ATLAS PUB Note

ATL-PHYS-PUB-2020-014

July 28, 2020

Deep Sets based Neural Networks for Impact Parameter Flavour Tagging in ATLAS

The ATLAS Collaboration

This work introduces a new architecture for Flavour Tagging based on Deep Sets, which models the jet as a set of tracks, in order to identify the experimental signatures of jets containing heavy flavour hadrons using the impact parameters and kinematics of the tracks. This approach is an evolution with respect to the Recurrent Neural Network (RNN) currently adopted in the ATLAS experiment, which treats track collections as a sequence. The Deep Sets model comprises a permutation-invariant and highly parallelisable architecture, leading to a significant decrease in training and evaluation time, and thus allowing for much faster turn-around times for optimisation. Additionally, this permutation invariance encoded in the model is more physically motivated than the sequence-based RNN. We compare the Deep Sets algorithm with the RNN benchmark, probe the model to interpret the information learned, and provide studies optimising the Deep Sets algorithm by loosening the track selection and including additional inputs.

28 July 2020: Updated references

1 Introduction

For the physics program of the ATLAS experiment at the Large Hadron Collider (LHC), the identification of jets initiated by b-quarks, or b-tagging, is a fundamental tool. Ensuring its optimal performance is particularly important for the study of the Higgs boson and the top quark [1, 2], as well as many exotic extensions of the Standard Model with resonances preferentially decaying to heavy quarks [3].

The characteristically long lifetime of hadrons containing b-quarks (b-hadrons) of the order of 1.5 ps [4] leads to two classes of b-tagging algorithms: *vertexing* based algorithms which explicitly reconstruct a production point, or vertex, of the b-hadron decay displaced from the primary interaction point, and *impact parameter (IP)* based algorithms which exploit the displacement of the reconstructed charged particles trajectories (tracks) produced in b-hadron decays from the primary interaction point.

This work builds on that of the RNNIP algorithm [5], which uses impact parameter information and recurrent neural networks (RNNs) for b-tagging, and provides improvements over other IP-based algorithms by accounting for the correlations between the track features, and the inclusion of additional discriminating variables. Here a new algorithm is introduced, Deep Impact Parameter Sets (DIPS), based on the Deep Sets architecture [6] and on the application of the Deep Sets formalism within particle physics known as Energy / Particle Flow Networks [7]. DIPS solves the same task as RNNIP but treats the tracks in the jet as an unordered, variable-sized set rather than as a sequence, avoiding the need to specify a sequence ordering and the slow processing of RNNs. Given that the b-hadron decay products do not exhibit any intrinsic sequential ordering, the Deep Sets architecture is also better physically motivated.

DIPS is demonstrated to be as performant as RNNIP but faster to train, decreasing evaluation time and reducing turn-around time for optimization. Therefore, optimization studies of the track selection criteria and new track features are also included. In addition, a discussion on how to measure the algorithm's efficiency in data, in particular for jets that do not contain a b or a c-hadron, is presented. Finally, one avenue of research in deep learning models is exploring the interpretability of the models, or trying to dissect what information the network is learning. Diagnostic studies from the machine learning literature are presented to demonstrate the well known characteristics from b-quark fragmentation and hadronization process that the network has gleaned.

This note is organized as follows: Section 2 describes the datasets and selections used to train and evaluate the algorithms, while section 3 details impact parameter based taggers, the Deep Sets algorithm and our specific implementation. Section 4 shows investigations of what the network has learned, results for the timing metrics, discussion on calibrating the algorithm, and the optimization studies conducted. Finally, section 5 summarizes the conclusions.

2 ATLAS detector and training datasets

The ATLAS detector [8] at the LHC covers nearly the entire solid angle around the collision point.[1] It consists of an inner tracking detector surrounded by a thin superconducting solenoid, electromagnetic and

[1] ATLAS uses a right-handed coordinate system with its origin at the nominal interaction point in the centre of the detector and the z-axis along the beam pipe. The x-axis points from the interaction point to the centre of the LHC ring, and the y-axis points upwards. Cylindrical coordinates (r, ϕ) are used in the transverse plane, ϕ being the azimuthal angle around the z-axis. The pseudorapidity is defined in terms of the polar angle θ as $\eta = -\ln\tan(\theta/2)$. Angular distance is measured in units of $\Delta R \equiv \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$.

hadronic calorimeters, and a muon spectrometer incorporating three large superconducting toroidal magnets. The inner detector system (ID) is immersed in a 2 T axial magnetic field and provides charged-particle tracking in the range $|\eta| < 2.5$. The high-granularity silicon pixel detector covers the vertex region and typically provides four measurements per track, the first hit being normally in the insertable B-layer (IBL) installed before Run 2 [9, 10]. It is followed by the silicon microstrip tracker (SCT) which usually provides eight measurements per track. These silicon detectors are complemented by the transition radiation tracker (TRT), which enables radially extended track reconstruction up to $|\eta| = 2.0$.

Algorithm training and evaluation is performed with simulated $t\bar{t}$ events, produced by $\sqrt{s} = 13$ TeV proton-proton collisions, in which at least one of the W bosons, from the top quark decay, decays leptonically. Events are generated using the PowhegBox [11–14] v2 generator at next-to-leading order with the NNPDF3.0NLO [15] parton set of distribution functions (PDF) and the h_{damp} parameter[2] set to 1.5 m_{top} [16], with $m_{\text{top}} = 172.5$ GeV. The events are interfaced to Pythia 8.230 [17] to model the parton shower, hadronisation, and underlying event, with parameters set according to the A14 tune [18] and using the `NNPDF2.3lo` set of PDFs [19]. The decays of b and c-hadrons are performed by EvtGen v1.6.0 [20]. Particles are passed through the ATLAS detector simulation [21] based on GEANT4 [22].

Tracks are reconstructed from energy deposits, or hits, in the inner detector system and are required to pass a quality selection: each track must have at least 7 hits in the silicon layers (pixel and SCT, where dead sensors are not penalised), no more than two missing hits where expected in the silicon layers, no more than one hit shared by multiple tracks, at least one hit in the pixel detector, and $|\eta| < 2.5$. The event's selected primary vertex (PV) is defined as the reconstructed primary vertex with largest $\sum p_{\text{T}}^2$ of the associated tracks.

Jets are reconstructed from particle flow objects [23] using the anti-k_T algorithm [24] with $R = 0.4$. The jet energy scale is calibrated according to [25]. Jets used for training and evaluation have $p_{\text{T}} \geq 20$ GeV, $|\eta| < 2.5$, and are required not to overlap with a generator-level electron or muon from W boson decays. Additionally, the contamination of jets from other interactions in the beam crossing (pile-up) is surpressed by applying the jet vertex tagger [26] optimized for particle flow jets.

Tracks are associated to jets using a ΔR association cone which decreases as a function of jet p_{T}, with a maximum association ΔR(track, jet) of approximately 0.45 for a jet with $p_{\text{T}} = 20$ GeV and ΔR(track, jet) of approximately 0.25 when the jet $p_{\text{T}} = 200$ GeV. If a track is within the association cones of more than one jet, it is assigned to the jet which has a smaller ΔR(track, jet).

The impact parameter of the track characterises the point-of-closest approach of a track to the PV in the longitudinal ($z_0 \sin\theta$) and transverse (d_0) planes. Of particular use in b-tagging is the IP significance defined as the impact parameter divided by its uncertainty, $s_{d0} = d_0/\sigma_{d0}$ and $s_{z0} = z_0 \sin\theta/\sigma_{z0 \sin\theta}$. The track's IP and its significance are signed according to the track's direction with respect to the jet axis and the primary vertex [27]. A positive IP is expected to be consistent with a track produced from a displaced vertex. This procedure is referred to as lifetime signing. The nominal track selection considered in the algorithms to be described requires tracks with $p_{\text{T}} > 1$ GeV, $|d_0| < 1$ mm, and $|z_0 \sin\theta| < 1.5$ mm.

The jets are labelled as b-jets if they are matched to at least one b-hadron having $p_{\text{T}} \geq 5$ GeV within ΔR(b-hadron, jet)< 0.3 of the jet axis. If this condition is not satisfied, then c-hadrons and then τ leptons are searched for, with similar selection criteria. If a jet is matched to a c-hadron (τ-lepton), it is labelled a c-jet (τ-jet). A jet that does not meet any of these conditions is called a light-flavour jet.

[2] The h_{damp} parameter is a resummation damping factor and one of the parameters that controls the matching of Powheg matrix elements to the parton shower and thus effectively regulates the high-p_{T} radiation against which the $t\bar{t}$ system recoils.

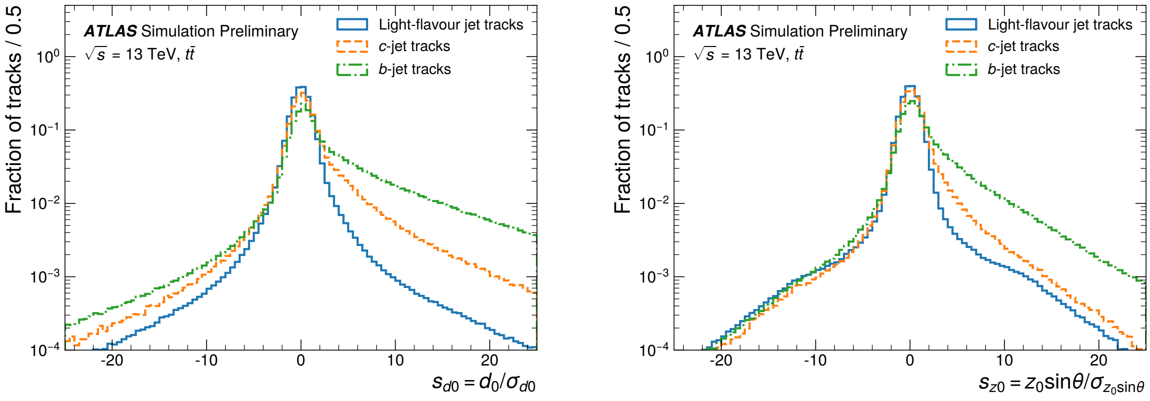
3 Algorithm description

3.1 Current IP-based *b*-taggers

ATLAS employs several IP-based algorithms which are later combined with vertexing algorithms to produce a "high-level" tagger for general use. The IP3D algorithm [28] assigns probabilities to tracks based on two-dimensional likelihood templates, with the tracks' $z_0 \sin\theta$ and d_0 lifetime signed significances, built from simulated jets. These templates are obtained in 14 exclusive categories defined by the hit patterns of the tracks, and separately for tracks in *b*-jets, *c*-jets and light-flavour jets. The inclusive distribution of $z_0 \sin\theta$ and d_0 lifetime signed significances for the different jet flavours are shown in Figure 1. By assuming that the track probabilities inside a jet are independent, jet-level likelihoods can be constructed by multiplying the individual probabilities. The IP3D *b*-tagging discriminants are therefore defined as:

$$D_{\mathrm{IP3D,l}} = \log \prod_{i\,\in\mathrm{tracks}} \frac{p_b^i}{p_l^i} \qquad D_{\mathrm{IP3D,c}} = \log \prod_{i\,\in\mathrm{tracks}} \frac{p_b^i}{p_c^i}. \tag{1}$$

RNN based IP algorithms aim to overcome this overly simplistic assumption of independence, and offer the possibility to employ more features than only the IP significance in the discriminant [5].



(a)

(b)

Figure 1: Lifetime signed transverse (a) and longitudinal (b) significances for *b*-jets, *c*-jets and light-flavour jets.

RNNs operate on variable length *sequences* by iterating over the sequence elements, processing them with a neural network, and using previously processed elements when processing new ones. It then outputs a fixed size vector that can be used for classification. The RNNIP algorithm utilizes a Long Short Term Memory (LSTM) cell for the RNN to preserve long range correlations between the elements of the sequence [29]. As shown in [5], the accounting for these correlations allows the RNN to be more performant than IP3D even when trained on the same inputs. The use of neural networks instead of histograms allows one to avoid the "curse of dimensionality" when using additional variables sensitive to the kinematics of the *b*-hadron decay which significantly improve performance [5].

An implementation of the RNNIP algorithm is used as the baseline for comparison to DIPS, but has further optimisations with respect to [5]. The RNNIP architecture comprises a 100 dimensional LSTM hidden state and a dropout layer, with dropout fraction of 0.2, before a 20 unit fully connected layer for classification, uses track IP significance, kinematics, and the number of hits in the silicon detectors as features (described in Table 1), and orders the tracks by s_{d_0}.

3.2 Sets-based architectures

The Deep Sets architecture [6], which treats the elements as a set without any specific order, maintains the benefits of the RNNIP algorithm, while avoiding the required element ordering (which for b-tagging is empirically driven rather than strictly dictated by the inputs of the problem). This architecture was first employed in particle physics in a phenomenological study on the identification of different types of jets [7]. Adopting the formalism of [7], if p_i is the vector representing the inputs associated with the i^{th} track in the jet, then the Deep Sets architecture applies a neural network (NN) Φ to each track, sums over the tracks, and then applies additional processing on the summed representation with a feed forward NN F, as described in equation 2,

$$\mathcal{O}\left(\{p_1, \ldots, p_n\}\right) = F\left(\sum_{i=1}^{n} \Phi\left(p_i\right)\right), \tag{2}$$

where $\mathcal{O}\left(\{p_1, \ldots, p_n\}\right)$ represents the b-, c-, and light-flavour class probabilities derived from the inputs for the n tracks in the jet. The architecture bifurcates the problem into operations over inputs and operations over sets, where the track-network Φ extracts the relevant track features, and the jet-network F accounts for the correlations between the tracks. The permutation invariance of the set is encoded with the permutation invariant sum operation, although other permutation invariant operations such as the max or average could be used as well. The presence of this aggregation layer in the architecture encodes information about track multiplicity inside the jet, which is a useful information for identifying b-jets.

This Deep Sets architecture offers the same advantages as RNNIP but encodes permutation invariance between the tracks in the jet, giving a more natural representation of the data and allowing the algorithm to be trained more efficiently with fewer parameters and less data [30]. In addition, Deep Sets offers a major additional advantage over RNNs in that the operation of processing the tracks in the jet with the Φ network can be easily parallelised. This allows training and evaluation to make significantly more efficient use of GPUs over the non-parallelisable iterative processing of the RNN. The timing performance comparison between DIPS and RNNIP is further discussed in Section 4.2.

3.3 Implementation details

All algorithms are trained with a sample of simulated $t\bar{t}$ events (described in Section 2) for multi-class classification between b-jets, c-jets and light-flavour jets. To avoid classification based on the differing kinematic spectra of the jet classes, the p_{T} spectra for b-jets and c-jets is reweighted to the light-flavour jet spectra, as described in reference [5].

The class probabilities predicted by the model outputs (p_b, p_c, and p_l), are combined into a b-tagging discriminant:

$$D_b = \log \frac{p_b}{(1 - f_c)p_l + f_c p_c} \tag{3}$$

where f_c is a free parameter that balances between the rejection of light-flavour vs c-jets for a given efficiency of selecting b-jets, and can be optimized post-training. A value of $f_c = 0.07$ was used in these studies as this is representative of the fraction of c-jets relative to non b-jets in $t\bar{t}$ events.

For the timing comparisons in Section 4.2, the same input features are used for both RNNIP and the DIPS. The features used in each algorithm are described in Table 1. The track variables related to the track reconstruction quality focus on the IBL and the next-to-innermost pixel layer (PIX1) due to their strong impact on the IP significance distributions. In particular, the number of split hits, which are hits being created by multiple charged particles [31], is used to help identify dense tracking environments, in which distinguishing tracks from heavy flavour decays is generally more difficult.

Input	**Description**
s_{d0}	d_0/σ_{d0}: Transverse IP significance
s_{z0}	$z_0 \sin\theta/\sigma_{z0\sin\theta}$: Longitudinal IP significance
$\log p_T^{frac}$	$\log p_T^{track}/p_T^{jet}$: Logarithm of fraction of the jet p_T carried by the track
$\log \Delta R$	Logarithm of opening angle between the track and the jet axis
IBL hits	Number of hits in the IBL: could be { 0, 1, or 2 }
PIX1 hits	Number of hits in the next-to-innermost pixel layer: could be { 0, 1, or 2 }
shared IBL hits	Number of shared hits in the IBL
split IBL hits	Number of split hits in the IBL
nPixHits	Combined number of hits in the pixel layers
shared pixel hits	Number of shared hits in the pixel layers
split pixel hits	Number of split hits in the pixel layers
nSCTHits	Combined number of hits in the SCT layers
shared SCT hits	Number of shared hits in the SCT layers

Table 1: Track features used as inputs for RNNIP and DIPS algorithms.

After applying the track selections described in Section 2, the tracks are ordered by decreasing s_{d0}, and the first 15 tracks are kept for processing. The ordering plays a limited role in the algorithm, since typical jets in the topology investigated should have an average number of tracks that is smaller than the maximum allowed number of tracks (see Table 4). Since the p_T^{frac} and ΔR variables have a tail at larger values, the natural log of the value for these variables is used as the feature in order to improve the convergence time of the training. Variable normalisation to zero mean and unit variance is frequently used for preprocessing of features in ML algorithms. As many of our input variables already have near zero mean, only a subset of the track features are normalised: $\log p_T^{frac}$, $\log \Delta R$, nPixHits, nSCTHits, as well as d_0 and $z_0 \sin\theta$ for the optimised DIPS training.

A simplified scheme of the DIPS architecture is shown in Figure 2, which is based on the architecture in reference [7]. A grid search over the hyperparameters including the number of layers in the Φ and F networks, the number of nodes in the Φ and F networks and the dimension of the track latent space revealed similar performance for many different choices of these hyperparameters. Both batch normalisation [32] and dropout [33] were tested, and it was found that batch normalisation was helpful for the DIPS b-tagging performance while dropout was not.

The RNNIP and DIPS trainings were performed with 3 million jets, with 20% of these jets held out as a validation set to determine when to stop the training. After 10 consecutive training epochs (or iterations through the training dataset) without finding a new validation loss minimum, the training is terminated

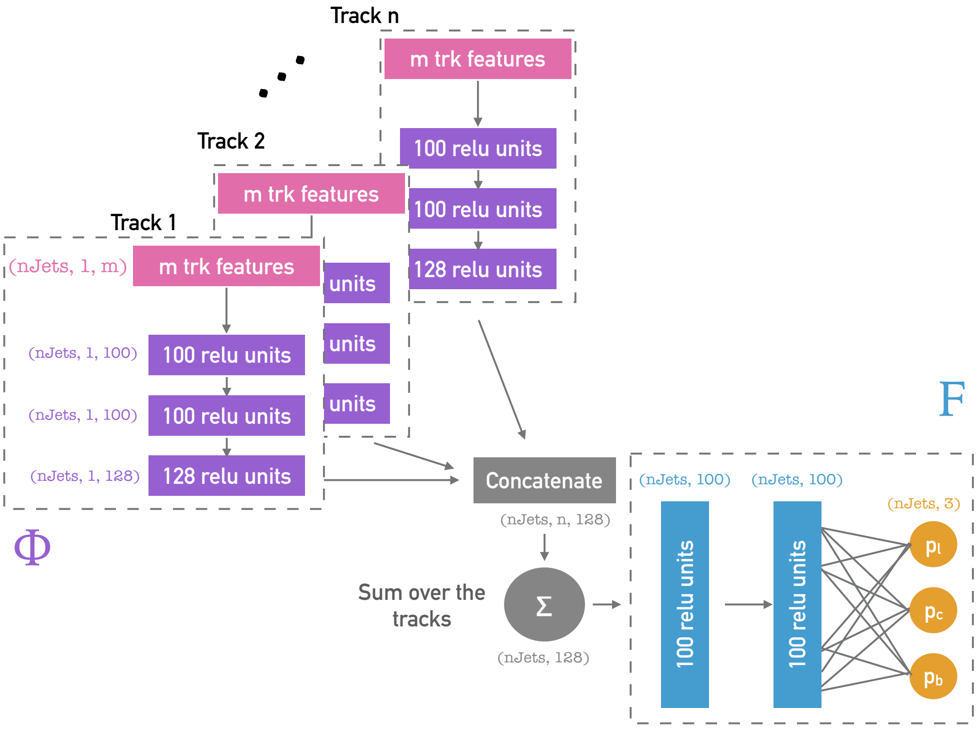


Figure 2: Architecture for the DIPS algorithm. The number of hidden units in the different neural network layers correspond to the final optimized architecture.

and the model with the best validation loss was selected. Both the RNNIP and DIPS architectures were implemented in Keras [34] and trained with the TensorFlow backend [35]. Algorithms were trained with the Adam optimiser [36] with a learning rate of 10^{-3} and a batch size of 256. The performance metrics shown in the following sections are obtained with a statistically independent dataset of 3 million jets.

4 Results

4.1 Baseline Performance

The distribution of the DIPS discriminant D_b (defined in Equation 3) for each of the jet flavours is shown in Figure 3. The peak at $D_b = -1.3$ is due to jets without any selected tracks. Clear separation between the distribution of b-jets and light-flavour jets can be seen, as well as a strong but smaller separation between b-jet and c-jets as expected due the similarities between b-hadron and c-hadron decays.

The performance of taggers can be examined and compared through a Receiver Operator Characteristic (ROC) curve: a scan is performed for a threshold τ on D_b, and the efficiency for b-jets at each threshold is computed as the fraction of b-jets with $D_b > \tau$, while the rejection of c-jets or light-flavour jets is computed as one over the fraction of c-jets or light-flavour jets (inverse mistag efficiency), respectively, with $D_b > \tau$. The b-jet efficiency and light-flavour (or c) jet rejection for the same τ are then plotted. Each model is trained five times and for a given b-jet efficiency, the mean of the rejections is used as the nominal

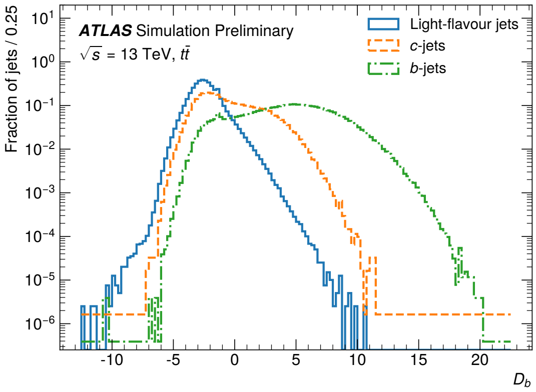


Figure 3: Distributions of DIPS b-tagging discriminant, as defined in Equation 3, for b-jets, c-jets and light-flavour jets.

value and the standard deviation of the rejections is used for the width of the curve. This ensemble of trainings is known to assess the predictive uncertainty of machine learning-based algorithms [37].

The ROC curves for b-jet efficiency versus light-flavour jet rejection and for b-jet efficiency versus c-jet rejection of the DIPS and RNNIP algorithms are shown in Figure 4. The lowest b-jet efficiency displayed corresponds to the lowest efficiency benchmark used in physics analyses within the ATLAS experiment. The DIPS algorithm provides up to a 15% additional light-flavour jet rejection and a 5% additional c-jet rejection at a given b-jet efficiency over the RNNIP algorithm. Notably, as will be discussed in Section 4.2, this similar performance comes with a significant decrease in training and evaluation time.

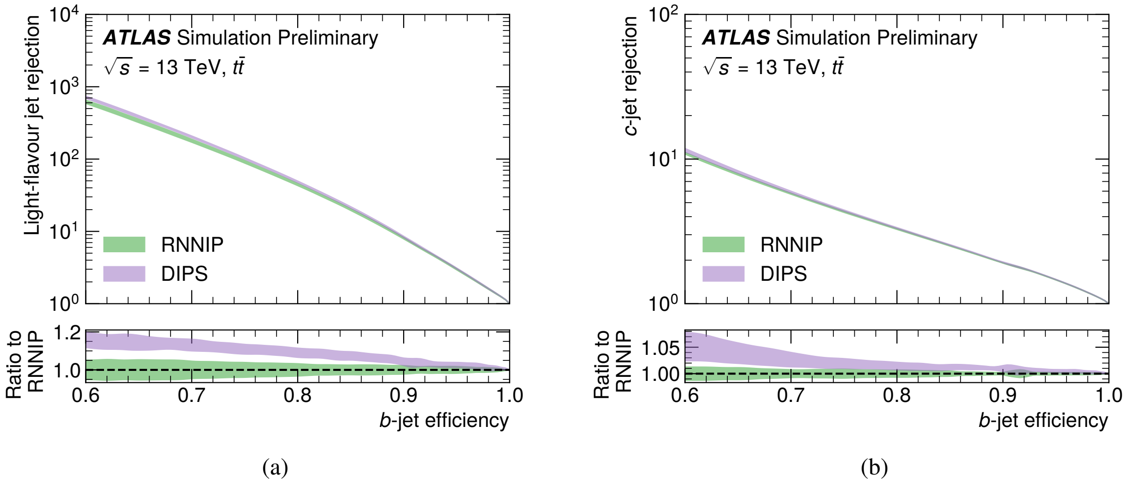


Figure 4: Light-flavour jet rejection as a function of b-jet efficiency (a) and c-jet rejection as a function b-jet efficiency (b) of the RNNIP (green) and DIPS (purple) algorithms. The central curves and error bands show the mean and standard deviation, respectively, of the rejection at each b-jet efficiency for 5 trainings. The ratios are computed with respect to the RNNIP ROC curve.

In order to explore what DIPS is learning in the correlations between features that aids the classification performance, the average saliency map for b-jets with 8 associated tracks and failing a threshold corresponding to 77% b-tagging efficiency is shown in Figure 5. The saliency map is computed as

$$\frac{\partial D_b}{\partial x_{ik}} = \frac{1}{N} \sum_{j=1}^{N} \frac{\partial D_b^{(j)}}{\partial x_{ik}^{(j)}}, \tag{4}$$

and is the gradient of the discriminant value $D_b^{(j)}$ with respect to each track feature input $x_{ik}^{(j)}$, averaged over jets (j) in a sample of N jets [38]. In this case, the feature inputs are normalized to zero mean and unit variance, in a similar way to the training procedure. The saliency map gives a linearised view of how the discriminant value is sensitive to changes in the inputs. Figure 5 thus shows how this sample of b-jets which failed tagging could be modified to make them more b-jet like. One can see there is a reasonably strong positive gradients for the significances (s_{d0} and s_{z0}) extending up to 5 tracks, which is the average number of charged particle tracks in a b-hadron decay. Beyond 5 tracks, the gradients for all features are either nearly zero or negative, indicating that either these tracks provide no further information or that tracks with large feature values are more indicative of background. In addition, DIPS is highly sensitive to the $\log p_{\mathrm{T}}^{frac}$ and $\log \Delta R$ of the leading s_{d_0} track, which is consistent with the harder fragmentation of b-quarks with respect to light-flavour and charm jets. Interestingly, this strong correlation with $\log p_{\mathrm{T}}^{frac}$ and $\log \Delta R$ for the highest s_{d_0} track also indicates that simply enlarging the IPs of a track in a jet would not directly lead to a jet passing a tagging threshold, as the track must also be consistent with the kinematic expectations from b-jet fragmentation. The gradients for the shared and split hits of the high s_{d0} tracks are strongly negative since tracks formed from random combinations of hits are more likely in highly dense environments. It can also be seen that the correlation with the overall number of hits in the inner most pixel layers, IBL and PIX1, is positive but small. Such features are of high importance to the estimate of the IP and IP resolution. However such information is also encapsulated in the IP significance features which are strongly correlated with the discriminant. We suspect these correlations are observed to be relatively small due to the discriminator heavily relying on the IP significance for the first order estimate of the quality of the track and the track's utility for classification.

4.2 Time comparison

A further key comparison metric between the RNNIP and DIPS algorithms is the time needed for training and evaluation. The training time limits the ability to critically perform optimisation tests and compare model variants, while the evaluation time impacts ATLAS reconstruction time when deployed at scale and the ability to use such algorithms in low-latency environments such as the trigger. The DIPS and RNNIP models with comparable numbers of parameters are compared in terms of their speed of training and evaluation in Tables 2 and 3, respectively. Training comparisons are done on an NVIDIA 2080 Ti GPU, while evaluation comparisons are performed on an NVIDIA Titan X GPU. Five versions of each model are trained and evaluated, and the mean and standard deviation of the training and evaluation time is reported. A significant speed up of more than a factor of 2 for the DIPS algorithm over RNNIP is observed. As training also involves the early stopping procedure, and thus each algorithm may train for a different number of epochs, the training time per epoch is also reported and shows more than a factor of 3 faster speed for DIPS over RNNIP. This is similar to evaluation time, where DIPS is seen to be nearly a factor of 4 faster than RNNIP.

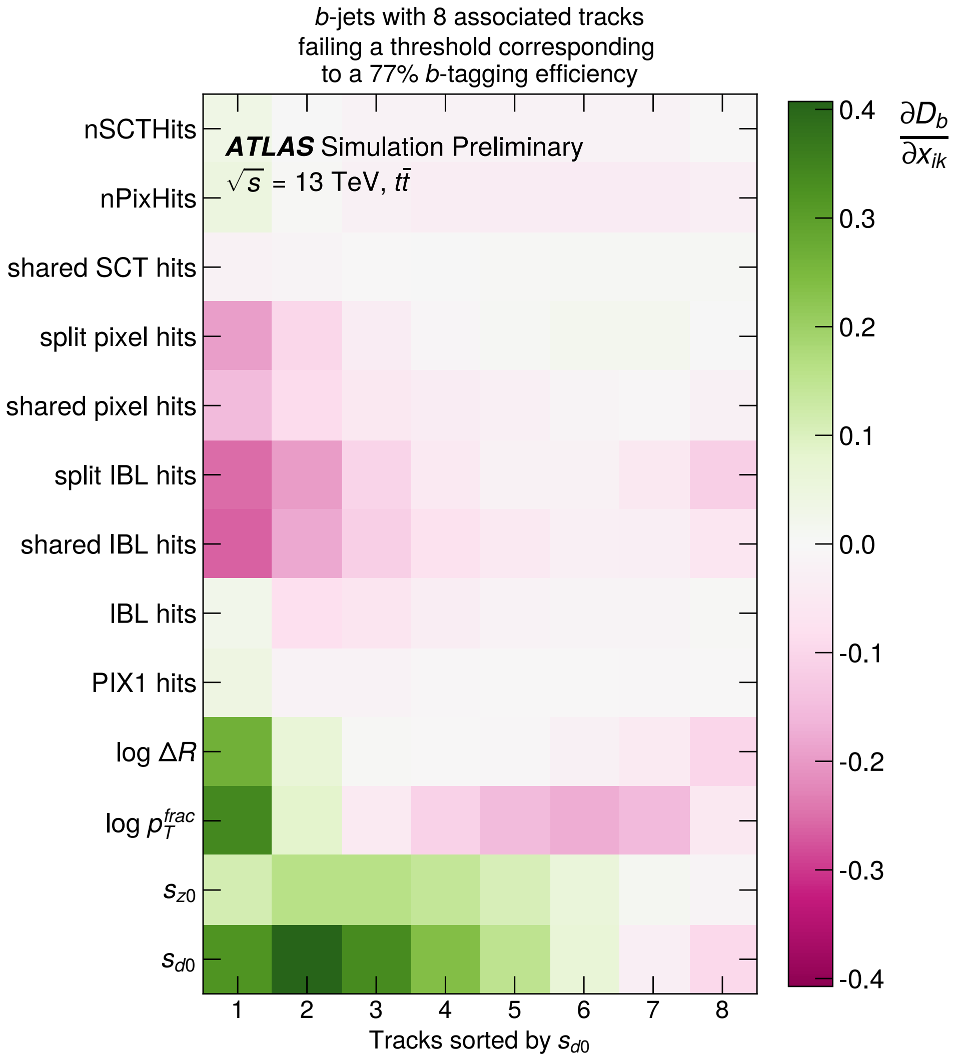


Figure 5: Saliency map for b-jets with 8 tracks. The track features are shown on the y-axis, the tracks (ordered by s_{d_0}) are listed on the x-axis. The colors in each pixel represent the gradient defined in Equation 4.

Model	**Parameters**	**Training time [min]**	**Time / epoch [s]**
RNNIP	47k	86 ± 13	241 ± 14
DIPS	49k	44 ± 4	78 ± 4

Table 2: Timing metrics for trainings performed on Nvidia 2080 Ti GPUs. The nominal value denotes the mean of five independent trainings, while the error bar is the standard deviation.

Model	**Parameters**	**GPU Evaluation time [s]**	**CPU evaluation time [s]**
RNNIP	47k	170 ± 2	685 ± 84
DIPS	49k	46 ± 2	206 ± 98

Table 3: Timing metrics for the full test dataset (3 million jets) with GPU evaluations on an NVIDIA Titan X GPU. The nominal value denotes the mean of five independent trainings, while the error bar is the standard deviation.

4.3 Calibratability

While performance in simulation gives an important view of an algorithm's performance, ultimately its efficiency must be calibrated to data. This is done using control samples built with specific event selections for each flavour of jet and comparing the observed and simulated efficiency. This is especially challenging for light-flavour jets, as it is difficult to identify a highly pure sample of such jets after the b-tagging requirement.

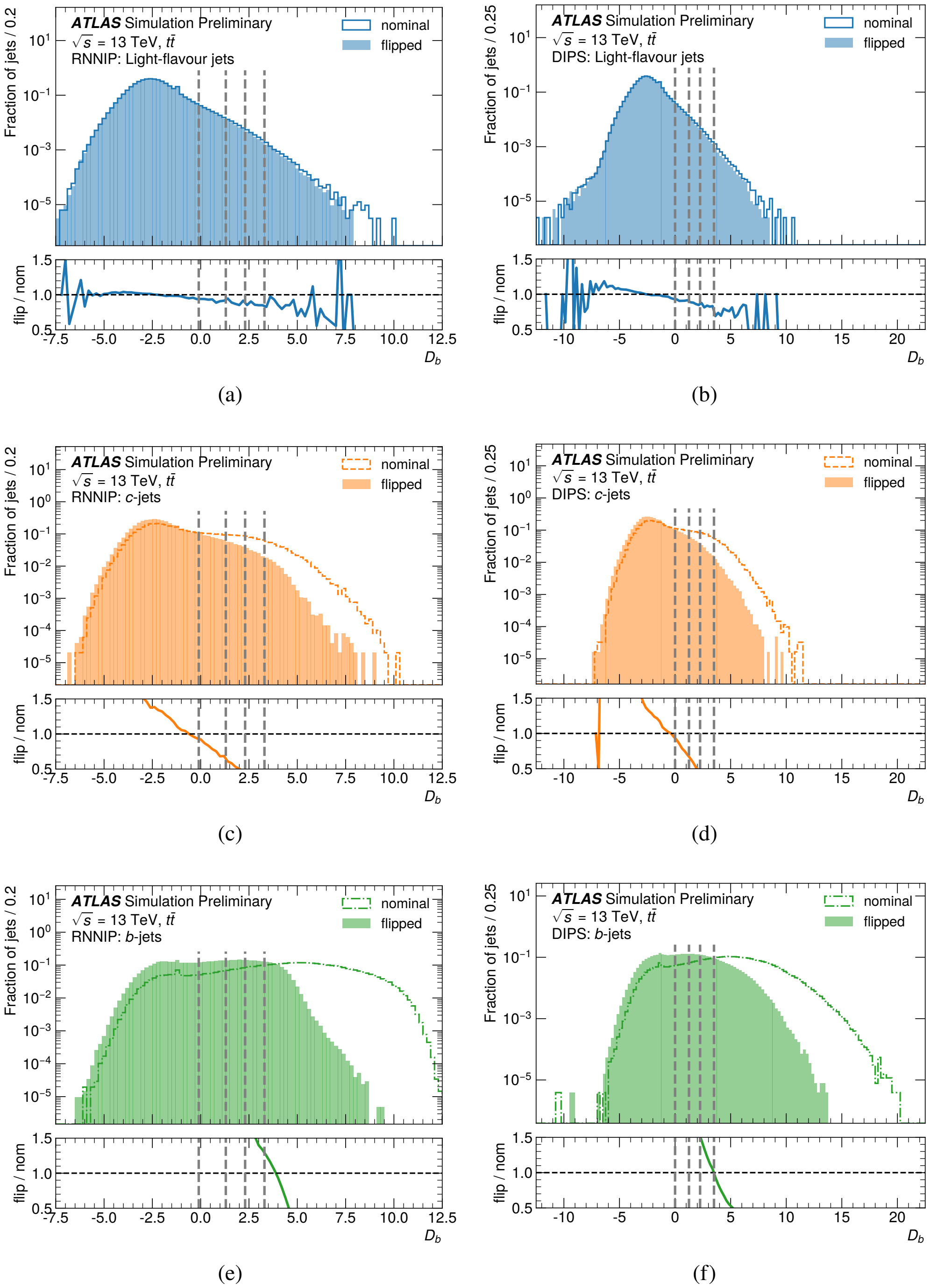
A large fraction of light-flavour jets are wrongly classified as b-jets due to tracks being on the tail of their IP distribution and are thus mismeasured. This effect is mostly coming from sources, such as detector resolution and pile-up collisions, which have equal probability for mismeasuring a track as having positive or negative lifetime sign, leading to mostly symmetric IP distributions (as seen in Figure 1). As such, a data augmentation procedure called *flipping* can be applied whereby the sign of track IPs (and that of secondary vertices) is multiplied by -1, without affecting the overall light-flavour jets IP distributions [39]. The tagger evaluated on flipped inputs, the *flipped* tagger, will then have an approximately equal performance in light-flavour jets as the nominal tagger. However, for b-jets and c-jets with real large IP tracks, the flipping will lead to large changes in their asymmetric IP distribution, with significantly fewer large IP tracks, causing the flipped tagger to be inefficient for identifying these jets. Therefore, applying a b-tagging requirement on the flipped tagger will generate a dataset with a higher fraction of light-flavour jets, when compared to the dataset built with the nominal tagger, such that the light-flavour jet efficiency can be obtained in data. In order for this to succeed, the b-tagging algorithms must uphold this approximate flipping symmetry of the light-flavour jets in their prediction, while reducing b-jets and c-jets tagging efficiencies.

The discriminant distributions of b-jets, c-jets and light-flavour jets with nominal and flipped inputs for the RNNIP and DIPS algorithms are shown in Figure 6. The dashed vertical lines represent the discriminant requirement for 85%, 77%, 70% and 60% inclusive b-jet efficiencies, corresponding to the efficiency benchmarks used at analysis level. The desired properties are found for both DIPS and RNNIP, the flipped distribution for light-flavour jets is nearly unchanged, while there is a significant decrease in flipped b-jets and c-jets at high discriminant values. Using these distributions, the efficiencies of the different jet flavours as a function of the RNNIP or DIPS discriminants can be examined, as in Figure 7. For both DIPS and RNNIP, one can see the large reduction on the efficiency for selecting b-jets and c-jets for a fixed light-flavour jet rejection as desired.

4.4 Track Selection Optimisation

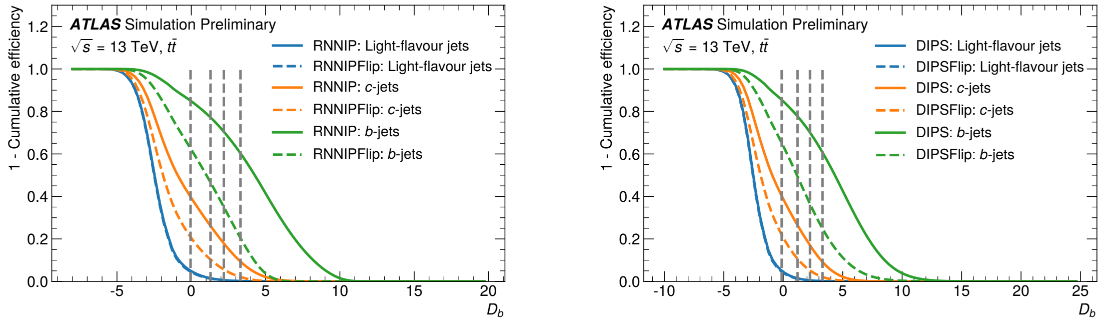
A major benefit of the reduced training time for DIPS is that it facilitates critical optimisation studies which require retraining the algorithm for each change one would like to examine. Two classes of optimisation are presented here: 1) varying the selection of tracks given to DIPS for processing, and 2) providing additional features per track.

The DIPS implementation described so far relies on the same track selection as the IP3D and RNNIP algorithms. This selection, denoted *nominal*, selects tracks with $p_{\mathrm{T}} > 1$ GeV, $|d_0| < 1$ mm, $|z_0 \sin\theta| < 1.5$ mm. This is a relatively strict selection that is used to keep the number mismeasured and pile-up tracks low, as the IP3D algorithm can be sensitive to such tracks. At the same time, this selection removes some of the key tracks from heavy flavour decays that are vital for classification. With the larger expressive power of the DIPS neural network over the IP3D model, DIPS will have more power to learn which tracks are useful for



(a) (b) (c) (d) (e) (f)

Figure 6: D_b discriminant distributions for the nominal and flipped taggers. The vertical dashed lines correspond to the discriminant requirements for 85%, 77%, 70% and 60% inclusive b-jet efficiencies, corresponding to the efficiency benchmarks used at analysis level. Plots (a), (c) and (e) refer to the RNNIP performance, while (b), (d) and (f) refer to DIPS. Plots (a) and (b), (c) and (d), (e) and (f) show light-flavour jets, c-jets and b-jets respectively.



(a)

(b)

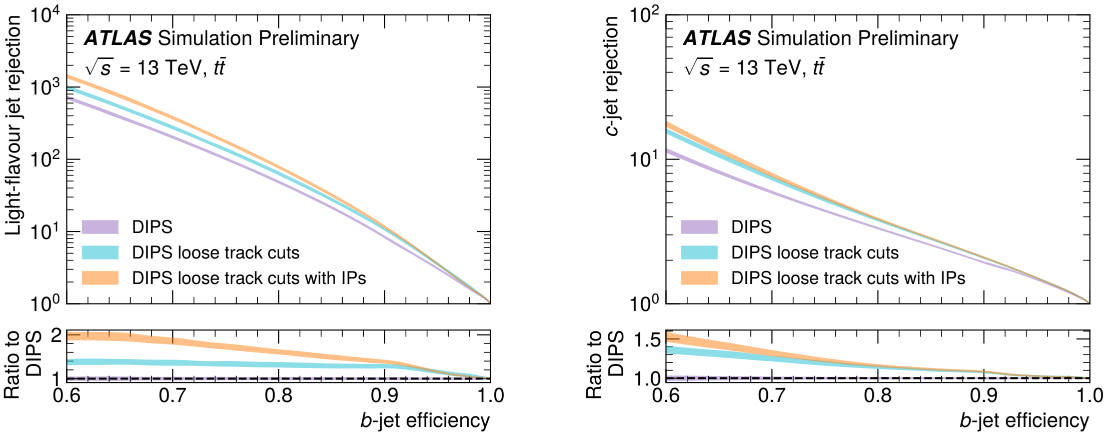
Figure 7: 1 - Cumulative efficiency as a function of b-tagging discriminant for RNNIP (a) and DIPS (b). In both cases, the performance remains nearly unchanged for light-flavour jets when comparing nominal and flipped taggers, while the b-jet and c-jet efficiencies drop.

tagging and thus will potentially be less sensitive to such tracks. As a result, a *loose* selection is examined, defined as $p_\mathrm{T} > 0.5$ GeV, $|d_0| < 3.5$ mm, $|z_0 \sin\theta| < 5$ mm, which utilises a lower p_T threshold and a wider allowance on the impact parameter thresholds in order to capture more tracks from the heavy flavour decay. In addition, DIPS with the *loose* selection examines up to the 25 highest s_{d_0} tracks, rather than 15 tracks as in the *nominal* selection, to further increase the ability to select tracks from heavy flavour decays.

The average number of tracks of different origin per jet is shown in Table 4 for the *nominal* and *loose* selections, and is shown separately per jet flavour. The total number of tracks (n_{trk}), the number of tracks from heavy flavour decays (n_{trk}^{HF}), the number of tracks from hadronisation, excluding those from heavy flavour decays (n_{trk}^{hadr}), and the number of tracks from mismeasurement, material interactions, and pile-up (n_{trk}^{other}), are compared. The *loose* selection increases the average number of tracks per jet from heavy flavour decay by ≈ 15% over the *nominal* selection. However, for all flavours, the *loose* selection also increases the number of fragmentation and other tracks per jet. As can be seen in the ROC curves in Figure 8, DIPS with the *loose* selection (shown in pink) outperforms the nominal DIPS (shown in purple) by up to ≈ 40% for light-flavour jet and charm jet rejection.

Jet Flavour	**Track selection**	n_{trk}	n_{trk}^{HF}	n_{trk}^{hadr}	n_{trk}^{other}
b-jets	*nominal*	5.9 ± 2.7	3.4 ± 1.8	2.0 ± 1.9	0.4 ± 0.8
	loose	8.1 ± 3.2	3.9 ± 1.8	2.5 ± 2.1	1.7 ± 1.7
c-jets	*nominal*	5.1 ± 2.5	1.7 ± 1.0	2.9 ± 2.2	0.4 ± 0.8
	loose	7.1 ± 3.1	1.8 ± 1.0	3.6 ± 2.4	1.7 ± 1.7
Light-flavour jets	*nominal*	4.6 ± 2.6	-	4.1 ± 2.5	0.5 ± 0.9
	loose	6.8 ± 3.3	-	5.0 ± 2.7	1.8 ± 2.0

Table 4: The average per jet total number of tracks (n_{trk}), the number of tracks from heavy flavour decays (n_{trk}^{HF}), the number of tracks from hadronisation, excluding those from heavy flavour decays (n_{trk}^{hadr}), and the number of tracks from mismeasurement, material interactions, and pile-up (n_{trk}^{other}), are shown for the *nominal* and *loose* selections for each jet flavour.



(a)

(b)

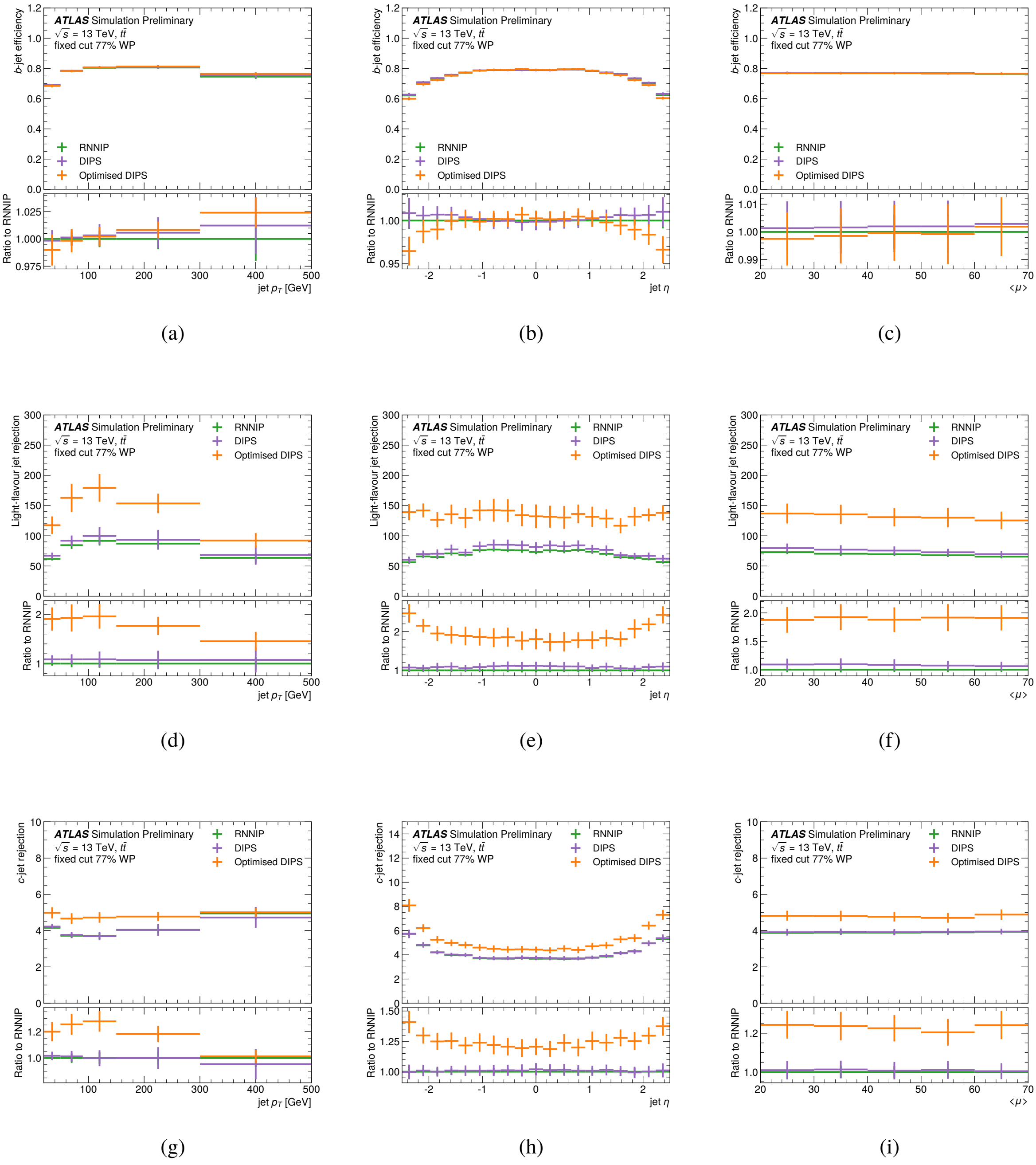
Figure 8: Light-flavour jet rejection as a function of b-jet efficiency (a) and c-jet rejection as a function of b-jet efficiency (b) of the nominal DIPS setup, DIPS with *loose* track selection, and Optimised DIPS with the *loose* track selection and additional IP inputs. The central curves and error bands show the mean and standard deviation, respectively, of the rejection at each b-jet efficiency for 5 trainings. The ratios are computed with respect to the DIPS ROC curve.

4.5 Optimised DIPS Performance

Beyond the *loose* selection, the impact of adding more per-track features is also examined, namely the impact parameters d_0 and $z_0 \sin\theta$. The DIPS with additional features and *loose* track selection, denoted *Optimised DIPS*, can be seen in orange in the ROC curves in Figure 8, compared to a reference of the nominal DIPS or RNNIP trainings, respectively. For the following studies, Optimised DIPS is built with the same architecture described in Section 3.3. The Optimised DIPS outperforms the nominal DIPS by up to a factor of 2 in light-flavour jet rejection and a factor of 1.5 in the c-jet rejection.

While ROC curves give a global view of an algorithm's performance, the behavior of the b-tagging efficiency and the background rejection as a function of key kinematic variables is also vital to performance within analyses. To explore this metric, a threshold defining an inclusive 77% b-tagging efficiency for each algorithm is determined, and the b-jet efficiency and background rejections with this fixed threshold are examined as a function of kinematic quantities. The b-jet efficiency as well as the c-jet and light-flavour jet rejections versus jet p_{T} and η are shown in Figure 9, for the RNNIP, DIPS, and Optimised DIPS algorithms. The behavior of DIPS and RNNIP are nearly the same across the p_{T} and η range, with DIPS providing a slightly higher light-flavour jet rejection. The Optimised DIPS delivers a factor of 1.5 to 2.5 in additional light-flavour jet rejection and up to ≈ 33% additional charm jet rejection. Loosening the track requirements for Optimised DIPS could potentially have the drawback of increasing the performance dependency on pile-up. We therefore check the b-jet efficiency, c-jet and light-flavour jet rejection as a function of the average number of proton-proton collisions per bunch crossing $\langle\mu\rangle$, also shown in Figure 9. The Optimised DIPS performance dependency on $\langle\mu\rangle$ is not found to be significantly stronger than the baseline DIPS or RNNIP.

One challenge in comparing background rejections with a fixed threshold is that the b-tagging efficiency is not the same for each algorithm in each kinematic region. As an alternative, the threshold on the



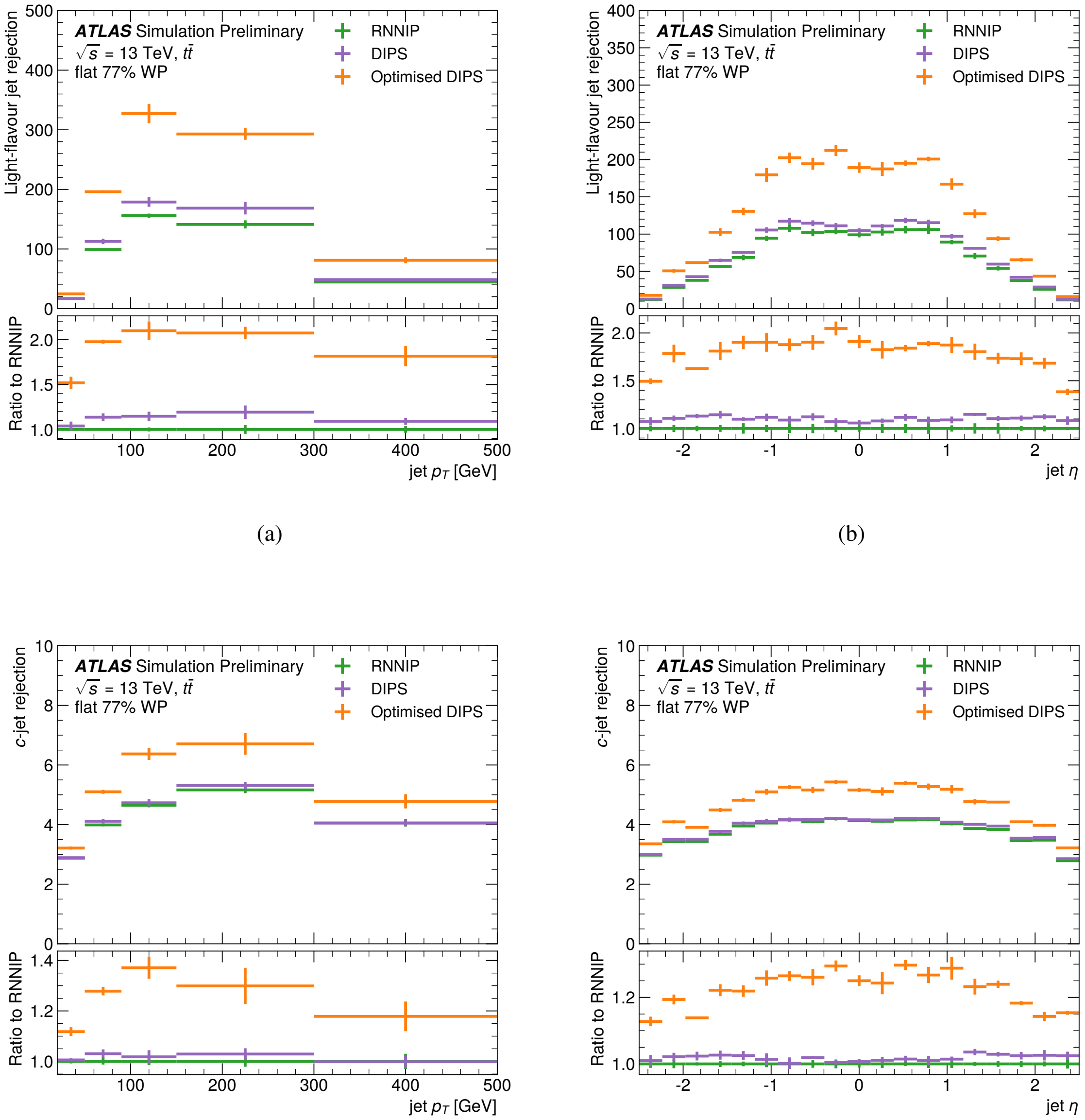
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 9: Performance plots using a fixed cut with 77% b-jet efficiency. Plots (a), (b) and (c) show the b-jet efficiency as a function of jet p_T, η and average number of proton-proton collisions per bunch crossing $\langle\mu\rangle$. Plots (d), (e) and (f) show the light-flavour rejection as a function of the same quantities, while plots (g), (h) and (i) show the c-jet rejection.

b-tagging discriminant can be tuned in each kinematic region to give a constant 77% *b*-tagging efficiency. A comparison of the *c*-jet and light-flavour jet rejections as a function of p_{T} and η for the DIPS, RNNIP, and Optimised DIPS algorithms with flat 77% *b*-tagging efficiency can be seen in Figure 10. While DIPS and RNNIP are seen to be quite similar, DIPS provides up to ≈ 20% additional light-flavour jet rejection in some regions of jet p_{T}. The Optimised DIPS shows more than a factor of 2 increase in light-flavour jet rejection and up to ≈ 50% additional charm jet rejection of the DIPS, for jets with p_{T} between 50 and 300 GeV.



(a)

(b)

(c)

(d)

Figure 10: Performance plots using a requirement where the b-jet efficiency is 77% in each bin. Plots (a) and (b) show the light-flavour rejection as a function of jet p_{T} and η, while plots (c) and (d) show the c-jet rejection as a function of the same quantities.

5 Conclusion

DIPS, a new algorithm for identifying heavy flavour jets with impact parameter information and based on the Deep Sets architecture, has been introduced and is shown to be comparable in performance and up to a factor of 3 to 5 faster to train and evaluate over the baseline recurrent neural network based algorithm RNNIP when using the same inputs. The large speed-up of the algorithm facilitates optimisation, and an optimised DIPS with loosened track selections and additional per-track features was shown to improve light-flavour jet rejection by up to a factor of 2.5 and c-jet rejection by up to a factor of 1.5 over the baseline DIPS algorithm, which already outperforms the current RNNIP algorithm by up to 15%. As such, DIPS represents a promising future direction for neural network-based flavour tagging algorithms. Moreover, the parallelisability and increased speed of DIPS not only has the potential to reduce the computational load of the ATLAS reconstruction, but also makes DIPS an excellent candidate for trigger applications where extremely low latency is required.

References

[1] ATLAS Collaboration, *Observation of $H \to b\bar{b}$ decays and VH production with the ATLAS detector*, Phys. Lett. B **786** (2018) 59, arXiv: 1808.08238 [hep-ex] (cit. on p. 2).

[2] ATLAS Collaboration, *Observation of Higgs boson production in association with a top quark pair at the LHC with the ATLAS detector*, Phys. Lett. B **784** (2018) 173, arXiv: 1806.00425 [hep-ex] (cit. on p. 2).

[3] ATLAS Collaboration, *Search for new resonances in mass distributions of jet pairs using 139 fb^{-1} of pp collisions at $\sqrt{s}$ = 13 TeV with the ATLAS detector*, JHEP **03** (2020) 145, arXiv: 1910.08447 [hep-ex] (cit. on p. 2).

[4] Tanabashi, M. et al, *Review of Particle Physics*, Phys. Rev. D **98** (2018) 030001 (cit. on p. 2).

[5] ATLAS Collaboration, *Identification of Jets Containing b-Hadrons with Recurrent Neural Networks at the ATLAS Experiment*, ATL-PHYS-PUB-2017-003, 2017, URL: https://cds.cern.ch/record/2255226 (cit. on pp. 2, 4, 5).

[6] M. Zaheer et al., *Deep Sets*, CoRR (2017), URL: http://arxiv.org/abs/1703.06114 (cit. on pp. 2, 5).

[7] P. T. Komiske, E. M. Metodiev, and J. Thaler, *Energy flow networks: Deep Sets for particle jets*, JHEP **01** (2019) 121, URL: http://dx.doi.org/10.1007/JHEP01(2019)121 (cit. on pp. 2, 5, 6).

[8] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3** (2008) S08003 (cit. on p. 2).

[9] ATLAS Collaboration, *ATLAS Insertable B-Layer Technical Design Report*, ATLAS-TDR-19, 2010, URL: https://cds.cern.ch/record/1291633 (cit. on p. 3).

[10] B. Abbott et al., *Production and integration of the ATLAS Insertable B-Layer*, JINST **13** (2018) T05008, arXiv: 1803.00844 [physics.ins-det] (cit. on p. 3).

[11] S. Frixione, P. Nason, and G. Ridolfi, *A positive-weight next-to-leading-order Monte Carlo for heavy flavour hadroproduction*, JHEP **09** (2007) 126, arXiv: 0707.3088 [hep-ph] (cit. on p. 3).

[12] P. Nason, *A new method for combining NLO QCD with shower Monte Carlo algorithms*, JHEP **11** (2004) 040, arXiv: hep-ph/0409146 (cit. on p. 3).

[13] S. Frixione, P. Nason, and C. Oleari, *Matching NLO QCD computations with parton shower simulations: the POWHEG method*, JHEP **11** (2007) 070, arXiv: 0709.2092 [hep-ph] (cit. on p. 3).

[14] S. Alioli, P. Nason, C. Oleari, and E. Re, *A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX*, JHEP **06** (2010) 043, arXiv: 1002.2581 [hep-ph] (cit. on p. 3).

[15] R. D. Ball et al., *Parton distributions for the LHC Run II*, JHEP **04** (2015) 040, arXiv: 1410.8849 [hep-ph] (cit. on p. 3).

[16] ATLAS Collaboration, *Studies on top-quark Monte Carlo modelling for Top2016*, ATL-PHYS-PUB-2016-020, 2016, URL: https://cds.cern.ch/record/2216168 (cit. on p. 3).

[17] T. Sjöstrand et al., *An introduction to PYTHIA 8.2*, Comput. Phys. Commun. **191** (2015) 159, arXiv: 1410.3012 [hep-ph] (cit. on p. 3).

[18] ATLAS Collaboration, *ATLAS Pythia 8 tunes to* 7 *TeV data*, ATL-PHYS-PUB-2014-021, 2014, URL: https://cds.cern.ch/record/1966419 (cit. on p. 3).

[19] R. D. Ball et al., *Parton distributions with LHC data*, Nucl. Phys. B **867** (2013) 244, arXiv: 1207.1303 [hep-ph] (cit. on p. 3).

[20] D. J. Lange, *The EvtGen particle decay simulation package*, Nucl. Instrum. Meth. A **462** (2001) 152 (cit. on p. 3).

[21] ATLAS Collaboration, *The ATLAS Simulation Infrastructure*, Eur. Phys. J. C **70** (2010) 823, arXiv: 1005.4568 [physics.ins-det] (cit. on p. 3).

[22] S. Agostinelli et al., *Geant4 – a simulation toolkit*, Nucl. Instrum. Meth. A **506** (2003) 250 (cit. on p. 3).

[23] ATLAS Collaboration, *Jet reconstruction and performance using particle flow with the ATLAS Detector*, Eur. Phys. J. C **77** (2017) 466, arXiv: 1703.10485 [hep-ex] (cit. on p. 3).

[24] M. Cacciari, G. P. Salam, and G. Soyez, *The anti-k_t jet clustering algorithm*, JHEP **04** (2008) 063, arXiv: 0802.1189 [hep-ph] (cit. on p. 3).

[25] ATLAS Collaboration, *Jet energy scale measurements and their systematic uncertainties in proton–proton collisions at $\sqrt{s}$ = 13 TeV with the ATLAS detector*, Phys. Rev. D **96** (2017) 072002, arXiv: 1703.09665 [hep-ex] (cit. on p. 3).

[26] ATLAS Collaboration, *Tagging and suppression of pileup jets with the ATLAS detector*, ATLAS-CONF-2014-018, 2014, URL: https://cds.cern.ch/record/1700870 (cit. on p. 3).

[27] ATLAS Collaboration, *Performance of b-jet identification in the ATLAS experiment*, JINST **11** (2016) P04008, arXiv: 1512.01094 [hep-ex] (cit. on p. 3).

[28] ATLAS Collaboration, *ATLAS b-jet identification performance and efficiency measurement with $t\bar{t}$ events in pp collisions at $\sqrt{s}$ = 13 TeV*, (2019), arXiv: 1907.05120 [hep-ex] (cit. on p. 4).

[29] S. Hochreiter and J. Schmidhuber, *Long Short-Term Memory*, Neural Computation **9 8** (1997) (cit. on p. 4).

[30] P. W. Battaglia et al., *Relational inductive biases, deep learning, and graph networks*, CoRR (2018), URL: http://arxiv.org/abs/1806.01261 (cit. on p. 5).

[31] ATLAS Collaboration, *A neural network clustering algorithm for the ATLAS silicon pixel detector*, JINST **9** (2014) P09009, arXiv: 1406.7690 [hep-ex] (cit. on p. 6).

[32] S. Ioffe and C. Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, CoRR (2015), URL: http://arxiv.org/abs/1502.03167 (cit. on p. 6).

[33] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, *Improving neural networks by preventing co-adaptation of feature detectors*, CoRR (2012), URL: http://arxiv.org/abs/1207.0580 (cit. on p. 6).

[34] F. Chollet et al., *Keras*, https://keras.io, 2015 (cit. on p. 7).

[35] M. Abadi et al., *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, Software available from tensorflow.org, 2015, URL: https://www.tensorflow.org/ (cit. on p. 7).

[36] D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, 2014, arXiv: 1412.6980 [cs.LG] (cit. on p. 7).

[37] B. Lakshminarayanan, A. Pritzel, and C. Blundell, *Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles*, 2016, arXiv: 1612.01474 [stat.ML] (cit. on p. 8).

[38] K. Simonyan, A. Vedaldi, and A. Zisserman, *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*, 2013, arXiv: 1312.6034 [cs.CV] (cit. on p. 9).

[39] ATLAS Collaboration, *Calibration of light-flavour b-jet mistagging rates using ATLAS proton–proton collision data at* $\sqrt{s} = 13$ *TeV*, ATLAS-CONF-2018-006, 2018, URL: https://cds.cern.ch/record/2314418 (cit. on p. 11).