New Jet Clustering (and its Performance in Physics Studies)

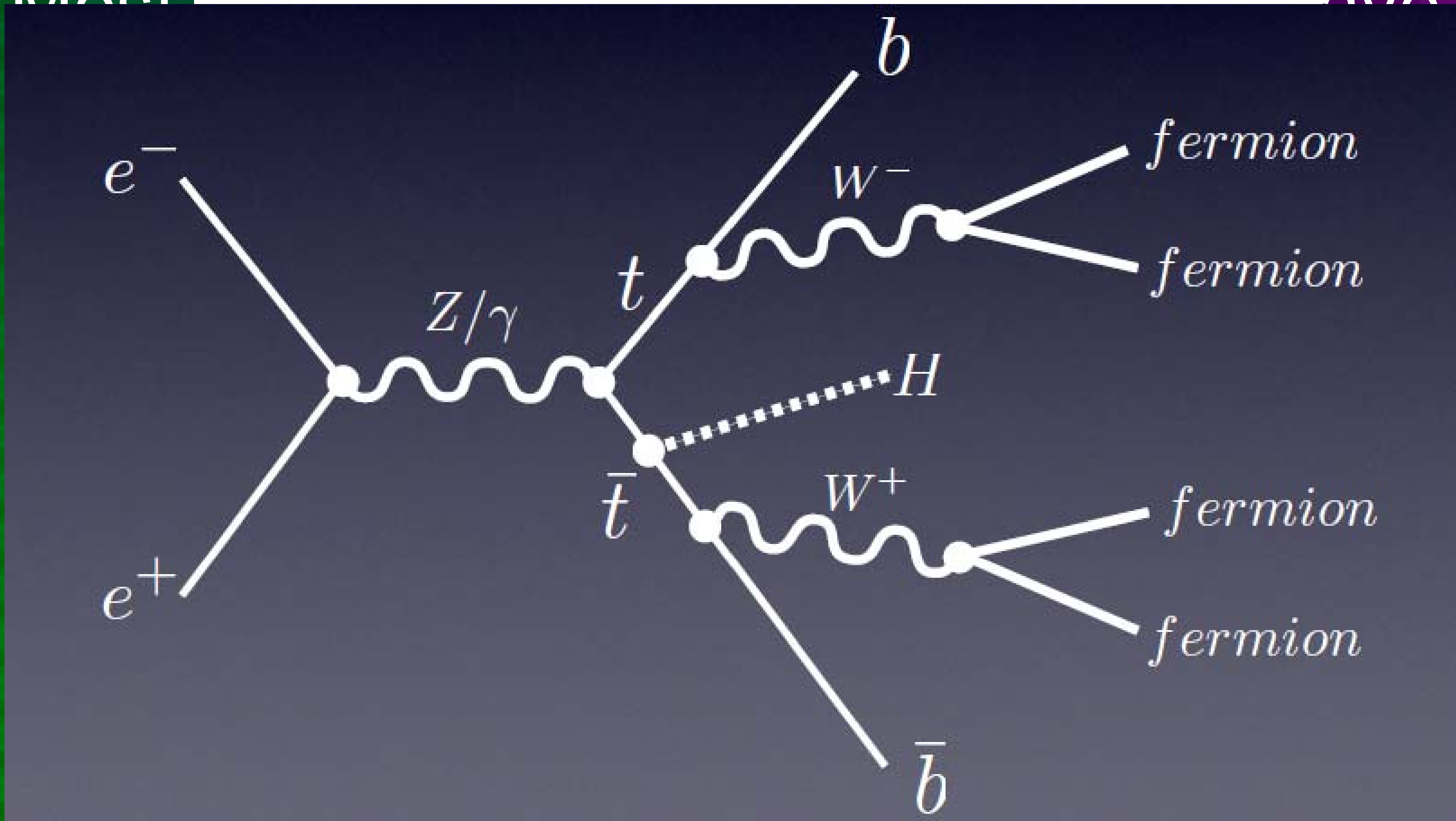
Taikan Suehara, Hiroki Kawahara
(The Univ. of Tokyo)

with support of
K. Fujii(KEK), A. Miyamoto(KEK)
& all ILD optimization, ILC-Asia physics group

Notice

- We started development of the new jet clustering algorithm just a month ago.
- We worked hard for a month, but most of results obtained are still not good, need more study.

Motivation of new jet clustering



– tt: (mostly as background) 6 jets with 2 b-jets

Existing method (Jade/Durham)

1. List all reconstructed particles
2. Calculate 'y' value of every pair of reconstructed particles using their energies and momenta.
   – Jade: $$y = \frac{2E_1E_2(1-\cos\theta_{ij})}{Q^2}$$
   Q is cms energy (constant in a event) Effectively y = (invmass)^2
   – Durham: $$y = \frac{2\min(E_1,E_2)^2(1-\cos\theta_{ij})}{Q^2}$$
3. Pairs of 'y' less than threshold value are associated into one jet.
   – Association order is least-order of y or opening angles
4. Repeat clustering with associated particles treated as a single 'particle'.

An idea of new clustering

- Heavy-quark jets have secondary vertices around jet center.
- Secondary vertices can be a ‘jet-core’.
- Clustering with vertex information can reduce mis-clustering and improve flavor tagging performance.

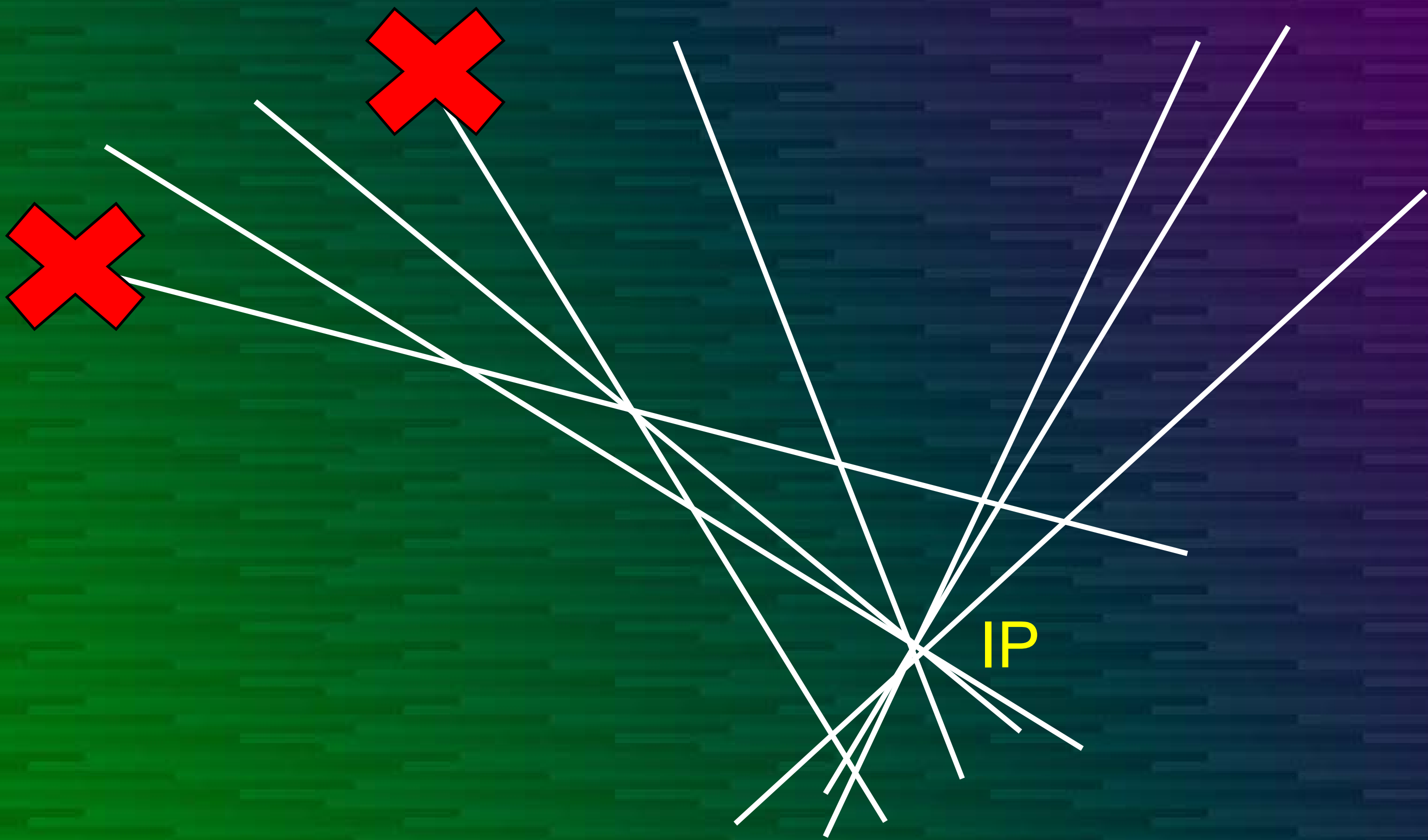
→ Vertex clustering!

Recipe

1. Finding secondary vertices.
   – Currently simple line-track calculation is implemented.
   – Vertex finder used for flavor tagging (ZVRES etc.) will be utilized.
2. Define 'jet-core' lists using secondary vertices.
   – Method to find core of light quarks should be developed.
3. Associate all particles with appropriate cores.
   – Using Y-like measure. Need optimization.
4. Decide # of jets by 'goodness' of clustering.
   – Can be used for selections and reclustering (gluon association) by analysis needs.

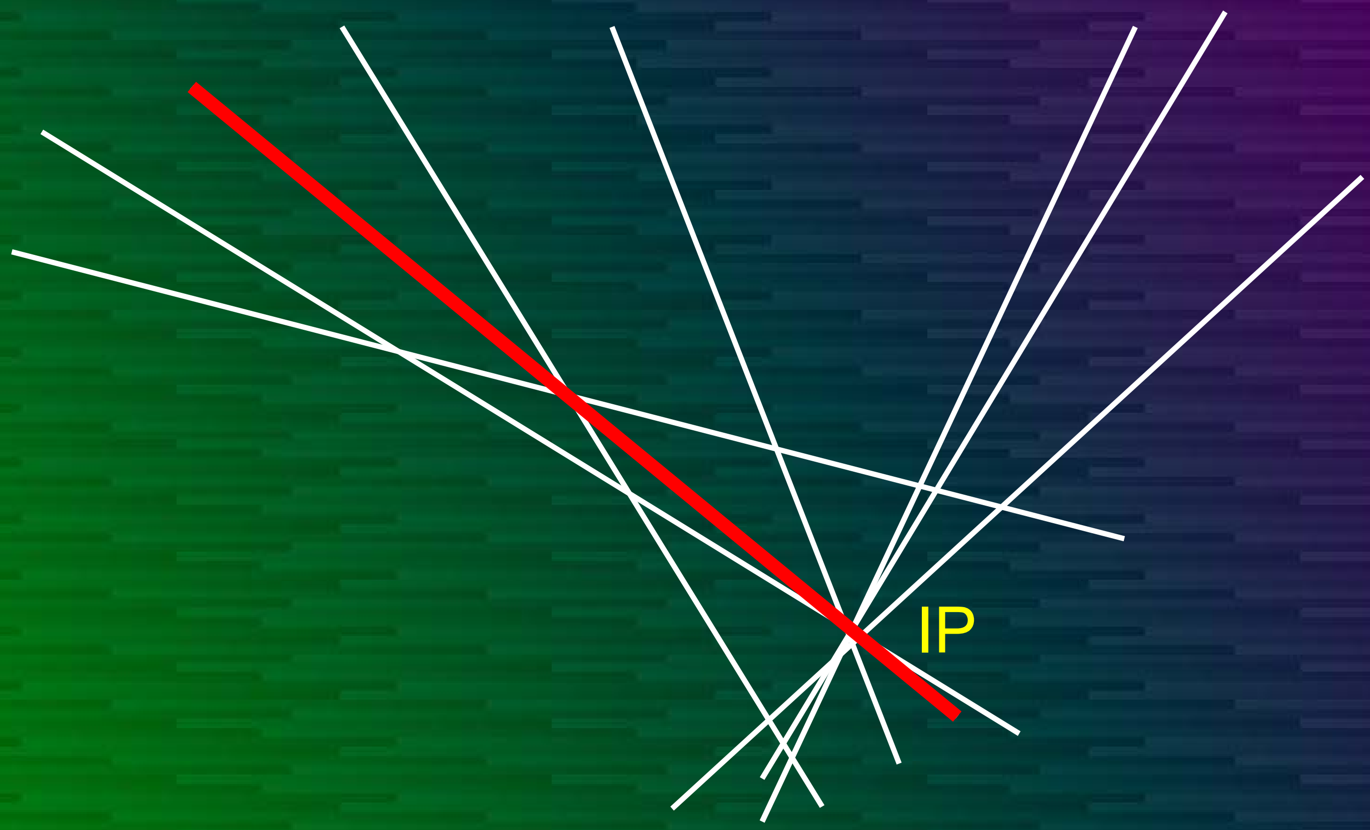
Sample

Traditional jet clustering (not using vertex info)



Sample

Jet clustering with Vertex info



Vertex finder

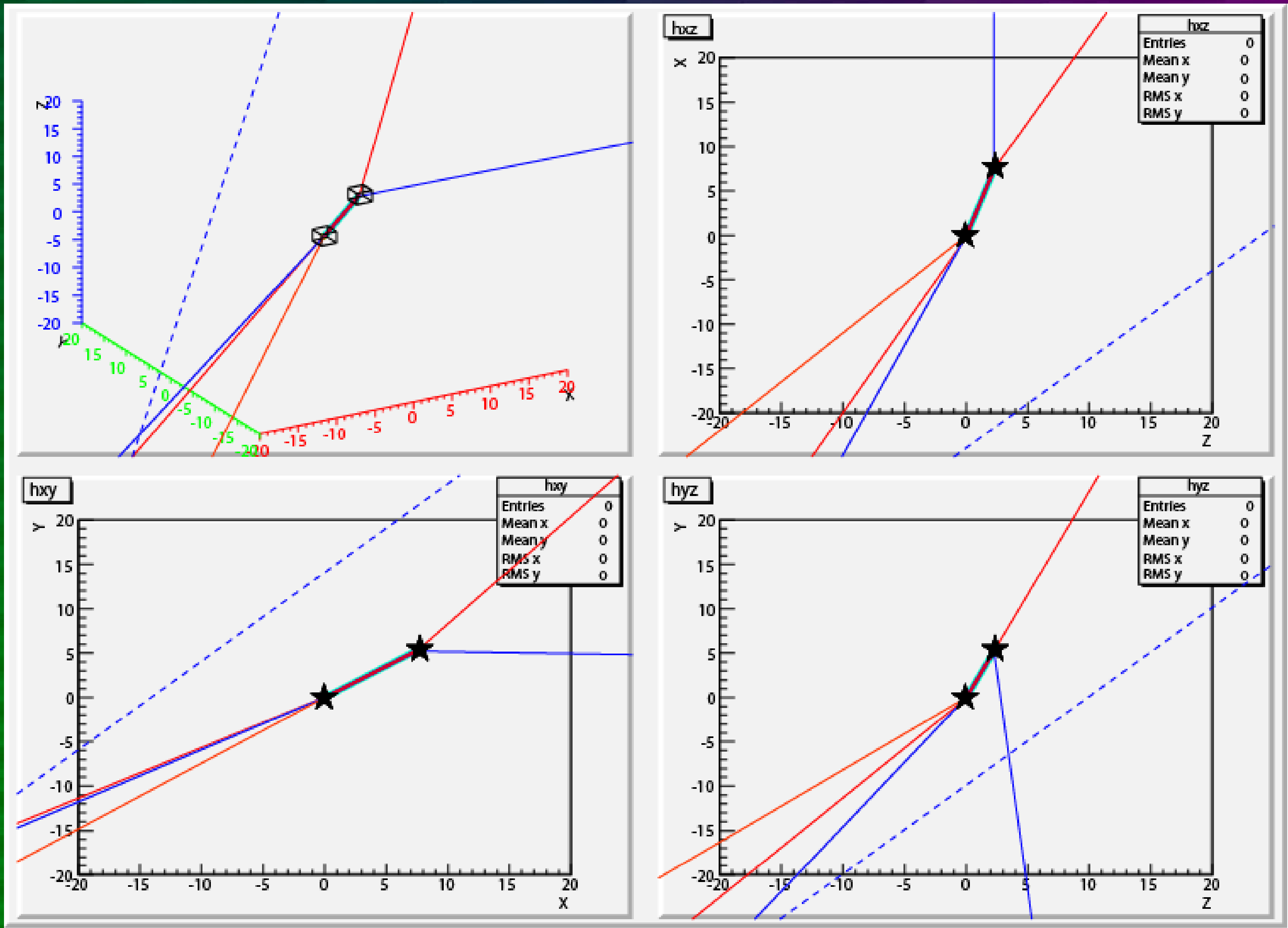
Basic method:

1. List off-vertex tracks (> 100 um)
2. Calculate PCA (point of closest approach) of tracks with the listed off-vertex tracks
3. Points where good PCAs (small distance) are concentrated should be reconstructed vertices. Concentrated tracks (>3) are associated to one ‘Combined particle’.
   Momentum direction is decided by vertex position, not sum of momenta of particles.
4. PCAs away from the IP with very small distance between tracks are also treated as vertices.
   (even if no other particles pass near the PCA).

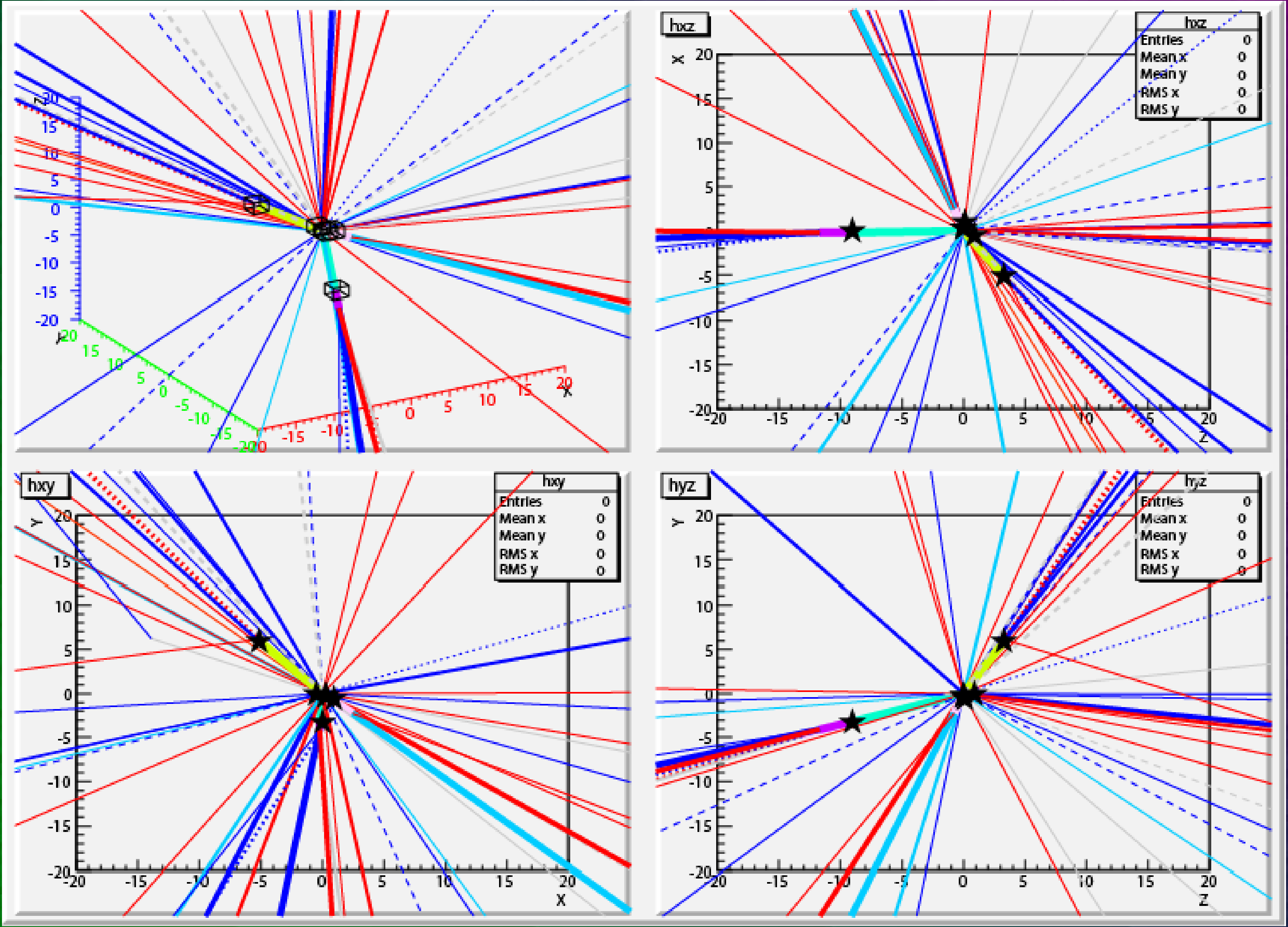
Event samples

- bb (20+20, 50+50, 100+100, 250+250 GeV)
  - No ISR
  - MC: Jupiter, GLD' with MarlinReco
  - 10000 events each
- bb (Slac SM sample)
  - With ISR
  - MC: Mokka, LDC' with MarlinReco
- bbcssc (Slac SM sample)
  - ttbar sample for performance study
  - ~100000 events

Sample (bb50+50 noISR event)



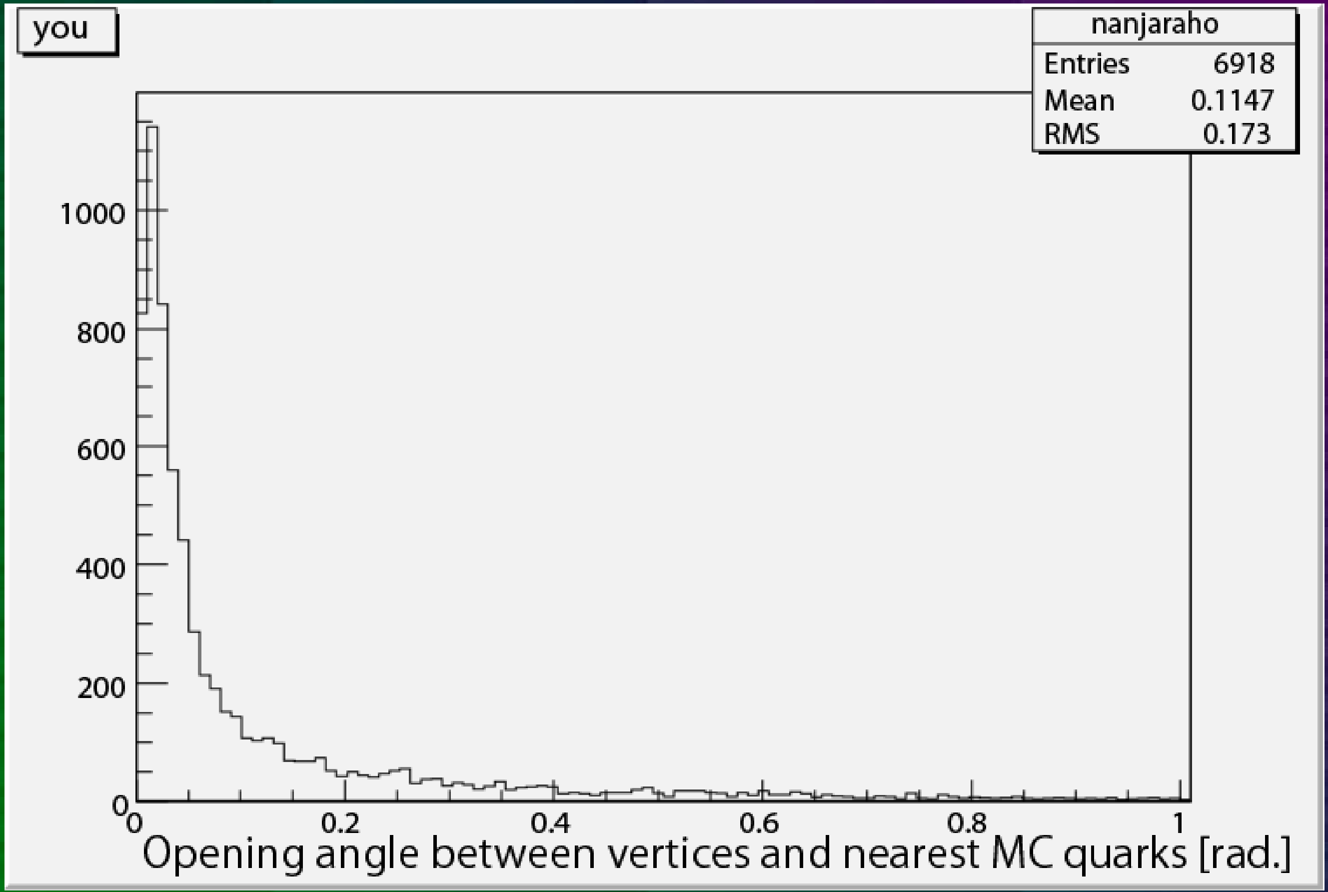
Sample (tt event)



Performance of Vertex finding

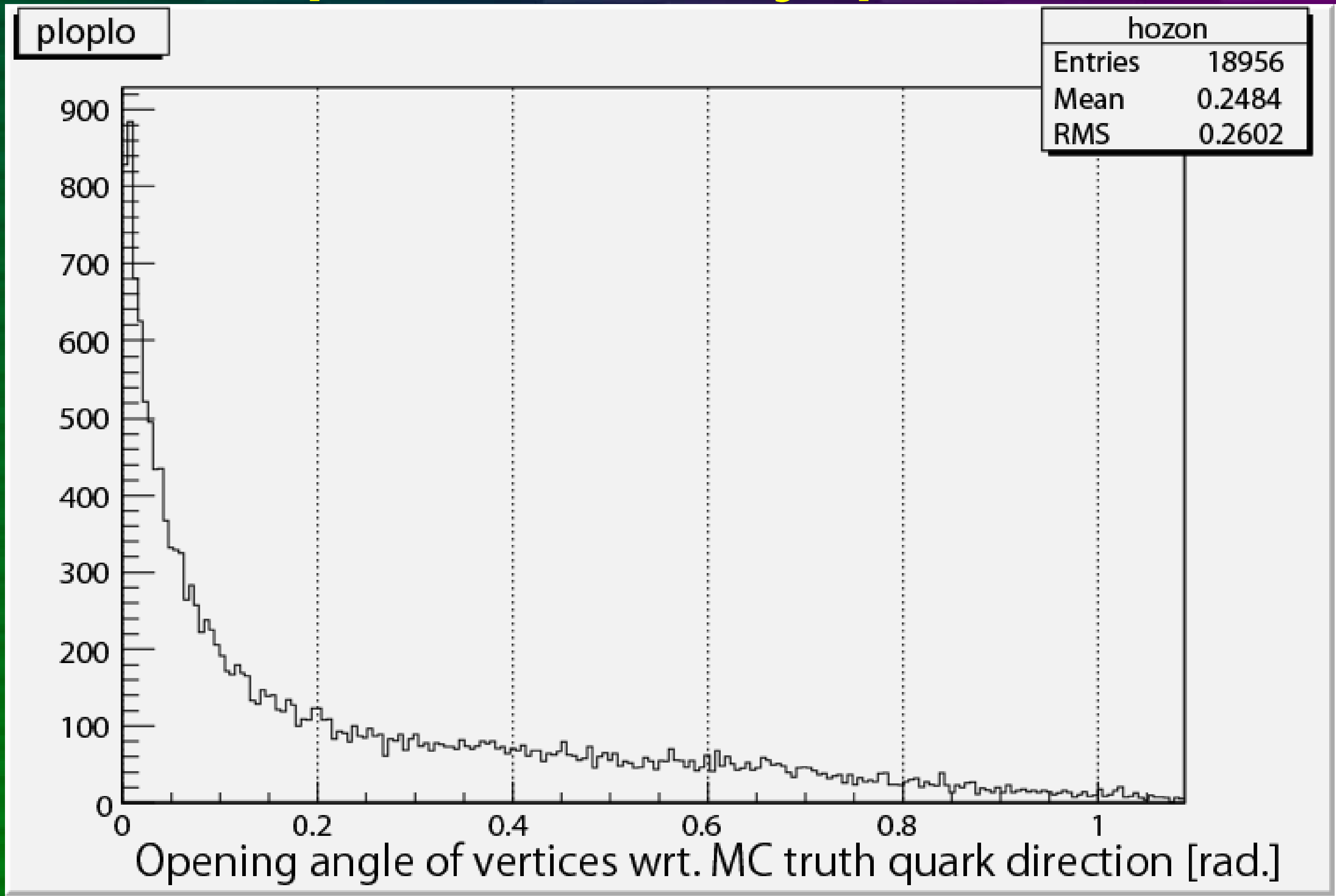
- bb- Jupiter events
  - ‘MC truth b-quark’ directions are compared to reconstructed vertex direction
  - In Jupiter, MC truth quark direction is recorded after gluon emissions so quark directions are almost the same as B-meson (or baryon) directions.
- tt- Mokka events
  - In Mokka, MC truth quark directions are before gluon emissions, and quark directions are sometimes far away from B-mesons due to hard gluon emissions.

bb-Jupiter preliminary performance



~70% of reconstructed vertices have < 0.1 rad to MC quarks.
Vertex finding efficiency is ~70%/b quark.
Optimization is ongoing, should be improved further.

tt-Mokka preliminary performance



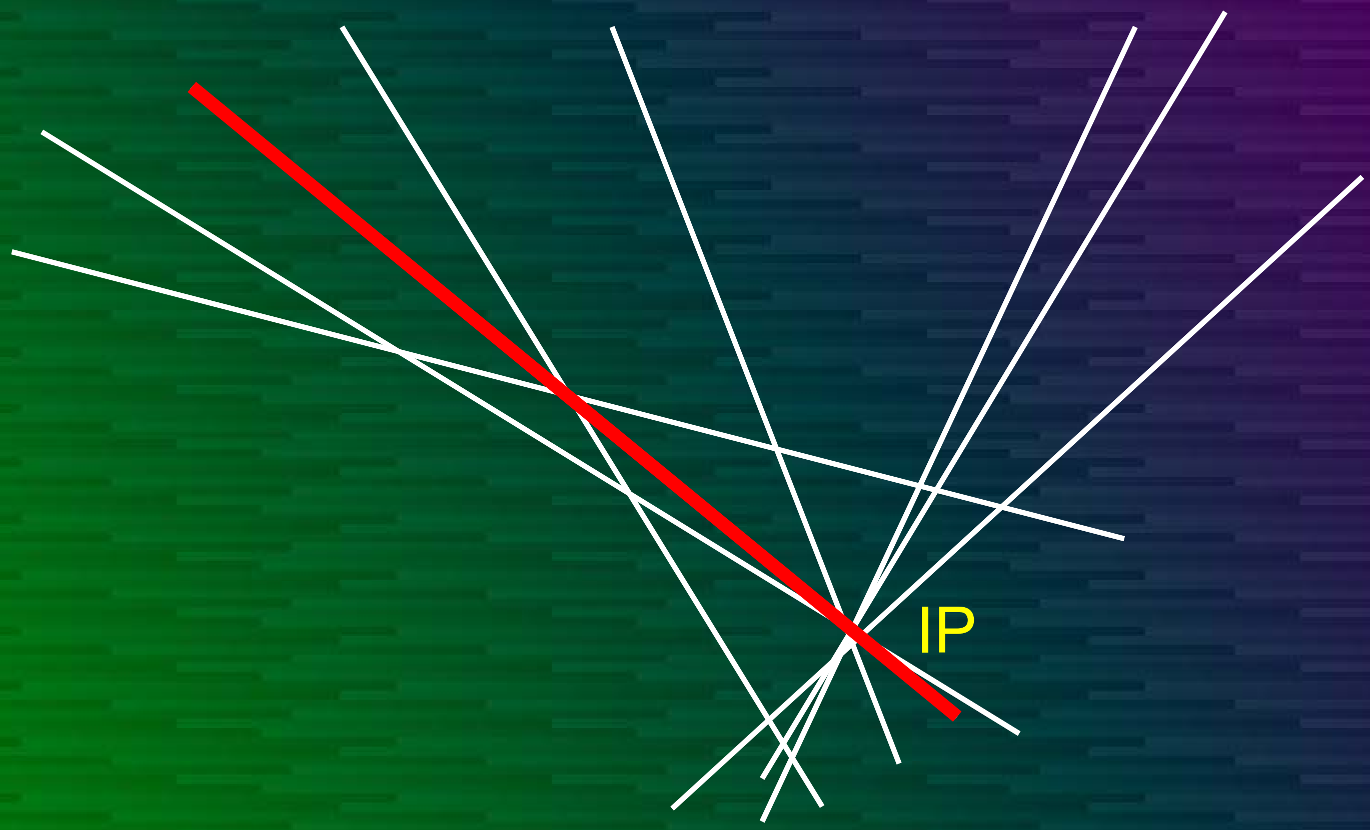
- Slightly worse angular distribution than bb

Defining 'Jet core-candidates'

1. List each reconstructed vertex as a jet core-candidate, as 'a (virtual) particle' with
   - Energy summing energy of all element particles
   - Momentum direction the same as vertex direction
2. Finding other jet cores among rest particles
   1. List all particles not from secondary vertices.
   2. Remove particles with 'y' to existing jet core candidates less than a threshold from the list.
   3. Move most energetic particles remained to a core-candidate.
   4. Go back to 2.

Sample

Jet clustering with Vertex info



Finding secondary vertex
Off-vertex tracks
Forming jet cores

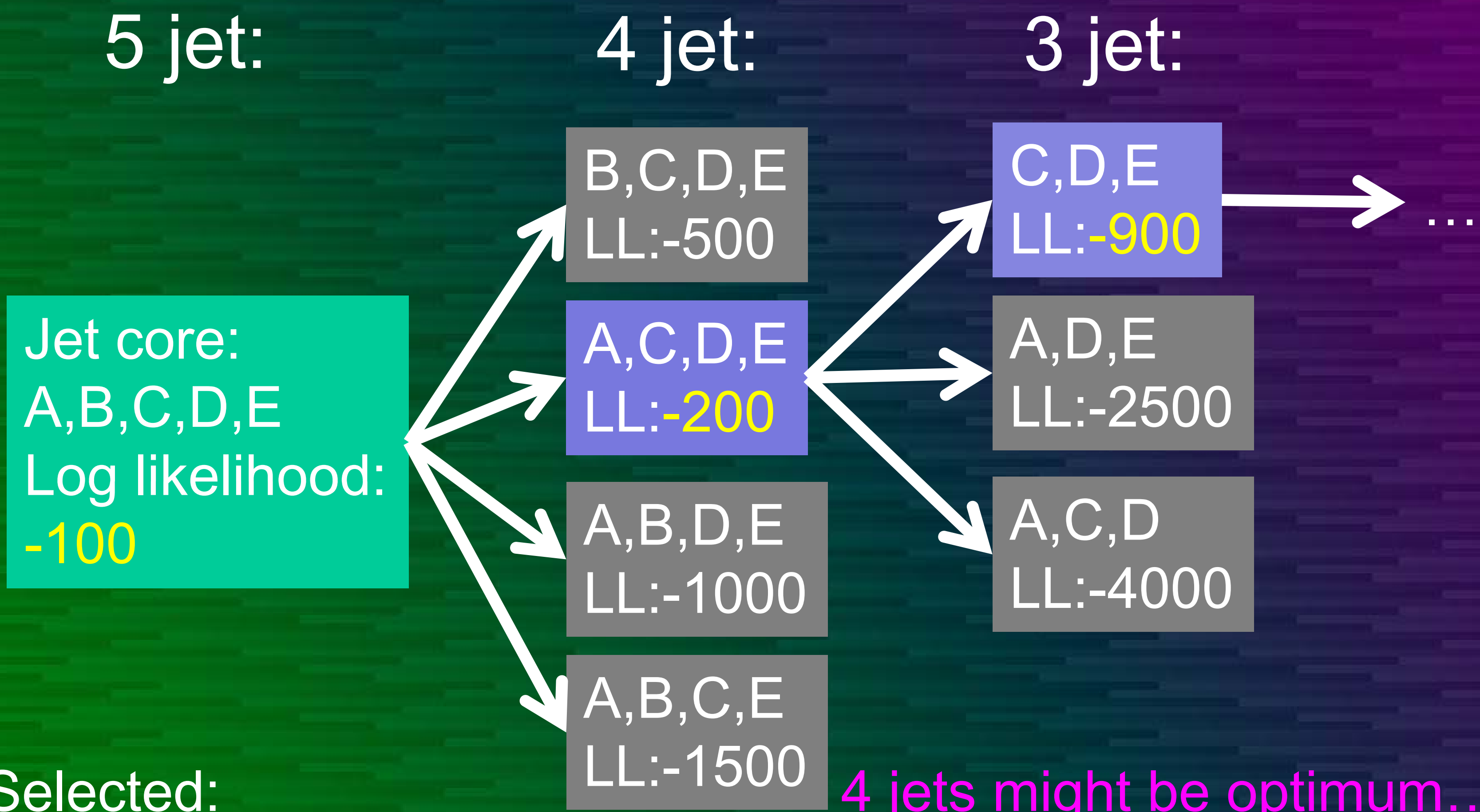
Associate fragments

1. List non-core particles as an energy order.
2. Associate the head particle of the list to a core-candidate which gives least y value with it.
3. Combine energy and momentum of associated particle to the associated jet-core.
4. Repeat from 2.

Reducing core-candidates

- Number of core-candidates is usually much larger than expected number of jets (depending y definition and threshold).
- Reduction method
  1. Move one of candidates to the non-core list.
  2. Associate all fragments (as previous slide).
  3. Calculate 'likelihood' value based on y.
  4. Sum loglikelihood of all particles with weight on energy of each particle.
  5. Try every candidates to move to non-core.
  6. Remove a candidate which gives largest sum of loglikelihood when it is removed from candidate list.
  7. Repeat until obtaining preferred number of jets.

An example for jet reduction



4 jets might be optimum…

Selected:

ABCDE: LL=-100 **ACDE: LL=-200** CDE: LL=-900

The least LL values for each # of jets
can be used to identify 'optimum' number of jets.

Considering Y definitions

- Three y definition has been examined.
  1. Durham
  2. Jade
  3. KS (original)

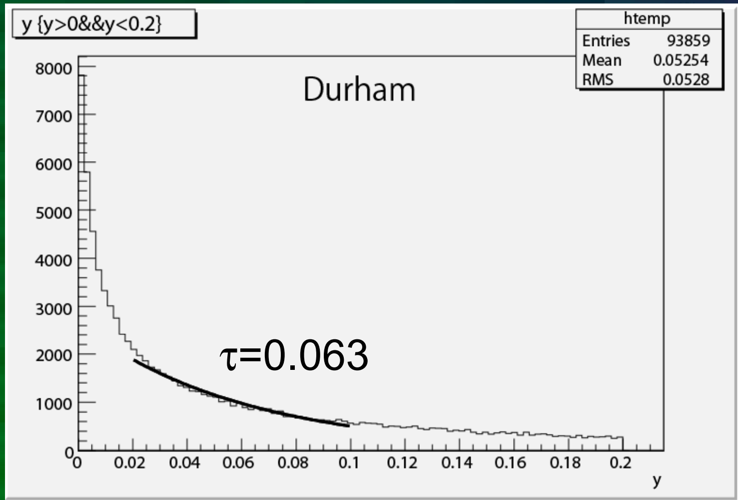
$$y = \frac{2\min(E_1, E_2)^2(1-\cos\theta_{ij})}{Q^2}$$

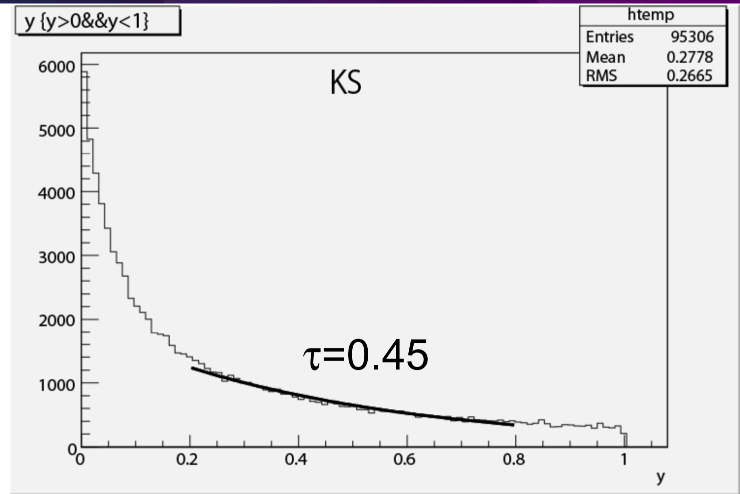
$$y = \frac{2E_1E_2(1-\cos\theta_{ij})}{Q^2}$$

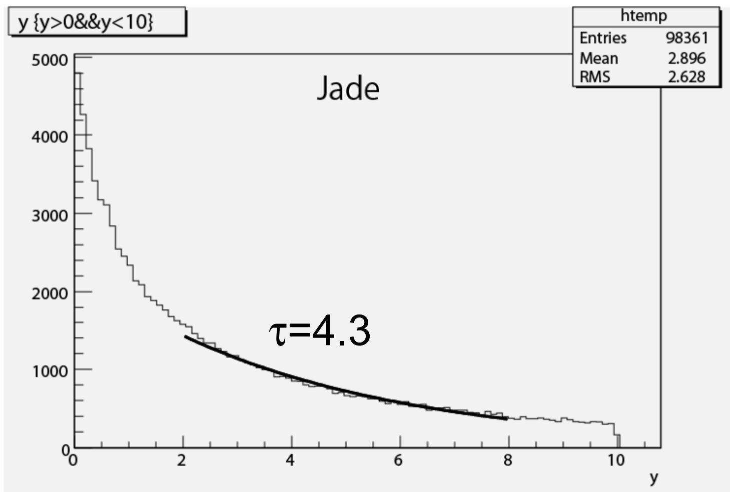
$$y = \frac{2E_1E_2(1-\cos\theta_{ij})}{\sqrt{E_1E_2}}$$

- Likelihood function is determined from y distribution by each definition in ttbar events.
- Performance on clustering is compared.

Distribution of Y in ttbar events



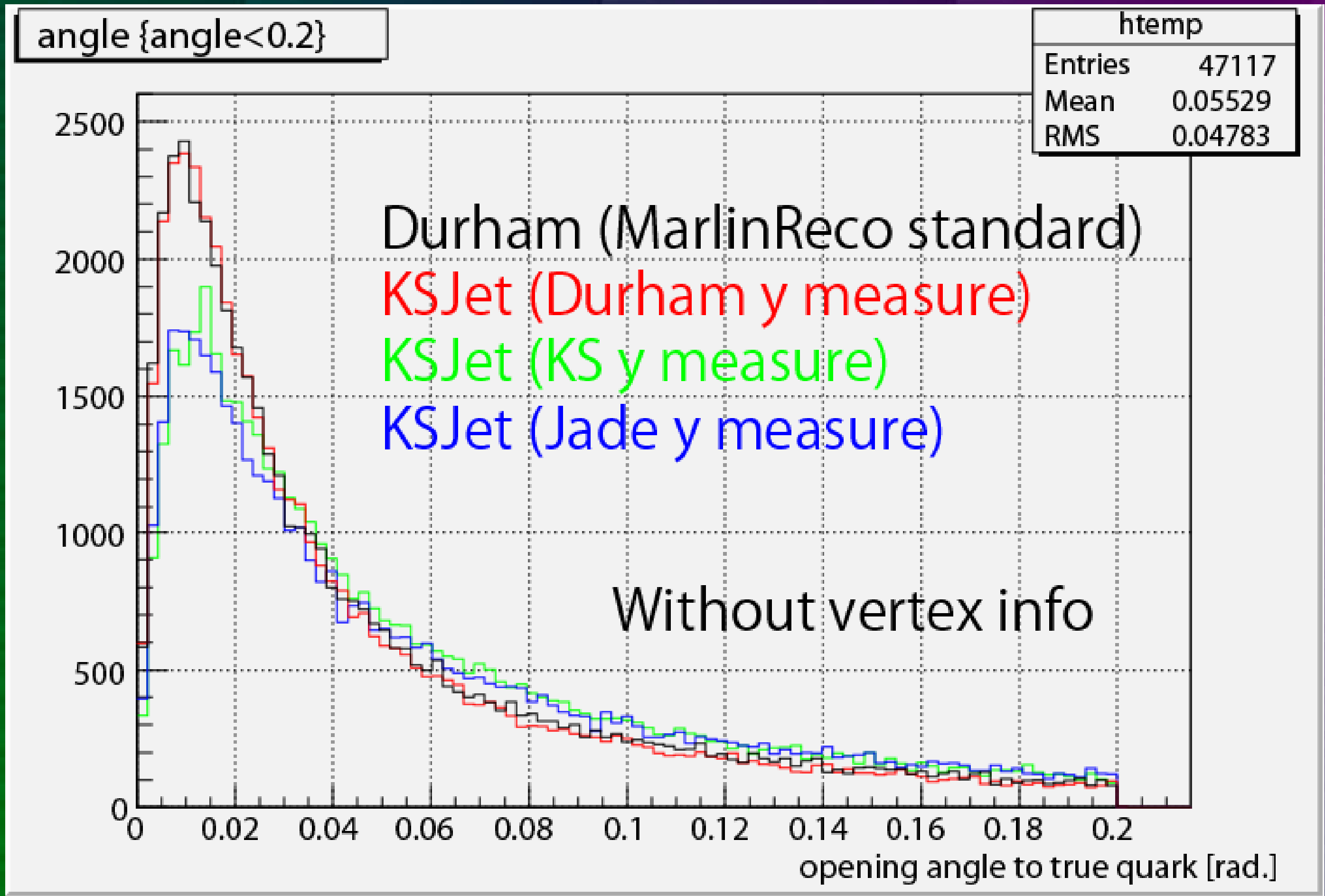




Use integrals (to infinite) of exponential fit functions as Likelihood functions. (normalized to 1 at y=0)

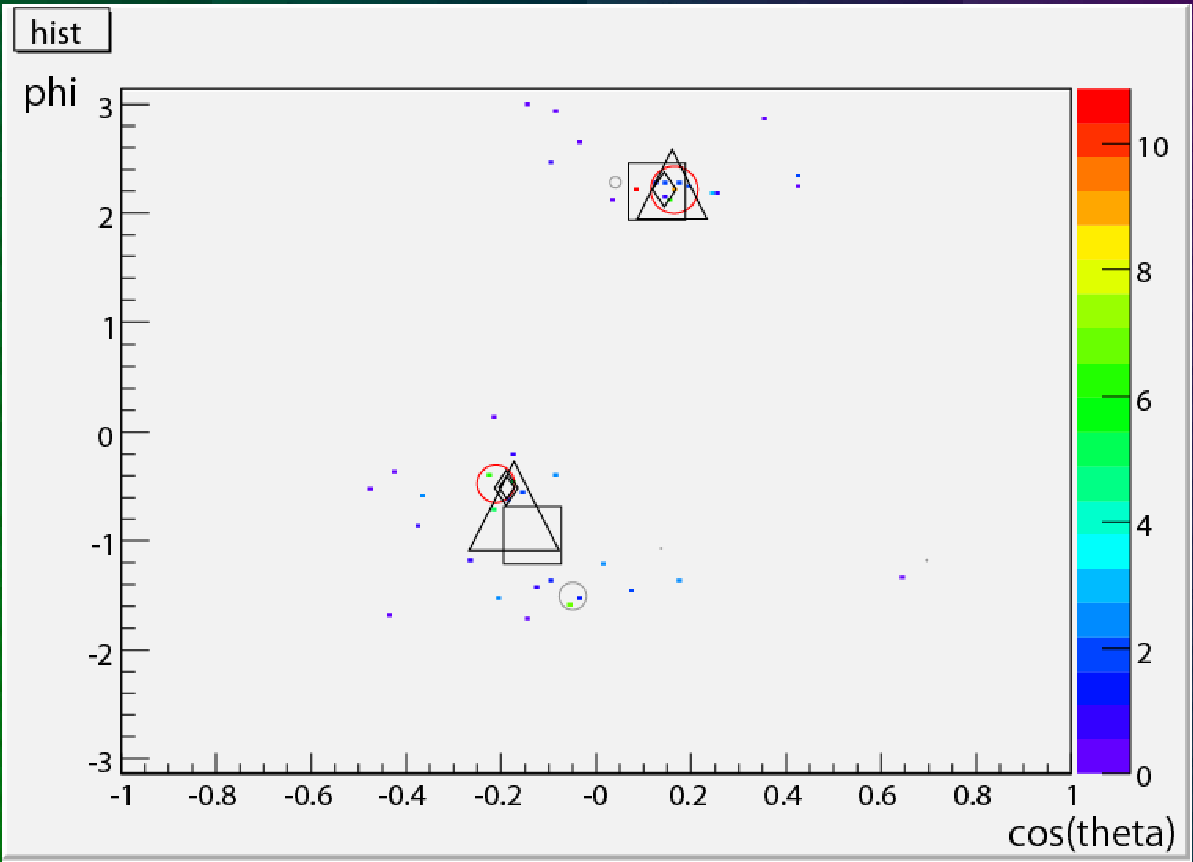
$$\mathcal{L} = \int_y^\infty \frac{1}{\tau} \exp\left(-\frac{t}{\tau}\right) dt = \exp\left(-\frac{y}{\tau}\right)$$

Clustering without vertex info



Standard Durham and KSJet with Durham y give almost the same results. Jade and KS y give worse results.

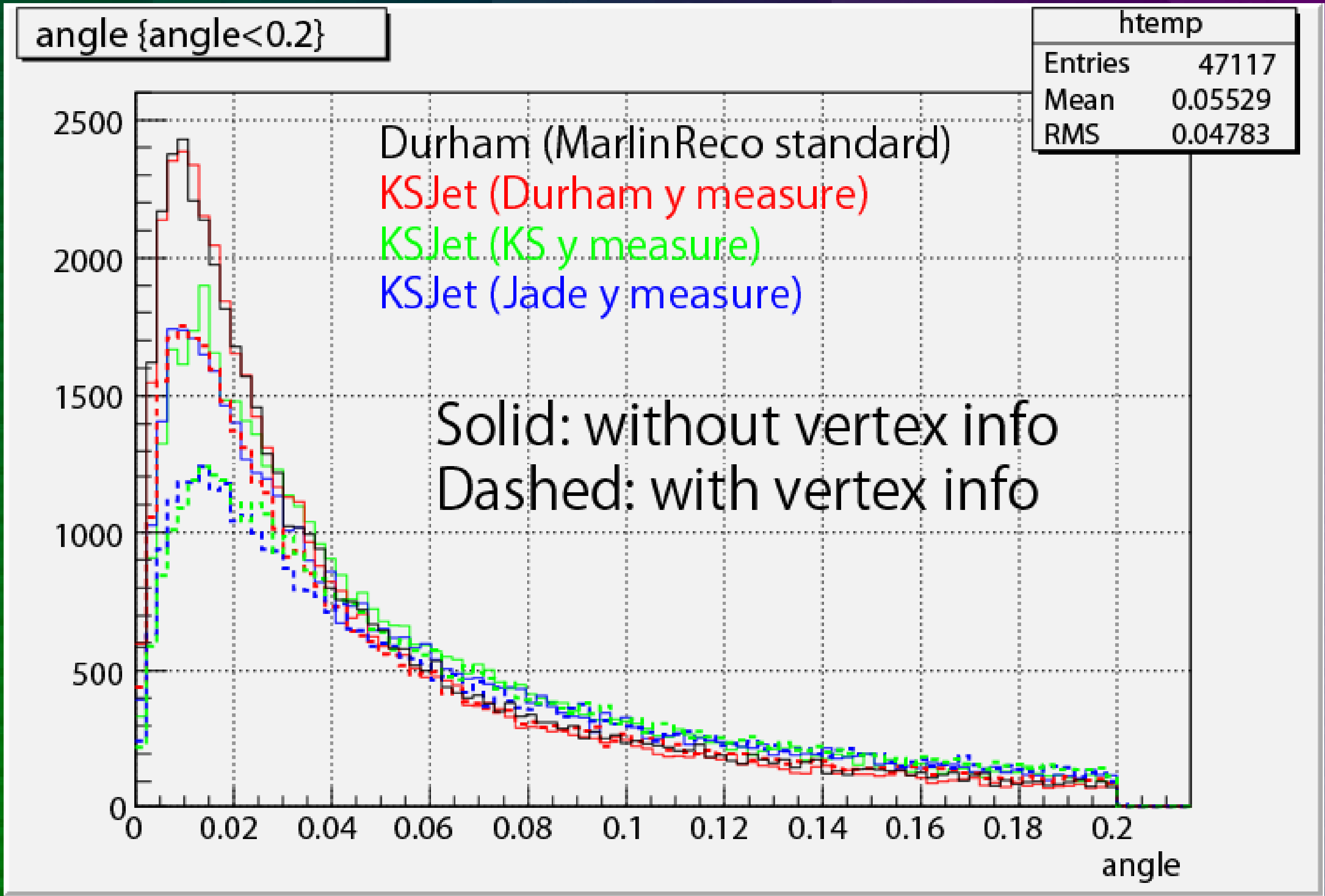
Clustering with vertex: bb sample



Red circle: b quark position (size stands for energy)
Gray circle: gluon position
Rhombus: reconstructed vertex position
Square: reconstructed jet without vertex info
Triangle: reconstructed jet with vertex info

However…

Clustering with vertex info



Vertex clustering should not worked correctly now…

Problems and measures

- Vertex finder performance is not good enough
  - Need to reduce spurious vertices by parameter tuning
  - Need to use helix tracks for lower energies
  - Need to use covariant matrices
- Utilization of vertex info to clustering can be improved
  - More emphasis on vertex position?
  - Avoid effects of spurious vertices?
- Need to check performance of flavor tagging
- Slow clustering/vertex finder
  - Need to optimize codes for fast processing

Summary & prospects

- Development of a new jet clustering with vertex information has started.
- First implementation has been done. Core-clustering seems not bad, but combining vertex information does not work correctly now.
- We plan to establish better-than-durham vertex clustering by end of this year and use it for dense-jet analyses (tt and hopufully ZHH) in LOI.