s_\bullet^\pm$ and $s^\pm$) by the following relationship between them:

$$s = s_\bullet - q\delta \tag{9}$$

Note that with above defn, the assumption (shared with [Aad+20]) is that there is one universal bias correction, which is applied one way for positive muons and the other way for negative muons. E.g. if the symbols $\pm$ were used to indicate the (η,ϕ)-bins of the lepton of the specified charge, then (9) would look like:

$$s^+ = s_\bullet^+ - \delta^+ \tag{10}$$

$$s^- = s_\bullet^- + \delta^- \tag{11}$$

in which the signs **before** the deltas are accounting for charge, while the signs in the $\delta^\pm$ are merely labelling the bin for the relevant lepton.[7] The above defs lead to:

$$p = \frac{1}{s_\bullet - q\delta} \tag{12}$$

$$= \frac{1}{s_\bullet}\frac{1}{1-\frac{q\delta}{s_\bullet}} \tag{13}$$

$$= \frac{1}{s_\bullet}\left(1+\frac{q\delta}{s_\bullet}+O\left(\left(\frac{\delta}{s_\bullet}\right)^2\right)\right) \tag{14}$$

$$= p_\bullet\left(1+qp_\bullet\delta+O\left((p_\bullet\delta)^2\right)\right) \tag{15}$$

[6] An explanation for this non-standard and soon-to-be-removed choice may be found in Section 3.1.

[7] In Section 6.2 we discuss the compatibility (or otherwise) between the sign conventions used here and those in [Aad+20]. If subsequent clarification of the conventions of [Aad+20] suggests it may be useful, we may decide to revise our sign conventions.

3.4 Working

Using the above definitions we get:

$$m_{ll}^2 = 2p^+p^-(1-\cos\theta_{ll}) \tag{16}$$

$$= 2p_\bullet^+\left(1+p_\bullet^+\delta^+ + O\left(\left(p_\bullet^+\delta^+\right)^2\right)\right)p_\bullet^-\left(1-p_\bullet^-\delta^- + O\left(\left(p_\bullet^-\delta^-\right)^2\right)\right)(1-\cos\theta_{ll}) \tag{17}$$

$$= m_{ll\bullet}^2\left(1+p_\bullet^+\delta^+ - p_\bullet^-\delta^- + O\left(\left(p_\bullet^+\delta^+\right)^2\right) + O\left(\left(p_\bullet^-\delta^-\right)^2\right) + O\left(\left(p_\bullet^+\delta^+\right)\left(p_\bullet^-\delta^-\right)\right)\right) \tag{18}$$

which we will abbreviate as

$$m_{ll}^2 \approx m_{ll\bullet}^2\left(1+p_\bullet^+\delta^+ - p_\bullet^-\delta^-\right) \tag{19}$$

or equivalently as

$$m_{ll\bullet}^2 \approx m_{ll}^2\left(1-p_\bullet^+\delta^+ + p_\bullet^-\delta^-\right) \tag{20}$$

$$\approx m_{ll}^2\left(1-p^+\delta^+ + p^-\delta^-\right) \tag{21}$$

which are our analogues the (2) which we reproduced from [Aad+20].

3.5 Statement of Goal(s)

The key difference between how we will proceed compared to how things are described in [Aad+20] is that we will try to set out an unambiguous 'goal' which will define what the measured sagitta biases should actually be. Then, given such a goal (or goals) one may then attempt to derive a calculation procedure that will achieve that goal.

We have presented examples of two such goals (see below) and have shown how the calculation would be performed to measure the biases for each case.

Note that it is perfectly possible that the example goals we have set are incompatible with your needs – possibly even laughably so. We are, after all, not paid-up MCP members and have neither your experience of nor an awareness of the key challenges your Muon Calibration work faces.

If our stated goals are not suited to your needs (whether laughably so or not!) then we would argue that you should replace *our* goals with ones which is more suited to *your* needs, and then you should repeat similar derivations to those we have set out below. This should allow you to adapt our method to your needs.

3.5.1 Our initial goal

We initially decided to *define* the **measured sagitta biases**, δ, to be

those which **minimise** Var $\left[m_{ll\bullet}^2\right]$.

This variance is to be calculated over all events in some calibration sample. The 'principle' being assumed here is that accidental sagitta biases can only broaden the Z-mass peak.[8]

3.5.2 Secondary goal

As an example of how one could go about varying the goals, as a second case we have investigated what happens if one instead *defines* the **measured (fixed scale) sagitta biases**, δ, to be

> those which **minimise** $\mathrm{Var}\left[m^2_{ll\bullet}\right]$ **subject to** $E[m^2_{ll\bullet}]$ **remaining fixed**.

Again, these variances and expectations are to be calculated over all events in some calibration sample, and should be replaced by other goals you set if you don't like them.

3.5.3 Motivation for the secondary goal

The goal of section 3.5.1 allows (in principle) sagitta biases to take on any values they please, so long as the variance of the Z-peak is thereby minimised. It is therefore possible (in principle) for the section 3.5.1 objective to move the mean mass of the Z-boson peak up or down.[9]

Such movements might be undesirable. For example, if mass-scale calibrations were fixed and applied *before* sagitta correction determination was begun,[10] then the sagitta corrections determined via the goal of section 3.5.1 would have the ability to "partly undo" those earlier mass-scale calibrations.

This motivated the consideration of the secondary goal (of section 3.5.2) which restricts sagitta biases to values which leave the mean of the Z-peak unaltered. [The reader could, of course, propose more complicated alternatives at his or her pleasure.]

3.6 Working towards those goal(s)

To assist in the calculation of the variances needed by the above goal statements, we will create B dimensional vectors and $B \times B$ dimensional matrices where B is the number of bins over which one wishes to discretise the sagitta biases. Specifically we will place the B biases which we aim to find into a vector $\vec{\delta}$

[8] Perhaps that is a principle which you will find laughable and would need modifying as suggested in the introduction to Section 3.5.

[9] Though such movements can occur (and indeed will occur if doing so would make the width of the distribution smaller) such sagitta induced shifts are presumably small? Any change in a bias bin will tend to increase the momenta of some muons and will correspondingly decrease the momenta of muons of the opposite charge by a similar amount, so all in all the mass peak will not shift at first order. Such shifts will only occur at second order as a result of the difference in spectrum between positive and negative muons.

[10] I don't know in what order these operations are performed.

having B components:

$$\vec{\delta} = \begin{pmatrix} \delta_1 \\ \vdots \\ \delta_B \end{pmatrix}. \tag{22}$$

Furthermore, for every event $e_i \in E$ (with E being the set of all events) we define a B-dimensional vector $\vec{e_i}$ as follows:

$$\vec{e_i} \equiv \begin{pmatrix} 0 \\ \vdots \\ 0 \\ m^2_{lli} p_i^+ \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ m^2_{lli} p_i^- \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{23}$$

in which the p_i^+ term appears in the b_i^+th row and the p_i^- term appears in the b_i^-th row, where $b_i^\pm$ values contain the numbers corresponding to the (η, ϕ)-bins of the positive and negative muons in event e_i. With that notation in mind we may write

$$m^2_{ll\bullet i} = m^2_{lli} - \vec{e_i} \cdot \vec{\delta} \tag{24}$$

and hence

$$\begin{aligned}
\mathrm{Var}\left[m^2_{ll\bullet}\right] &= \mathrm{Var}\left[m^2_{lli} - \vec{e_i} \cdot \vec{\delta}\right] && (25)\\
&= \mathrm{Var}\left[m^2_{lli}\right] + \mathrm{Var}\left[\vec{e_i} \cdot \vec{\delta}\right] - 2\mathrm{Cov}\left[m^2_{lli}, \vec{e_i} \cdot \vec{\delta}\right] && (26)\\
&= \mathrm{Var}\left[m^2_{lli}\right] + \mathrm{Cov}\left[\vec{e_i} \cdot \vec{\delta}, \vec{e_i} \cdot \vec{\delta}\right] - 2\mathrm{Cov}\left[m^2_{lli}, \vec{e_i}\right] \cdot \vec{\delta} && (27)\\
&= \mathrm{Var}\left[m^2_{lli}\right] + \sum_{s,t} \mathrm{Cov}\left[(\vec{e_i})_s, (\vec{e_i})_t\right] \delta_s \delta_t - 2\sum_s \mathrm{Cov}\left[m^2_{lli}, (\vec{e_i})_s\right] \delta_s && (28)
\end{aligned}$$

and so the minimum variance occurs when

$$\begin{aligned}
0 &= \frac{\partial}{\partial \delta_k} \mathrm{Var}\left[m^2_{ll\bullet i}\right] && (29)\\
&= \sum_t \mathrm{Cov}\left[(\vec{e_i})_k, (\vec{e_i})_t\right] \delta_t + \sum_s \mathrm{Cov}\left[(\vec{e_i})_s, (\vec{e_i})_k\right] \delta_s - 2\mathrm{Cov}\left[m^2_{lli}, (\vec{e_i})_k\right] && (30)
\end{aligned}$$

i.e. when

$$\sum_t \mathrm{Cov}\left[(\vec{e_i})_k, (\vec{e_i})_t\right] \delta_t = \mathrm{Cov}\left[m^2_{lli}, (\vec{e_i})_k\right] \tag{31}$$

which is a linear algebra problem of the form

$$M\vec{\delta} = \vec{k} \tag{32}$$

where M is the $B \times B$-matrix having components:

$$(M)_{st} \equiv \mathrm{Cov}\left[(\vec{e}_i)_s, (\vec{e}_i)_t\right] \tag{33}$$

and $\vec{k}$ is the B-vector having components

$$(\vec{k})_t \equiv k_t \equiv \mathrm{Cov}\left[m^2_{lli}, (\vec{e}_i)_t\right]. \tag{34}$$

If we can compute the components of M and $\vec{k}$ efficiently, we can therefore solve the linear system (32) to find the sagitta biases in $\vec{\delta}$. Let us consider each in turn:

$$\begin{aligned} k_t &\equiv \mathrm{Cov}\left[m^2_{lli}, (\vec{e}_i)_t\right] && (35)\\ &\equiv \mathrm{Cov}\left[m^2_{lli} - u, (\vec{e}_i)_t\right] \qquad \text{(for any constant } u\text{)} && (36)\\ &= \left(\frac{1}{N}\sum_i (m^2_{lli} - u)(\vec{e}_i)_t\right) - \left(\frac{1}{N}\sum_i (m^2_{lli} - u)\right)\left(\frac{1}{N}\sum_i (\vec{e}_i)_t\right) && (37) \end{aligned}$$

and

$$\begin{aligned} (M)_{st} &\equiv \mathrm{Cov}\left[(\vec{e}_i)_s, (\vec{e}_i)_t\right] && (38)\\ &= \left(\frac{1}{N}\sum_i (\vec{e}_i)_s(\vec{e}_i)_t\right) - \left(\frac{1}{N}\sum_i (\vec{e}_i)_s\right)\left(\frac{1}{N}\sum_i (\vec{e}_i)_t\right). && (39) \end{aligned}$$

The above forms motivate the creation of the following four quantities (one a scalar g, two B-vectors $\vec{E}$ and $\vec{F}$, and one symmetric $(B \times B)$-matrix H) defined as follows:

$$g = \frac{1}{N}\sum_i (m^2_{lli} - u) \tag{40}$$

$$\vec{E} = \frac{1}{N}\sum_i \vec{e}_i \tag{41}$$

$$\vec{F} = \frac{1}{N}\sum_i (m^2_{lli} - u)\vec{e}_i \tag{42}$$

$$H = \frac{1}{N}\sum_i (\vec{e}_i)(\vec{e}_i)^T \tag{43}$$

Note that each of the above quantities can be incrementally filled in a single pass over the events reading only one event at a time. It is not necessary to store all events in memory at once. Their filling time is evidently proportional to the number of events N in all cases. Less obviously the filling time does not

grow with B or $(B \times B)$ since each $\vec{e}$ contains at most two non-zero elements. In terms of those quantities, the desired (symmetric) matrix M and vector $\vec{k}$ are then simply found to be:

$$M = H - \vec{E}\vec{E}^T \tag{44}$$

$$\vec{k} = \vec{F} - g\vec{E}. \tag{45}$$

Note that the quantity u may take (in principle) any constant value. However, in practice, the calculation of $\vec{k}$ will have the greatest numerical accuracy (given floating point rounding issues) if u is chosen to be approximately $91.2GeV$ (i.e. the Z-boson mass) since this will tend to keep $|g|$ and $|\vec{F}|$ small ensuring that (45) does not end up performing a subtraction between two positive numbers.

3.7 Implementation and timing

The implementation of the proof-of-principle may be found here: `https://gitlab.cern.ch/emus/OSDFChargeFlavourAsymmCode/-/blob/master/sagitta/lester/root_to_matrix.cc`.

That most important lines of code in that file are extracted and presented schematically below.

The parts which compute the four helper quantities are, schematically, as follows:

```
1   Event event;
2
3   // Helper quantities:
4   double mean_mll2 = 0;
5   Eigen::MatrixXd H_mean_ee_mat = Eigen::MatrixXd::Zero(bins,
        bins);
6   Eigen::VectorXd E_mean_e_vec = Eigen::VectorXd::Zero(bins);
7   Eigen::VectorXd F_mean_mll2e_vec = Eigen::VectorXd::Zero(bins)
        ;
8   double          g_mean_mll2 = 0;
9
10
11
12  // Loop over events:
13  long int num_events=0;
14  while (true) {
15    event = readNextEvent();
16    ++num_events;
17
18    // (tweaking of event omitted for clarity)
19
20    // Now do some record keeping ...
21    const double safetified_value_of_mll2 = (
22         (config.safety_factor) ? // Improves numerical
                precision of k calc. Not to be used in the M calc.
                 So only use in g_mean_mll2 and F_mean_mll2e_vec
23         (event.mll2 - (Z_mass*Z_mass) * config.safety_factor) :
24         (event.mll2)
25       );
```

```
26      g_mean_mll2 += safetified_value_of_mll2;
27      for (const Lepton & lepS : event.leps) {
28         const double thingS = lepS.new_thing() * event.mll2;
29         const int binS = config.bins2D.find_bin(lepS);
30         E_mean_e_vec(binS) += thingS;
31         F_mean_mll2e_vec(binS) += safetified_value_of_mll2 *
               thingS;
32         for (const Lepton & lepT : event.leps) {
33             const double thingT = lepT.new_thing() * event.mll2;
34       const int binT = config.bins2D.find_bin(lepT);
35       H_mean_ee_mat(binS, binT) += thingS * thingT;
36         }
37      }
38   } // loop over events
39
40   // Tidy up normalisations now that we know how many events
         there were:
41   g_mean_mll2 /= static_cast<double>(num_events);
42   E_mean_e_vec /= static_cast<double>(num_events);
43   F_mean_mll2e_vec /= static_cast<double>(num_events);
44   H_mean_ee_mat /= static_cast<double>(num_events);
```

The above step is limited by file access. It costs me about 1 second per 1,000,000 events read.

The lines computing M and $\vec{k}$ from the above are these which are a no-op as far as time constraints are concerned:

```
1   // Now finish the computations:
2   Eigen::VectorXd tmpK = F_mean_mll2e_vec - g_mean_mll2 *
        E_mean_e_vec;
3   Eigen::MatrixXd tmpM = H_mean_ee_mat - E_mean_e_vec *
        E_mean_e_vec.transpose();
```

The linear system of equations is solved (in about 10 seconds for a 40x40 binned set of biases) in the line reading:

```
1 Eigen::VectorXd deltaVec = M.colPivHouseholderQr().solve(K);
```

.

3.8 Timing summary

The proof-of-principle implementation takes approximately 13 seconds to calculate a 40x40 binned bias correction based on 1,000,000 dimuon events. About 1 second of that is spent reading the data, and 10 seconds are spent on fixed-time operations that would not increase with larger datasets (but would increase if the binning structure were made finer) and the remainder goes on debug output. Ultimately, in the limit of large datasets, the scaling time is proportonal to dataset size. Therefore the ultimate scaling is approximately 1 second per 1,000,000 events if a 40x40 binning is used.

3.9 The decrease Δ in the variances cased by the bias correction.

Note that with the matrices as defined, we could have written (28) as:

$$\mathrm{Var}\left[m^2_{ll\bullet}\right] = \mathrm{Var}\left[m^2_{lli}\right] + \vec{\delta}^T M\vec{\delta} - 2\vec{k}\cdot\vec{\delta} \tag{46}$$

therefore the **decrease** Δ in variance obtained by choosing good values of δ (which we hope is as large and positive as possible) is given by:

$$\Delta \equiv \mathrm{Var}\left[m^2_{lli}\right] - \mathrm{Var}\left[m^2_{ll\bullet}\right] = 2\vec{k}\cdot\vec{\delta} - \vec{\delta}^T M\vec{\delta}. \tag{47}$$

We will later see in (73) that the above formula for Δ may be considerably simplified.

3.10 Enforcing the constraint $\langle m^2_{ll\bullet}\rangle = \langle m^2_{ll}\rangle$

To meet the secondary goal of section 3.5.2 it is necessary for us to use a new Lagrangian which enforces the constraint in the title of this section.

The quantity $\mathcal{L}_0$ we have previously been minimising with respect to $\vec{\delta}$ could have been written as:

$$\mathcal{L}_0 = \frac{1}{2}\vec{\delta}^T M\vec{\delta} - \vec{k}\cdot\vec{\delta}. \tag{48}$$

If, instead, we had wanted to minimise $\mathrm{Var}\left[m^2_{ll\bullet}\right]$ subject to the constraint $\left\langle m^2_{ll\bullet}\right\rangle = \left\langle m^2_{ll}\right\rangle$ then using a Lagrange multiplier λ we could instead have minimised $\mathcal{L}_1$ with respect to $\vec{\delta}$ and λ with:

$$\begin{aligned}
\mathcal{L}_1 &= \frac{1}{2}\vec{\delta}^T M\vec{\delta} - \vec{k}\cdot\vec{\delta} - \lambda(\left\langle m^2_{ll\bullet}\right\rangle - \left\langle m^2_{ll}\right\rangle & (49)\\
&= \frac{1}{2}\vec{\delta}^T M\vec{\delta} - \vec{k}\cdot\vec{\delta} - \lambda\left(\left\langle m^2_{lli} - \vec{e}_i\cdot\vec{\delta}\right\rangle - \left\langle m^2_{ll}\right\rangle\right) & (50)\\
&= \frac{1}{2}\vec{\delta}^T M\vec{\delta} - \vec{k}\cdot\vec{\delta} + \lambda\left\langle \vec{e}_i\cdot\vec{\delta}\right\rangle & (51)\\
&= \frac{1}{2}\vec{\delta}^T M\vec{\delta} - \vec{k}\cdot\vec{\delta} + \lambda\vec{E}\cdot\vec{\delta}. & (52)
\end{aligned}$$

The quantity $\mathcal{L}_2$ is stationary with respect to $\vec{\delta}$ and λ when:

$$\begin{aligned}
0 &= \frac{\partial\mathcal{L}_1}{\partial\vec{\delta}} = M\vec{\delta} - \vec{k} + \lambda\vec{E} & (53)\\
0 &= \frac{\partial\mathcal{L}_1}{\partial\lambda} = \vec{E}\cdot\vec{\delta}. & (54)
\end{aligned}$$

The two conditions above may be written as one matrix constraint as follows:

$$\begin{pmatrix} M & \vec{E} \\ \vec{E}^T & 0 \end{pmatrix}\begin{pmatrix} \vec{\delta} \\ \lambda \end{pmatrix} = \begin{pmatrix} \vec{k} \\ 0 \end{pmatrix} \tag{55}$$

or as

$$M_1'\vec{\delta}' = \vec{k}' \tag{56}$$

if we define the symmetric $(B+1)\times(B+1)$-matrix M_v' and the $(B+1)$-vectors $\vec{k}'$ and $\vec{\delta}'$ as follows:

$$M_v' = \begin{pmatrix} M & v\vec{E} \\ v\vec{E}^T & 0 \end{pmatrix} \tag{57}$$

$$\vec{\delta}' = \begin{pmatrix} \vec{\delta} \\ \lambda \end{pmatrix} \tag{58}$$

$$\vec{k}' = \begin{pmatrix} \vec{k} \\ 0 \end{pmatrix}. \tag{59}$$

Note that the matrix equation:

$$M_0'\vec{\delta}' = \vec{k}' \tag{60}$$

just encodes

$$M\vec{\delta} = \vec{k}, \qquad \text{and} \tag{61}$$

$$0 = 0 \tag{62}$$

which are just the constraints one needs to solve for the case lacking the $\left\langle m_{ll\bullet}^2 \right\rangle = \left\langle m_{ll}^2 \right\rangle$ constraint. The one equation

$$M_v'\vec{\delta}' = \vec{k}' \tag{63}$$

therefore encompasses both cases:

- when $v = 0$ its solutions are those which extremalize $\mathcal{L}_0$, and
- when $v = 1$ its solutions are those which extremalize $\mathcal{L}_1$.

Returning to the quantity Δ already defined as

$$\Delta \equiv \mathrm{Var}\left[m_{lli}^2\right] - \mathrm{Var}\left[m_{ll\bullet}^2\right] = 2\vec{k}\cdot\vec{\delta} - \vec{\delta}^T M\vec{\delta} \tag{64}$$

we may ask ourselves what value, Δ_0, it takes for the unconstrained mean problem ($v = 0$ $\mathcal{L}_0$), and value, Δ_1, it takes for the constrained mean problem ($v = 1$ $\mathcal{L}_1$). From the results already established we can see that

$$\Delta_0 = 2\vec{k}\cdot\vec{\delta} - \vec{\delta}^T M\vec{\delta} \tag{65}$$

$$= 2\vec{k}\cdot\vec{\delta} - \vec{\delta}^T\vec{k} \qquad \text{(when (32) is satisfied)} \tag{66}$$

$$= \vec{k}\cdot\vec{\delta}, \tag{67}$$

and

$$\Delta_1 = 2\vec{k} \cdot \vec{\delta} - \vec{\delta}^T M \vec{\delta} \tag{68}$$

$$= 2\vec{k} \cdot \vec{\delta} - \vec{\delta}^T (\vec{k} - \lambda \vec{E}) \tag{69}$$

$$= \vec{k} \cdot \vec{\delta} + \lambda \vec{E} \cdot \vec{\delta} \qquad \text{(when (53) is satisfied)} \tag{70}$$

$$= \vec{k} \cdot \vec{\delta} + 0 \qquad \text{(when (54) is satisfied)} \tag{71}$$

$$= \vec{k} \cdot \vec{\delta}. \tag{72}$$

We see that in **both** cases, the improvement in the variance is equal to $\vec{k} \cdot \vec{\delta}$. [Aside: this does not mean that the improvement in the variance is the same in both cases, of course, since $\vec{\delta}$ is not, in general, the same in the two cases.] We therefore present a simpler version of the variance improvement as follows:

$$\Delta = \vec{k} \cdot \vec{\delta}. \tag{73}$$

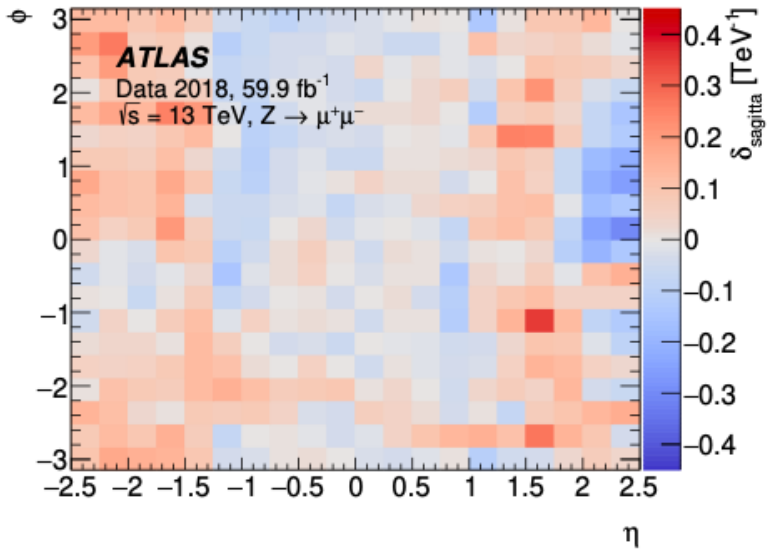

Figure 1: Sagitta biases calculated using the **baseline method**. This figure is figure 18 of [Aad+20].

4 Sagitta biases calculated for the two example goals which were set

Alas I have not yet written text in this section to explain the results shown in the figures of this document. However, the captions of the figures may be sufficient for this first iteration of the document.

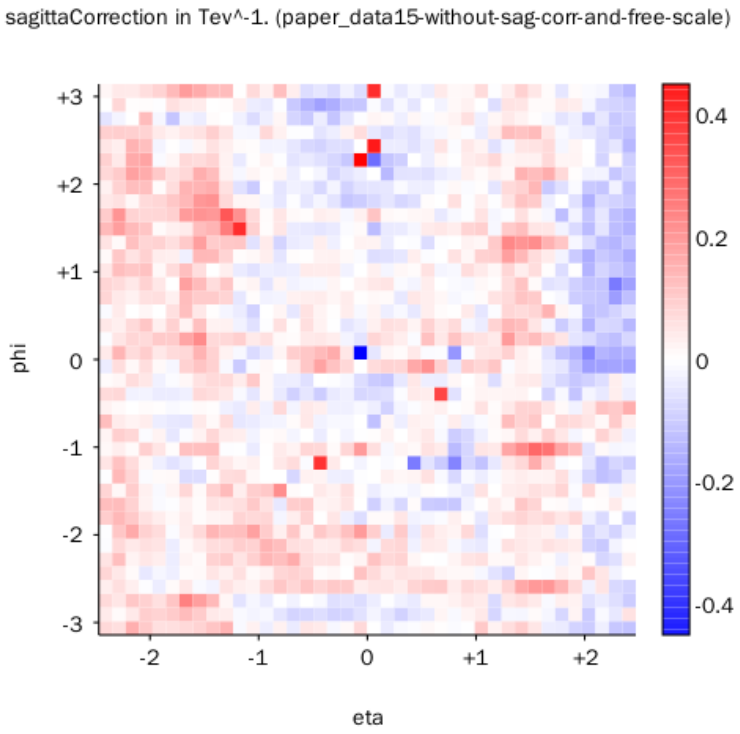

Figure 2: This is the equivalent of Figure 1 but **calculated with our own non-iterative method**. The biases shown here use the **goal of section 3.5.1**. Recall that this goal makes **no special requirement on the mean mass of the sagitta-corrected dimuon events**. Compare it with Figure 3 which is calculated for the other goal. This figure was generated from approximately 1,000,000 events in data15.

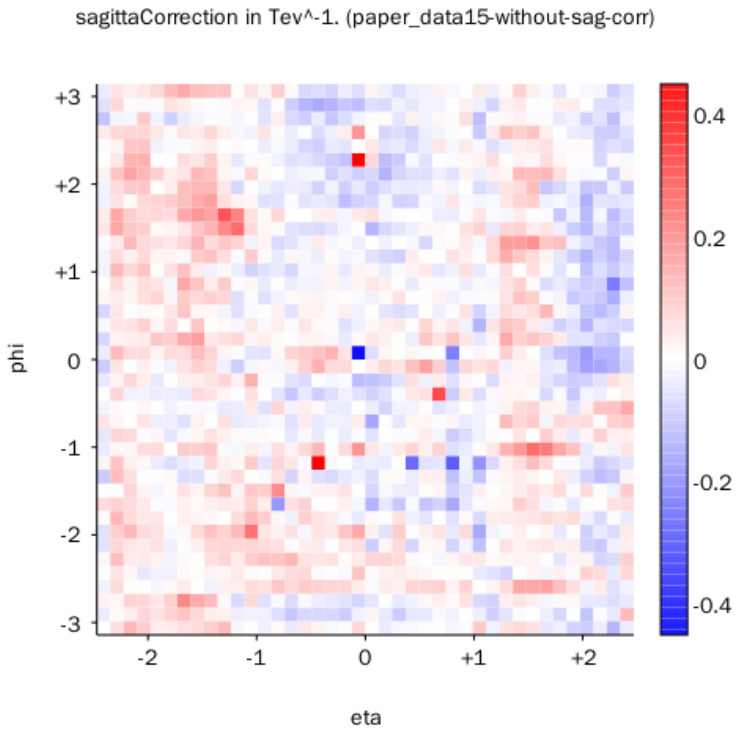

Figure 3: This is the equivalent of Figure 1 but **calculated with our own non-iterative method**. The biases shown here use the **goal of section 3.5.2**. Recall that this goal **requires that the mean mass of the sagitta-corrected dimuon events is not affected by the sagitta correction**. Compare it with Figure 2 which is calculated for the other goal. This figure was generated from approximately 1,000,000 events in data15.

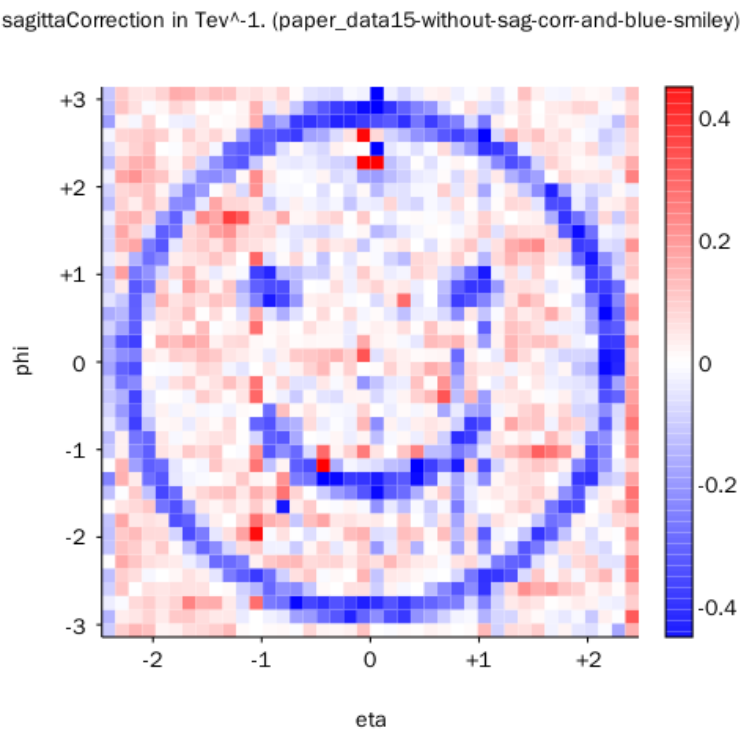

Figure 4: This plot is identical to that shown in Figure 2, except that an artificial sagitta bias (in the shape of a popular emoticon) has been injected into the data with strength -0.4/TeV, on top of any existing biases. This figure establishes that the method is indeed measuring sagitta biases, not something else. This figure was generated from approximately 1,000,000 events in data15.

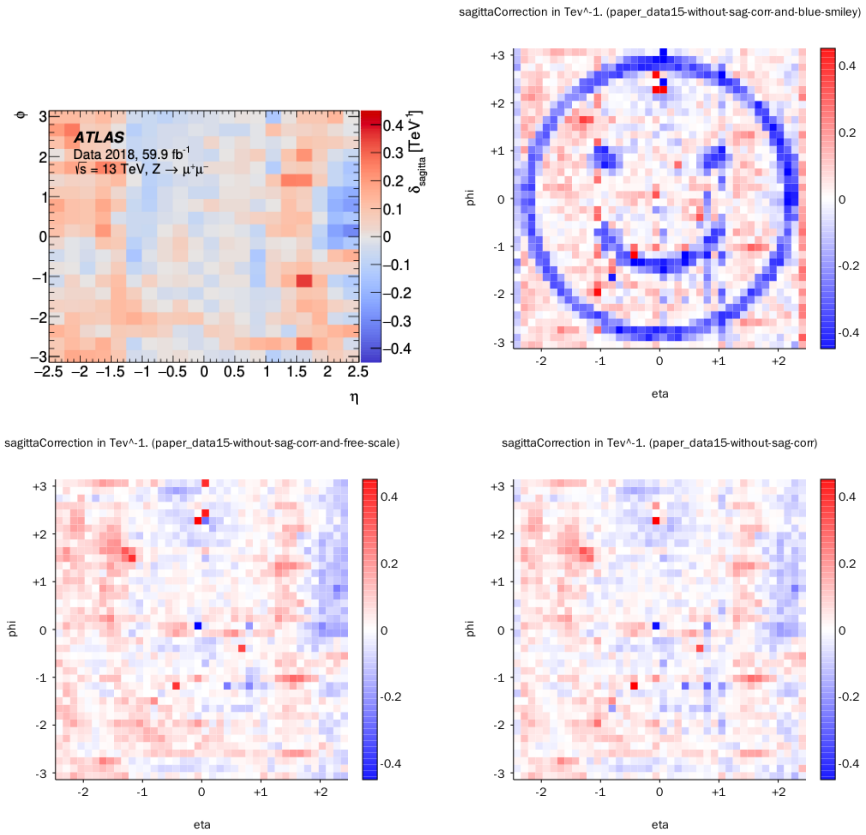

Figure 5: This figure shows the plots of Figures 1 to 4 side by side. The lower left plot uses the primary goal. The lower right plot uses the secondary goal.

5 Discussion and Conclusions

My only conclusion is that it is not (yet) obvious to me what the **baseline method** gains from its iterative nature. I am sure there is a good reason why the **baseline method** is set up to be iterative – however my attempt to look at the problem did not naturally lead me to an iterative answer. Instead it lead me to solving a linearised system.

Is it the case, perhaps, that the iterative approach of the **baseline method** is necessary to account for non-linear effects in some way I have not understood?

It is certainly the case that linear systems can be attempted by iterative methods (such as `https://en.wikipedia.org/wiki/Gauss\OT1\textendashSeidel_method` or `https://en.wikipedia.org/wiki/Jacobi_method` which also come in over- or under-relaxed forms. It has occurred to me that it is possible that the iterative part of the **baseline method** may be doing something like Jacobi or Gauss-Siedel on my linear system.

It is also clear that for sufficiently complex constraints, the solutions of the Lagrange system would no longer be linear. In such cases non-linear solvers (which are inevitably iterative) would have to be employed. It is not obvious to me, though, that you ever need such constraints since sagitta bias corrections ought (in principle) to be 'small' if the ID is sufficiently well aligned.

6 Appendix

6.1 Notes

My original derivation is archived here: `https://gitlab.cern.ch/emus/OSDFChargeFlavourAsymmCode/-/blob/master/sagitta/lester/DOCS/sketch_of_working.pdf`

6.2 Comparison to ATLAS baseline method

We may re-write (1) (but keeping it in the notation of [Aad+20]) as:

$$p = p'(1 + qp_T\delta). \tag{74}$$

If we compare (74) above with our own (15) we see that the sagitta bias sign conventions and definitions in [Aad+20] are agree with those here if

- looks only at first order in δ,
- glosses over differences between p and p_T (recall that this requirement may easily be removed for the reasons explained in Section 3.1),

and one either:

(a) treats our "$\bullet$" (sagitta corrected) and the **baseline method's** "$'$" (reconstructed but not true) quantities as meaning the same thing, or:

(b) reverses our sign convention for δ and then associates our "$\bullet$" (sagitta corrected) with the **baseline method's** unprimed (true not reconstructed) quantities.

Which of those is the better convention to adopt will depend on what the **baseline method** actually means by its primed and unprimed quantities, which at present is not clear. It remains possible that after clarity over the **baseline method** is established we may alter our own sign conventions to match if required.

Nonetheless, sign conventions aside, it is clear that although the **baseline method** and method presented here are different, they should agree to the level of the leading order approximation which both make regarding the bias definitions, modulo the caveat of Section 3.1.

References

[Aad+20] Georges Aad et al. "Alignment of the ATLAS Inner Detector in Run-2". In: (July 2020). arXiv: `2007.07624 [hep-ex]`.