Journal of Physics: Conference
Series

A new inclusive secondary vertex algorithm for b-jet tagging in ATLAS

To cite this article: G Piacquadio and C Weiser 2008 *J. Phys.: Conf. Ser.* **119** 032032

View the article online for updates and enhancements.

A new inclusive secondary vertex algorithm for b-jet tagging in ATLAS

Giacinto Piacquadio, Christian Weiser

Albert-Ludwig University of Freiburg, Hermann-Herder-Str. 3, D-79104 Freiburg

E-mail: giacinto.piacquadio@physik.uni-freiburg.de

Abstract. A new inclusive secondary vertex reconstruction algorithm which exploits the topological structure of weak b- and c-hadron decays inside jets is presented. The primary goal is the application to b-jet tagging. The fragmentation of a b-quark results in a decay chain composed of a secondary vertex from the weakly decaying b-hadron and typically one or more tertiary vertices from c-hadron decays. The decay lengths and charged particle multiplicities involved in these decays, as well as the instrumental resolution, do not allow to separately reconstruct and resolve these vertices efficiently using conventional secondary vertexing algorithms based on the assumption of a common geometrical vertex. These difficulties are partially overcome in the algorithm presented in this paper, that is based on the hypothesis that the primary event vertex and the vertices of the weak b- and c-hadron decays lie on the same line, the flight direction of the b-hadron. The algorithm provides detailed information on the topology of the decay cascade. The algorithm based on this hypothesis is implemented mathematically as an extension of the Kalman Filter formalism for vertex reconstruction and technically as a set of flexible software modules integrated in the ATLAS software framework Athena, which make use of the existing Event Data Model for vertex reconstruction and b-tagging. The application of the algorithm to b-jet tagging is shown.

1. Motivation

Tagging of b-quark jets plays an important role in many analyses where signal events with one or more b-quark jets in the final state have to be separated from background events which mainly contain up-, down- and strange-quark jets (which are here defined as *light quark* jets), c-quark jets or gluon jets, e.g.:

- $t\bar{t}$ events: Because of $BR(t \rightarrow Wb) \approx 1$, one b-quark is expected per top quark decay.
- Searches for the Higgs boson: For a Standard Model Higgs boson, the dominant decay for low Higgs boson masses (up to $\approx$ 135 GeV/c^2) is into a $b\bar{b}$ pair. This decay mode is investigated in the $pp \rightarrow t\bar{t}H \rightarrow t\bar{t}b\bar{b}$ channel, where additional b-quarks from the decays of the top quarks are present. B-tagging plays a crucial role in the analysis of such channels.

As a general rule, a displaced vertex reconstructed inside a jet provides a strong signature for a b-quark jet. Further properties of such a displaced vertex, like a large track multiplicity or invariant mass of the charged particle tracks associated to it, can then be used to suppress vertices not originating from b-hadron decays. These can be either real vertices of e.g. c-hadron, K^0_S, and Λ^0 decays, photon conversions or hadronic interactions in the detector material, or fake vertices from random combinations of tracks from the primary interaction vertex.

Two b-tagging methods based on the lifetime of b-hadrons are implemented in ATLAS so far. They rely either on the displacement of charged particle tracks relative to the primary event vertex or the explicit reconstruction of the secondary b-hadron decay vertex. The standard algorithm uses a combination of these two.

2. Vertex reconstruction inside b-jets

A b-jet originates from a b-quark, which produces a b-hadron in the fragmentation. The b-hadron then decays due to electroweak interactions, which cause the transition of the b-quark preferably into a c-quark ($|V_{cb}|^2 \gg |V_{ub}|^2$), which then subsequently also undergoes a weak decay. As a result, the typical topology of the particles in a b-jet seen in the detector is a decay chain with two vertices, one stemming from the b-hadron decay and at least one from c-hadron decays. The eventual intermediate presence of excited b- or c-hadron states does not change this picture because their strong or electromagnetic decays do not cause measurable lifetimes.

The default secondary vertex based b-tagging algorithm in ATLAS relies on a very robust classical vertexing algorithm [1], in which displaced tracks are selected and an inclusive single vertex is obtained using a Kalman based χ^2 fit (Fig. 1). However, the underlying hypothesis of having a single geometrical vertex is not correct: when the distance between the b- and c-hadron decay vertices is significant compared to the vertex resolution in flight direction, the tracks from one of the two vertices can be lost in the fit. Some of these tracks can be kept by using a looser cut on the $Prob(\chi^2)$ of the vertex when rejecting the outliers, but this also results in a reduced rejection power against random combination of tracks in light quark jets.

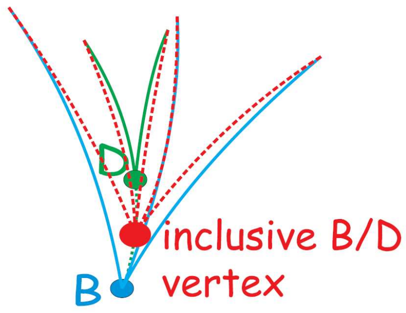

Figure 1. The ATLAS default secondary vertex finder fits all displaced tracks to an inclusive vertex.

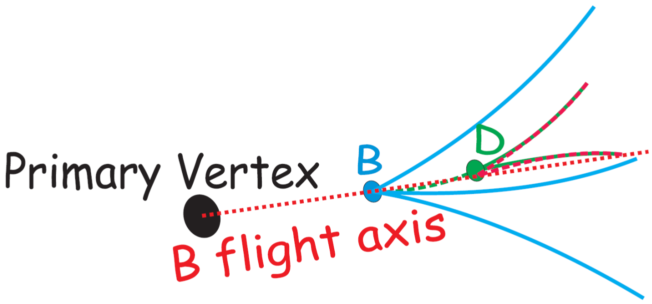

Figure 2. JetFitter performs a multi-vertex fit using the b-hadron flight direction constraint.

Trying to resolve the b- and c-hadron vertices separately is very difficult for the following reasons:

- The probability to have at least two reconstructed charged particle tracks both from the b- and c-hadron decays is by far not reaching 100%. This is both because of the genuine charged particle multiplicities as well as the limited track reconstruction efficiency, mainly because of interactions in the detector.
- The resolutions of the relevant track parameters, especially at low transverse momenta, are not sufficient to separate the two vertices efficiently.

The algorithm presented here, denoted *JetFitter* in the following, is based on a different hypothesis. It assumes that the b- and c-hadron decay vertices lie on the same line defined through the b-hadron flight path. All charged particle tracks stemming from either the b- or c-hadron decay thus intersect this b-hadron flight axis. There are several advantages of this method:

- Incomplete topologies can also be reconstructed (in principle even the topology with a single track from the b-hadron decay and a single track from the c-hadron decay is accessible).
- The fit evaluates the compatibility of the given set of tracks with a b-c-hadron like cascade topology, increasing the discrimination power against light quark jets.
- Constraining the tracks to lie on the b-hadron flight axis reduces the degrees of freedom of the fit, increasing the chance to separate the b/c-hadron vertices.

From the physics point of view this hypothesis is justified through the kinematics of the particles involved as defined through the hard b-quark fragmentation function and the masses of b- and c-hadrons. The lateral displacement of the c-hadron decay vertex with respect to the b-hadron flight path is small enough not to violate significantly the basic assumption within the typical resolutions of the tracking detector (see Fig. 2).

This hypothesis, extensively used in the *JetFitter* algorithm, was explored for the first time in the *ghost track algorithm* developed by the SLD Collaboration [2], where the already defined b-hadron flight axis is substituted by a *ghost track* and where a numerical global χ^2 minimisation procedure was used to perform the multi-vertex fit.

3. JetFitter

3.1. Standard Kalman Filter for Vertex Reconstruction

The nowadays very commonly used Kalman Filter based algorithm for vertex reconstruction was formalised for the first time in [3] and will be briefly summarised here.

As input a set of tracks in form of helix track parameters and covariance matrices is provided, the variables being defined at the *perigee* position (point of closest approach in the transverse plane with respect to a chosen reference point). In ATLAS these track parameters are:

$$\vec{q} = \left(d_0, z_0, \phi, \theta, \frac{q}{p} \right)$$

with:

- d_0 and z_0: the distances in the transverse and longitudinal planes with respect to the reference point.
- ϕ and θ: the azimuthal and polar directions of the track momenta at the point of closest approach.
- $\frac{q}{p}$: the signed curvature of the track defined through the charge (q) divided by the momentum (p) of the particle.

These measurements are used to iteratively update the vertex position $\vec{x} = (x, y, z)$, performing one Kalman update step for every track participating in the vertex reconstruction. A linearized *measurement equation* has to be provided, which represents the relation between the track parameters $\vec{q}$ and the vertex position $\vec{x}$ plus the momentum $\vec{p}$ of the track at the vertex position $\vec{x}$, linearized around some previously guessed or estimated $\vec{x}_0, \vec{p}_0$:

$$\begin{aligned} \vec{q} &= \vec{q}(\vec{x}, \vec{p}) = \vec{q}(\vec{x}, \vec{p})_{\vec{x}=\vec{x}_0, \vec{p}=\vec{p}_0} + \left. \frac{\delta \vec{q}(\vec{x}, \vec{p})}{\delta \vec{x}} \right|_{\vec{x}=\vec{x}_0} (\vec{x} - \vec{x}_0) + \left. \frac{\delta \vec{q}(\vec{x}, \vec{p})}{\delta \vec{p}} \right|_{\vec{p}=\vec{p}_0} (\vec{p} - \vec{p}_0) \\ &= \vec{C} + A\vec{x} + B\vec{p} \end{aligned} \tag{1}$$

The constant parameters $\vec{C}$ and the Jacobian matrices A and B (one set of these values for each track, representing the *linearised track* around $\vec{x}_0, \vec{p}_0$) need to be typically calculated only once before the first cycle of fit iterations.

The first order term of the Taylor expansion of the Jacobians A and B as a function of the curvature ($O\left(\frac{q}{p}\right)$) has been calculated in [4]. The full calculation, using the complete equation of motion of a helix trajectory and the ATLAS convention for the track parameters, was performed to be used in the *Adaptive Multi-Vertex Finder* adopted for primary vertex finding in ATLAS [5] and is also implemented in *JetFitter*.

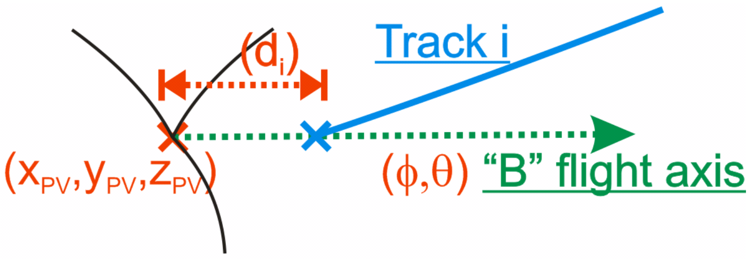

Figure 3. The JetFitter algorithm iteratively updates the primary vertex position, the b-hadron flight direction and the intersections of this axis with the fitted tracks.

3.2. The JetFitter algorithm

In *JetFitter* the variables to be updated describe directly the primary vertex and b- to c-hadron decay cascade (see Fig. 3):

$$\vec{d} = (x_{PV}, y_{PV}, z_{PV}, \phi, \theta, d_1, d_2, ..., d_N), \tag{2}$$

with:

- (x_{PV}, y_{PV}, z_{PV}): the primary vertex position
- (ϕ, θ): the azimuthal and polar directions of the b-hadron flight axis.
- $(d_1, d_2, ..., d_N)$: the distances of the fitted vertices, defined as the intersections of one or more tracks and the b-hadron flight axis, to the primary vertex position along the flight axis (N representing the number of vertices).

Before starting the fit, the variables are initialised with their prior knowledge:

- The primary vertex position (with covariance matrix), as provided by the primary vertex finding algorithm.
- The b-hadron flight direction, approximated by the direction of the jet axis, the error being provided by the convolution of the jet direction resolution with the average displacement of the jet axis relative to the b-hadron flight axis, as determined from Monte Carlo simulations.

The fit is then performed by adding iteratively as input the track parameters of the individual tracks, specifying at each iteration the vertex number to be updated.

The new measurement equation to be used to update the fit variables is:

$$\begin{aligned}\vec{q} &= \vec{q}(\vec{d},\vec{p}) = \vec{q}(\vec{d},\vec{p})_{\vec{d}=\vec{d}_0,\vec{p}=\vec{p}_0} + \left.\frac{\delta\vec{q}(\vec{d},\vec{p})}{\delta\vec{d}}\right|_{\vec{d}=\vec{d}_0}\left(\vec{d}-\vec{d}_0\right) + \left.\frac{\delta\vec{q}(\vec{d},\vec{p})}{\delta\vec{p}}\right|_{\vec{p}=\vec{p}_0}(\vec{p}-\vec{p}_0) \\ &= \widehat{\vec{C}} + \hat{A}\vec{d} + \hat{B}\vec{p}\end{aligned} \tag{3}$$

where the linearisation has now to be performed around some initial $\vec{d}_0$.

Before the first sequence of Kalman iterations is performed, N is initialized to the number of tracks, since at this stage only single-track vertices are considered, and a good estimation of $\vec{d}_0$ is provided by the minimum distance of the flight axis to the single tracks. Since this task cannot be solved analytically, this minimum distance is obtained through the implementation of a *Newton* based iterative method which minimises the distance between a straight line (the b-hadron flight axis) and a helix (the track).

Defining as $\vec{x} = (x, y, z)$ the position of the vertex on the b-hadron flight axis to be updated by the track added to the fit, the relation between $\vec{x}$ and $\vec{d}$ is:

$$\begin{aligned} x &= x_{PV} + d_i \cdot sin(\theta)cos(\phi) \\ y &= y_{PV} + d_i \cdot sin(\theta)sin(\phi) \\ z &= z_{PV} + d_i \cdot cos(\theta), \end{aligned} \tag{4}$$

where d_i is the flight length corresponding to the vertex considered, so that the measurement equations (3) and (1) are identically equal, once the equality

$$\vec{x} - \vec{x_0} = \vec{x}(\vec{d}) - \vec{x}(\vec{d_0}) = \left.\frac{d\vec{x}(\vec{d})}{d\vec{d}}\right|_{\vec{d}=\vec{d_0}} \left(\vec{d} - \vec{d_0}\right) \tag{5}$$

is used.

While $\widehat{\vec{C}} = \vec{C}$ and $\widehat{B} = B$, the position Jacobian $\widehat{A}$ to be used in the fit is different from the old one, but can simply be obtained by computing $\widehat{A} = A \left.\frac{d\vec{x}(\vec{d})}{d\vec{d}}\right|_{\vec{d}=\vec{d_0}}$, where the transformation matrix $\frac{d\vec{x}(\vec{d})}{d\vec{d}}$ is:

$$\frac{d\vec{x}(\vec{d})}{d\vec{d}} = \begin{pmatrix} 1 & 0 & 0 & -d_i sin(\theta)cos(\phi) & d_i cos(\theta)cos(\phi) & ... & sin(\theta)cos(\phi) & ... \\ 0 & 1 & 0 & d_i sin(\theta)cos(\phi) & d_i cos(\theta)sin(\phi) & ... & sin(\theta)sin(\phi) & ... \\ 0 & 0 & 1 & 0 & -d_i sin(\theta) & ... & cos(\theta) & ... \end{pmatrix} \tag{6}$$

and the columns number $(6, ..., d + 4, d + 6, ..., N + 5)$ related to the vertices not considered in the current iteration $(d_1, d_2, ..., d_{i-1}, d_{i+1}, ..., d_N)$ are identically zero.

While the original Jacobian A is usually quite insensitive to small changes of the position of the vertex $\vec{x}$ (the linear approximation for a track around a vertex holds quite well), the transformation matrix and thus the Jacobian $\widehat{A}$ has a highly non-linear dependence on the distance from the primary vertex d_i of the vertex formed by one or more intersecting tracks.

In order for the fit to converge, more cycles of iterations will be needed due to the non-linearity of the measurement equation and during each new iteration the position Jacobian $\widehat{A}$ needs to be recomputed. However, this operation can be consistently sped up by relinearising the measurement equation (3) only in the part of $\widehat{A}$ which involves the transformation matrix around the new linearisation position $\vec{d_1}$, while keeping the Jacobian A at the old linearisation position $\vec{d_0}$:

$$\begin{aligned} \vec{q} \;&=\; \vec{q}(\vec{x}(\vec{d}), \vec{p})_{\vec{d}=\vec{d_0}, \vec{p}=\vec{p_0}} + \left.\frac{\delta\vec{q}(\vec{x}(\vec{d}), \vec{x})}{\delta\vec{x}}\right|_{\vec{x}=\vec{x}(\vec{d_0})} [\vec{x}(\vec{d}) - \vec{x}(\vec{d_1}) + \vec{x}(\vec{d_1}) - \vec{x}(\vec{d_0})] \\ &+\; \left.\frac{\delta d\vec{q}(\vec{d}, \vec{x})}{\delta\vec{p}}\right|_{p=p_0} (\vec{p} - \vec{p_0}) = \widehat{\vec{C}} + \hat{A}\vec{d} + B\vec{p} \end{aligned} \tag{7}$$

where the transformation matrix can be applied directly at the new linearisation point $\vec{d_1}$, to relate $\vec{x}(\vec{d}) - \vec{x}(\vec{d_1})$ to $\vec{d} - \vec{d_1}$:

$$\vec{x}(\vec{d}) - \vec{x}(\vec{d_1}) = \frac{d\vec{x}}{d\vec{d}}_{\vec{d}=\vec{d_1}} (\vec{d} - \vec{d_1}) \tag{8}$$

Comparing Eq. (7) with Eq. (1), the coefficients $\widehat{\vec{C}}$ and the Jacobian $\widehat{A}$ can be obtained from the coefficients $\vec{C}$ and the Jacobian A:

$$\widehat{\vec{C}} = \vec{C} + A \left[\vec{x}_1 - \left. \frac{d\vec{x}}{d\vec{d}} \right|_{\vec{d}=\vec{d}_1} \right] \tag{9}$$

$$\widehat{A} = A \left. \frac{d\vec{x}}{d\vec{d}} \right|_{\vec{d}=\vec{d}_1}$$

The prescription for a fast linearisation is simple:

(i) Linearise the measurement equation (Eq. (1)), computing $\vec{C}$, A and B.

(ii) At each new iteration typically only recompute the transformation matrix (Eq. (6)) and obtain the new $\widehat{\vec{C}}$ and $\widehat{A}$ to be used in the Kalman update step (Eq. (9)).

3.3. Vertex Probability Estimation

While in the standard Kalman based fit of a vertex the *smoothing* procedure provides the correct χ^2 contribution of a track to the fitted vertex, in *JetFitter* the corresponding quantity to estimate is the χ^2 contribution of a certain vertex on the b-hadron flight axis to the fit of the whole decay chain. This can be obtained by slightly modifying the usual smoothing procedure, as described in the following steps:

- The final fitted decay chain is taken
- All tracks belonging to the chosen vertex on the b-hadron flight axis are removed iteratively from the decay chain and their χ^2 contributions are added up
- The resulting decay chain is compared with the final fitted one, considering this as an additional contribution to the χ^2.

The χ^2 obtained has then $2*N_t-1$ degrees of freedom, where N_t is the number of tracks involved in the vertex considered.

4. Finding algorithm

After the primary vertex and the b-hadron flight axis have been initialised, a first fit is performed under the hypothesis that each track represents a single vertex along the b-hadron flight axis, until χ^2 convergence is reached, obtaining a first set of fitted $(\phi, \theta, d_1, d_2, ..., d_N)$.

A clustering procedure is then performed, where all combinations of two vertices (picked up among the vertices lying on the b-hadron flight axis plus the primary vertex) are taken into consideration, filling a *table of probabilities*.

In order to take the correlations of the positions of the vertices with the direction of the b-hadron flight axis correctly into account, but avoiding to repeat the complete fit frequently, the hypothesis of merging each pair of vertices is tested adding on top of the previous complete fit a single Kalman update step, applying the constraint $d_i = d_j$ (for merging two vertices on the b-hadron flight axis or $d_i = 0$ (to merge vertex i with the primary vertex). This is technically realized by switching from the Kalman *weighted mean* formalism to the *gain* formalism, where a constraint can be imposed as a measurement without error (as demonstrated formally in [6]). After the constraint is applied to the fit, the vertex probability is estimated as already described in Section 3.3.

This then represents the probability of the two vertices under consideration to form a single vertex and at the same time the probability for this single vertex to be compatible with the rest of the decay chain. Avoiding the complete fit to be performed for each pair of vertices reduces the number of iterations needed from $\propto N^3_{tracks}$ to $\propto N_{tracks}$.

After the table of probabilities is filled, the vertices with the highest compatibility are merged, a new complete fit is performed and a new table of probabilities is filled. This procedure is then iterated until no pairs of vertices with a probability above a certain threshold exist anymore.

The result of this clustering procedure is a decay topology with a well defined association of tracks to vertices along the b-hadron flight axis, with at least one track each.

4.1. Negative flight lengths

When the fit is applied to tracks belonging to a real b-jet, it may sometimes happen that the b-hadron flight axis converges to a non physical direction pointing opposite to the real b-hadron flight direction, on which most or all of the fitted vertices show negative flight lengths.

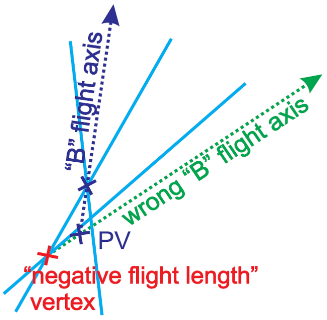

Figure 4. The fit can end up forming a vertex behind the primary interaction vertex, with a negative flight length, which represent a local minimum for the *B flight axis* direction

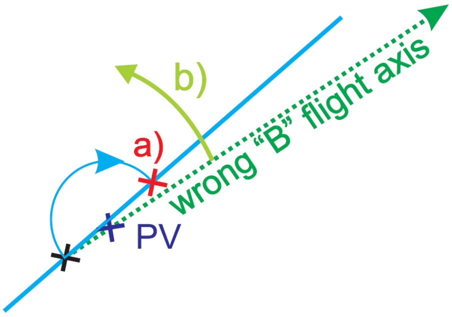

Figure 5. Relinearising the *negative flight length* tracks with the opposite flight length (a) results in a global attempt to move the fit out of the wrong minimum (b).

The following solution has been implemented to avoid this problem:

(i) A first fit is performed until convergence is reached

(ii) A second fit is done under the hypothesis that the relinearisation of the track is performed as if the track would have had a opposite flight length.

(iii) The fit is iterated again until χ^2 convergence is reached (which also states that no more tracks have been swapped between positive and negative flight lengths).

(iv) The default fit is performed until final convergence is reached (usually only one more iteration of the fit).

The relinearisation of the tracks with negative flight length as to have positive flight length does not only change the sign of the derivative in the Jacobian $\widehat{A}$ of the b-hadron flight direction with respect to a shift of the fitted vertex position, but also considerably affects the fit because the constant term $\widehat{C}$ in Eq. (3) contains the residual (which the fit tries to minimise) between the track and the vertex position on the b-hadron flight axis being now *artificially* chosen to be opposite to the negative fitted one.

Tracks originating from the primary interaction vertex, even with their flight length swapped, will still have small residuals and continue to barely affect the fit of the b-hadron flight direction, while tracks coming from secondary vertices which randomly have formed negative flight length vertices will generally show consistent residuals and will try to move the fit towards succeeding in having their vertex on the right side, out of the wrong minimum, as is schematically illustrated in Fig. 5.

5. B-Tagging algorithm

The b-tagging algorithm implemented as a first application of *JetFitter* is based on separating b-jets from c- and light-quark (u,d,s) jets (denoted as b,c and l-jets in the following) by means of the definition of a likelihood function.

The decay topology is described by the following discrete variables:

(i) Number of vertices with at least two tracks.

(ii) Total number of tracks at these vertices.

(iii) Number of additional single track vertices on the b-hadron flight axis.

while the vertex information is condensed in the following variables:

(i) Mass: the invariant mass of all charged particle tracks attached to the decay chain.

(ii) Energy Fraction: the fraction of energy of these particles divided by the sum of the energies of all charged particles matched to the jet.

(iii) Flight length significance $\frac{d}{\sigma(d)}$ of the weighted average position divided by their errors of the displaced vertices.

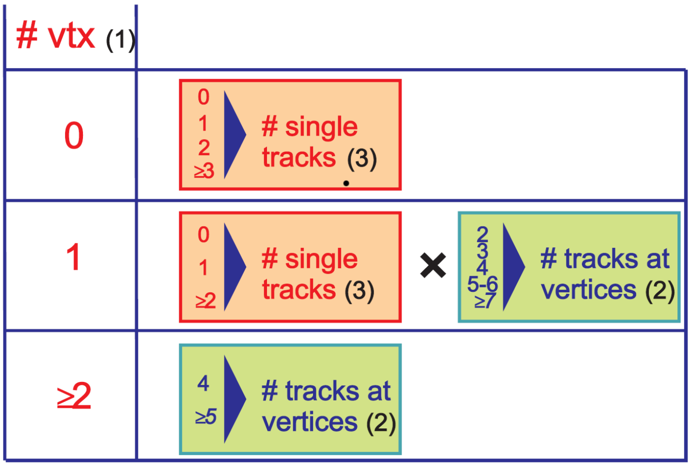

Figure 6. 13 different topologies are defined, combining the three discrete variables in such way as to reduce their correlations. Variable (1) represents the number of vertices with at least two tracks. For the case of one single vertex with at least two tracks, the discrete *PDFs* for both the other two variables, total tracks at vertices (2) and single additional tracks (3), are used, but they're then considered as uncorrelated, so that their corresponding coefficients are just multiplied.

The use of these variables permits to define a likelihood function of the form:

$$L^{\mathrm{b,c,l}}(x) = \sum_{cat} \mathit{coeff}(cat) \cdot PDF_{cat}(mass) \cdot PDF_{cat}(energyFraction) \cdot PDF_{cat}\left(\frac{d}{\sigma(d)}\right), \qquad (10)$$

which has to be parametrised separately for each of the three different flavours (b, c or l).

The information about the decay topology of the jet as reconstructed by *JetFitter* is represented by the category (the coefficient *coeff*(cat) representing how probable it is to find a certain topology for a given flavour), while the vertex information is contained in the probability distribution functions (*PDFs*).

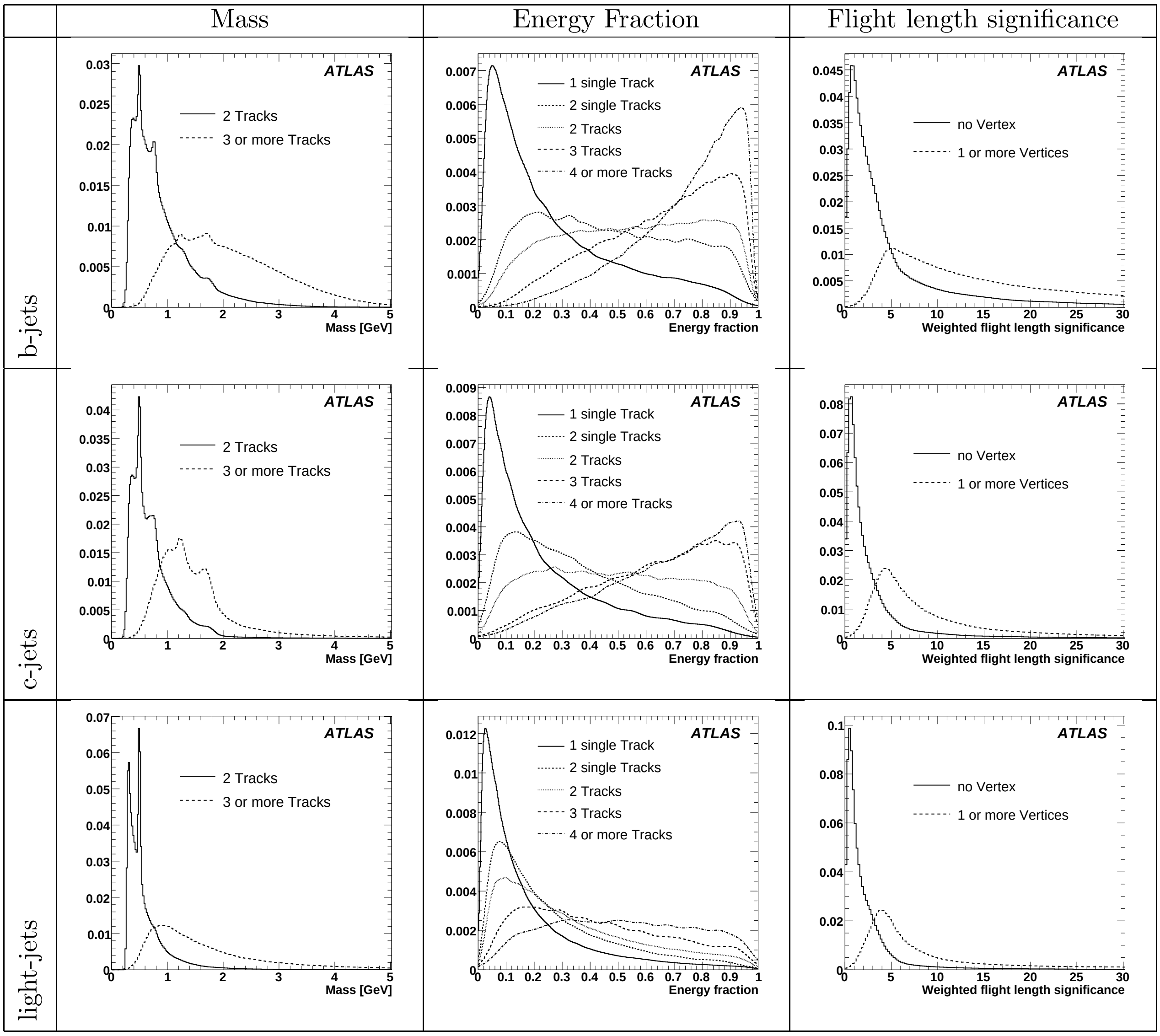

Figure 7. The *PDFs* for the mass, the energy fraction and the flight length significance are shown, separately for the three different jet-flavours and split according to the decay chain topology found by *JetFitter*. A vertex is defined here as having at least 2 tracks. Single tracks are considered only in the case no vertex was found.

The discrete variables describing the decay topology are combined according to the scheme of Fig. 6 into 13 category coefficients.

In order to reduce the correlations between the decay topology and vertex related variables and to increase the discrimination power, the *PDFs* are made category dependent. This splitting of *PDFs* is done only when strictly needed.

The templates for the vertex variables are shown in Fig. 7 for all three flavours. Each *PDF* was split independently into the categories it showed to be most correlated with, in order to maintain the number of split *PDFs* to be determined on Monte Carlo simulated events as low as possible. They were determined running the ATLAS full detector simulation on $pp \to WH(m_H = 120$ GeV$/c^2)$ Monte Carlo events with $H \to b\bar{b}$, $H \to c\bar{c}$ and $H \to u\bar{u}$ and on $pp \to t\bar{t}$ Monte Carlo events.

5.1. Performance

The default b-tagging algorithm in ATLAS uses combined information from both impact parameter based and secondary vertex based b-tagging algorithms not taking explicitly into account correlations between the algorithms. A description of the method can be found in [7]. *JetFitter* has been combined exactly in the same way with the pure impact parameter based b-tagging algorithm.

The performance of the combined default algorithm with the combined JetFitter based algorithm has been compared. The comparison was performed running the reconstruction chain on a sample of 550.000 $pp \to t\bar{t}$ Monte Carlo events, passed through the full simulation of the ATLAS detector, using the official data sample.

b-tagging efficiency	Default	JetFitter
40%	1247 ± 38	2012 ± 78
50%	477 ± 9	765 ± 18
60%	156 ± 2	228 ± 3
70%	50.0 ± 0.3	50.2 ± 0.3

Table 1. Comparison of light-quark rejection rates between the default ATLAS b-Tagging algorithm and *JetFitter*. The numbers are preliminary.

The results in terms of rejection rate (the rejection rate is defined as the inverse mistagging efficiency for non-b jets at a given b-jet efficiency) against light-quark jets for b-tagging efficiencies between 40% and 70% are shown in Table 1. At b-tagging efficiencies of 50% and 60%, an increase of light-quark jets rejection of about 60% and 45%, respectively, is seen. This makes the *JetFitter* based algorithm very promising. The higher the b-tagging efficiency, the smaller the observed increase in rejection is. The reason for this behaviour is that the Impact Parameter based algorithm becomes more and more relevant for large b-tagging efficiencies with respect to the secondary vertex based one.

The numbers shown here are preliminary since they are not yet officially approved by the ATLAS Collaboration and represent the state of the ATLAS Monte Carlo simulation and event reconstruction algorithms in September 2007.

References

[1] V. Kostyukhin, ATL-PHYS-2003-031 (08/2003)

[2] K. Abe *et al.* [SLD Collaboration], in *Proc. of the 19th Intl. Symp. on Photon and Lepton Interactions at High Energy LP99* ed. J.A. Jaros and M.E. Peskin,

[3] R. Frühwirth, Nucl. Instrum. Meth. A **262** (1987) 444

[4] P. Billoir and S. Qian, Nucl. Instrum. Meth. A **311** (1992) 139

[5] G. Piacquadio, K. Prokofiev, A. Wildauer, Proc. CHEP 2007, Victoria, Canada (these Proceedings)

[6] W. D. Hulsbergen, Nucl. Instrum. Meth. A **552** (2005) 566 [arXiv:physics/0503191].

[7] S. Corrèard *et al.*, ATL-PHYS-2004-006 (11/2003)