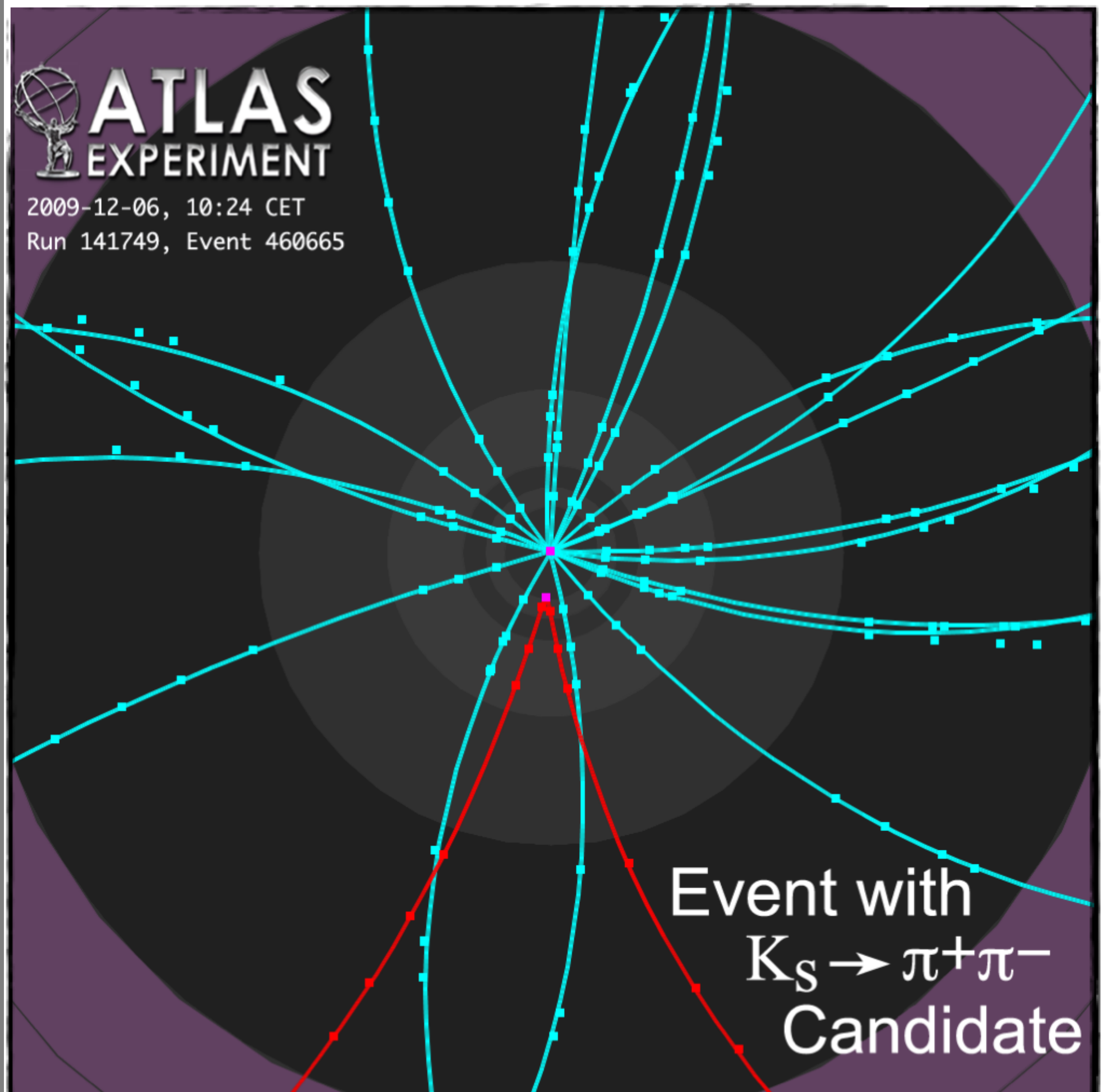
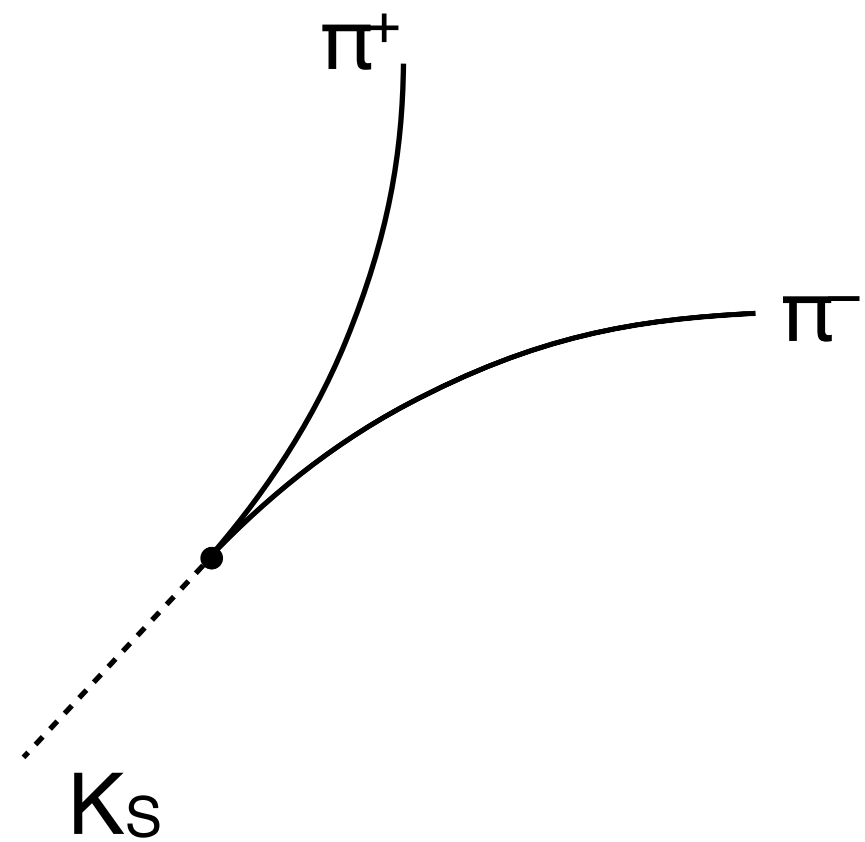
Checking the Momentum Scale

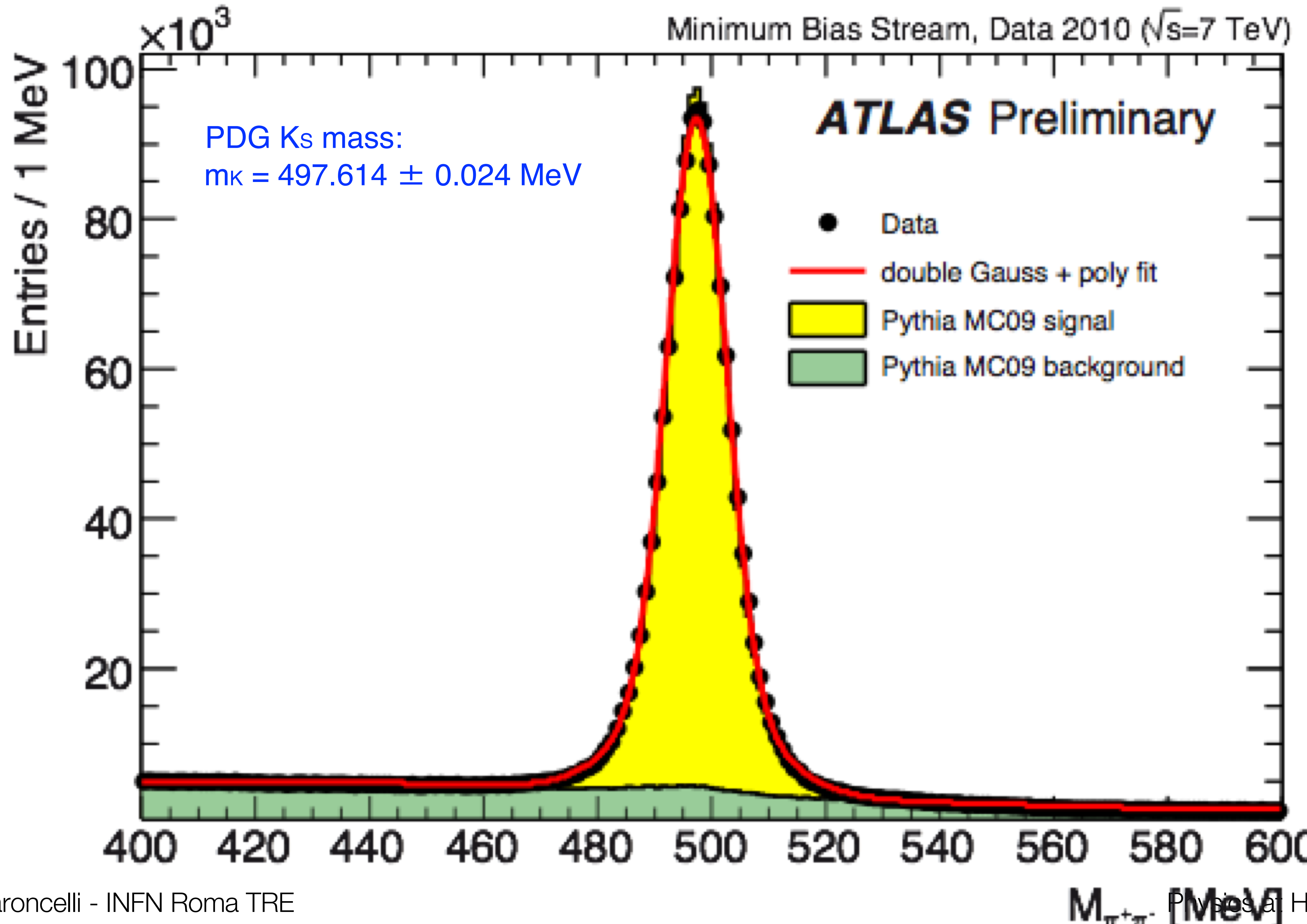


Momentum Reconstruction with Inner Tracker



$$m_K^2 = (p_{\pi^+} + p_{\pi^-})^2$$

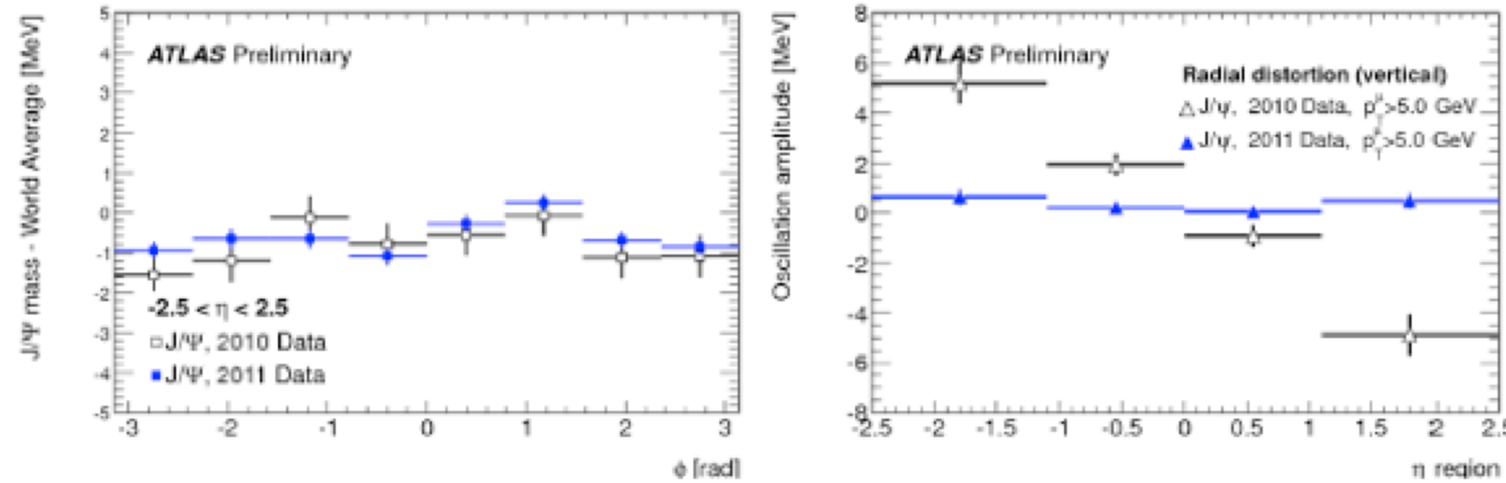
Checking the Momentum Scale



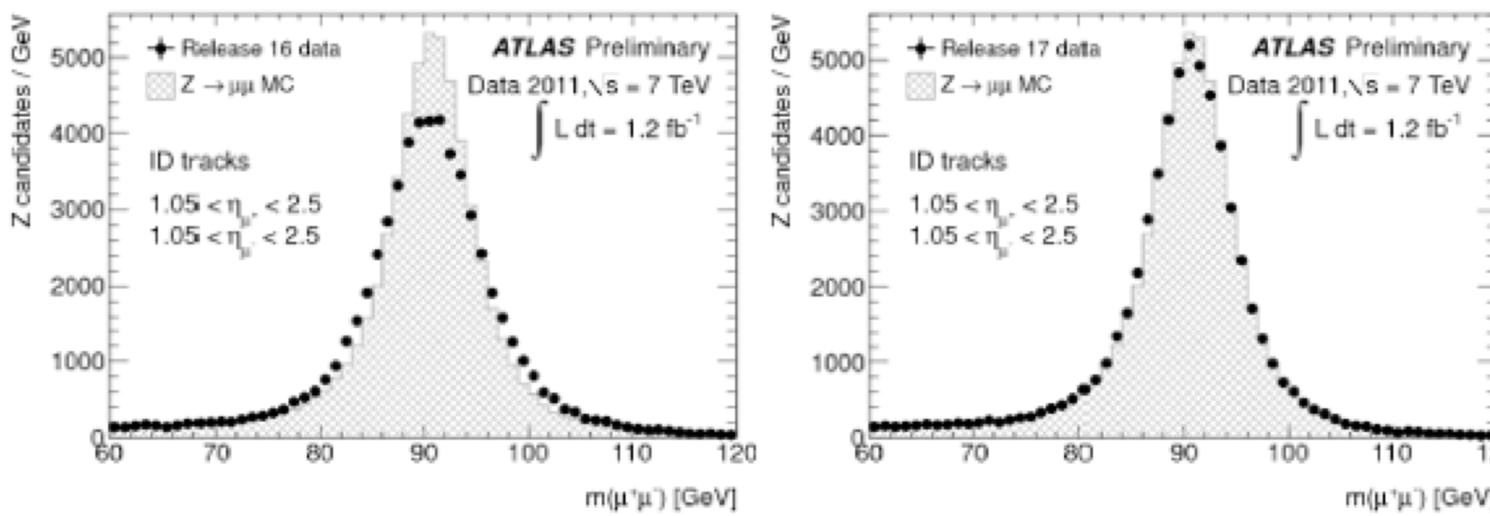
Mass resolution

Systematics on tracking and measured momentum due to misalignment have been studied using

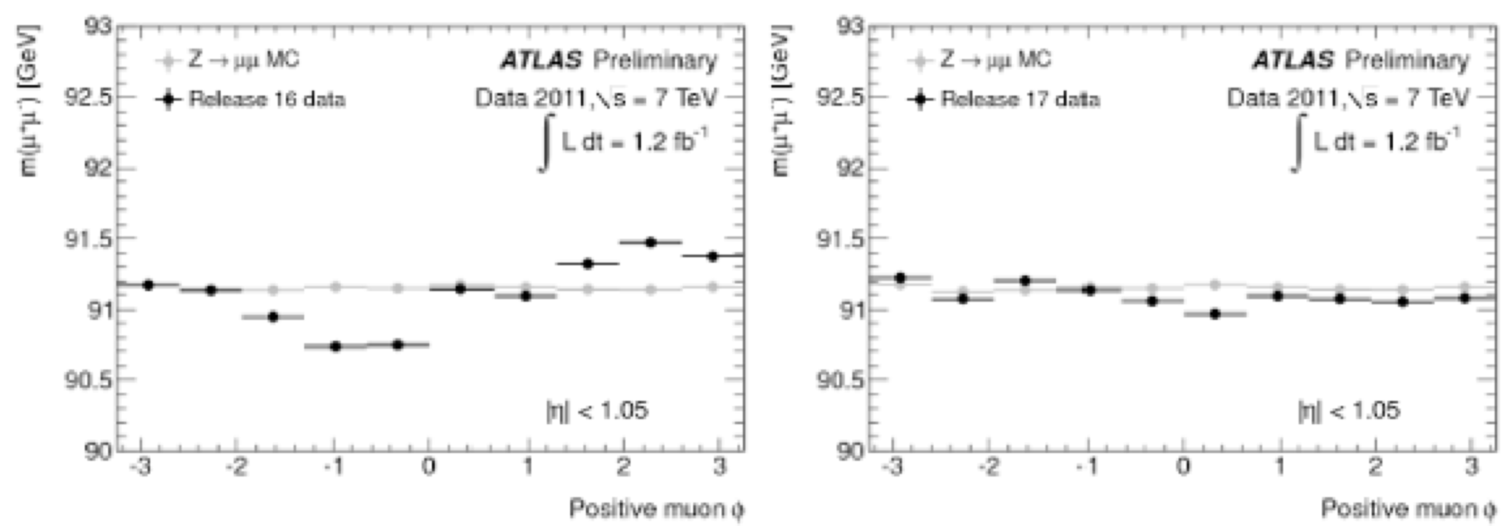
J/Psi->μμ and
Z->μμ decays.



The difference between central value of the fitted mass and PDG World Average mass for J/ψ (left) and comparison of the fitted radial-deformation amplitudes in 2010 and 2011 data for the J/ψ (right) [5].



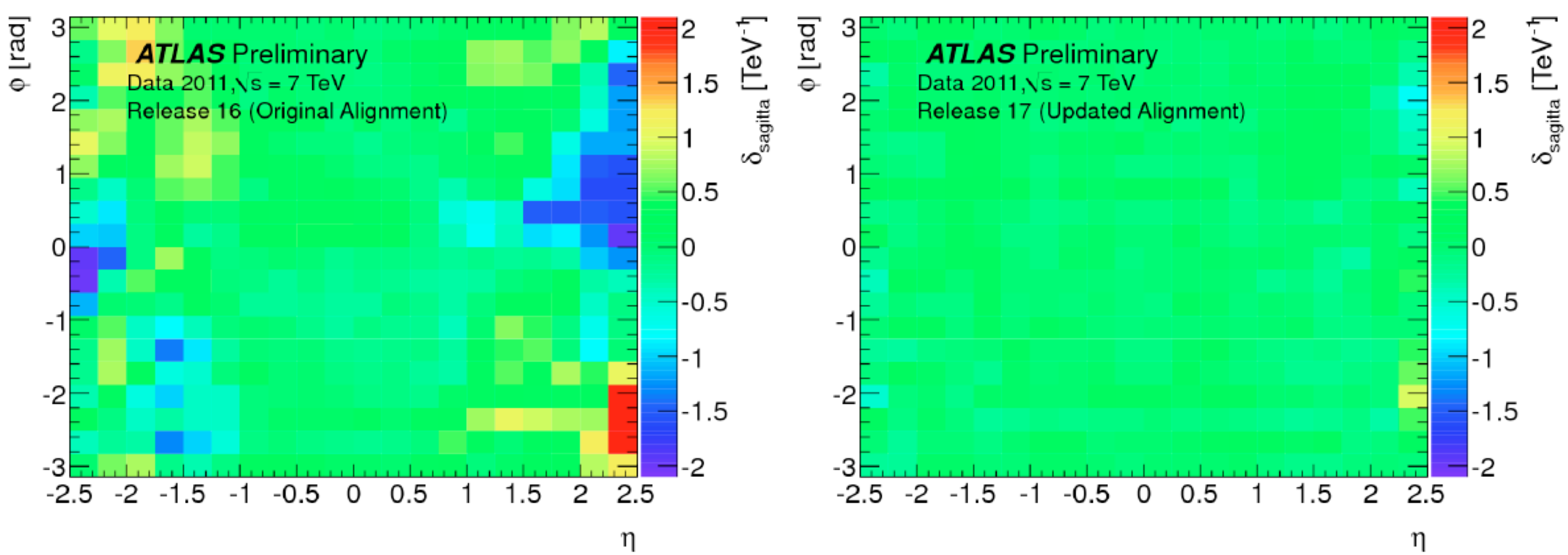
Reconstructed Z mass distributions before (left) and after alignment corrections and perfectly aligned MC (grey) for the end-cap A [5].



Fitted mean Z mass as a function of ϕ of the positive muon, for collision data before (left) and after charge-antisymmetric alignment corrections and compared to MC with perfect Inner Detector alignment (grey) [5].

Momentum bias (systematic differences in p_T)

Momentum bias is extracted from data for electrons and positrons from Z and W decays using E/p method with parametrization of momentum $q/p = q/p[1+qp_T \, \text{delta(sagitta)}]$. Except for a few isolated spots in very forward region, local biases stay within $|\text{delta}| < 0.5\ \text{TeV}^{-1}$, corresponding to < 2% bias at 40 GeV.



Momentum bias parameter for the charge-antisymmetric parametrization with old (left) and updated alignment (right) [5].

Vertex reconstruction & robust technique

2 Reconstruction of primary vertices

Finding (first) Fitting (then)

The reconstruction of primary vertices is organized in two steps: a) the primary vertex finding algorithm, dedicated to associate reconstructed tracks to the vertex candidates, and b) the vertex fitting algorithm, dedicated to reconstruct the vertex position and its corresponding error matrix. It also refits the associated tracks constraining them to originate from the reconstructed interaction point. The detailed implementation of the *finding* and *fitting* algorithms is described below.

In this analysis, reconstructed tracks fulfilling the following quality requirements are used for the primary vertex reconstruction:

- $p_T > 150$ MeV,
- $|d_0| < 4$ mm,
- $\sigma(d_0) < 5$ mm,
- $\sigma(z_0) < 10$ mm,
- at least 4 hits in the SCT detector,
- at least 6 hits in the pixel and SCT detectors.

Here d_0 and z_0 denote the transverse and longitudinal impact parameters of tracks with respect to the centre of the luminous region, and $\sigma(d_0)$ and $\sigma(z_0)$ denote the corresponding uncertainties as estimated in the track fit. The symbol p_T denotes the reconstructed track transverse momentum. The selection criteria based on the impact parameters are designed to remove a good fraction of the tracks originating from secondary interactions. As estimated from simulations, based on results obtained with the PYTHIA Monte Carlo program [6] and the full simulation of the ATLAS detector, in non-diffractive pp collisions

Robust vertex reconstruction

at 7 TeV, the above requirements are fulfilled by $(83.5 \pm 0.1)\%$ of reconstructed tracks corresponding to primary particles.

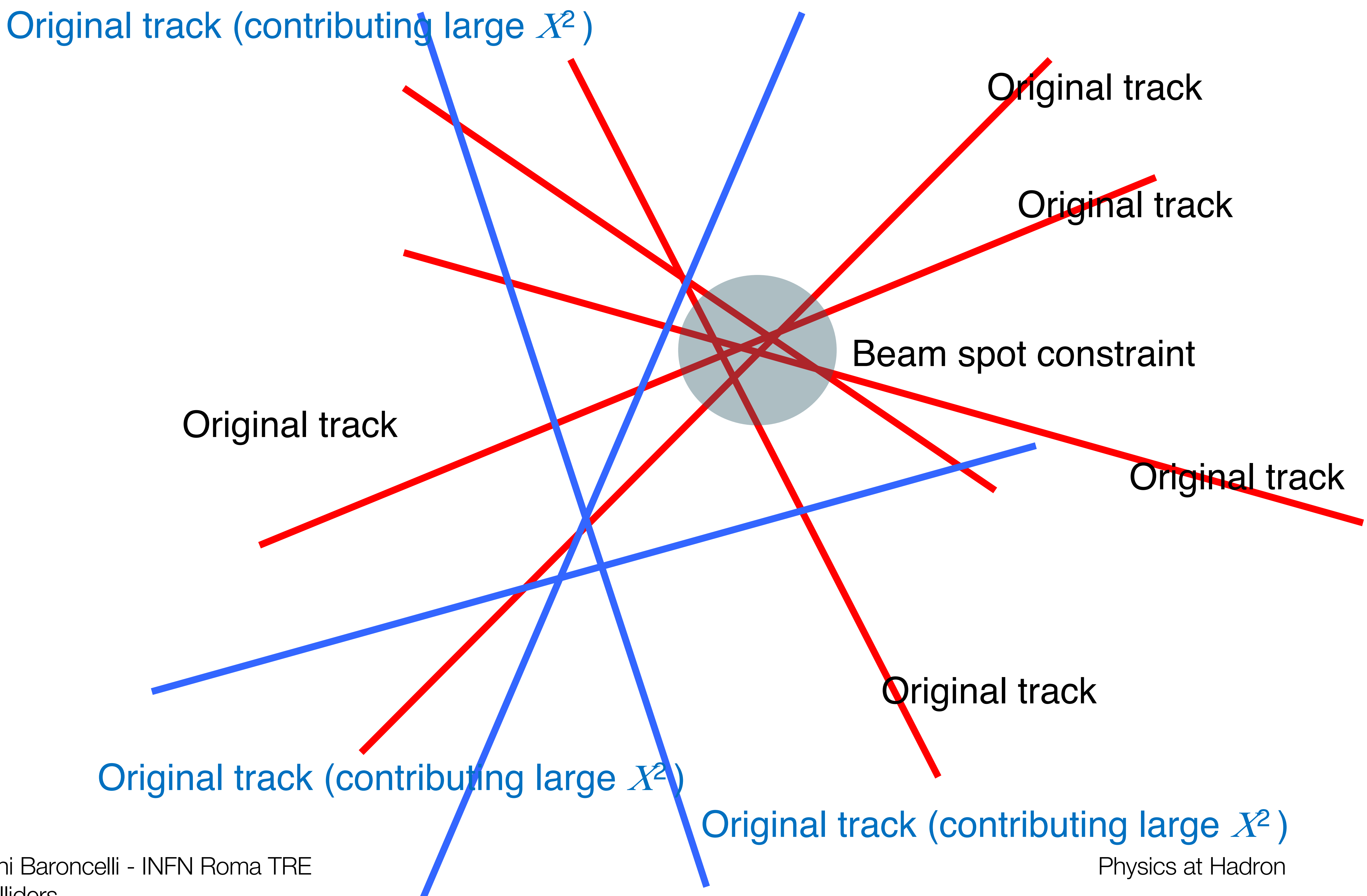
The luminous region in ATLAS is determined during a physics run, typically every $\approx$ 10 minutes, by applying an unbinned maximum likelihood fit to the distribution of primary vertices recorded in this period of time, where the same primary vertex reconstruction algorithm is used as described in the following, but without applying the beam-spot constraint. A detailed description of how the beam-spot is determined and on the uncertainties connected with its determination can be found in Ref. [7].

The *Iterative Vertex Finding* approach used for this study works as follows:

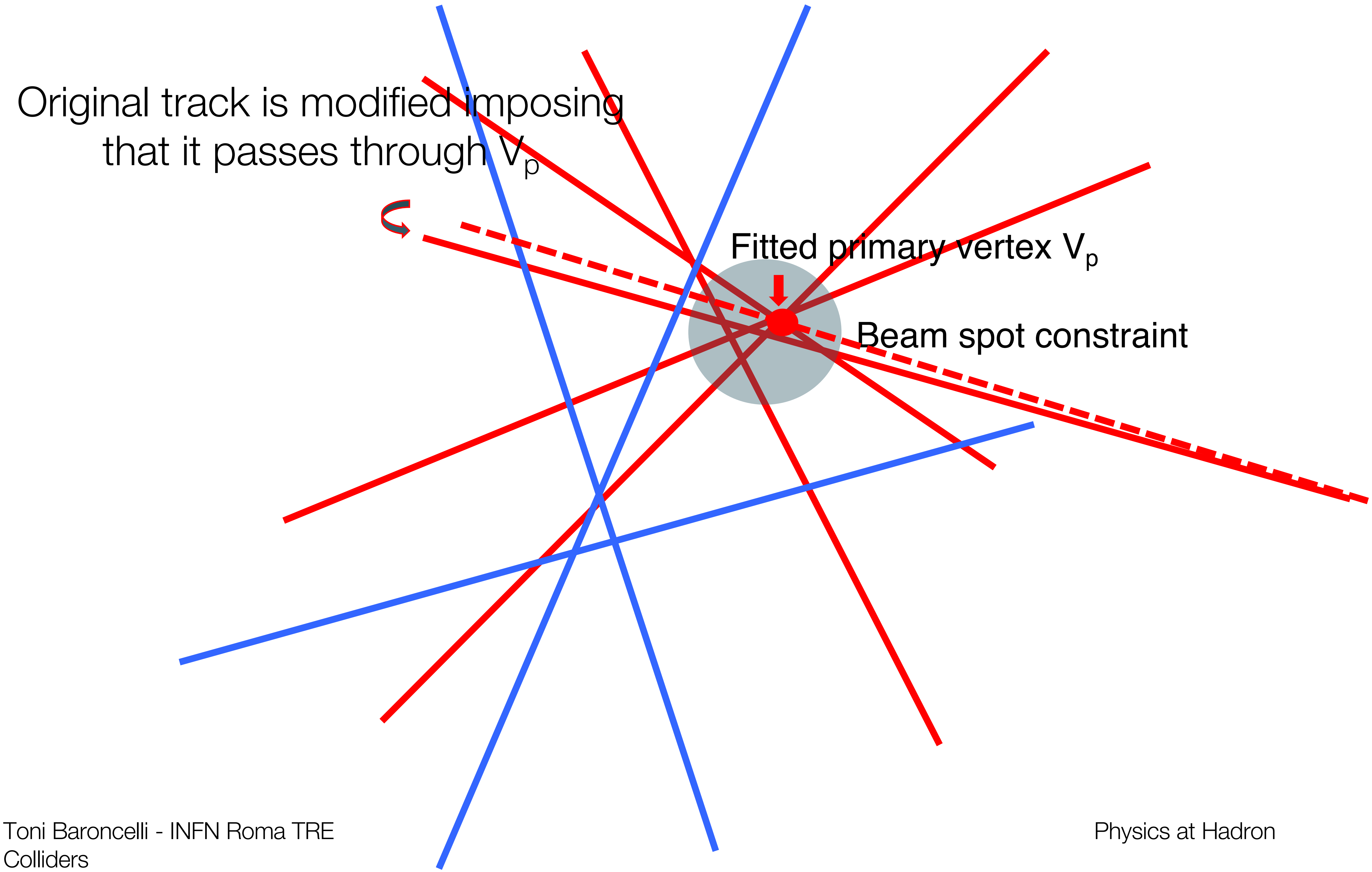
- Reconstructed tracks compatible with originating from the interaction region are pre-selected according to the criteria listed above.
- A vertex seed is found by looking for the global maximum in the distribution of z coordinates of the tracks, computed at the point of closest approach to the beam spot center.
- The vertex position is determined using the *adaptive vertex fitting* algorithm [8], which takes as input the seed position and the tracks around it. The adaptive vertex fitter is a robust χ^2 based fitting algorithm which deals with outlying track measurements by down-weighting their contribution to the overall vertex χ^2. The down-weighting is performed progressively, while the fit iterations proceed according to a fixed number of steps (deterministic annealing scheme [8]).
- Tracks incompatible with the vertex by more than approximately 7 σ are used to seed a new vertex. The compatibility of the track to the vertex is expressed in terms of a χ^2 with 2 degrees of freedom. The present cut is $\chi^2 > 49$. This procedure is repeated until no unassociated tracks are left in the event or no additional vertex can be found.

The very loose cut of $\chi^2 > 49$ is intended to reduce the number of single vertices which split into two due to the presence of outlying track measurements.

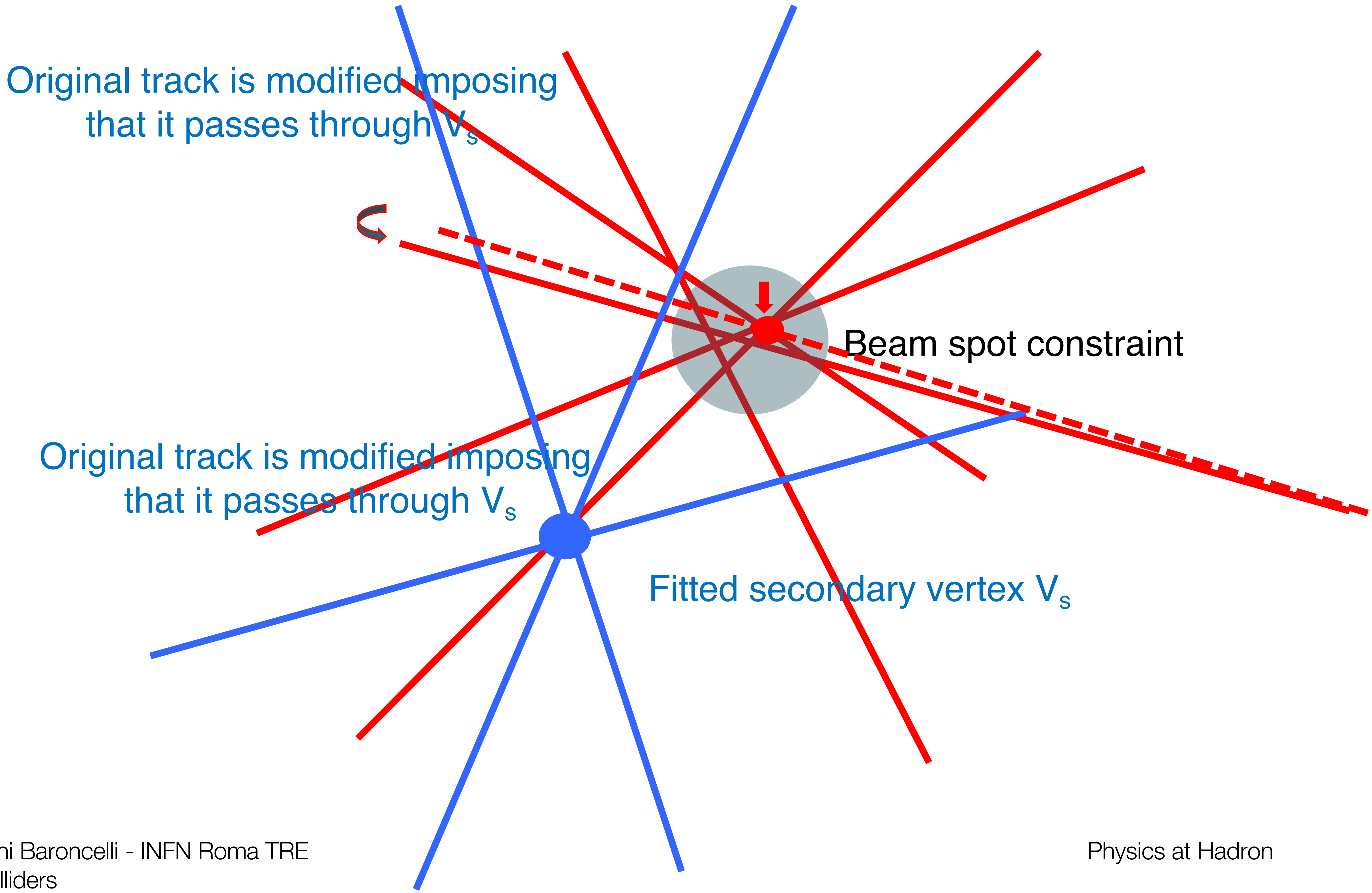
Vertex fitting - 0



Vertex fitting - 1

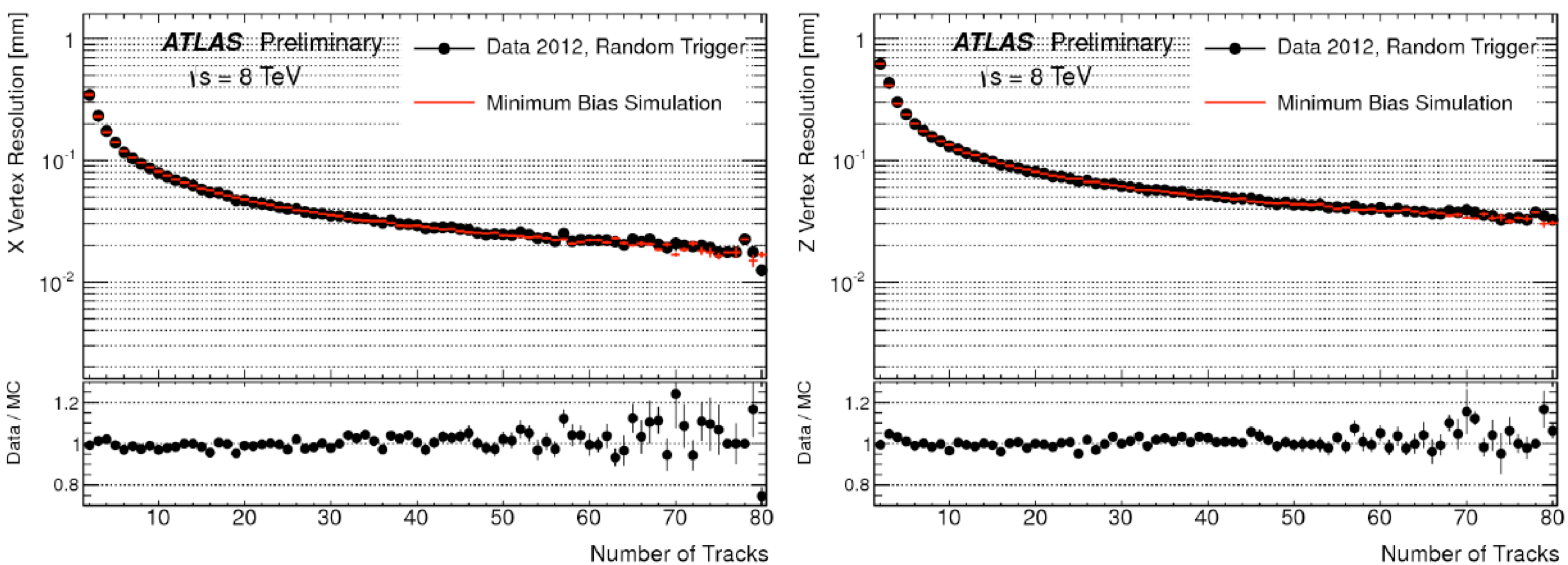


Vertex fitting - 2



ATLAS vertexing

takes z-position of track at beam-line as seed
iterative Chi^2 fit of nearby tracks
new seed from tracks displaced by more than 7
beam-spot used as a constraint



Vertex position resolution in data and MC for the transverse (left) and longitudinal coordinate as a function of number of tracks in vertex [2].

High pile-up events: μ and N_{PV} (or N_{Vx})

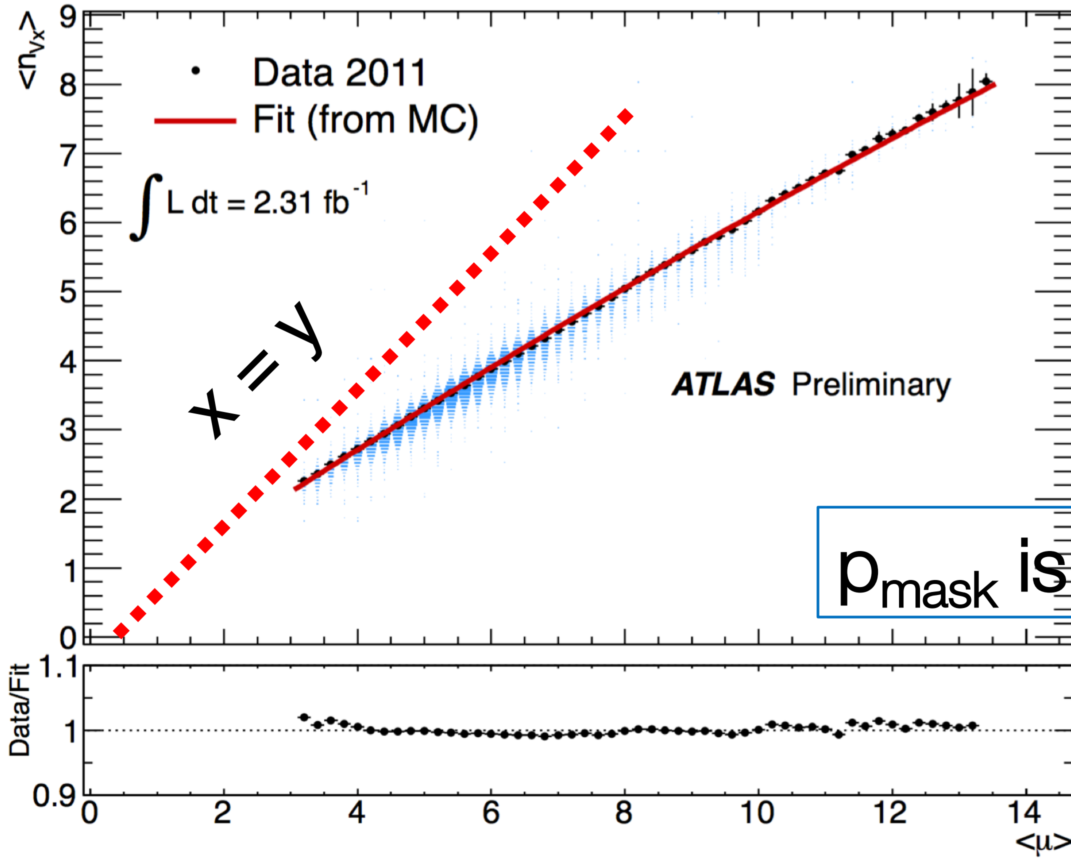
The number of proton-proton interactions per bunch crossing follows a Poisson distribution with mean value μ. During a fill, μ decreases with decreasing beam intensity and increasing emittance, such that the quoted peak value, or μ^{peak}, is the highest value in a single bunch crossing at the start of the stable beam period of the fill. The number of interactions per bunch crossing also varies between bunches. The number of interactions averaged over all bunch crossings and averaged over the data analysed will be referred to as $\langle\mu\rangle$.

In data, μ is calculated using the following formula:

$$\mu = \frac{L \times \sigma_{\text{inel}}}{n_{\text{bunch}} f_{\text{r}}} \qquad \mathcal{L} = \frac{N_1 N_2 f}{A} \qquad (1)$$

where L is the luminosity, σ_{inel} is the total inelastic cross-section, n_{bunch} the number of colliding bunches and f_{r} the LHC revolution frequency. The uncertainty on μ depends on the uncertainties on the luminosity and the total inelastic cross-section. The luminosity measurement is performed with dedicated detectors and calibrated using special LHC fills. The uncertainty on the integrated luminosity is $\sim 3.9\%$ [5] for the 2011 physics data. The high-intensity runs studied have an additional 1% uncertainty to account for the extrapolation of direct luminosity measurements from lower intensity runs. The total inelastic cross-section used, $\sigma_{\text{inel}} = 71.5$ mb, is taken from Pythia [6]. The value is $\sim$3% lower than the measurement from TOTEM of 73.5 ± 1.9 mb [7]. The total cross-section has also been measured by ATLAS to be $69.1 \pm 2.4(\text{exp.}) \pm 6.9(\text{extr.})$ mb [8, 9] by extrapolating a measurement of the cross-section for events in the acceptance of scintillators in the forward region. The difference between the ATLAS and TOTEM measurements and the nominal value from Pythia is taken as a systematic uncertainty on μ of 3%.

$\langle n_{Vx} \rangle$ vs μ



In an ideal detector with perfect resolution and acceptance you would expect to have

$\langle n_{Vx} \rangle$ ~ m

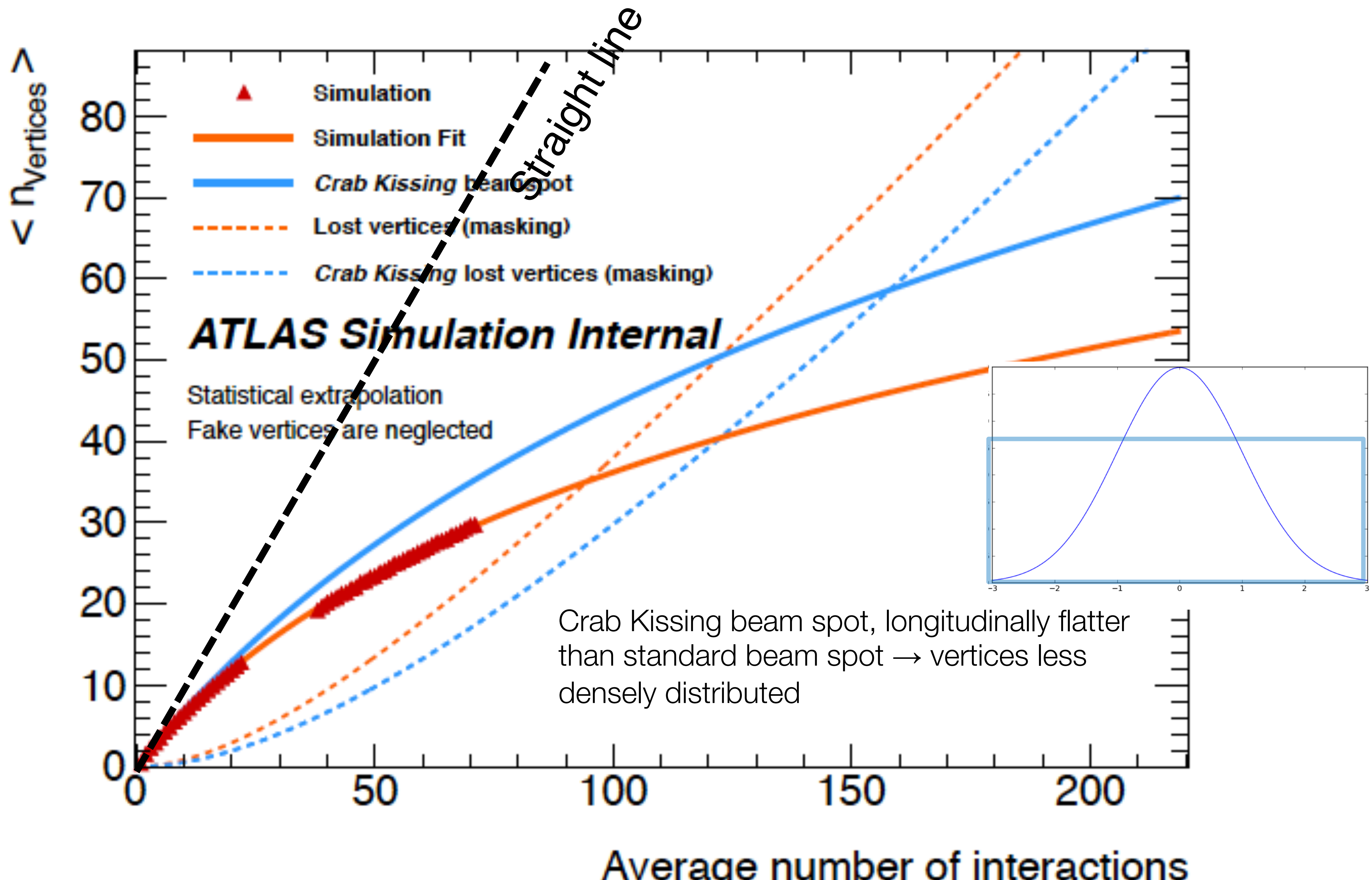
p_{mask} is the inability of resolving nearby vertices

The average number of reconstructed vertices is shown as a function of the average number of interactions μ. The red triangles show the simulation prediction for minimum-bias events without any trigger bias using the 2012 setup for beam and detector conditions. The simulation covers the μ range up to 22 and a higher interval which was observed in some special high-μ data taking runs.

The simulation has been fitted (orange line) with a function taking into account the vertex reconstruction efficiency ε and the inability of resolving nearby interactions in distinct vertices (vertex masking): $\boxed{\langle n_{vertices} \rangle = \varepsilon\mu - F(\varepsilon\mu, p_{mask})}$. $F(\varepsilon\mu, p_{mask})$ is a function that estimates the correction to the number of reconstructed vertices for masking effects. The probability of not resolving two interactions in distinct vertices, p_{mask}, depends on the detector and vertex reconstruction algorithm performance (assumed to be independent from pile-up) and on the density of the interactions (which depends on the beamspot longitudinal profile).The latter dependence is exploited to test the effect of a different beamspot longitudinal profile (approximately flat, called *Crab Kissing*); this is shown in the Figure as a blue line.

Fakes have been found to be negligible in the range covered by simulation and are assumed to be equally negligible in the extrapolated regions. The correction to the number of reconstructed vertices for masking effects $F(\varepsilon\mu, p_{mask})$ is show by the dotted lines for the two configurations.

High pile-up events

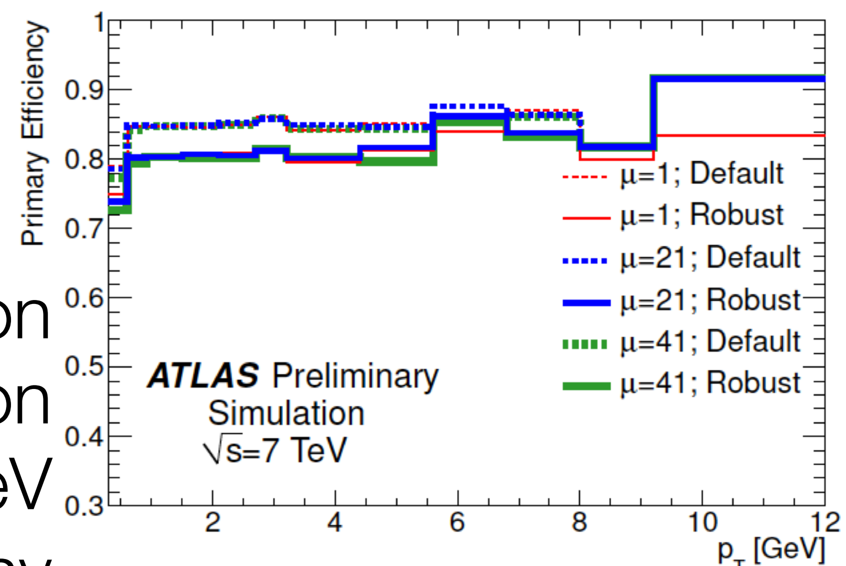


Crab Kissing beam spot, longitudinally flatter than standard beam spot → vertices less densely distributed

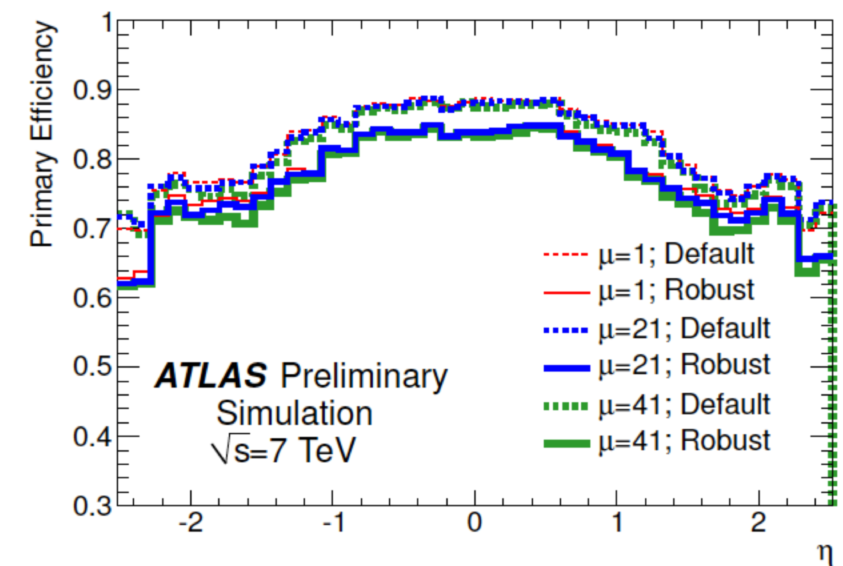
High pile-up events & robust reconstruction

non-primary = not pointing to a primary vertex

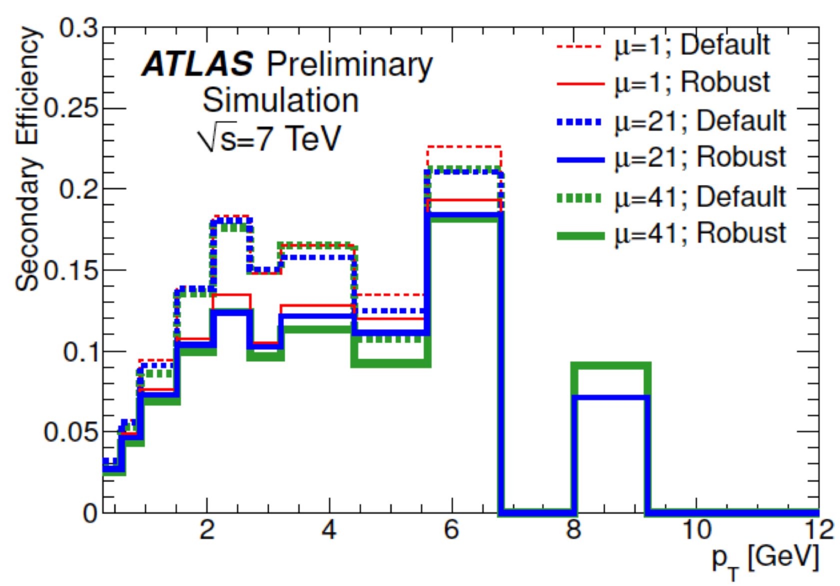
A special track selection allows the reconstruction of tracks with p_T>400 MeV in events with many superimposed vertices



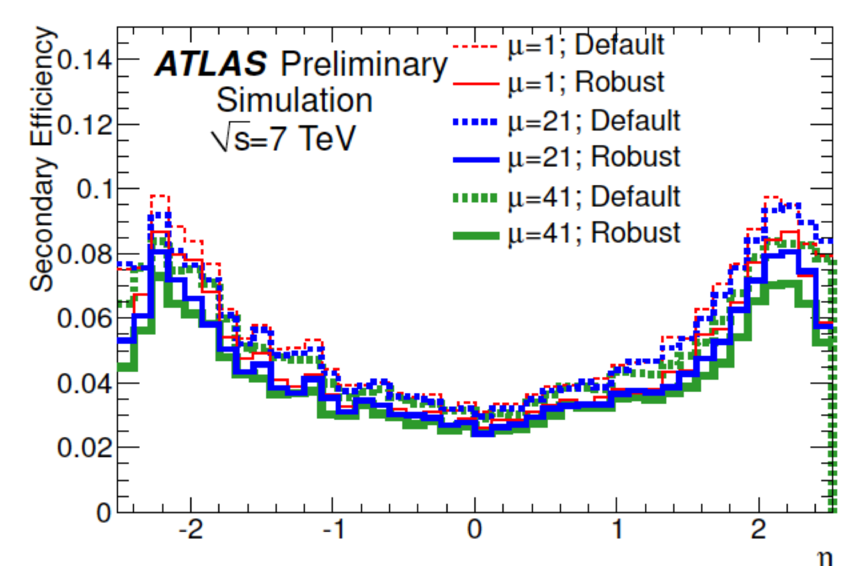
(a) Primary track reconstruction efficiency vs p_T



(b) Primary track reconstruction efficiency vs η

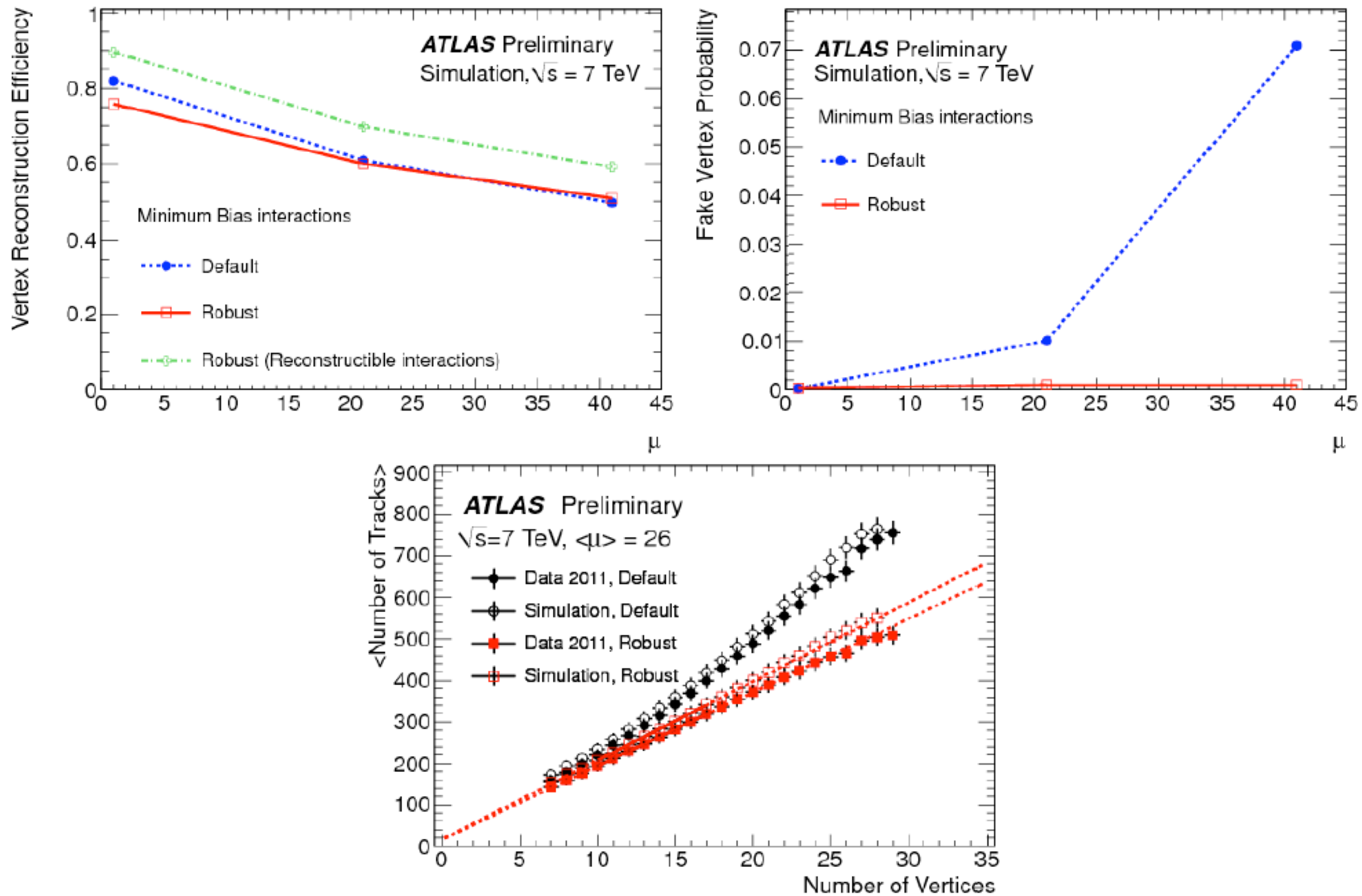


(c) Secondary efficiency vs p_T



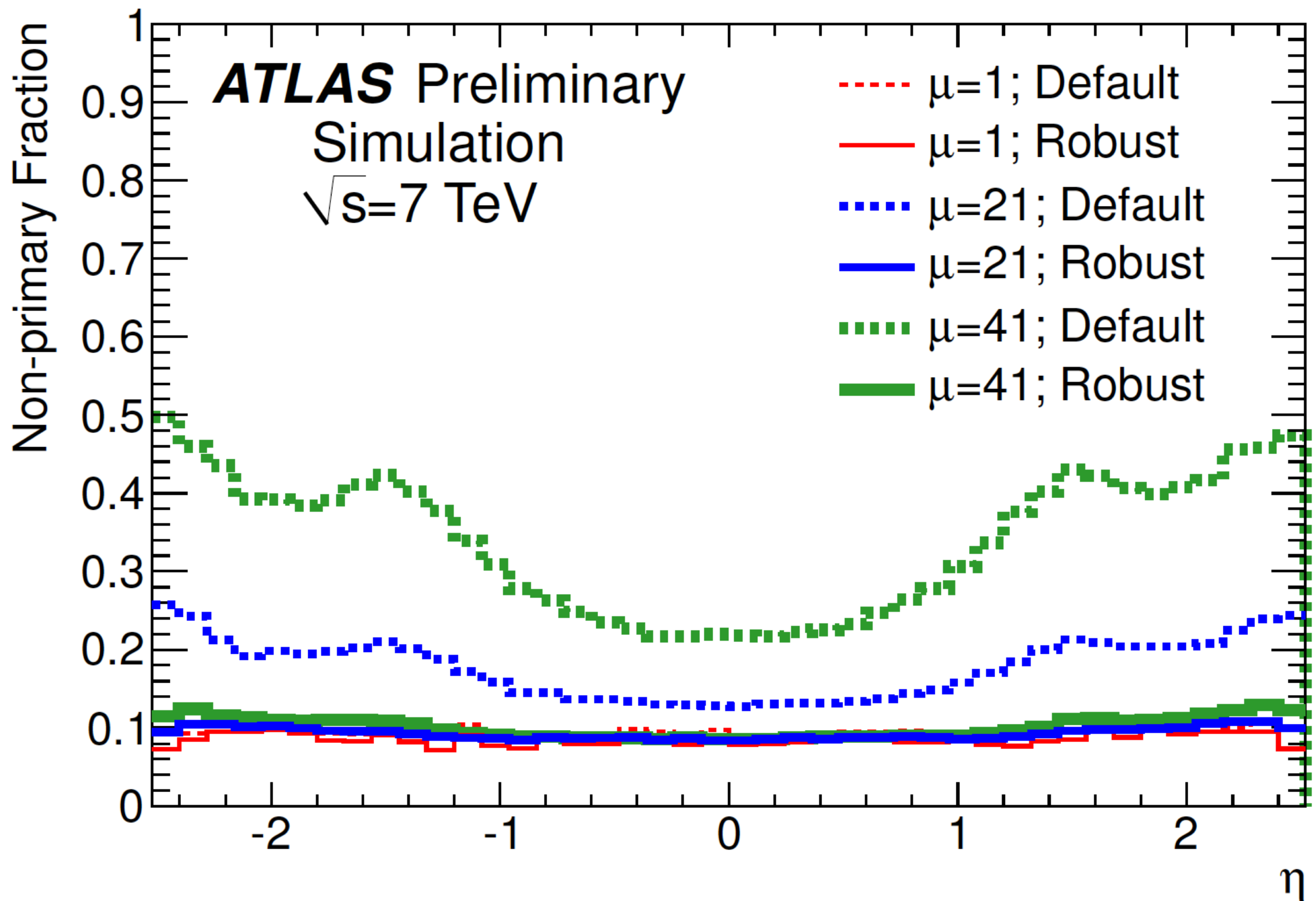
(d) Secondary efficiency vs η

Vertices in high pile-up events



Vertex reconstruction efficiency (top left) and fake probability (top right) as a function of the average number of interactions in minimum bias MC and the average number of tracks per event as a function of the number of vertices for data and simulation (bottom) [4].

Fake vertices vs η



(a) Non-primary fraction vs η

Vertex resolution

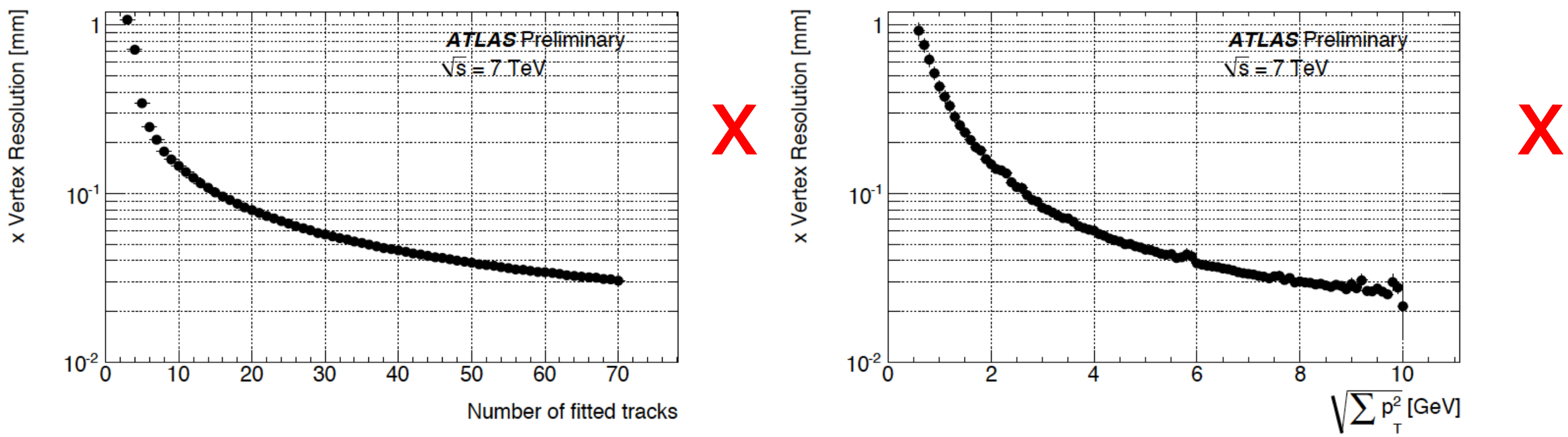


Figure 8: Estimated vertex resolution $\sigma_{x\mathrm{PV,true}}$ in 7 TeV data as a function of the number of tracks N_{trk} (left) or as a function of the value of $\sqrt{\sum_{\mathrm{trk}} p_{\mathrm{T}}^2}$ (right).

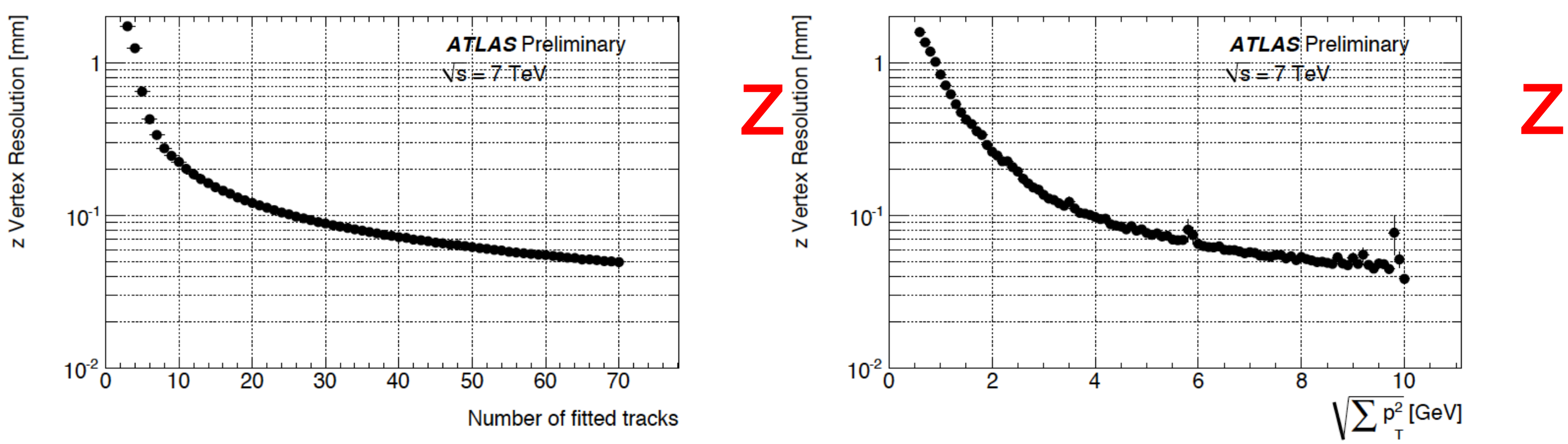


Figure 9: Estimated vertex resolution $\sigma_{z\mathrm{PV,true}}$ in 7 TeV data as a function of the number of tracks N_{trk} (left) or as a function of the value of $\sqrt{\sum_{\mathrm{trk}} p_{\mathrm{T}}^2}$ (right).