b-tagging at Muon Collider

Lorenzo Sestini
INFN-Padova

Muon Collider Workshop – CERN, 11-9-2019

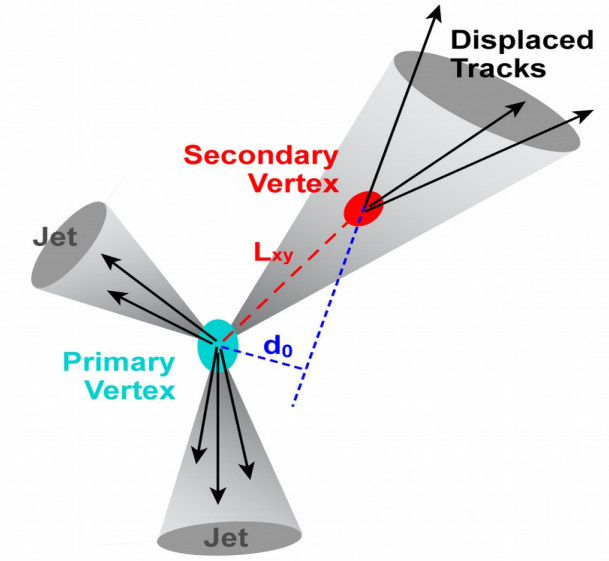
b-jet tagging algorithm

b-jet tagging algorithm inspired by LHCb:

1) Tracks with p_T>500 MeV and impact parameter>0.04 cm are selected

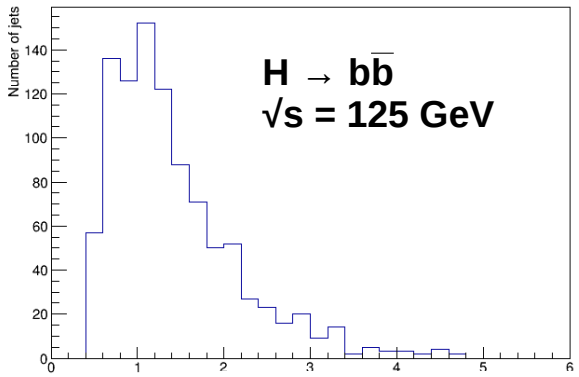
2) 2-tracks vertices are formed by requiring a distance of closest approach < 0.02 cm and p_T(2-tracks) > 2 GeV

3) 2-tracks vertices are linked if they share one track → 3-tracks vertices are formed

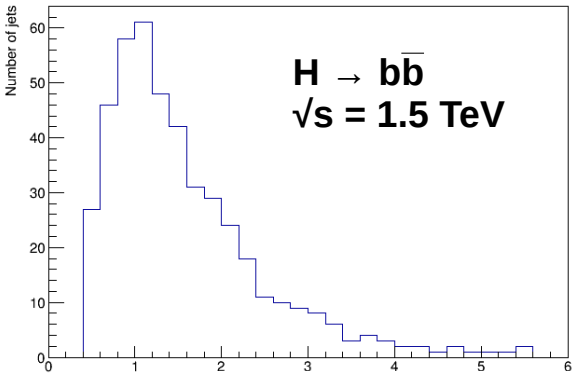


Without beam-induced background
(with complete tracking)

b-tagging $\sqrt{s}$ = 125 GeV vs $\sqrt{s}$ = 1.5 TeV



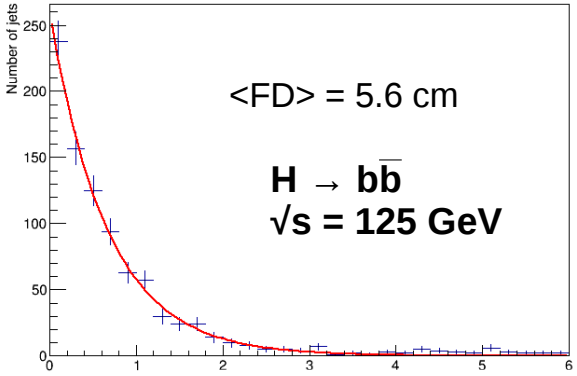
SV-tagging efficiency (125 GeV) = 63%

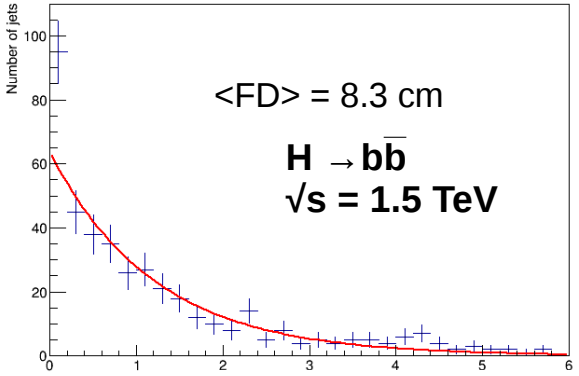


SV-tagging efficiency (1.5 TeV) = 69%

Without beam-induced background (with complete tracking)

→ Improved tagging efficiency at 1.5 TeV

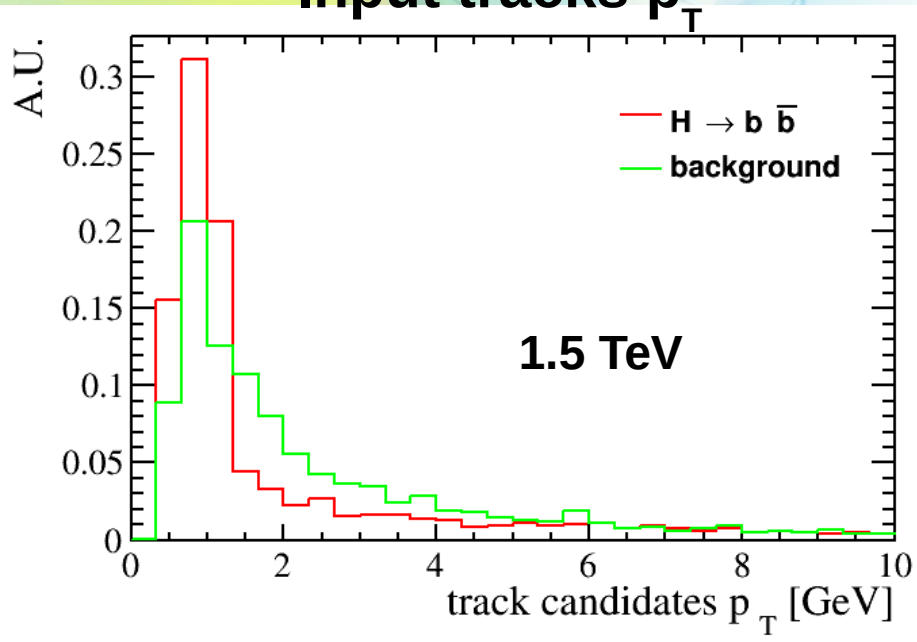




→ Jets are more boosted → higher flight distance

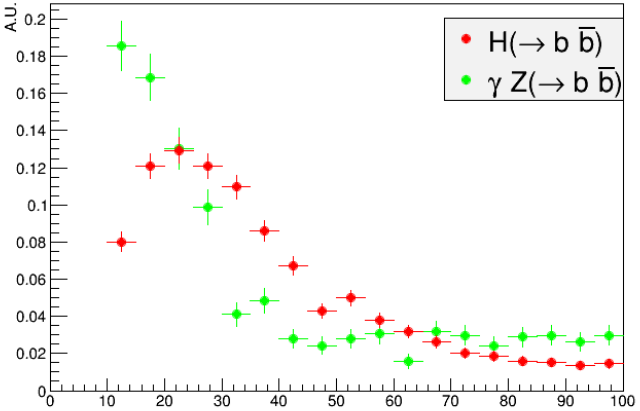
b-jet tagging studies

Input tracks p_T



- Due to framework limitations, we were not able to perform a **complete** in the presence of the beam-induced background.
- **Inefficiencies at low track p_T reduce significantly the tagging efficiency.**

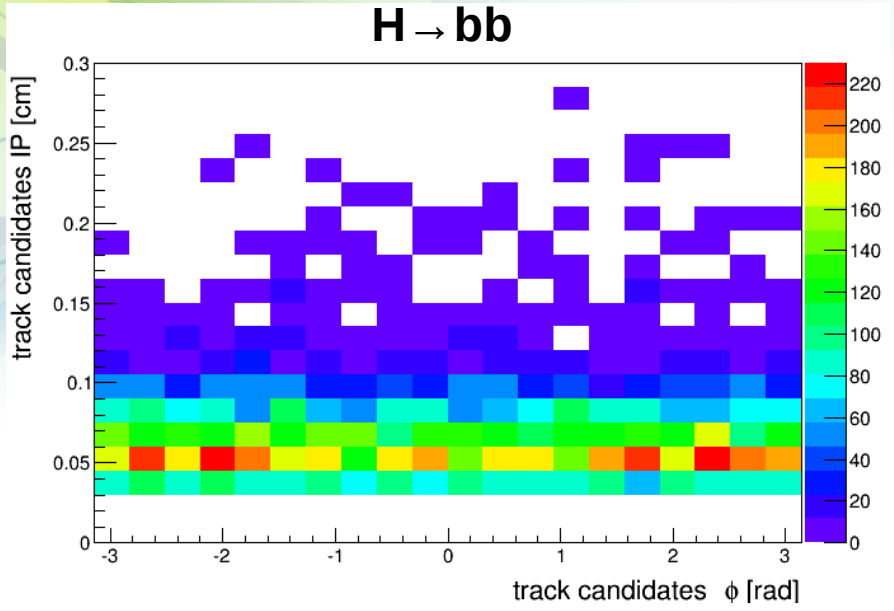
Remember that √s = 1.5 TeV is one of the most difficult cases.

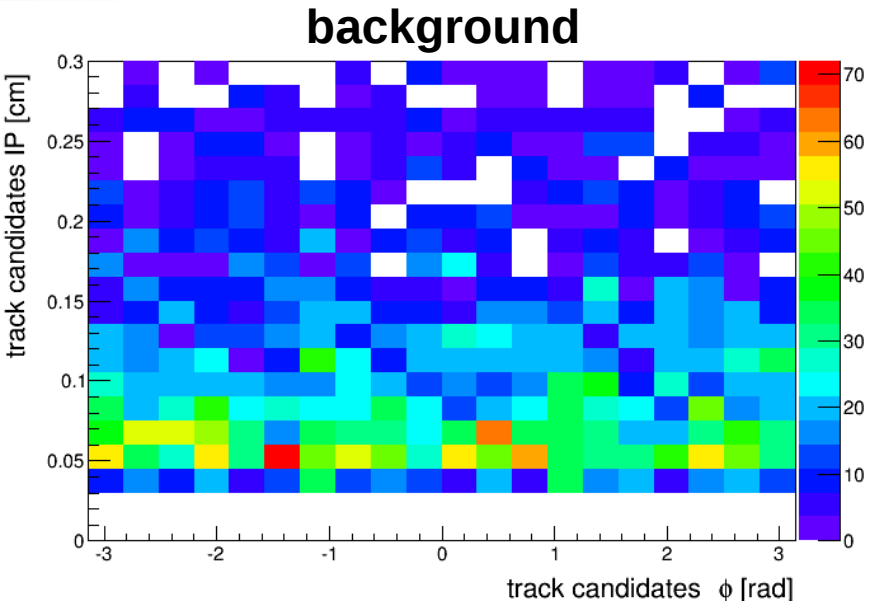


Jet transverse momentum [GeV]

b-tagging

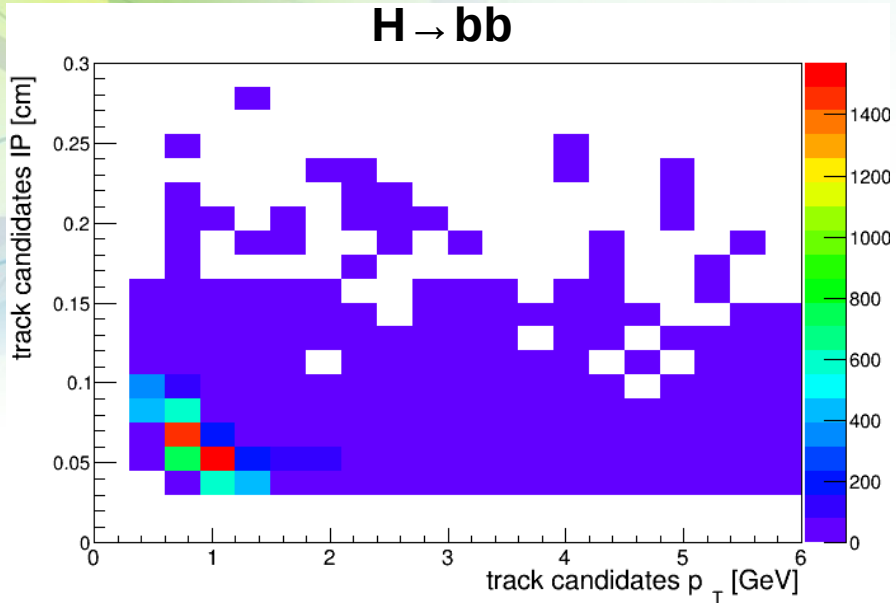
Input tracks IP vs φ

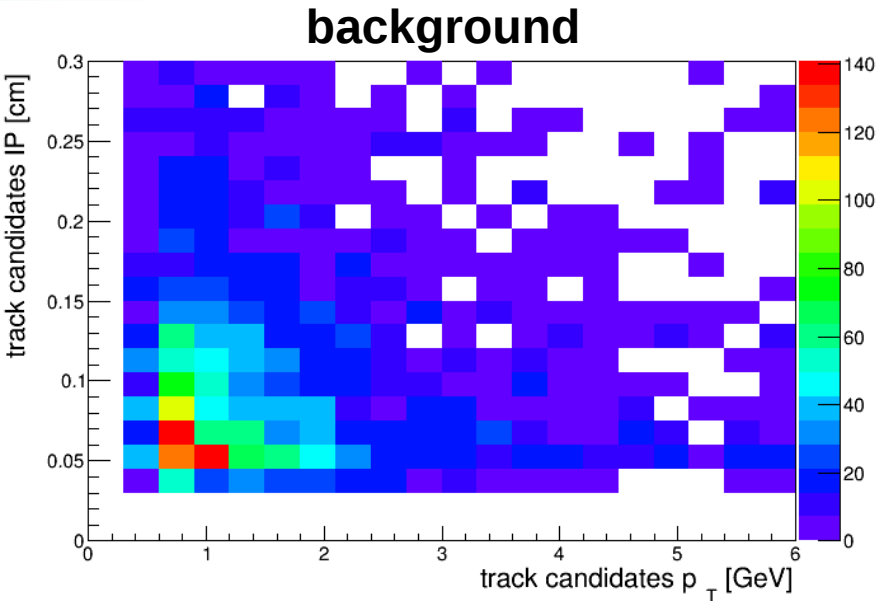




b-tagging

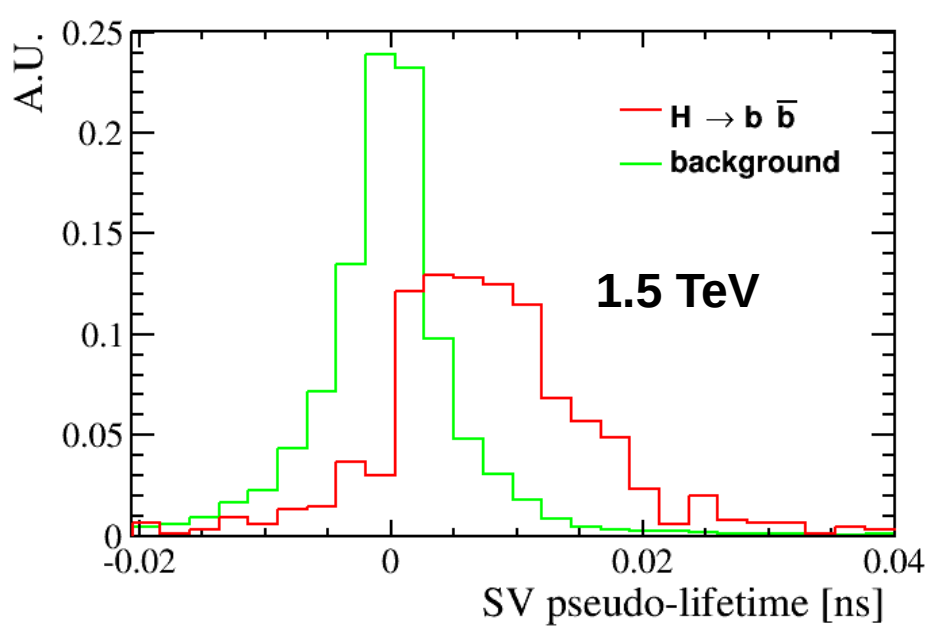
Input tracks IP vs p_T

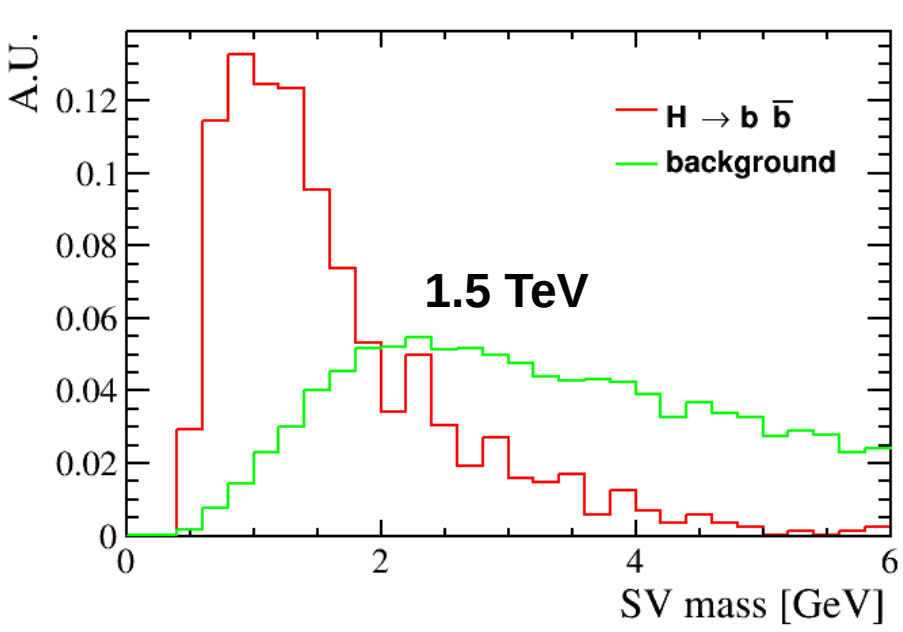




b-jet tagging studies

Secondary Vertex observables





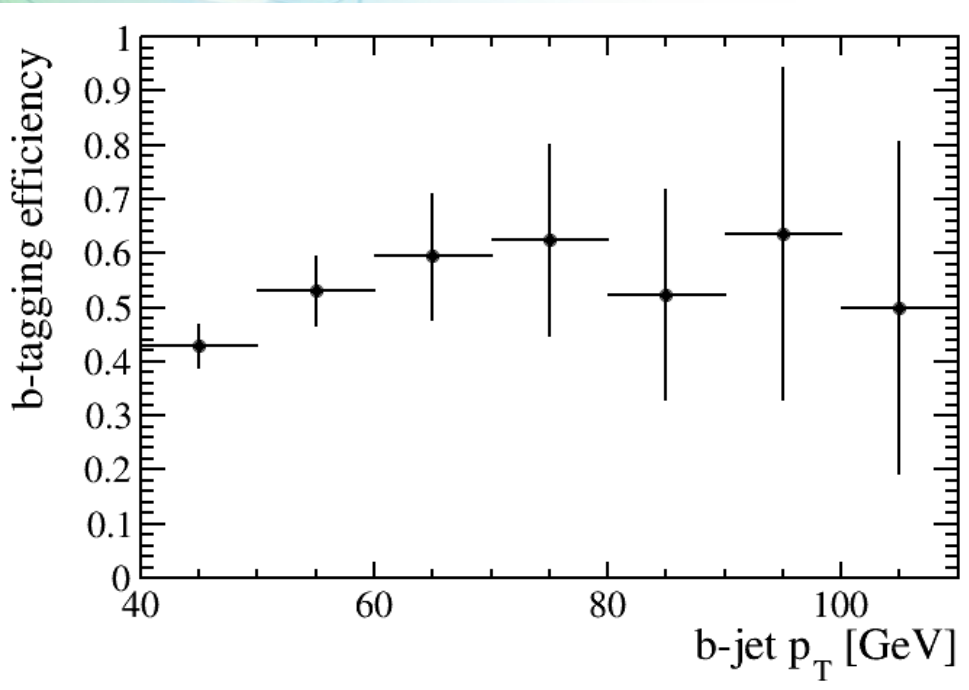
$$\tilde{t} = \frac{m(B) \cdot FD}{p}$$

Negative lifetime: negative SV projection along the jet axis (false tag)

- Beam-induced background produces fake SV
- **SV-related observables show different properties between signal and beam-induced background**.
- It is evident that a proper algorithm (e.g DNN) could remove the combinatorial preserving the efficiency.

b-tagging efficiency

$$\epsilon = \frac{N^{reco}_{tagged}}{N^{reco}}$$



Without the beam-induced background at 1.5 TeV the SV-tagging efficiency was 69%

- **Mistag**: the (η-φ) space is divided in cones with R=0.5 → $\omega = \frac{N^{cone}_{with\,SV}}{N^{cone}}$

The mistag is ~1-3%

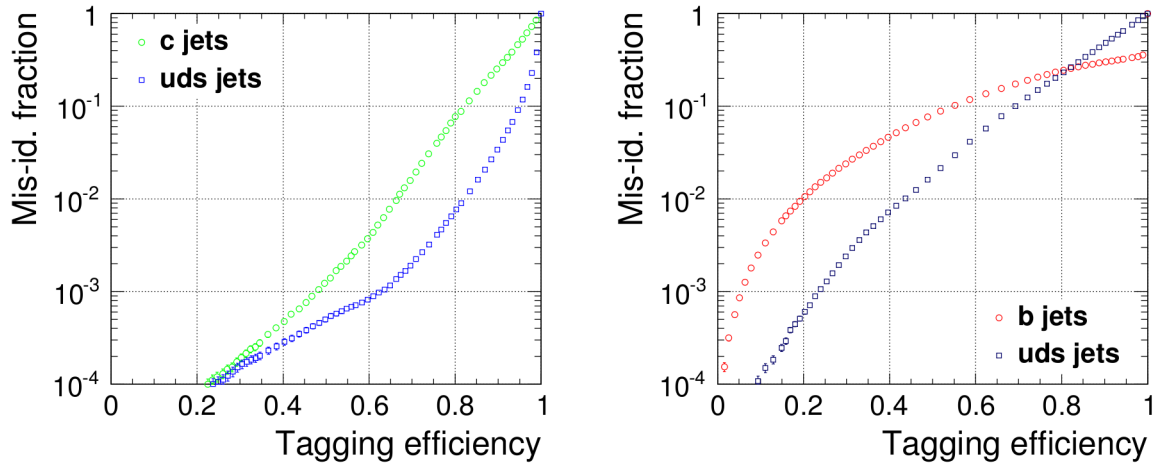
Implementation

- Until now for b-tagging a root macro has been used
- Not the best way to go: **in the future we should use the b-tagging embedded in the ILCsoft framework**

```
cout << "#good tracks in jet = " << ntracks << endl;

//2-tracks secondary vertices

Int_t nTwoTrVx=0;
Double_t TwoTrVx_X[2000000]={0};
Double_t TwoTrVx_Y[2000000]={0};
Double_t TwoTrVx_Z[2000000]={0};
Double_t TwoTrVx_TrId1[2000000]={-1};
Double_t TwoTrVx_TrId2[2000000]={-1};

for (int i=0; i<ntracks; i++){
  for (int j=i+1; j<ntracks; j++){

  Double_t x1[3];
  Double_t x2[3];

  IlcAODTrack *tr1 = ev->GetTrack(injet_flag[i]);
  IlcAODTrack *tr2 = ev->GetTrack(injet_flag[j]);

  tr1->GetXYZ(x1);
  tr2->GetXYZ(x2);

  TVector3 v1(tr1->Px(), tr1->Py(), tr1->Pz() );
  TVector3 p1( x1[0], x1[1], x1[2] );

  TVector3 v2(tr2->Px(), tr2->Py(), tr2->Pz() );
  TVector3 p2( x2[0], x2[1], x2[2] );

  TVector3 r = p1 - p2;

  Double_t a = v1.Mag2();
  Double_t b = v1.Dot(v2);
  Double_t c = v2.Mag2();
  Double_t d = v1.Dot(r);
  Double_t e = v2.Dot(r);

  Double_t t1 = (b*e - c*d)/(a*c - b*b);
  Double_t t2 = (a*e - b*d)/(a*c - b*b);

  TVector3 r1 = p1 + t1 * v1;
  TVector3 r2 = p2 + t2 * v2;

  TVector3 dv = r1-r2;

  TVector3 sv = 0.5*(r1+r2);

  Double_t FD = sv.Mag();

  Double_t dist = dv.Mag();

  TVector3 p_tot = v1+v2;

  Double_t pt_tot = sqrt (pow(p_tot.X(),2) + pow(p_tot.Y(),2));

  if (dist<0.02) h_pt2->Fill(pt_tot);
```

b-tagging in ILCsoft

- **Secondary vertex reconstruction**
- **BDTs trained with SV+ jet constituents inputs**



(a) b tag (b) c tag

Figure 1: The flavor tagging performance, evaluated on $Z \to q\bar{q}$ sample at $\sqrt{s} = 91.2$ GeV, is shown in terms of the mis-identification fraction versus the tagging efficiency. (a) The tagging efficiency is shown for b jets. The green (circle) points show the fraction of c jets being mistaken as a b jet. The blue (square) points show the fraction of uds jets being mistaken as a b jet. (b) The tagging efficiency is shown for c jets. The red (circle) points show the fraction of b jets being mistaken as a c jet. The blue (square) points show the fraction of uds jets being mistaken as a c jet.

Working plan

- **Tracking and jet reconstruction are needed!**
- **Setup the tagging algorithm already implemented in ILCsoft**
- **Measure the SV-tagging efficiencies**
- **Train the BDTs to improve the identification performance**

Backup slides