Particle ID study and its application – towards LCFIPlus improvement

Masakazu Kurata
The University of Tokyo

TYL Meeting, 03/02/2015-03/03/2015

Contents

- Introducing new variables
  - dE/dx from TPC
  - Shower profiles from Calorimeter(s)
- Particle ID
  - Construction of Particle ID
- Towards flavor tagging improvement
  - Vertex mass recovery
  - Checking the improvement of vertex mass recovery
- Top analysis related stuff – flavor separation improvement & vertex charge, study ongoing
- Summary

For analysis improvement

- All the analyses are saturated within the present framework
  - Needs new idea
  - Especially, improvement is necessary for Top physics and (small signal) Higgs Physics analysis
- Fundamental new variables might provide improvements of analysis tools @ILD, but not yet used well
  - dE/dx in TPC
  - Shower profiles in the calorimeters
- Particle ID will be available using those variables
- Will those variables give improvements to other analysis components?
  - Isolated lepton ID → of course! ～30% improvement for fake lepton rejection
  - Energy correction using PID → it is OK!
    - Correction is going to good direction, but effect is small
  - Flavor tagging using PID? → Looks hopeful! → LCFIPlus re-development
  - Hope for jet clustering?

→it is necessary to study them

dE/dx from TPC

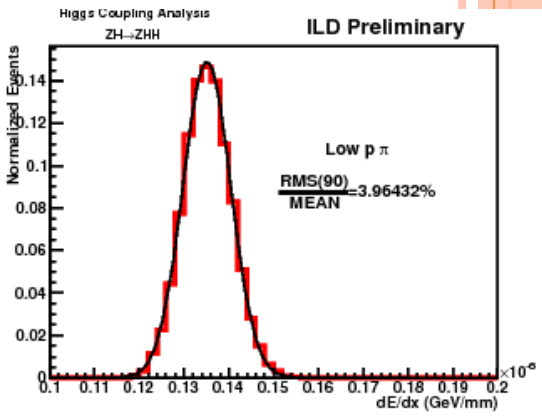
- For improvement, using dE/dx is one of the powerful tools
  - Particle ID for each track will give a large impact to the analysis
  - Application to general analysis component is very wide
    - Lepton ID
    - Track energy correction
    - Flavor tagging
    - Jet clustering?
- Important factor to use dE/dx is: fluctuation
  - TDR: measurement resolution is **5%**
  - So, natural fluctuation from simulation is within 5% without detector effect
- dE/dx definition:
  - $\frac{dE}{dx} = \frac{energy\ deposit}{flight\ path\ in\ the\ hit(TPC)}$
  - dE/dx can be calculated at any hit point
  - Truncated mean is calculated as track dE/dx

    $\left\langle \frac{dE}{dx} \right\rangle = \frac{1}{n}\sum_i^n \frac{dE_i}{dx_i}$ upper 30%, lower 8%(important!) hits are discarded to avoid Landau tail(next slide)

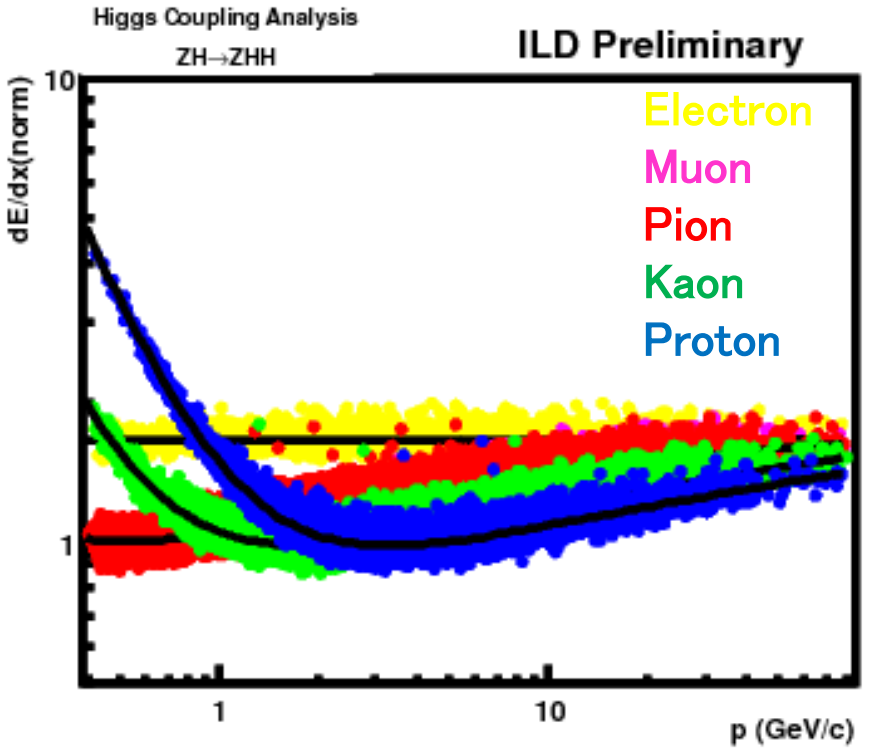
    →optimization is necessary

dE/dx

- Fluctuation of dE/dx using various type of tracks
  - Particles@MIP range, Particles with high momentum
  - Fluctuations of each particle/each momentum range:
    3 – (<5)%!!
  - Including detector effect is necessary



- Momentum dependence of dE/dx for each particle
  - Polar angle dependence corrected
  - Num. of Hits dependence corrected
  - Scale to $\left\langle \frac{dE}{dx} \right\rangle = 1.0$ for MIP pion



Shower profile

- Shower shapes in the calorimeter are different between electron/photon/muon/hadrons
  - So characters of the clusters will be a good tool to distinguish tracks
  - Especially, electromagnetic shower shape is well known
  - Grabbing those information will boost leptonID efficiency/fake rejection efficiency

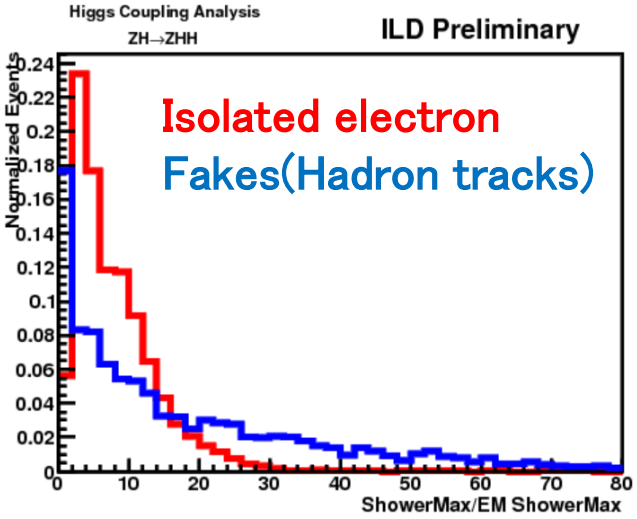
- Information extraction is based on fitting to cluster hits:
  - Well-known EM shower profile

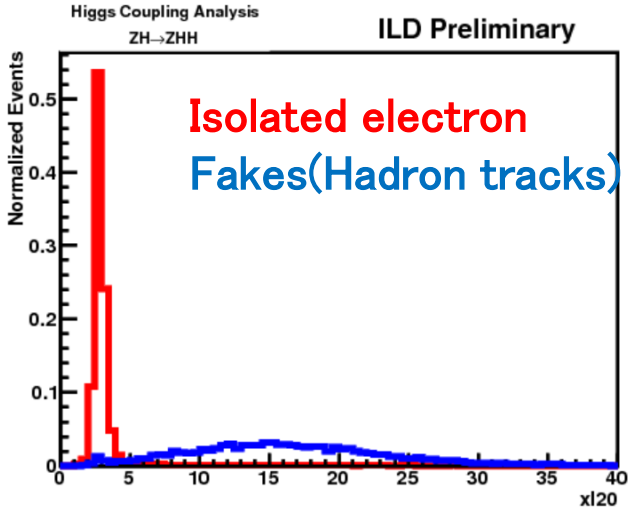
$$f(x_l, x_t) = ac\frac{(c(x - x_{l0}))^{b-1} \cdot \exp(-c(x - x_{l0})) \cdot \exp(-dx_t)}{\Gamma(b)}$$

- In addition, hit based variable is introduced to identify shower start
  - Xl20 – length from cluster start to 20% of total energy deposit

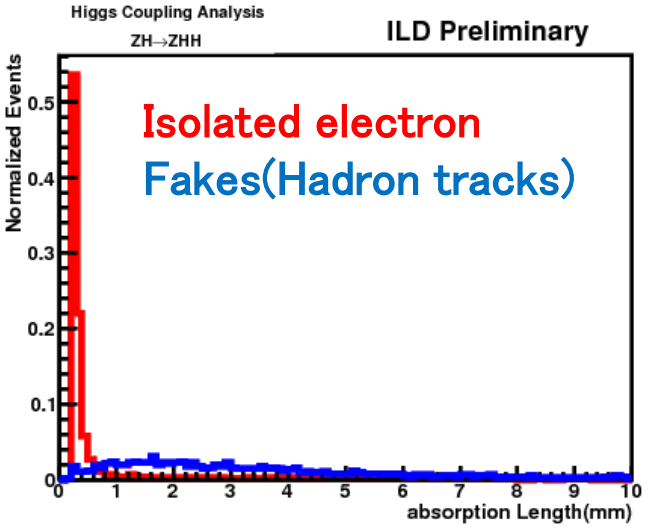
SHOWER PROFILE

- Longitudinal information – shower Max. & shower start position



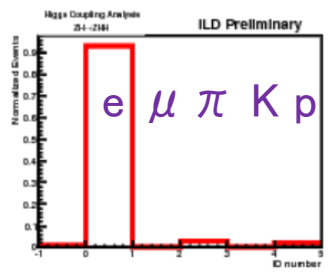


- Transverse information – Absorption length

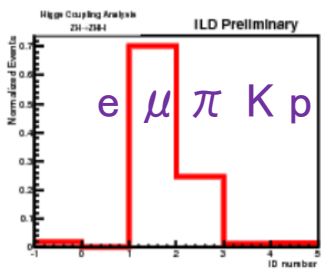


Particle ID

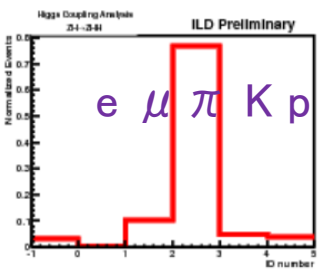
- New variables make Particle ID available
  - How are particles identified as each particle type?
- Construct Particle ID algorithm:
  - Based on Bayesian approach: define posterior probability
  - Make “rejected” category:
    - Track is rejected if its posterior probability is below threshold
    - Those tracks are moved to pions
- Overall ID efficiency – using tracks in jets:
  - Electron can be identified almost perfectly(>90%)
  - Muon ID eff. is ～70% →due to low energy muons(μ / π separation)
  - Hadron ID effs. are 62%～75%



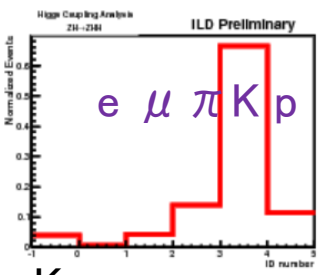
e



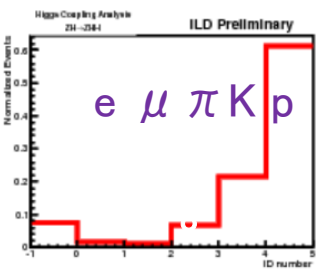
μ



π



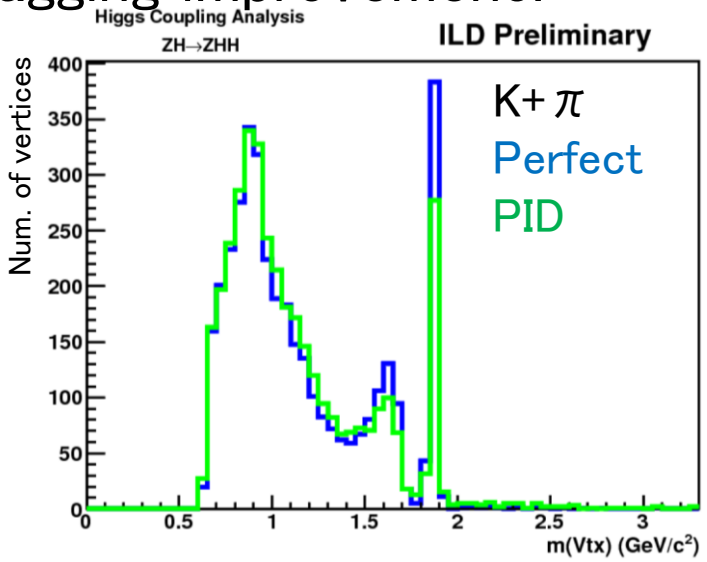
K



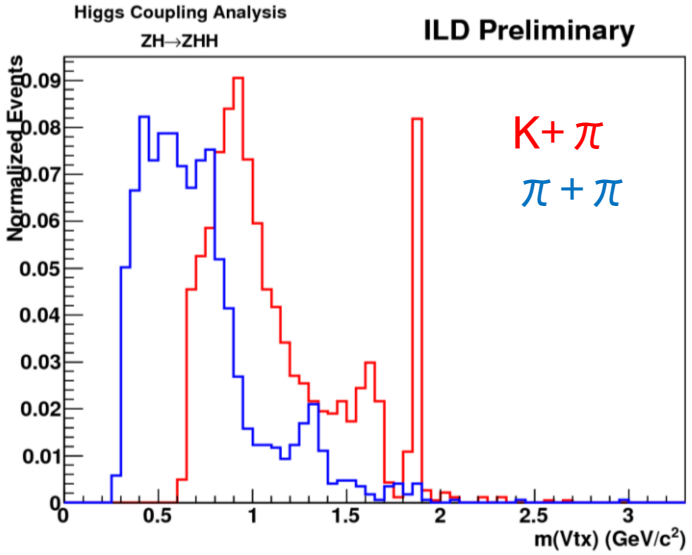
p

Vertex classification

- Can Particle ID be used for flavor tagging improvement?
  - Checking vertex mass distribution
  - Vertex is from LCFIPlus
  - How much effect on vertex mass?



- Classifying vertices with particle type using particle ID
  - Different vertex pattern has different vertex mass pattern
  - e.g.) K+ π v.s. π + π



Hope for flavor tagging improvement

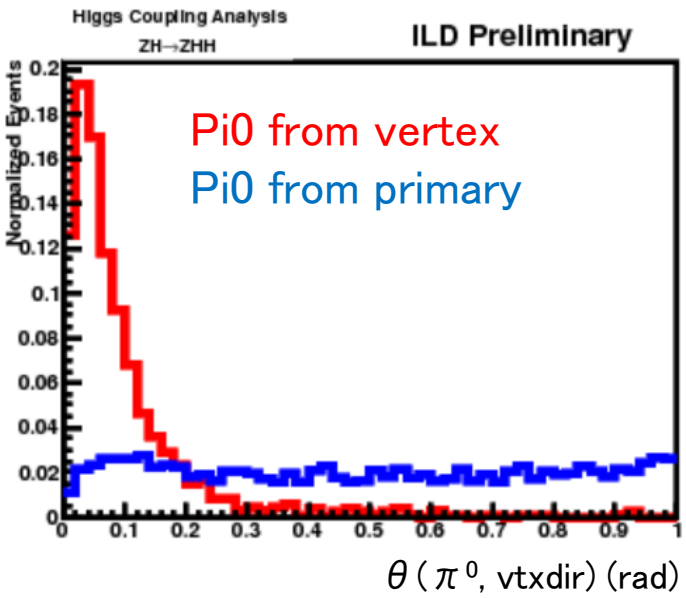
- For flavor tagging improvement
  - Vertex mass is the key to separate heavy/light flavor vertex
  - Many pi0s will escape from B/D vertex → checked that using MC truth
  - Mass resolution will be degrade due to escaping neutrals
  - Is there possibility to recover pi0s which escape from vertices?

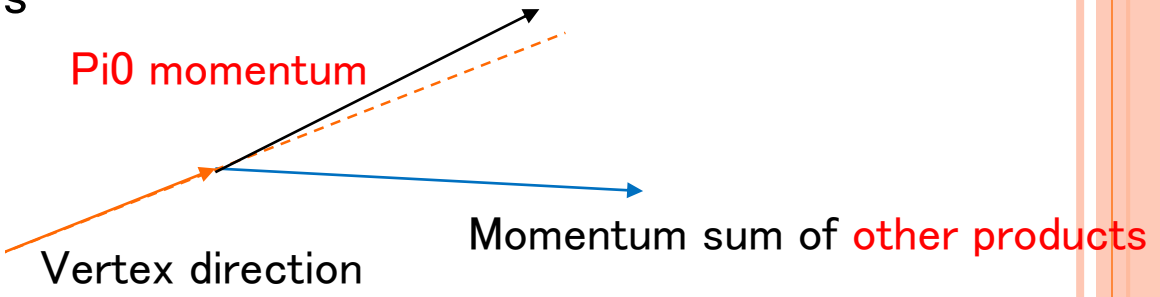
- We are studying the possibility of vertex mass recovery using pi0s
  - Pi0 vertex finder – which vertex is the π^0 coming?
- Finding vertex of pi0s
  - Very difficult to identify vertex – depends on detector configuration
  - Making the best of decay kinematics
  - Using TMVA to find pi0 candidates from the vertex
  - Comparing vertex mass distribution
  - Sample: using qqHH@500GeV samples(so many tracks & pi0s in events)

- Goal: flavor tagging efficiency improvement!

KEY ISSUES

- Pi0s from (secondary, third) vertices are very collinear to vertex direction
  - due to their small masses

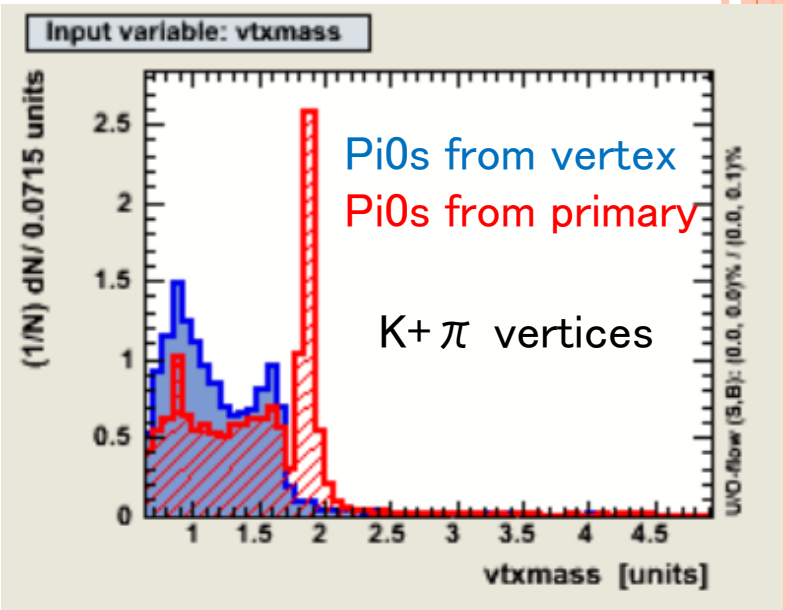




- But, there are many pi0s which come from primary vertex & are accidentally collinear to the vertex direction!
  - Ref.) In qqHH events, 50～60 pi0s will be produced!!

KEY ISSUES

- To avoid attaching too many pi0s:
  - Don't add pi0s in specific conditions →using vertex mass for MVA input
    e.g.) no pi0s will come on D meson peak



  - **Generality can't keep due to this variable!**

→vtxmass spectrum is a hint

※Particle pattern on vertex has different vertex mass pattern!

  - Making wrong mass shift effect smallest
    - Checking pi0s from large energy to small energy
      - Arrange pi0s in descending order of those energies
    - Update vertex momentum when a pi0 candidate is found

→add pi0 4-momentum to vertex momentum, and use it for next pi0 candidate check

Input variables to construct a general classifier

- Getting general – num. of particles are used as input variables
  - Num. of e/ μ / π /K/p in the vertices – using particle ID
  - Those variables are not variables for background rejection, but are variables for vertex classification

  → Do those variables work as variables for vertex classification in the MVA classifier?

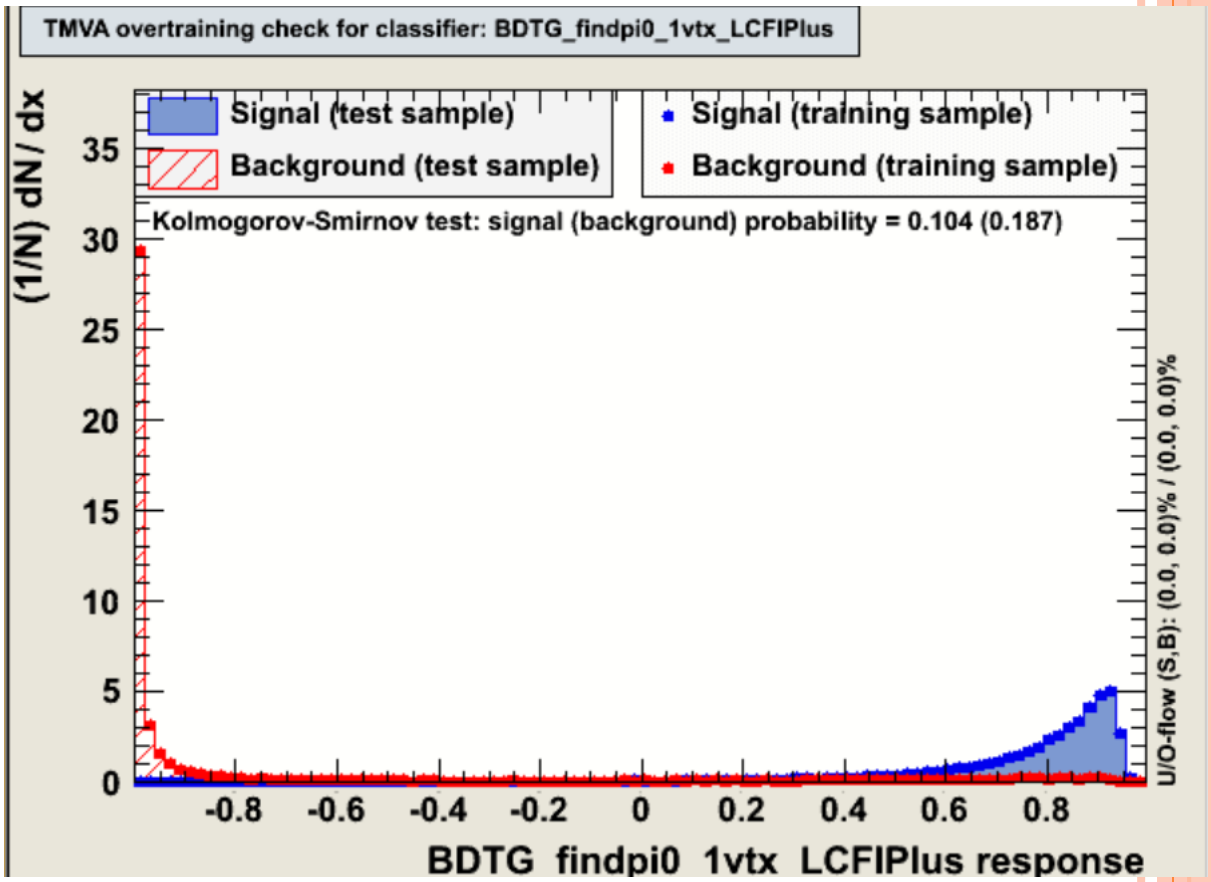
- Num. of tracks in vertices must not be a variable
  - Don't need the bias from num. of tracks on vertices

- 9 Kinematic Variables + 5 num. of particles on vertices for MVA

- Construct the 3 types of MVA classifiers – based on LCFIPlus vertex finding:
  - For third vertices
  - For secondary vertices which have third vertices
  - For secondary vertices which don't have third vertex
  - Using b jets

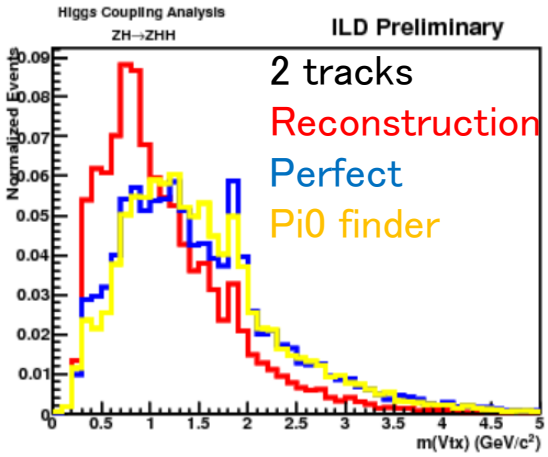
MVA OUTPUT EXAMPLE

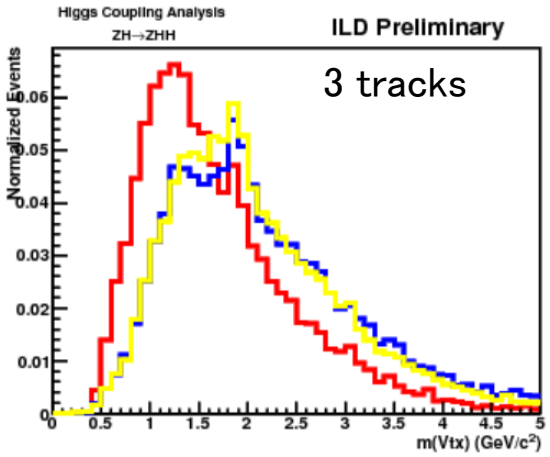
- Signal: pi0s from secondary vertices which don't have third vertex
- Background: pi0s from primary(L_{decay} from IP <0.3mm)
- All the pi0s are assumed to come from secondary vertex
  - Correct gammas & pi0 momentum
- Using Gradient BDT
- MVAcut is optimized

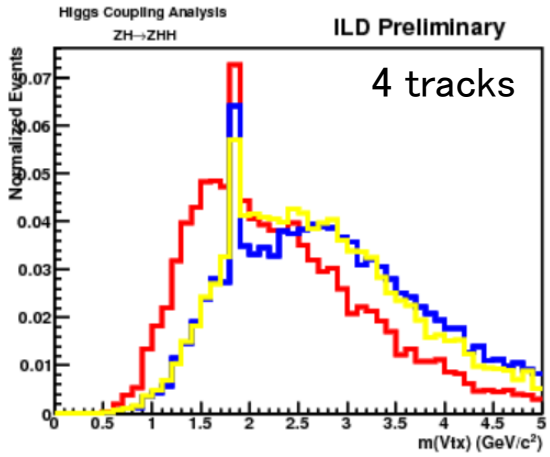


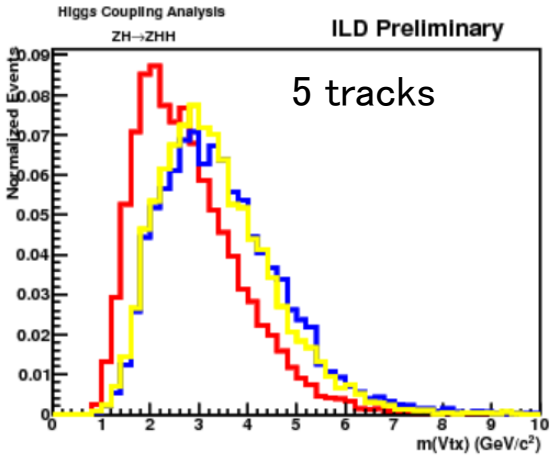
VTX MASSES

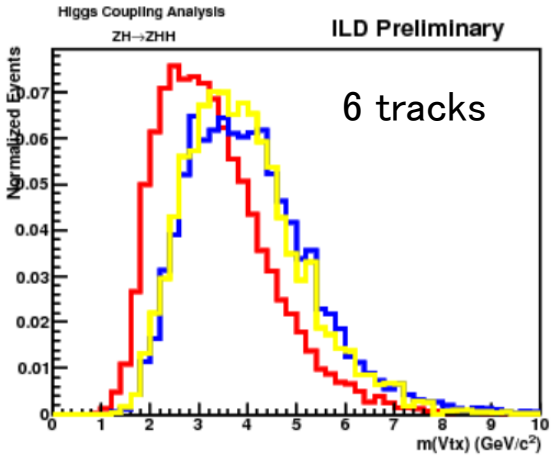
- Vtx mass distributions for each vertex pattern(ntrk)
  - not so bad
  - Difference is coming from mis-pairing of gammas and mis-attachment of pi0s

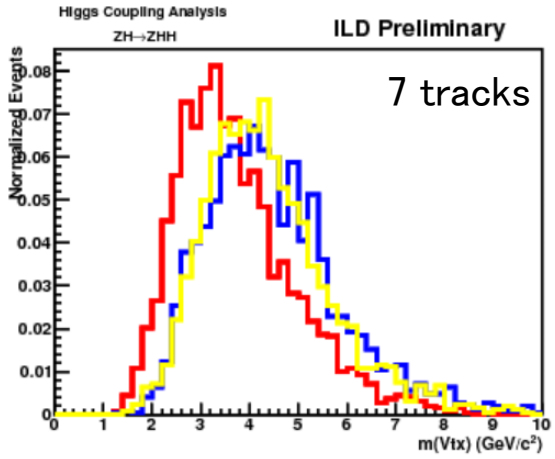






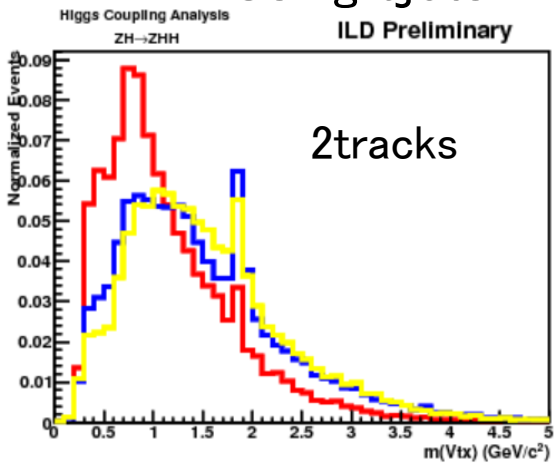


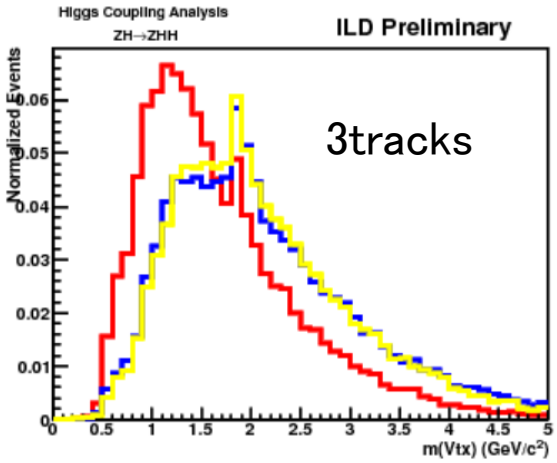


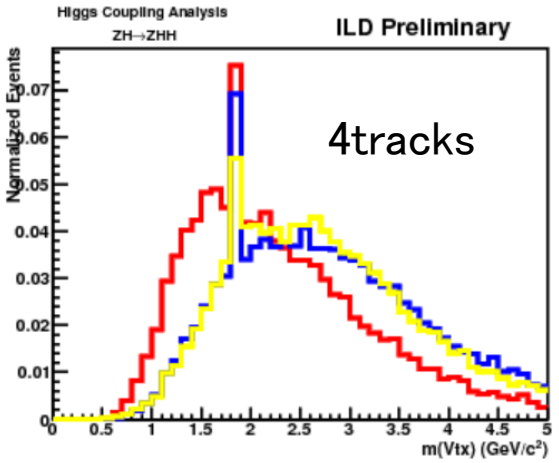


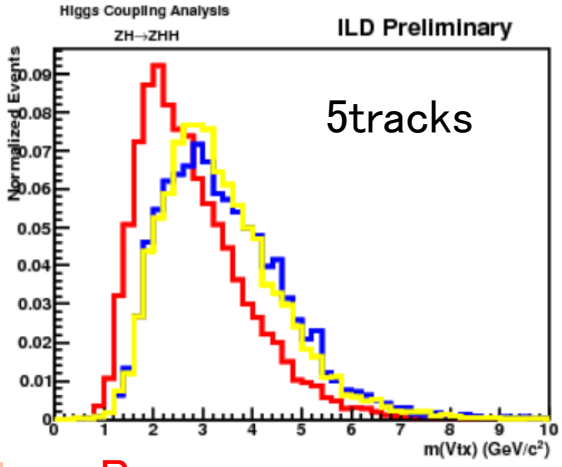
1VTX– SECONDARY VERTICES FROM TOP & Z BOSON

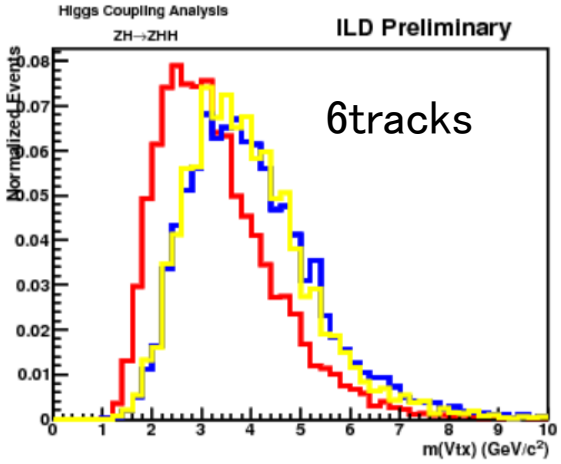
- Use Pi0VertexFinder to b jets from bbcssc sample
  - Using bjets

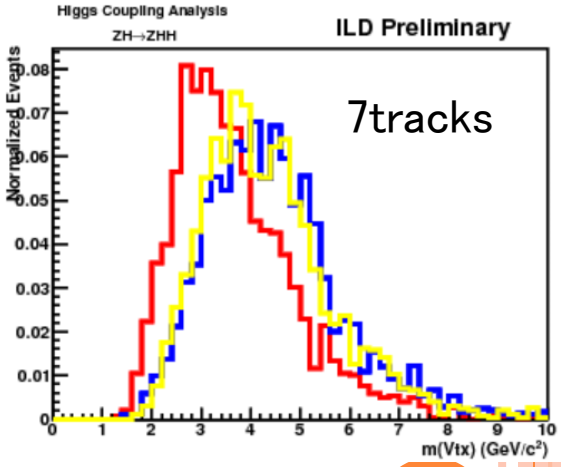












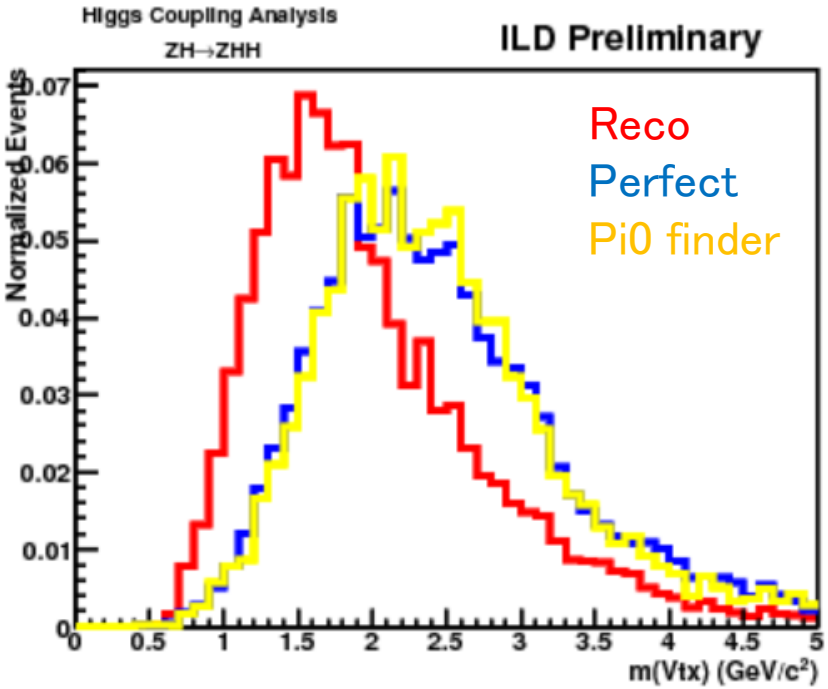
Reco

Pi0&attach perfect

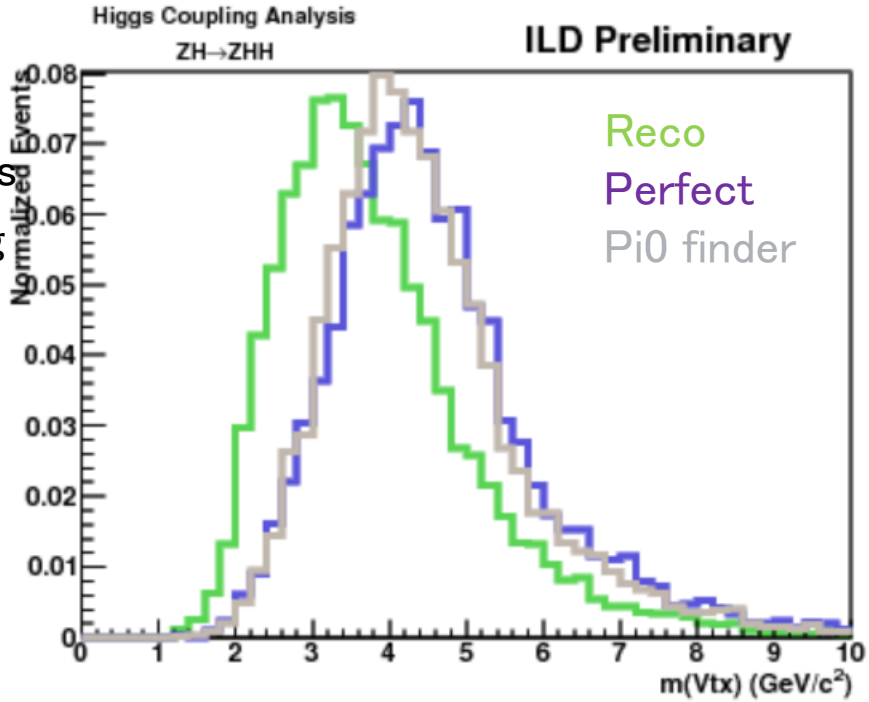
Pi0reco&vertex finder

GLANCE AT OTHER CASE

- 2 vertices in bjet
  - Secondary vertex – 4tracks case



  - Merging with third vertex
    - Third vertices allow all the track patterns
    - Attach pi0s to both of the vertices using pi0 vertex finder



VERTEX MASS RECOVERY EFFECT ON FLAVOR TAGGING

- **Can vertex mass recovery really improve flavor tagging?**
  - Try to construct flavor tagger using recovered vtx mass!
  - Note: this flavor tagger is very "toy" flavor tagger!

- First, checking single variable separation power $\langle S^2 \rangle$:

$$\langle S^2 \rangle = \frac{1}{2} \int \frac{(S(y) - B(y))^2}{S(y) + B(y)} dy$$

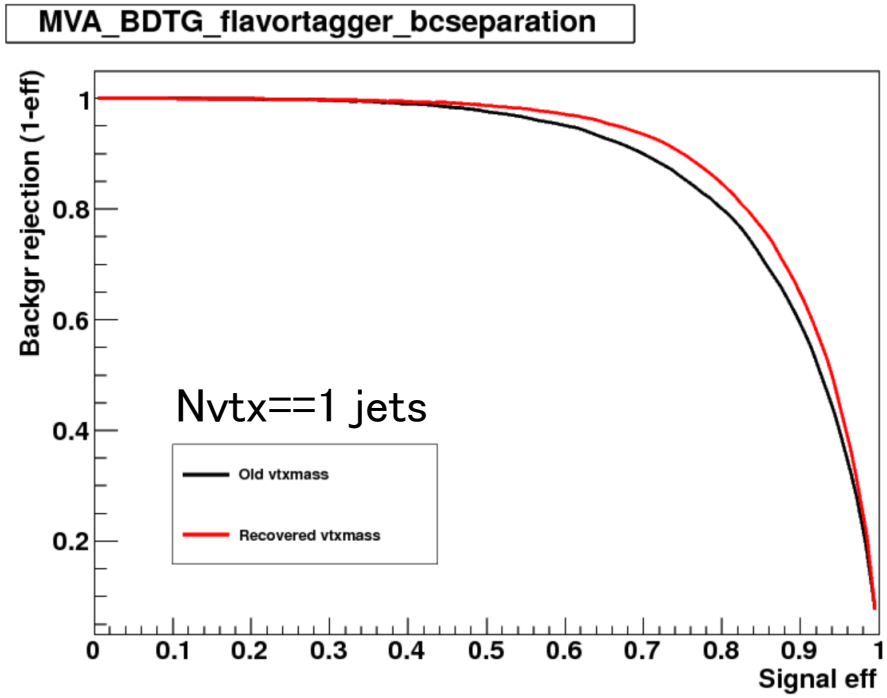
  - $\langle S^2 \rangle$ is from 0 to 1: 0 is no separation and 1 is perfectly separated

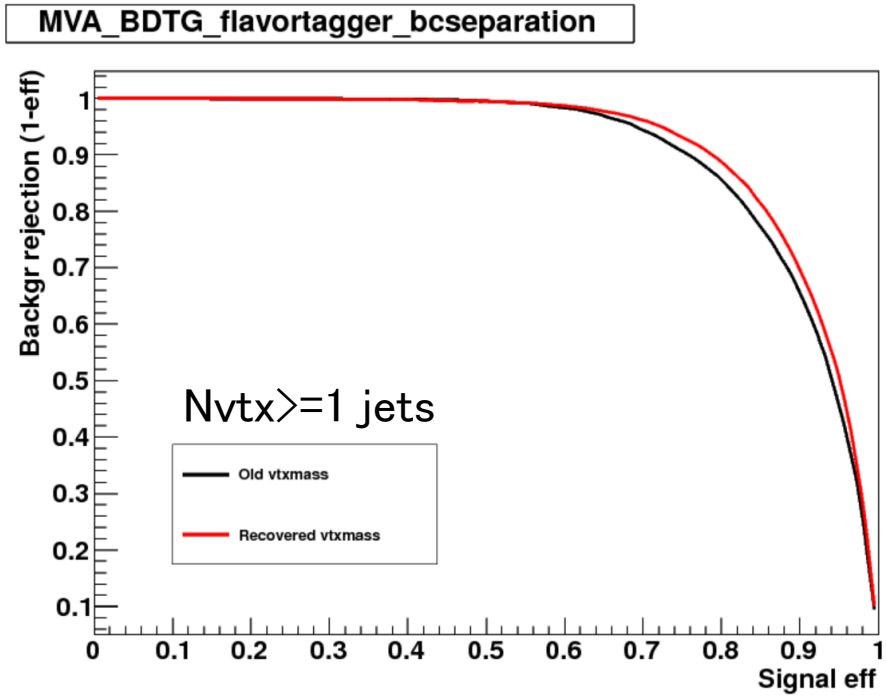
bc separation	Old vtxmass	Recovered vtxmass
2nd vtx (use 1vtx jet)	0.1654	0.2756
2nd vtx (use 2vtx jet)	0.2660	0.2870
3rd vtx (use 2vtx jet)	0.2714	0.3211
bl separation	**Old vtxmass**	**Recovered vtxmass**
2nd vtx (use 1vtx jet)	0.1652	0.1618

  - In b jet vs. l jet case, l jet statistics is too low

Vertex mass recovery effect on flavor tagging

- Construct a “toy” flavor tagger
  - Input variables are obtained from LCFIPlus
  - Input variable selection is too primitive!
  - Only vertex mass is replaced to recovered vertex mass
  - Compare with ROC curve





- For more precise study, need to step into LCFIPlus

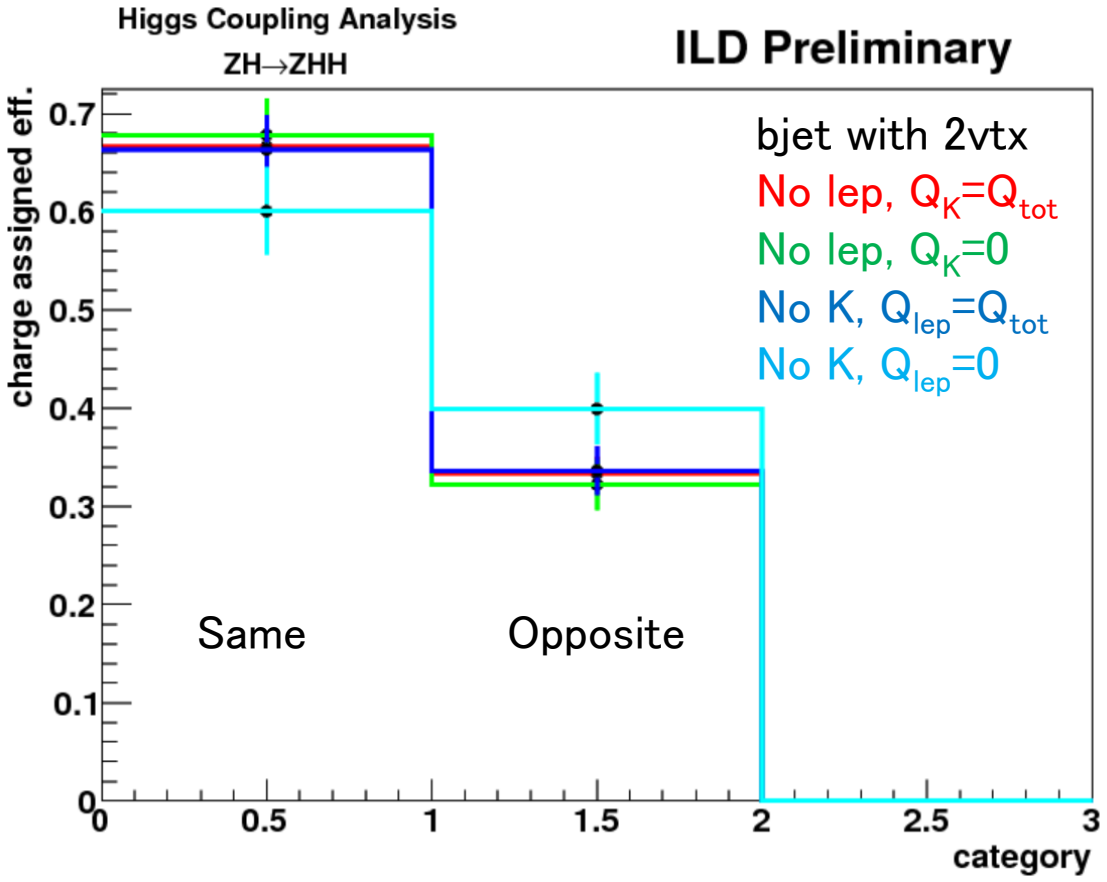
Flavor tagger improvement, on going

- Flavor tagging(LCFIPlus) is one of the key components to obtain better physics result
  - And there is much room to improve

- Now, focusing on
  - Vertex mass recovery → OK! Mention that
  - Flavor separation in 0 vtx case → using CDF style "BNess" tagger(not mention about the detail)
  - Vertex finding(with vtx charge assignment)
- Flavor separation of 0vtx jet is most difficult situation
  - Only impact parameter implies the existence of secondary vertices for flavor separation
- Perfect attachment of all the decay tracks is the key for vertex charge assignment
  - So, improvement of daughter track selection is important
    - Vertex track attachment and fake track rejection
  - As other benefit, vertex finding efficiency will be improved

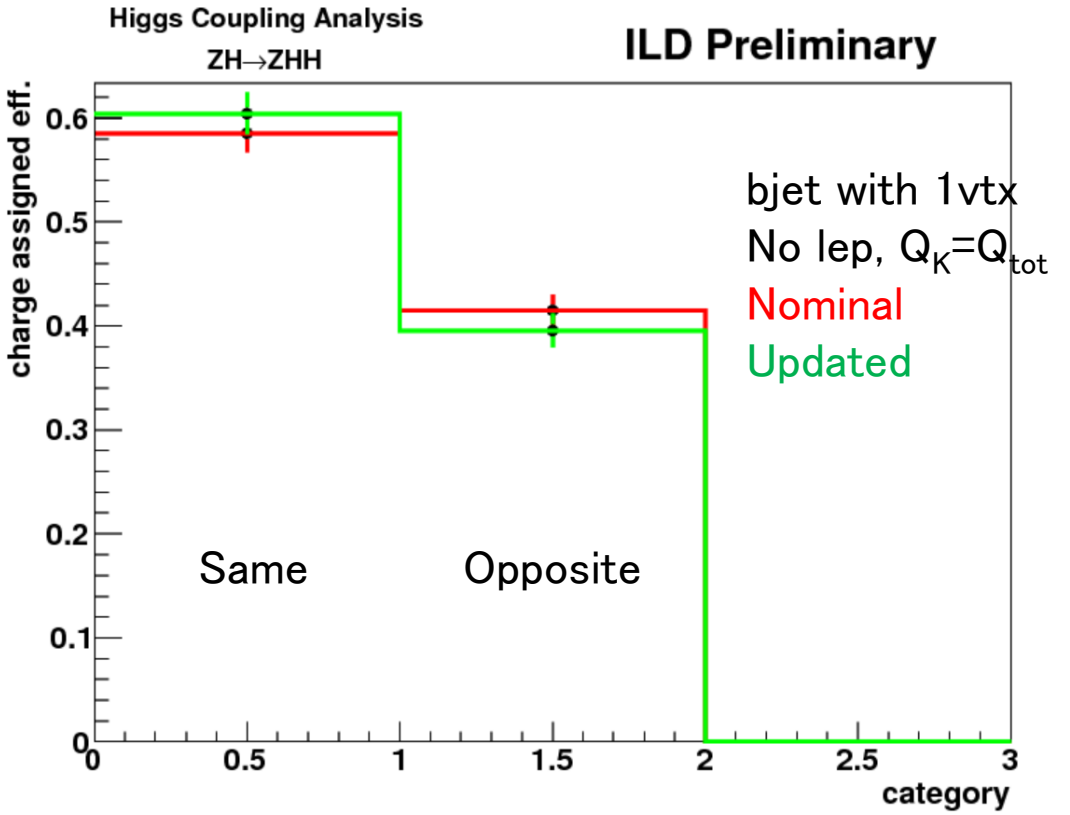
Using Particle ID – vertex classification

- Particle ID can classify vertices by particle set on vertices
- Is there any favor of vertex charge assignment efficiency among particle patterns?
- Example of vertex charge assignment eff.
  - Some tendency can be seen, but within 2σ difference
  - Vertex selection using PID?
  - Or other good idea?



For vertex charge assignment improvement

- Challenge of LCFIPlus improvement is on going
  - To obtain perfect track attachment on vertices
  - To reject fake tracks from vertices
- Some ideas on going:
  - Adaptive Vertex Fitting for vertex finding
  - Using BNess tagger to reject fake tracks
  - Etc.
- One feature is num. of jets with 2 vtx seems increased
  - Good for flavor separation?
  - Vertex quality is OK?
- Improvement can be seen, but not significant…
  - From very slight improvement up to 4% improvement
- need more study



Summary, Problems and Prospects

- Explore some fundamental variables for analysis improvement
  - dE/dx in TPC and shower profile
- Studying particle ID:
  - Hadron ID eff. is 62%～75%
  - Energy correction effect is small, but going to good direction
  - Vertex mass recovery is hopeful using particle ID
- Flavor tagger improvement:
  - There seems hope for attaching pi0s to vertices
    - Vertex mass recovery is reasonable
    - Vertex mass recovery will provide better separation on b/c jets!
  - Recovered vertex mass seems to bring better flavor tagger!
  - Need precise study in LCFIPlus → ongoing
- LCFIPlus improvement for vertex charge
  - So far, improvement seems very slight(～4%), need more study
  - And vertex finding eff. will be improved →leads to better flavor separation!

BACKUPS

First Application – Lepton ID

- Lepton ID for single lepton – using likelihood method
  - Lepton likeliness: $L = \frac{\prod s}{\prod s + \prod b}$,
  - Variables: traditional variables(Ecal/(Ecal+Hcal), E/P, D0, Z0, cone energy)
  - And using dE/dx(convert to χ^2) & shower profiles
- Signal detection efficiency – set almost same efficiency
- Signal is HH→(bb)(WW*)→(bb)(l ν jj)

method	Cut based	Likelihood_old	Likelihood_new
Signal(%)	98.1	98.1	97.8

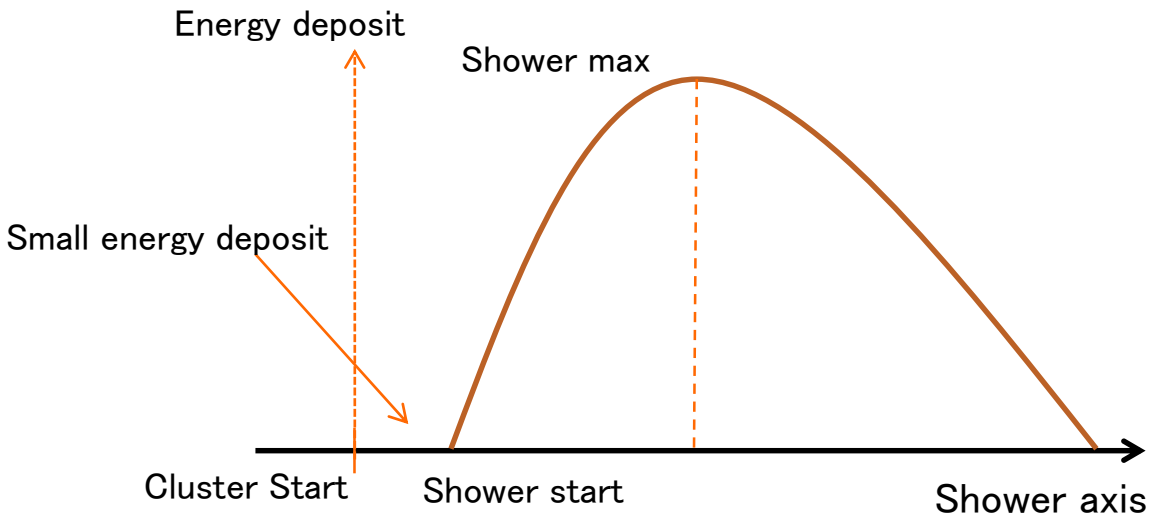
- Background rejection efficiency:

Single lepton ID	Cut based	Old likelihood	New likelihood
Signal(%)	98.1	98.1	**97.8**
ttbar – all hadronic(%)	7.9	3.1	**2.3**

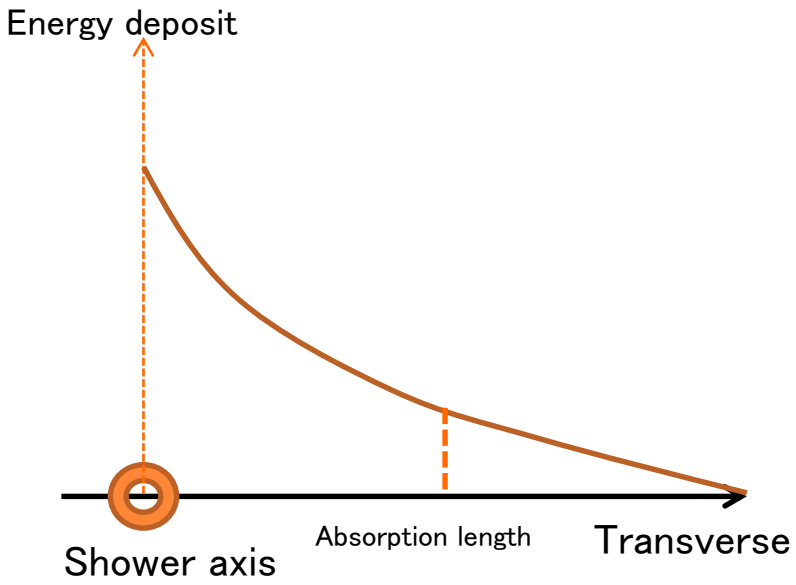
  - Improvement of all hadronic event rejection: ～30%
  - Note: lepton energy threshold is loosened on likelihood_new
    - From E(lep)>15GeV → E(lep)>10GeV

SHOWER PROFILE –STRUCTURE IN THE CLUSTER

longitudinal

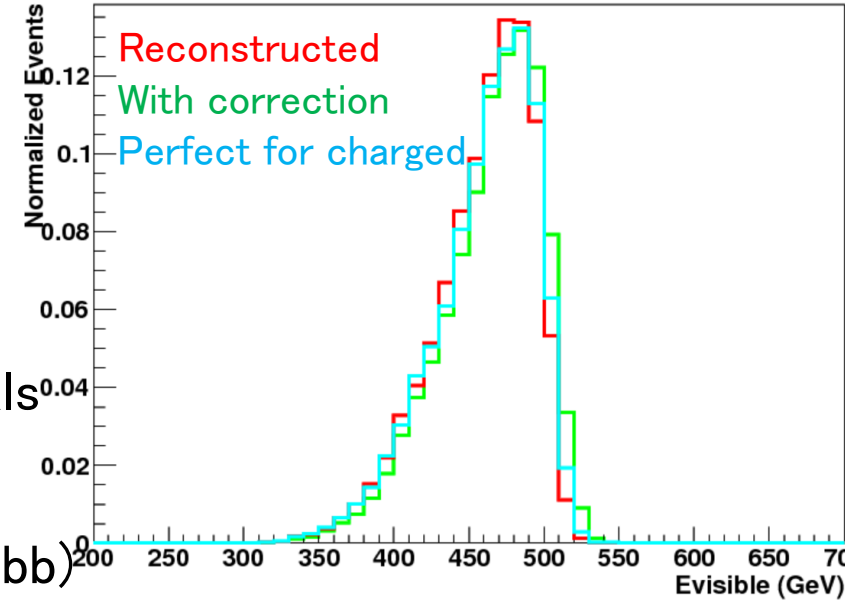


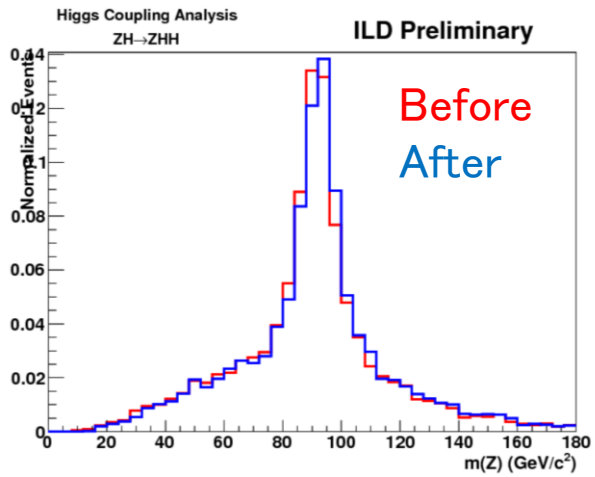
transverse

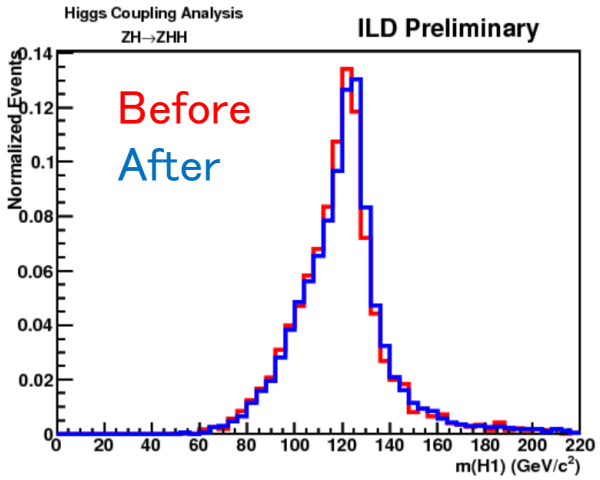


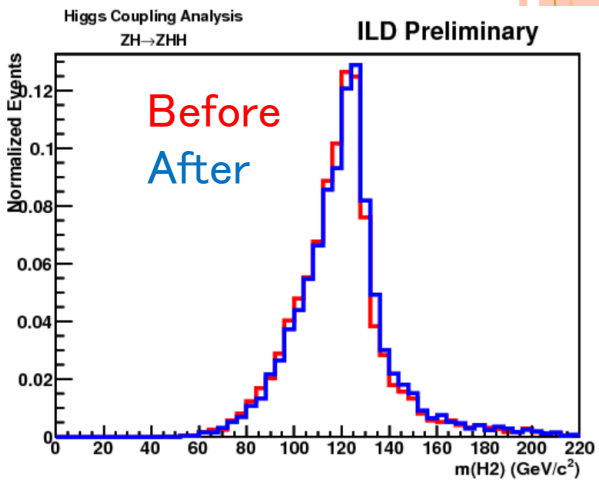
Track energy correction

- Track energies are corrected using those momentum & mass
  - Using particle ID to identify tracks
- Visible energy
  - Using qqHH→qq(bb)(bb)
  - So far, overestimated due to misID
  - Correction effect is small due to neutrals
- Mass distribution
  - Checking Z(Z→qq, q is light) and H(H→bb)
  - Jet matching with MC truth is applied
  - Effect is small too due to neutrals



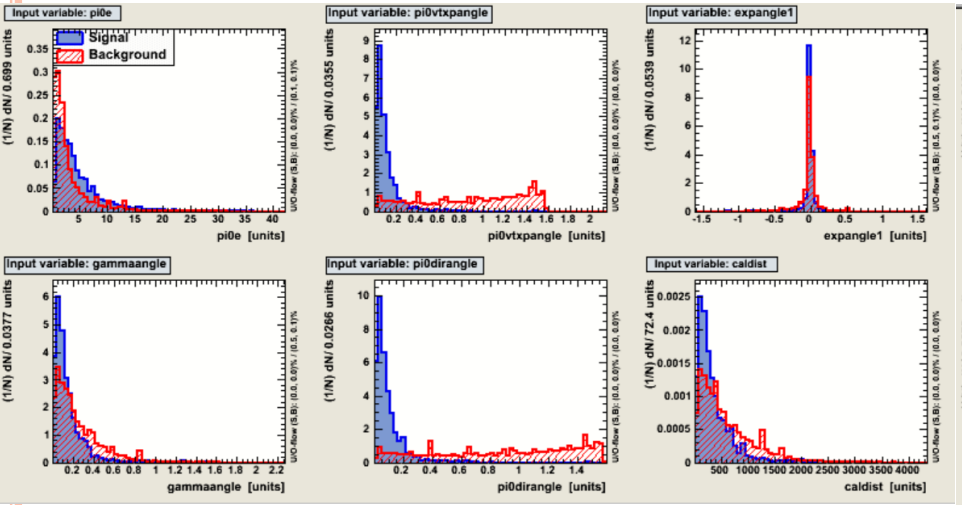


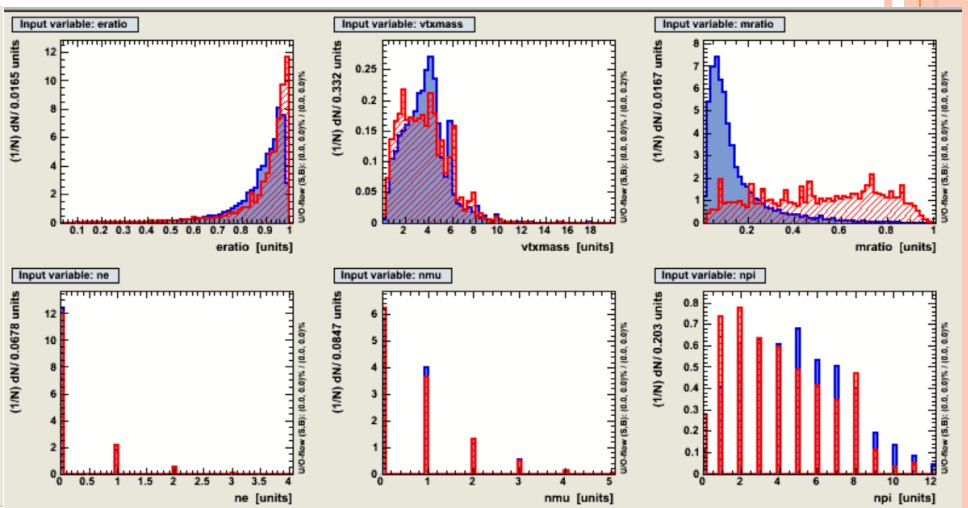




MVA – USING TMVA

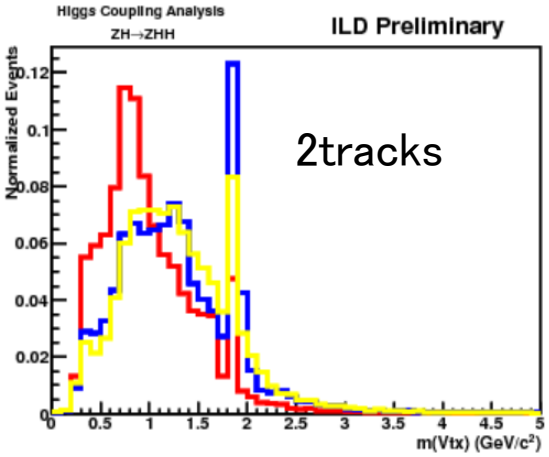
- Input variables to be used
  - Secondary vertices which don't have third vertex

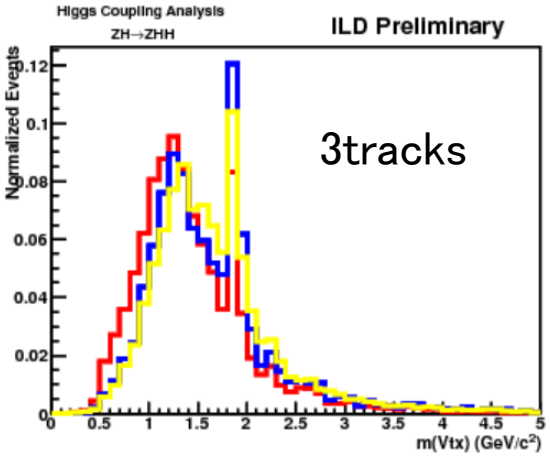


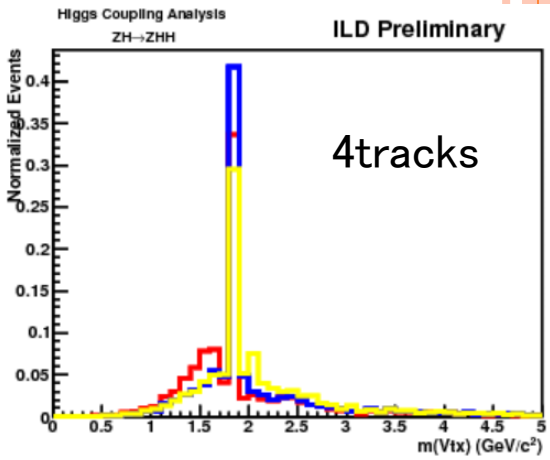


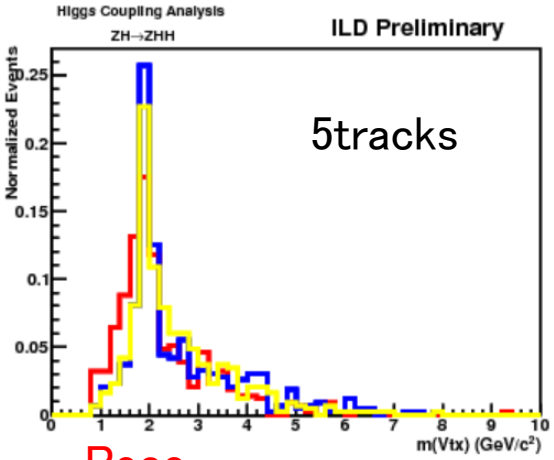
1VTX– SECONDARY VERTICES OF C JETS

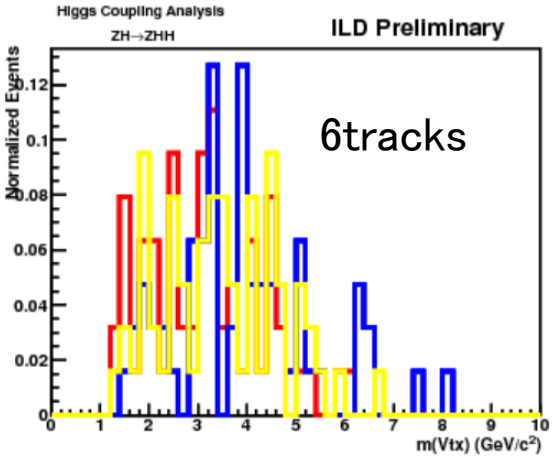
- Use Pi0VertexFinder to cjets
  - Seems to attaching too many pi0s… difficult to keep same performance to bjets and cjets at the same time

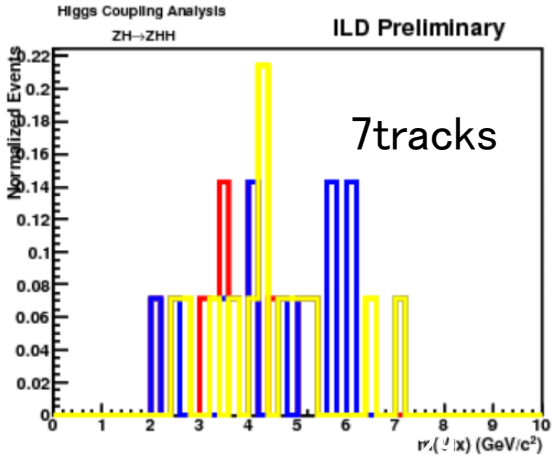












Reco

Pi0&attach perfect

Pi0reco&vertex finder

Some plots

- Num. of pi0s to be attached →determine MVAcut by it
- Where do pi0s really come from?
  - Many pi0s from primary are mis-attached to the vertices
  - Now, that is limited by detector configuration(can't determine exact gamma direction)
  - To some extent, an idea to catch gamma direction is necessary