Development and Application of Displaced Vertices Identification Methods Using Simulated Open Data of the ATLAS Experiment

Alexandros Tsagkaropoulos[1], Dimitris Fassouliotis[2], Stelios Angelidakis[3]

Athens, August 2022

[1] Department of Physics, National and Kapodistrian University of Athens, Bachelor Student, e-mail: sph1900190@uoa.gr
[2] Department of Physics, National and Kapodistrian University of Athens, Professor, Supervisor, e-mail: dfassoul@phys.uoa.gr
[3] Department of Physics, National and Kapodistrian University of Athens, Postdoc Researcher, Supervisor, e-mail: stylianos.angelidakis@cern.ch

Abstract

The identification of Displaced Vertices in ATLAS experiment is fundamental in verification of Beyond Standard Model Theories. The attempt to search for this kind of particles by a specific Algorithm is intended to compare the success of computer simulation to human input, which could provide viable information about changes that could be implemented in future reform of the Algorithm.

The paper focuses on the breakdown of the Algorithm in small parts to allow the interpretation of each function and the conditions used to get the final results. The implementation of it borrows concepts from Analytical Geometry, that are referenced in the main body and, whenever its is necessary, proofs are provided. Additionally, the values of the limits imposed in the Algorithm are backed up by information derived from processing of the data set and comments on the expected results it ought to have on the majority of data sets that could be analysed. Furthermore, its relative success is measured through some indexes defined in the Section 1, which are common in data from human input and the Algorithm's results.

The results of the Algorithm are superior than the ones aggregated by users' input in every aspect. Specifically, when the Algorithm manages to reconstruct equal or lower number than the Displaced Vertices that appear in an event, it is more than 99% accurate. Of course, this is not always the case, so the efficiency on the totality of events is approximately 80%. The decrease in efficiency is due to the exceeding number of Displaced Vertices that the Algorithm identifies in relation to the real number of them. Therefore, while the Algorithm provides finer results than the human input, there is room for improvement.

1 Introduction

1.1 Definitions

Primarily, there is a need to state few definitions so as the analysis in Section 2 becomes clear and concise.

Definition 1. The **Interaction Points** (IPs) are the points alongside LHC circumference where beams of protons collide and a detector is located.

In this paper, the IP examined is in the centre of ATLAS detector, so the name of the detector will be omitted.

The high energy proton collisions on the IP create a multitude of new particles that move outwards in all directions. The stable ones are detected by ATLAS. A part of the particles detected are assumed to be long-lived particles.

Definition 2. The **long-lived particles** are particles with lifetimes greater than the known Standard Model ones.

Due to their great lifetime, long lived particles decay several millimetres or centimetres away from the IP.

Definition 3. The decay points of long-lived particles are called **Displaced Vertices** (DVs), because of their distance from the IP.

In terms of the Algorithm's procedure, there are two types of DVs, called DV_{true} and DV_{reco}.

Definition 4. A $\mathbf{DV_{true}}$ is defined as a real DV that appears in data set's elements.

Definition 5. A $\mathbf{DV_{reco}}$[1] is defined as a DV that is reconstructed by the Algorithm exploiting data set's information about trajectories points.

In order to measure the proximity of a DV_{reco} and its corresponding DV_{true} the concept of "error" emerged.

Definition 6. The **error** of a DV_{reco} is called the distance between it and the corresponding DV_{true}.

By Definition 6 is obvious that errors cannot be calculated for all DV_{reco} in events where they outnumber DV_{true}. Also, the error limits that are imposed on users are used as a frame of reference and are:

[1] The index "reco" stands for reconstructed from the data set, without using any direct information about the DV_{true}.

- sz space: 35 mm,
- xy plane: 14 mm.

1.2 Goals of Project

Additionally, to provide further clarity about what would follow, the goals of the project are outlined:

1. Development of Algorithm that searches for and identifies $\mathrm{DV_{true}}$. For this quest the following indexes have been defined to quantify the results:

    - **Efficiency:** ratio of $\mathrm{DV_{true}}$, that are Matched[2] to a $\mathrm{DV_{reco}}$, to the total number of $\mathrm{DV_{true}}$.
    - **Purity:** ratio of Matched $\mathrm{DV_{reco}}$, to the total number of $\mathrm{DV_{reco}}$.
    - **Accuracy:** ratio of Matched $\mathrm{DV_{reco}}$, to the total number of $\mathrm{DV_{reco}}$, for which an error is calculated.

    Also, the histograms depicted in Figures on Section 3 aim to further enhance the understanding of results by visualising them.

2. Comparison of the Algorithm's results with those collected by human input.

1.3 Data Characteristics

The data set used for both the Algorithm and users' attempts contains computer simulated events and can be found on [1]. The reason for that is to measure with absolute certainty how close to the ideal comes the Algorithmic approach and human input. Furthermore, additional information about the events is provided below:

- Number of events: 4300.
    - Number of events with one $\mathrm{DV_{true}}$: 3359.
    - Number of events with two $\mathrm{DV_{true}}$: 934.
    - Number of events with three $\mathrm{DV_{true}}$: 7.
- Any other particle that decays to Standard Model ones, excluding long-lived particles, is eliminated.
- Every event includes at least one $\mathrm{DV_{true}}$.

Also, it is of profound importance to state the elements from the data set which are manipulated by the Algorithm. While the data set contains information about every aspect of events ($\mathrm{DV_{true}}$ position, $\mathrm{DV_{true}}$ number, number of trajectories, etc.) the Algorithm uses only the two given points for each trajectory to reconstruct a $\mathrm{DV_{reco}}$. Namely, the first and the last point of each trajectory that define a line segment.

[2]The word "Matched" refers to a $\mathrm{DV_{reco}}$ that respects both sz space and xy plane error's limits. On the other hand, the phrase "Not Matched" refers to a $\mathrm{DV_{reco}}$ that does not respects at least one of the error's limits. Also, $\mathrm{DV_{reco}}$ for which an error cannot be computed are placed in the last category.

2 Processing of Events

2.1 Distance Between Two Trajectories

It is impossible for two or more trajectories to converge perfectly to a single point, which would be the $\mathrm{DV_{true}}$. Thus, in order to decide if a couple or more trajectories came from a long-lived particle it is needed to compute the "distance" between them.

Definition 7. The **"distance"** between two trajectories is defined as the minimum distance between a point of the first one and the second trajectory.

Generally, let two lines ε_i and ε_j for which the only information given are two points, P_i, P_i' and P_j, P_j', that lie on each, respectively, as shown in Figure 1.

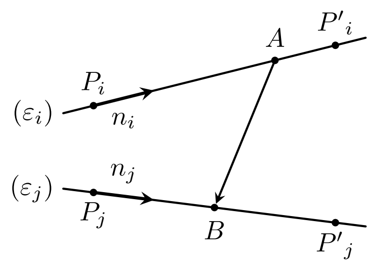


Figure 1: "Distance" between two straight lines

Also, let A and B be points on line ε_i and ε_j, respectively. If the length of the vector **AB** is the "distance" between the two lines and O stands for the IP, then the following equations hold true:

$$\mathbf{OA} = \mathbf{r_i} + \frac{\mathbf{u}\cdot(\mathbf{n_j}\times\mathbf{r_o})}{\|\mathbf{u}\|^2}\,\mathbf{n_i}, \tag{1a}$$

$$\mathbf{OB} = \mathbf{r_j} + \frac{\mathbf{u}\cdot(\mathbf{n_i}\times\mathbf{r_o})}{\|\mathbf{u}\|^2}\,\mathbf{n_j}, \tag{1b}$$

$$\mathbf{n_i}\equiv\mathbf{r_i}'-\mathbf{r_i},\ \mathbf{n_j}\equiv\mathbf{r_j}'-\mathbf{r_j},\ \mathbf{u}\equiv\mathbf{n_j}\times\mathbf{n_i},\ \mathbf{r_o}\equiv\mathbf{r_j}-\mathbf{r_i}.$$

To save space, vectors $\mathbf{OP_i}$, $\mathbf{OP_i}'$ and $\mathbf{OP_j}$, $\mathbf{OP_j}'$ are represented by $\mathbf{r_i}$, $\mathbf{r_i}'$ and $\mathbf{r_j}$, $\mathbf{r_j}'$, respectively. In addition, the vector **AB** is called "distance" vector and has fundamental role in identification of DVs.

The extensive proof of equation (1a) and (1b) can be found in Appendix A.

2.2 Conditions to Choose $\mathrm{DV_{reco}}$

The $\mathrm{DV_{reco}}$, which is constructed by two trajectories, is defined as the middle point of their "distance" vector. Expressed in mathematics, the vector $\mathbf{DV_{reco}}$ that connects the IP with the $\mathrm{DV_{reco}}$ is:

$$\mathbf{DV_{reco}} = \frac{1}{2}\left(\mathbf{OA}+\mathbf{OB}\right).$$

The implementation of the function that takes as arguments the two given points for every trajectory and returns the coordinates of the $\mathrm{DV_{reco}}$ is displayed in pseudocode[3]:

[3]Capital letters are used for arrays and lowercase letters for unidimensional values. Also, the two dot convention symbolises range.

Reconstructed-Displaced-Vertex(R_i, $R_i{}'$, R_j, $R_j{}'$)

Let $N_i[1\,..\,3]$ and $N_j[1\,..\,3]$ be new arrays
Let $R_o[1\,..\,3]$ and $U[1\,..\,3]$ be new arrays
for $k = 1$ **to** 3 **do**
    $N_i[k] = R_i{}'[k] - R_i[k]$
    $N_j[k] = R_j{}'[k] - R_j[k]$
    $R_o[k] = R_j[k] - R_i[k]$
U = Cross-Product(N_j, N_i)
t_o = Triple-Product(U, N_j, R_o)/Norm(U)^2
s_o = Triple-Product(U, N_i, R_o)/Norm(U)^2
Let $OA[1\,..\,3]$ and $OB[1\,..\,3]$ be new arrays
for $k = 1$ **to** 3 **do**
    $OA[k] = R_i[k] + t_o {*} N_i[k]$
    $OB[k] = R_j[k] + s_o {*} N_j[k]$
Let DV$[1\,..\,3]$ be new array
for $k = 1$ **to** 3 **do**
    $DV[k] = 0.5{*}(OA[k] + OB[k])$
return DV

The functions used in the procedure Reconstructed-Displaced-Vertex have the following uses:

- The Cross-Product(A_1, A_2) returns a pointer to an array which elements are the result of the cross product: $\mathbf{a_1} \times \mathbf{a_2}$.
- Triple-Product(A_1, A_2, A_3) returns a pointer to an array which elements are the result of the triple vector product: $\mathbf{a_1} \cdot (\mathbf{a_2} \times \mathbf{a_3})$.

Moreover, the trajectories whose given points are arguments in the Reconstructed-Displaced-Vertex function are of central importance. The following definition assigns to them a name.

Definition 8. The two trajectories utilised by the Algorithm so as to reconstruct a $\mathrm{DV_{reco}}$ are called **reconstructing trajectories**.

The $\mathrm{DV_{reco}}$ must satisfy three conditions in order not to get rejected:

1. The length of the "distance" vector must not be larger than `DVCut`[4] –equivalently, the "distance" of reconstructing trajectories must not be larger than `DVCut`.).
2. The two angles that are formed by connecting the $\mathrm{DV_{reco}}$ with the given points from reconstructing trajectories must not be larger than `thetaRel_max`.

   Relative-Angles(Dv, R_i, $R_i{}'$, R_j, $R_j{}'$)

   Let Theta$[1\,..\,2]$ be new array
   Let Dv$R_i[1\,..\,3]$ and Dv$R_i{}'[1\,..\,3]$ be new arrays
   Let Dv$R_j[1\,..\,3]$ and Dv$R_j{}'[1\,..\,3]$ be new arrays
   for $k = 1$ **to** 3 **do**
       Dv$R_i[k] = R_i[k] -$ Dv$[k]$
       Dv$R_i{}'[k] = R_i{}'[k] -$ Dv$[k]$
       Dv$R_j[k] = R_j[k] -$ Dv$[k]$
       Dv$R_j{}'[k] = R_j{}'[k] -$ Dv$[k]$
   dotProd1 = dotProduct(Dv$R_i{}'$, DvR_i)
   dotProd2 = dotProduct(Dv$R_j{}'$, DvR_j)
   Theta$[0]$ = acos(dotProd1)$*180/\pi$
   Theta$[1]$ = acos(dotProd2)$*180/\pi$
   return Theta

   The implementation of the function for the application of this condition is displayed in pseudocode. The function Relative-Angles takes as arguments the coordinates of the $\mathrm{DV_{reco}}$, in the array Dv, and coordinates of the four given points of the reconstructing trajectories, in the arrays R_i, $R_i{}'$, R_j, $R_j{}'$, and returns an array Theta containing the relative angles, shown in Figure 2.

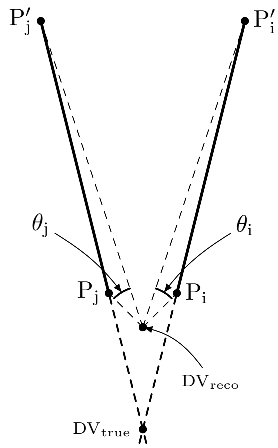


Figure 2: Relative angles between $\mathrm{DV_{reco}}$ and given points of reconstructing trajectories.

3. The distance from IP to $\mathrm{DV_{reco}}$ must be smaller than distance from IP to any of the reconstructing trajectories' given points.

2.3 Multiple Trajectories

Whilst a $\mathrm{DV_{reco}}$ is constructed using two trajectories, its product particles might be more that two. Thus, it is needed to take into account multiple trajectories that may converge to a single $\mathrm{DV_{reco}}$.

A way of deciding, despite the reconstructing trajectories, if another trajectory belongs to it, is to calculate the distance d_i between the i-th trajectory and the $\mathrm{DV_{reco}}$.

[4] There have been a remark from Researcher Stelios Vourakis to apply an exponential decline rule to `DVCut`, since it is expected the number of $\mathrm{DV_{true}}$ to fall exponentially as plural they are in an event. While this additional condition improves the results, by eliminating several $\mathrm{DV_{reco}}$ that exceed the number of $\mathrm{DV_{true}}$, it was rejected as "biased" condition.

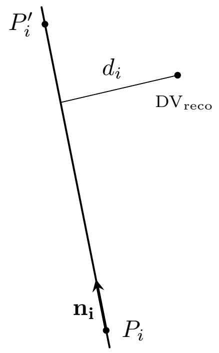


Figure 3: Distance between a $\mathrm{DV}_{\mathrm{reco}}$ and the corresponding straight line to a trajetory, excluding the reconstructing ones.

As seen in Figure 3 the distance d_i is the projection of the vector $\mathbf{r_i}$ to a vector perpendicular to the trajectory and can be though as:

$$d_i = \frac{\|(\mathbf{DV_{reco}} - \mathbf{n_i}) \times \mathbf{n_i}\|}{\|\mathbf{n_i}\|},$$

where $\mathbf{n_i} = \mathbf{r_i}' - \mathbf{r_i}$.

For the i-th trajectory to be included with the first two that reconstructed the $\mathrm{DV}_{\mathrm{reco}}$ the distance d_i must be less or equal to `TrajectoryCut = DVCut/2`.

2.4 Trajectories that Have Been Used

In order to exclude trajectories used either for constructing a $\mathrm{DV}_{\mathrm{reco}}$ or due to belonging to it, indexes are matched to each of them and saved in an array. Of course, the indexes refresh in every event studied by the Algorithm.

Index-Used(usedLineIndex, index)

used = FALSE
for $k = 0$ **to** $k =$ usedLineIndex.length **do**
  if index == usedLineIndex[k] **then**
    used = TRUE
    break
return used

The implementation of the procedure that takes as arguments the array usedLineIndex, containing the indexes used, and an index and returns `TRUE` if the index is used or `FALSE` if it was not is shown above.

2.5 Error Calculation

Certainly, it is not trivial to find out to which $\mathrm{DV}_{\mathrm{true}}$ each one of the $\mathrm{DV}_{\mathrm{reco}}$ found corresponds to. Therefore, for every reconstructed $\mathrm{DV}_{\mathrm{reco}}$ the following method is implemented:

- Every probable error between the $\mathrm{DV}_{\mathrm{reco}}$ and $\mathrm{DV}_{\mathrm{true}}$ is calculated, using every $\mathrm{DV}_{\mathrm{true}}$ that remains not matched to any $\mathrm{DV}_{\mathrm{reco}}$.

- The representative error for the $\mathrm{DV}_{\mathrm{reco}}$ is the least of all errors calculated.

- The $\mathrm{DV}_{\mathrm{true}}$ used to produced the representative error is saved as index in usedErrorIndex array. In this way, used $\mathrm{DV}_{\mathrm{true}}$ are excluded for further error calculation[5], if another $\mathrm{DV}_{\mathrm{reco}}$ arises.

3 Results and Evaluation

3.1 Results

The results for indexes mentioned in Introduction are shown in Output 1. Efficiency and Purity on xy plane and sz space refer to Matched $\mathrm{DV}_{\mathrm{true}}$ and $\mathrm{DV}_{\mathrm{reco}}$ only in the corresponding space and plane, respectively.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Efficiency:

A_xy: 0.8153    Total
A_xy: 0.8180    One DVtrue
A_xy: 0.8110    Two DVtrue

A_sz: 0.8290    Total
A_sz: 0.8273    One DVtrue
A_sz: 0.8330    Two DVtrue
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Purity:

Pu_xy: 0.6998   Total
Pu_xy: 0.6488   One DVtrue
Pu_xy: 0.8132   Two DVtrue

Pu_sz: 0.7116   Total
Pu_sz: 0.6561   One DVtrue
Pu_sz: 0.8352   Two DVtrue
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Accuracy:

Ac: 0.9995      Total
Ac: 0.9993      One DVtrue
Ac: 1.0000      Two DVtrue
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Time taken: 0.97s
```

Output 1: Results on results.txt file when the Algorithms halts.

The next pages are dedicated to histograms produced by the Algorithm so as to enhance the comprehension of its results. Particularly, the reasons this specific histograms were selected are analysed Figure by Figure:

Figure 4 The errors are separated in two categories: one concerning errors in sz space (polar coordinates) and the other concerning xy plane (cartesian coordinates). The histograms that refer to the first are located in the first row of the Figure and the second on the second row. This separation in categories is intended to make more approachable the comparison of human input and the results from the algorithm. Specifically, the user interface [2] provides

[5] In order to exclude $\mathrm{DV}_{\mathrm{true}}$ that are matched to a $\mathrm{DV}_{\mathrm{reco}}$ the procedure Index-Used(usedErrorIndex, errorIndex) is called.

the user with two classification options. One in xy plane (transversal view) and one in sz space (longitudinal view). Thus, the data collected from the users are separated in the same categories as the histograms in the Figure are. Furthermore, the error boundary in sz space is $35\,\text{mm}$ and in xy plane is $14\,\text{mm}$, since user input in considered correct only if their DV_{reco} approximation ranges within those limits.

Figure 5 The minimum distance from the DV_{reco} to the closest given reconstructing tajectory's point is a measure of how close the DV_{reco} is to them. Due to the fact that the implementation of the Algorithm uses the distance between lines and not line segments (where trajectories subsume) to reconstruct a DV_{reco}, the distance on the graph provides significant information. If this distance is large it means that the DV_{reco} is far from the beginning of its reconstructing trajectories, so it must be a "false positive". Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 6 The histograms concerning the distance from DV_{reco} to another trajectory, excluding the reconstructing ones, have been displayed so as to decide if `TrajectoryCut`'s value have to change and how. One caveat is that its value must not be too large. If that was the case, the Algorithm would assign a lot of trajectories in a single DV_{reco} preventing the formation of others. Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 7 The two histograms display a difference in two distances that is a measure of how much closer is the DV_{reco} to the IP than the first point from each trajectory that have been used to reconstruct it. A condition have been in applied, as mentioned in Subsection 2.2, that forbids the formation of a DV_{reco} that is further from the IP than the first trajectory points that reconstruct it. Consequently, the quantity $R_{\text{min}} - R_{\text{DV}_{\text{reco}}} \geq 0$. The suggestion of this condition seemed logically correct, because, excluding the minor cases of a missed backward hit, it cannot be broken in the real world. Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 8 The distribution of the distances between the trajectories used to reconstruct a DV_{reco} provides significant information about the data. Specifically, it is expected that products of a DV_{true} travel in trajectories that converge to it. So, ideally, their distance must be zero. Of course, there would be a variance around it, but it cannot be very large. The form of the histograms should tell where `DVCut` value should be placed so as the range of distances that is defined by it to contain a fair amount of entries. Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 9 The aim of two histograms is to match the behaviour of DV_{reco} to the expected behaviour of DV_{true}. It is fair to argue that the probability for the angle displayed in the histograms to be either $0°$ or $180°$ is close to impossible. On the other hand, the most probable angles would be closer to the $90°$ and have symmetry line on this value. Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 10 The histograms have been printed so as to verify the obvious idea that the maximum relative angle ought to be close to $0°$. As it can be seen in Figure 2, the closer DV_{reco} gets to DV_{true} the more the relative angles will decrease. In the ideal case that the DV_{reco} coincides with DV_{true}, the relative angle would be $0°$. Of course, the distribution of the relative angles would suggest a value for `ThetaRel_Max`, so as angles to be restricted to low values. Also, the histograms are divided in two categories: Matched DV_{reco} and not Matched DV_{reco}.

Figure 11 There are four rows and three columns which contain twelve figures in total. Specifically, for the rows the following rule is applied:

Row 1: refers to the total number of DV_{true}

Row 2: refers to the total number of DV_{reco}

Row 3: refers to the number of Matched DV_{reco}.

Row 4: refers to the number of Not Matched DV_{reco}.

Additionally, for the columns the following rule applies:

Column 1: refers to the total number of DVs.

Column 2: refers to the events with one DV_{true}.

Column 3: refers to events with two DV_{true}.

The reason they have been stacked this way is to compare the DV_{reco} numbers with DV_{true} and outline in which cases the exceeding or subceeding number of DV_{reco} emerges. Also, the next Figure is intended to clarify further the distribution of DV_{reco}.

Figure 12 The relative number of DV_{reco} with respect the number of DV_{true} provides explicit information about the distribution of the DV_{true}'s number. Namely, asymmetries in DV_{reco} numbers can be observed and corrected by tweaking the conditions that choose if a DV_{reco} is acceptable or not. The ideal distribution would be to observe only zero values. Of course, there are events where DV_{reco} outnumber DV_{true} or the contrary, in which the difference on the histogram would be negative or positive, respectively. So, as narrow the distribution is around zero, the more precise the results would be.

Figure 13 The distance between DV_{true} in events with two DV_{true} is a measure of how close the trajectories of its product particles are. The results of this histogram would provide information about the condition `TrajectoryCut` and how its value would affect the reconstruction of an additional DV_{reco} in events with more that one DV_{true}. Specifically, if the value of `TrajectoryCut` would be large relative to the distance between two DV_{true}, then it would restrict the reconstruction of multiple DV_{true}. So, an upper bound for the value of `TrajectoryCut` would be provided by the mean of the histogram.

3.2 Evaluation and Comments

While indexes contain the most important information about the success of the Algorithm to deliver the desired results, Figures are intentionally commented first, so as to review every piece of information the can provide. Thus, the evaluation of the indexes' values would be explicit and complete.

3.2.1 Figures

Starting from the Figure 4 and continuing with ascending order the comments on each Figure are the following:

Figure 4 The error distributions in sz space is greater bounded from the boundaries set in histograms than the ones in xy plane. This observation can be computed analytically through taking into account the overflow in each histogram and comparing the different categories between them. While this behaviour could reveal an incompetence of the Algorithm, it might recommend a change in xy plane boundary so as to have approximately the same overflow in both cases.

Figure 5 It is observed that both histograms have similar form, but the one on the Figure 5a has greater standard deviation than the on on the Figure 5b. This observation is declarative of the fact that the implementation of the code does not disturb the DV_{reco} reconstruction. Thus, there is no need for an additional condition to be applied in the reconstruction of the DV_{reco}.

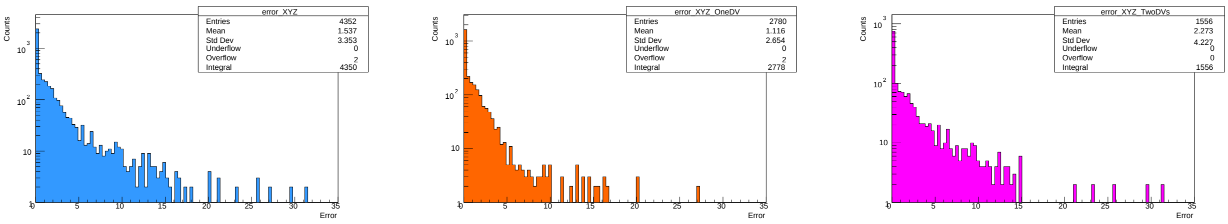
Figure 6 From histograms one can extract the information that in almost every event there is a additional track very close to DV_{reco}. Furthermore, the fact that the distribution is narrower in Non Matched DV_{reco} seems odd, since it is expected that not matched DV_{reco} would be further from any "exciting behaviour". Certainly, the fact that the distribution in narrower it a sign that the Non Matched DV_{reco} are concentrated closer to the IP, where the vast majority of trajectories emerge (precisely their linear expansion). So, if an additional condition would be applied, so as to prohibit DV_{reco} reconstruction close to the IP , the number of the Non Matched DV_{reco} might fell. Of course, one ought to be careful with this condition as it might interfere with Matched DV_{reco} as well. This condition has not been applied since the difference in standard deviation of the two distributions is approximately the same. Finally, due to the previous remark, there is no way of eliminating Not Matched DV_{reco} through the value of `TrajectoryCut`. The wiser selection for its value seemed the `DVCut/2`, because, if there are more trajectories than the two used to reconstruct the DV_{reco}, that belong to it, they would be in the same distance or closer to it than the second ones.

Figure 7 The distribution of the two histograms are identical. This behaviour suggests that there is no condition concerning the distance of DV_{reco} from the IP that can be applied and reduce the number of Non Matched DV_{reco}.

Figure 8 While the histograms in this Figure are drawn after the application of the value `DVCut = 0.2`, it can be seen that it manages to display the desirable results. It can be argued that an even lower value, such that `DVCut = 0.1`, would have similar results, but it was preferable to be a bit loose in this metric and exclude any "false positive" through other conditions.

Figure 9 Whereas the expected low probability to values 0° and 180° is observed in the histograms, there is a slight asymmetry left and right of the value 90°. Particularly, the angles less than 90° are more probable to be seen than the ones larger than 90°. There is not any effect that, taken into account, can explain this behaviour.

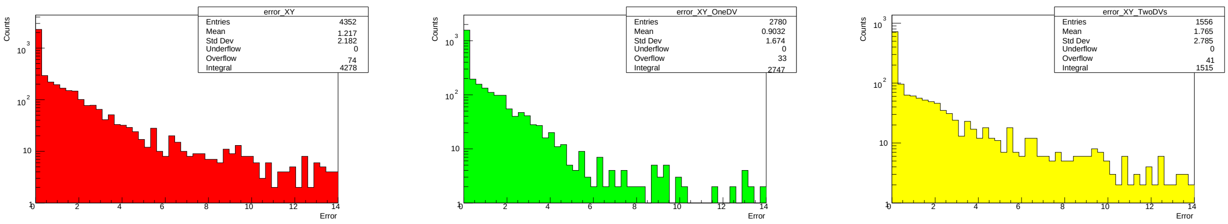
Figure 10 The histograms drawn have already set the value of `ThetaRel_Max = 90`. It is actually the largest angle that would produce a acceptable DV_{reco}. Surely, despite the relatively large range of angles allowed by the condition, the majority of histogram's entries is concentrated in the range 0° to 20°, in both Matched and Not Matched DV_{reco}. Additionally, the distribution form is quite similar in both histograms, so `ThetaRel_Max` cannot be confined to exclude Not Matched DV_{reco}.



(a) Errors in sz space

(b) Errors in sz space for events with one DV_{reco}

(c) Errors in sz space for events with two DV_{reco}

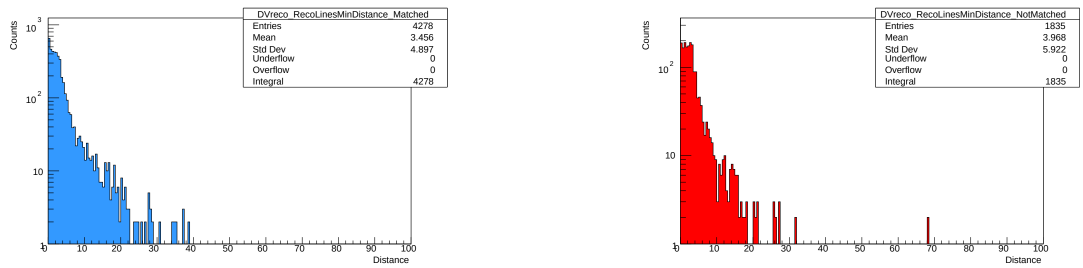


(d) Errors in xy plane

(e) Errors in xy plane for events with one DV_{reco}

(f) Errors in xy plane for events with two DV_{reco}

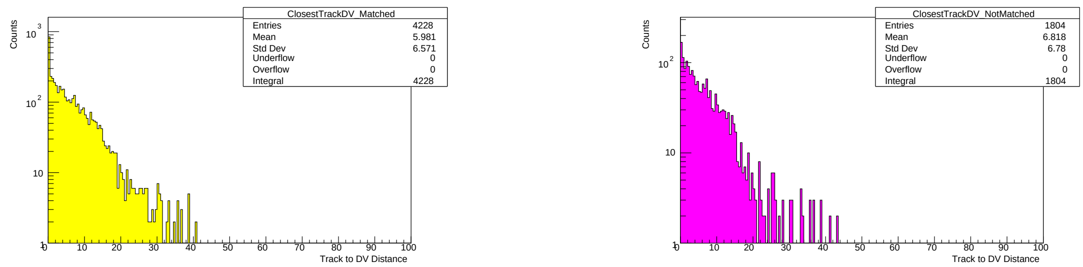
Figure 4: Errors in sz space are calculated by computing the distance between a DV_{reco} and its corresponding DV_{true} by taking into consideration the three coordinates of the points in xyz space. Respectively, errors in xy plane take into consideration only the x and y coordinated of DV_{reco} and DV_{true}. Also, errors are displayed in mm.



(a) Matched DV_{reco}

(b) Non Matched DV_{reco}

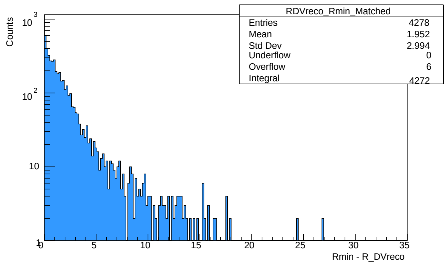
Figure 5: Minimum distance between DV_{reco} and the closest given point of the reconstructing trajecotries. Also, distances are displayed in mm.



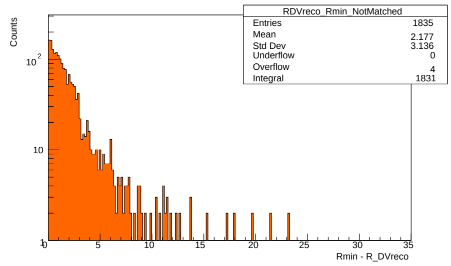
(a) Matched DV_{reco}

(b) Non Matched DV_{reco}

Figure 6: Minimum distance between DV_{reco} and a trajectory, excluding the ones used to reconstruct it (if it exists). The distances are displayed in mm.

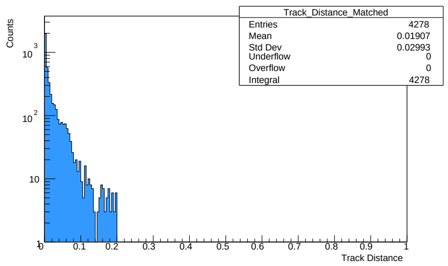


(a) Matched $\mathrm{DV}_{\mathrm{reco}}$

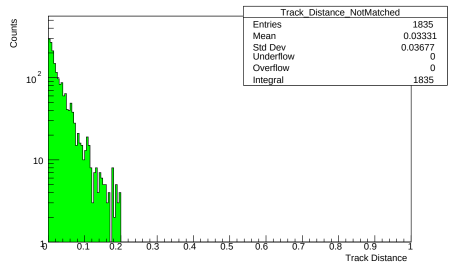


(b) Non Matched $\mathrm{DV}_{\mathrm{reco}}$

Figure 7: The distance between $\mathrm{DV}_{\mathrm{reco}}$ and IP is noted as $R_{\mathrm{DV}_{\mathrm{reco}}}$. Also, the minimum of the distances from IP to the given points of reconstructing trajectories is noted as R_{min}. The histogram displays the difference between the two distances: $R_{\mathrm{min}} - R_{\mathrm{DV}_{\mathrm{reco}}}$.

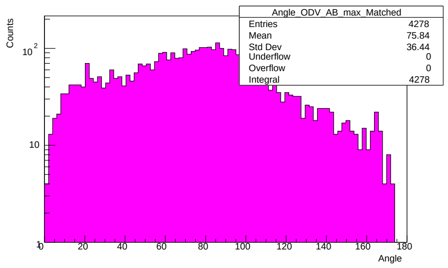


(a) Matched $\mathrm{DV}_{\mathrm{reco}}$

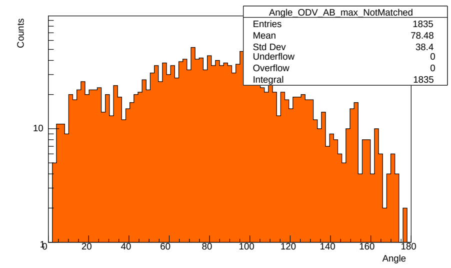


(b) Non Matched $\mathrm{DV}_{\mathrm{reco}}$

Figure 8: Distance between reconstructing trajectories. The distances displayed in the histograms are measured in mm.

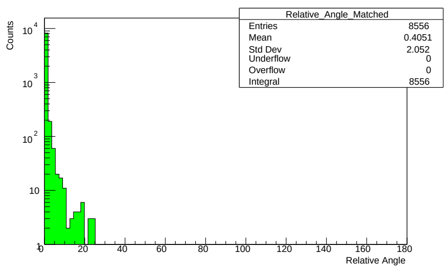


(a) Matched $\mathrm{DV}_{\mathrm{reco}}$

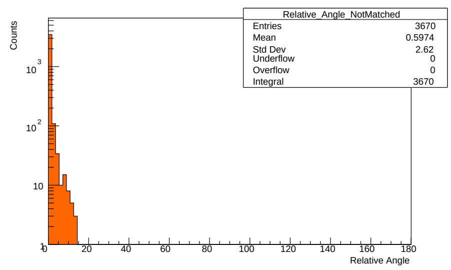


(b) Non Matched $\mathrm{DV}_{\mathrm{reco}}$

Figure 9: Maximum angle that is formed by two sides with common point the IP. The first one connects it to the $\mathrm{DV}_{\mathrm{reco}}$. The second one connects it to the first point of reconstructing trajectories. The angles in histograms are measured in degrees.

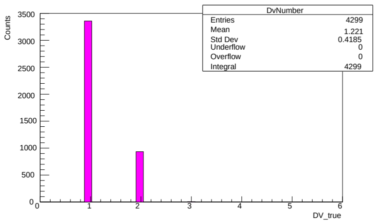


(a) Matched $\mathrm{DV}_{\mathrm{reco}}$

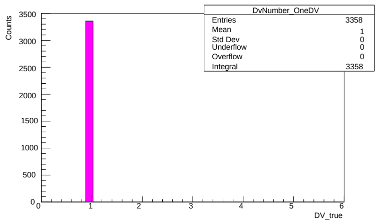


(b) Non Matched $\mathrm{DV}_{\mathrm{reco}}$

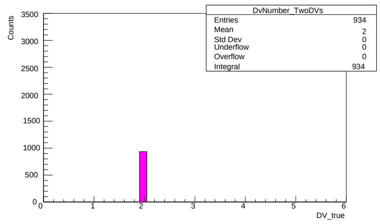
Figure 10: Maximum relative angle between $\mathrm{DV}_{\mathrm{reco}}$ and given points of reconstructing trajectories. The concept of the relative angle is mentioned in Subsection 2.2. The angles in histograms are measured in degrees.



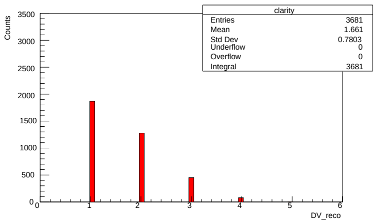
(a) Number of DV_{true}.



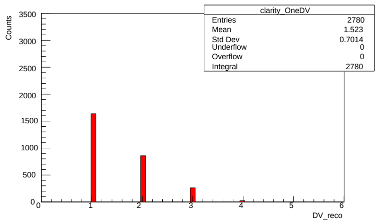
(b) Number of DV_{true} in events with one DV_{true}.



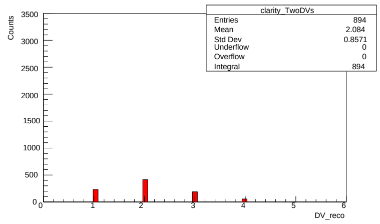
(c) Number of DV_{true} in events with two DV_{true}.



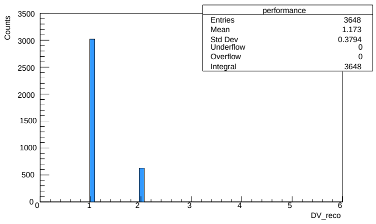
(d) Number of DV_{reco}.



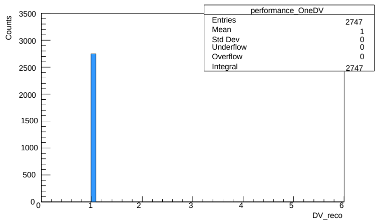
(e) Number of DV_{reco} in events with one DV_{true}.



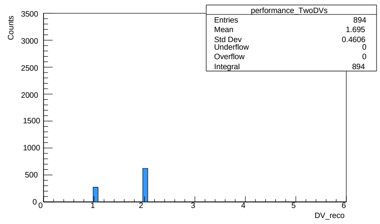
(f) Number of DV_{reco} in events with two DV_{true}.



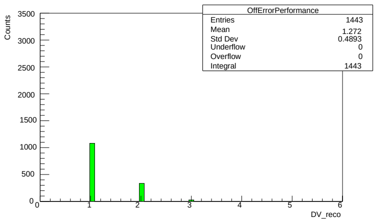
(g) Number of Matched DV_{reco}.



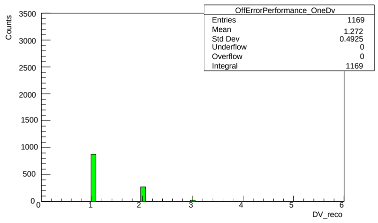
(h) Number of Matched DV_{reco} in events with one DV_{true}.



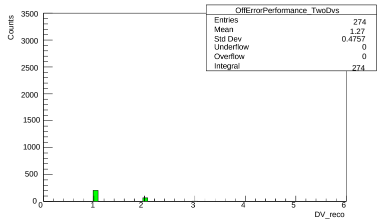
(i) Number of Matched DV_{reco} in events with two DV_{true}.



(j) Number of Not Matched DV_{reco}.

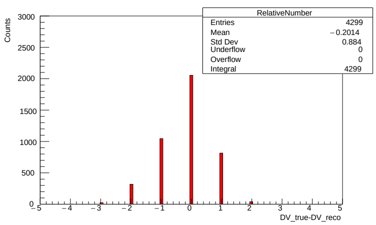


(k) Number of Not Matched DV_{reco} in events with one DV_{true}.

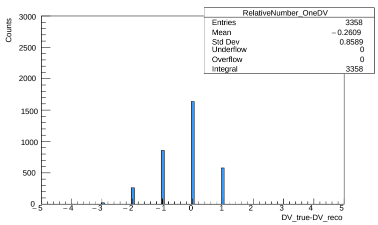


(l) Number of Not Matched DV_{reco} in events with two DV_{true}.

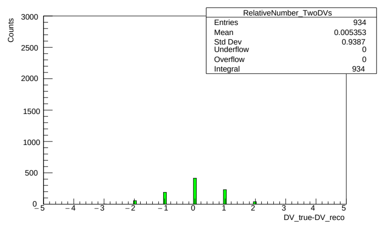
Figure 11: The DV_{true} and DV_{reco} numbers.



(a) All events



(b) Events with one DV_{true}



(c) Events with two DV_{true}

Figure 12: Relative number of DV_{reco} with respect the number of DV_{true}

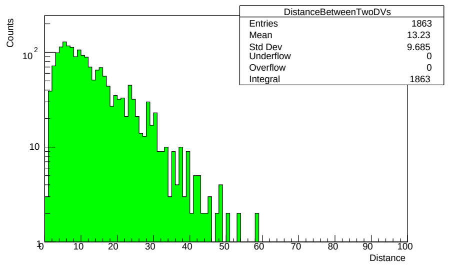


Figure 13: Distance between two DV_{true} in events with two DV_{true}

Figure 11 Undoubtedly, it can be derived from the distribution of the total number of DV_{reco} that, it is more common for DV_{reco} to exceed DV_{true} number than to subceed them. This behaviour would suggest that a stricter limit on `DVCut` value would reduce the number of DV_{reco}, thus making the algorithm more efficient. Notably, the majority of events contain a single DV_{true} and the minority contain two or three DV_{true}. Therefore, the previous change mentioned might produce better overall results in processing of this data set, but if the numbers of DV_{true} were reversed it would cause DV_{reco} to be outnumbered from DV_{true} in a lot of events. Since it is preferable for the algorithm to be more or less constant on its efficiency and purity, for whatever data set it processes, the `DVCut = 0.2`, is thought to be an ideal value.

Figure 12 It is noticed that there is a slight asymmetry in the distribution of the difference tilting to higher DV_{reco} numbers that DV_{true}, on the totality of events. On the other hand, it is noteworthy that in events with two DV_{true} the asymmetry tilts to the DV_{reco} be outnumbered from the DV_{true}. Consequently, if the value of `DVCut` becomes lower, the asymmetry in events with one DV_{true} would be fixed, but in events with two DV_{true}[6] would produce a greater asymmetry, because even lower number of DV_{reco} would be reconstructed. Similarly, if the value of `DVCut` becomes lower, the asymmetry on events with one DV_{true} would enlarge, whereas in events with two DV_{true}[7] would soften. Thus, the value of `DVCut` seems to treat every possible case with respect.

Figure 13 The mean value of the distribution is about 13mm, so the value `TrajectoryCut = DVCut/2 = 0.1` is a lot smaller that the range of distances on the histogram. For values less than 0.2 mm there are almost no DV_{true} with distances in this range. Therefore, it is impossible for the condition which uses the `TrajectoryCut` to stop a DV_{reco} forming.

[6]Assuming that this behaviour intensifies the larger the DV_{true} number is, this argument can be generalised.

[7]Using the same argument with footnote 6, this behaviour can be generalised.

3.2.2 Indexes

The indexes on Output 1 contain useful information about the success of the Algorithm to reconstruct the DV_{true} and its ability to stop when there are no data suggesting the existence of another one.

Efficiency

First of all, the efficiency is a measure of how many DV_{true} were found by the Algorithm, while it does not take into consideration the exceeding number of DV_{reco} that might appear. The values produced by the algorithm are satisfactory and does not fluctuate between xy plane and sz space. There is a slight advantage on efficiency in sz space due to the stricter limit on xy plane for the Matched DV_{reco}.

Purity

On the other hand, the index purity refers to DV_{reco}. To specify, it is a measure of how many of the reconstructed DV_{reco} where Matched with a DV_{true}. Thus, it takes into account both the limits on xy plane and sz space and considers if the number of DV_{reco}, in an event, exceeds the number of DV_{true}.

Similarly with efficiency, the results for sz space are slightly better than the ones referring to xy plane. Additionally, it is the events with two DV_{true} that produce the greater results in both xy plane and sz space. This behaviour could be explained by the histogram in Figure 12c, where the asymmetry around the value two tilts more on the greater values than lower. Namely, there are more events where DV_{true} outnumber DV_{reco}, so the only way for this events to decrease purity is to produce DV_{reco} that do not respect the xy plane and sz space limits. Considering that the most DV_{reco} which do not exceed the number of DV_{true} do respect those limits (see Figure 4), the purity is expected to be higher in events with two DV_{true}.

As mentioned in comments on Figure 11, the results on purity could be immensely augmented if `DVCut` would have a lower value, but the results would be specifically tweaked for the data set used for this paper. If another data set was to be processed, that would contain more events with multiple DV_{true}, the results would be abnormally different (of course worse). The value of `DVCut` is selected so as to provide consistence results in all possible data sets.

Accuracy

Finally, accuracy is a measure of how many of the DV_{reco} that did not exceed the number of DV_{true} where Matched to a DV_{true}. It can be argued that the results are significant and no further improvement needs to be done in this index. Of course, this values acknowledges with clarity that whatever inefficiency the Algorithm has is due to the exceeding number of DV_{reco} that are produced in some events. Thus, any future effort to augment the Algorithm's performance should reduce the additional DV_{reco}, relative to DV_{true}'s number.

4 Conclusions

Whereas all the results and Histograms produced by the Algorithm are commented thoroughly in the previous section, a concentrated summary of the main points is missing.

On the following numbered list the fundamental observations about indexes and graphs are presented with remarks about possible augmentation of the methods used.

1. There is greater efficiency in sz space than xy plane. This behaviour is due to the more strict limit on xy plane in relation to sz space and can be smoothed out if the limit becomes looser.

2. The purity is greater in sz space that xy plane. The argument follows the previous quantity. Also, purity is greater on events with two $\mathrm{DV}_{\mathrm{true}}$ because in these events it is more rare to find exceeding number of $\mathrm{DV}_{\mathrm{reco}}$. The overall purity could be ameliorated through decrease of `DVCut` value, though it is not suggested since it might produce worse results in other data sets.

3. The accuracy of the algorithm is almost perfect which suggests that if any improvement is to be made, it would be through the decrease in exceeding $\mathrm{DV}_{\mathrm{reco}}$ number.

4. It can be also argued that the error limit in xy plane needs to be regulated to lower values from the comparison of the first and the second row of histograms in Figure 4.

5. The data collected from the observation of histograms in Figures 5, 6 and 7 suggest that no additional conditions need to be applied, concerning the quantities that are displayed in those Figures.

6. The algorithm produces exceeding number of $\mathrm{DV}_{\mathrm{reco}}$ relative to $\mathrm{DV}_{\mathrm{true}}$ more often than the contrary. This behaviour could be abridged by applying stricter limits on `DVCut` value. However, in order for the Algorithm to be more consistent in its results, the value has been set `DVCut = 0.2`, to compensate both for events with one and multiple $\mathrm{DV}_{\mathrm{true}}$. Additionally, the value `DVCut = 0.2` goes along the suggestions set in Figure 8.

7. The definition of the `TrajectoryCut = DVCut/2` serves perfectly the conditions that Figure 13 sets.

A "Distance" Between Two Lines

For the sake of completeness the same conditions presented in Subsection 2.1 are repeated.

Let two lines ε_i and ε_j for which the only information given are two points, $P_i, P_i{}'$ and $P_j, P_j{}'$, that lie on each, respectively, as shown in Figure 1. Then, the equations which describe the two lines are the following:

$$(\varepsilon_i): \quad \mathbf{r_i} + t\,\mathbf{n_i}, \quad \mathbf{n_i} \equiv \mathbf{r_i}' - \mathbf{r_i}, \tag{2a}$$

$$(\varepsilon_j): \quad \mathbf{r_j} + s\,\mathbf{n_j}, \quad \mathbf{n_j} \equiv \mathbf{r_j}' - \mathbf{r_j}. \tag{2b}$$

Also, let A be a point of line ε_i and B be a point of line ε_j. Therefore, vector **AB** is given by:

$$\mathbf{AB} = \mathbf{r_j} - \mathbf{r_i} + s\,\mathbf{n_j} - t\,\mathbf{n_i},$$

which depicts a plane Π that goes through $\mathbf{r_o} \equiv \mathbf{r_j} - \mathbf{r_i}$ and is parallel to vectors $\mathbf{n_j}$ and $\mathbf{n_i}$, as seen in Figure 14. It is equivalent to say that plane Π is perpendicular to vector $\mathbf{u} \equiv \mathbf{n_j} \times \mathbf{n_i}$. In addition, the variables t and s are independent and their value determines which point in the plane **AB** points to.

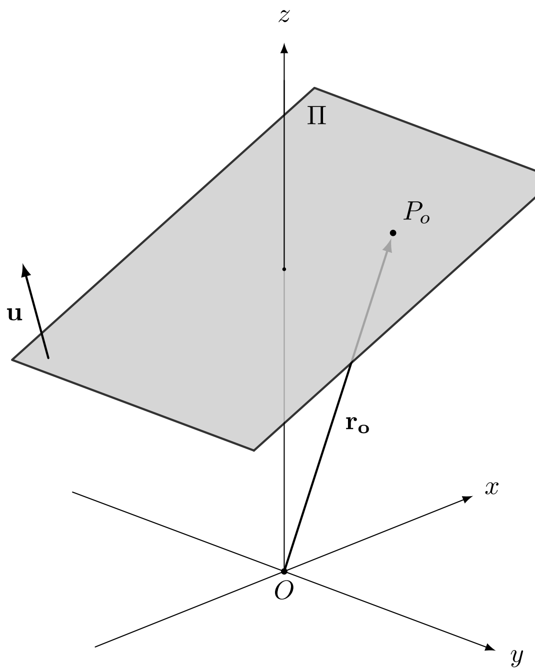


Figure 14: Plane Π to point P distance

Obviously, the "distance" d between the line ε_i and ε_j is the distance of the point O, the beginning of the axis, to the plane Π. The distance from O to the plane Π can be thought as the projection of vector $\mathbf{OP_o}$ on the vector **u**, so the following applies:

$$d = \frac{|\mathbf{r_o} \cdot \mathbf{u}|}{\|\mathbf{u}\|}.$$

Consequently, the "distance" vector **D** can be written as:

$$\mathbf{D} = \frac{|\mathbf{r_o} \cdot \mathbf{u}|}{\|\mathbf{u}\|}\,\mathbf{u}.$$

In order to find the vector **OA** the variable $t = t_o$ needs to be specified, for which applies the following:

$$\mathbf{r_i} + t_o\,\mathbf{n_i} + \mathbf{D} \in (\varepsilon_j) \Rightarrow$$
$$\mathbf{n_j} \times (\mathbf{r_i} - \mathbf{r_j} + t_o\,\mathbf{n_i} + \mathbf{D}) = 0 \Rightarrow$$
$$\mathbf{n_j} \times (\mathbf{r_i} - \mathbf{r_j}) + t_o\,\mathbf{u} + \mathbf{n_j} \times \mathbf{D} = 0 \Rightarrow$$
$$\mathbf{u} \cdot (\mathbf{n_j} \times (\mathbf{r_i} - \mathbf{r_j})) + t_o\|\mathbf{u}\|^2 + 0 = 0 \Rightarrow$$
$$t_o = \frac{\mathbf{u} \cdot (\mathbf{n_j} \times \mathbf{r_o})}{\|\mathbf{u}\|^2}.$$

Similarly, to find the vector **OB** the variable $s = s_o$ needs to be specified, for which applies the following:

$$\mathbf{r_j} + s_o\,\mathbf{n_j} - \mathbf{D} \in (\varepsilon_i) \Rightarrow$$
$$\mathbf{n_i} \times (\mathbf{r_j} - \mathbf{r_i} + s_o\,\mathbf{n_j} - \mathbf{D}) = 0 \Rightarrow$$
$$\mathbf{n_i} \times (\mathbf{r_j} - \mathbf{r_i}) - s_o\,\mathbf{u} - \mathbf{n_i} \times \mathbf{D} = 0 \Rightarrow$$
$$\mathbf{u} \cdot (\mathbf{n_i} \times (\mathbf{r_j} - \mathbf{r_i})) - s_o\|\mathbf{u}\|^2 + 0 = 0 \Rightarrow$$
$$s_o = \frac{\mathbf{u} \cdot (\mathbf{n_i} \times \mathbf{r_o})}{\|\mathbf{u}\|^2}.$$

Acknowledgements

Firstly, I would like to express my deepest gratitude to my supervisors, Professor Dimitris Fassouliotis and Postdoctoral Researcher Stylianos Angelidakis, for providing me their knowledge and expertise, guiding me throughout the semester with support and uninterrupted attention.

Additionally, I am grateful to the other members of the UoA ATLAS Team, Professor Emeritus Christine Kourkoumelis and Researcher Stelios Vourakis, for their helpful comments and the clarity they provided to the overall project.

References

[1] Alexandros Tsagkaropoulos. *ATLAS - Long Lived Particles*. 2022. GitHub.com.

[2] UoA ATLAS Team. *New Particle Search at CERN*. zooniverse/stage1/classify