Particle Flow in SGV: Implementation and Comparisons

Mikael Berggren, Jenny List, **Madalina Chera**

Fast Monte Carlo in HEP Workshop

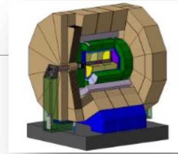
DESY Zeuthen, 15.01.13

Outline

1. Current ILC detector simulations
2. Particle Flow Parametrisations
3. Analysis based comparisons
    1. Light Higgsinos Analysis
    2. Higgs Self Coupling
    3. Chargino and Neutralino Pair Production
4. Conclusions & Outlook

Current Detector Simulations

> Full Sim – „LOI“



EVENT GENERATOR
WHIZARD

↓

DETECTOR SIMULATION
MOKKA
- based on GEANT4

↓

RECONSTRUCTION
MARLINRECO
- track finding and fitting
- particle flow – PandoraPFA
- flavour tagging

↓

OUTPUT
DST

> Full Sim – „DBD“

EVENT GENERATOR
WHIZARD

↓

DETECTOR SIMULATION
MOKKA
- **Increased realism**

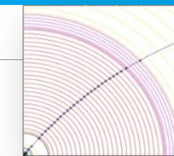
↓

RECONSTRUCTION
MARLINRECO
- track finding and fitting
- **PandoraPFA - rewritten**
- flavour tagging

↓

OUTPUT
DST

> SGV



EVENT GENERATOR
WHIZARD

↓

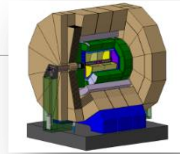
SGV
FAST SIM
- tracking from first principles
- PFlow Option

↓

OUTPUT
DST

Current Detector Simulations

> Full Sim – „LOI“



EVENT GENERATOR
WHIZARD

↓

DETECTOR SIMULATION
MOKKA
- based on GEANT4

↓

RECONSTRUCTION
MARLINRECO
- track finding and fitting
- particle flow – PandoraPFA
- flavour tagging

↓

OUTPUT
DST

> Full Sim – „DBD“

EVENT GENERATOR
WHIZARD

↓

DETECTOR SIMULATION
MOKKA
- **Increased realism**

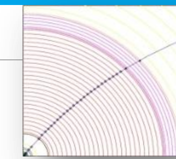
↓

RECONSTRUCTION
MARLINRECO
- track finding and fitting
- **PandoraPFA - rewritten**
- flavour tagging

↓

OUTPUT
DST

> SGV



Simulation à Grande Vitesse

- SGV: 10 ms to simulate one event
- Full sim: O(min)

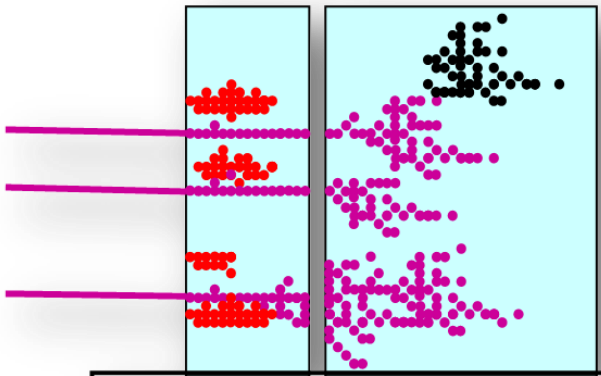
SGV is ~ 10^3 x faster than the full simulation!

A fast simulation like SGV is indispensible when:

- Simulating processes with large x-section: e.g. $\gamma\gamma$ events
- Performing SUSY parameter scans

High Precision at the ILC

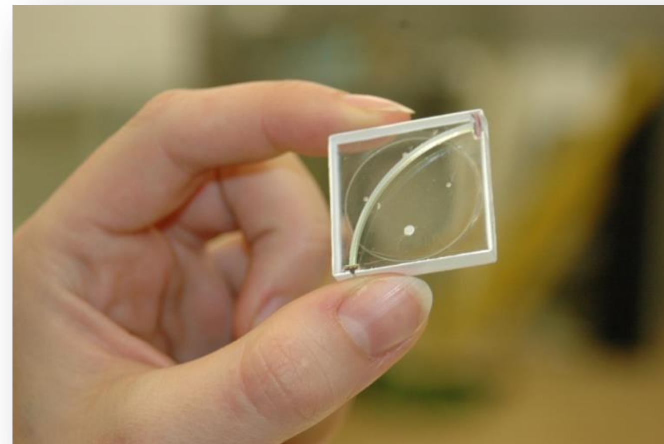
Particle Flow



$$E_{JET} = E_{TRACK} + E_{\gamma} + E_{n}$$

- jet energy = sum of the energies of all individual particles in the jet
- measure charged particles in tracker
- measure γ's in ECAL ($\frac{\sigma_E}{E} < \frac{20\%}{\sqrt{E}}$)
- measure neutrals ONLY in HCAL
- **ONLY 10% of jet energy from HCAL**

The International Large Detector

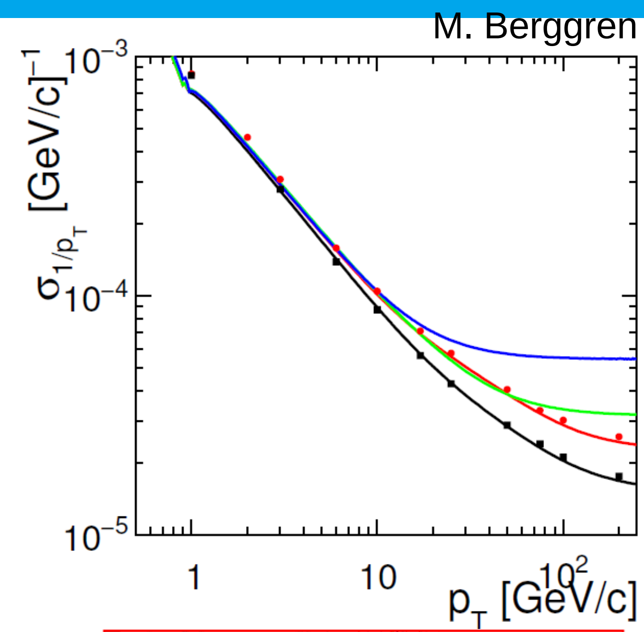


(ILD)

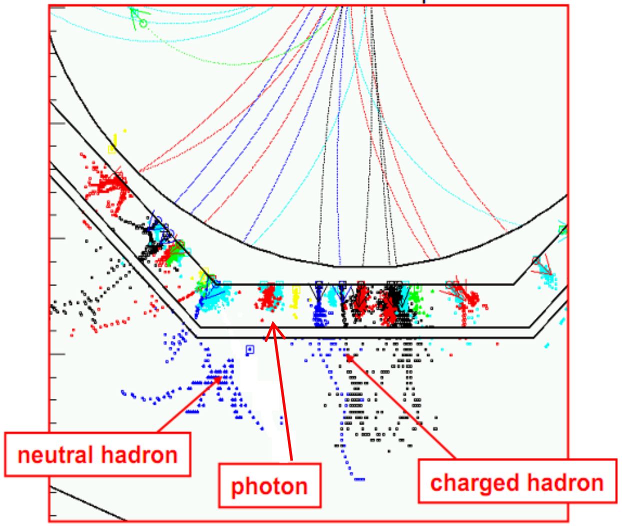
- design optimised for particle flow:
  $\rightarrow$ use full simulation
- **for physics studies: need fast simulation!**

Simulation à Grande Vitesse

- **The tracker + basic calorimeter simulation work(s) very well! However...**



- **Calorimeter**:
- The **goal** is to:
  - simulate a particle flow calorimeter
  - emulate the particle flow reconstruction
  - **PFlow in a fast simulation is a new approach!**



Calorimeter Implementation in SGV

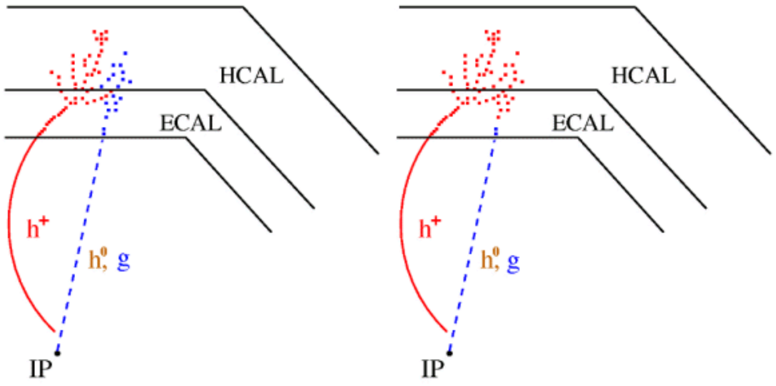
> Default features - random error on:

- detected energy
- shower position
- shower shape

> Association errors:

- calorimeter cluster merging
- cluster splitting
- cluster – track mis-association

Neutral cluster merging and mis-association



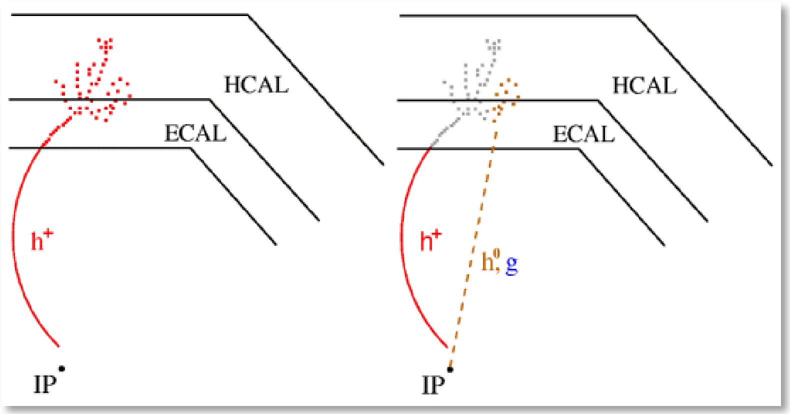
> Consequences:

- Associations errors have an impact on the total reconstructed energy.
- If a neutral cluster is (partially) associated to a track → energy is lost

Calorimeter Implementation in SGV

- Default features - random error on:
  - detected energy
  - shower position
  - shower shape
- Association errors:
  - calorimeter cluster merging
  - cluster splitting
  - cluster – track mis-association

Charged cluster splitting



- Consequences:
  - Associations errors have an impact on the total reconstructed energy.
  - If a neutral cluster is (partially) associated to a track → energy is lost
  - If a charged cluster is partially split → energy is double counted
  - Other errors (e.g. split neutral clusters, charged clusters mis-associations) do not give rise to significant errors in the total reconstructed energy or momentum

Tuning the SGV Calorimeter Implementation to PandoraPFA

- Goal: study and parametrise the association errors from data samples processed with FullSim
- Used: 8000 $e^+e^- \rightarrow udsc$ events simulated with the LOI version of Mokka and reconstructed with PandoraPFA
- Compare the *True particles* info with the *Reconstructed particles* (tracks, calorimeter hits, clusters and PFOs):
  - Make use of the MCTruth ↔ Reco information
  - Create **true clusters**: find all the calorimeter hits produced by one true particle and group them into one (or more) cluster(s) associated to the MC particle
  - Each of the true clusters **contributes to only one** reconstructed cluster.
- Study how PandoraPFA has associated the tracks and clusters:
  - Verify the link MC Particle ↔ Track
  - Verify true cluster ↔ reconstructed cluster

Tuning the SGV Calorimeter Implementation to PandoraPFA

> The study focuses on:

- True charged particles (partially) splitting their shower (double counting energy)
- True neutral particles (partially) being merged with charged showers (energy loss)

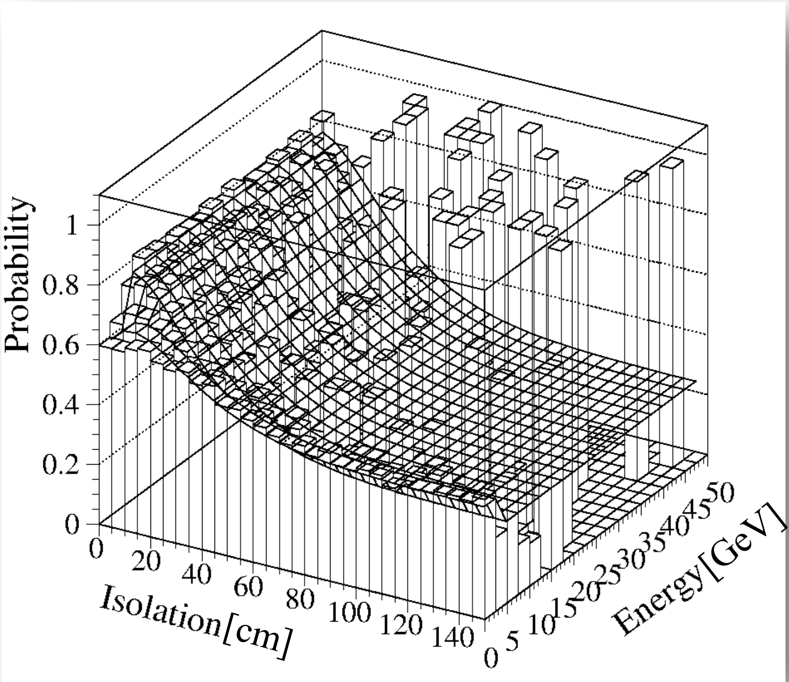
> The study ignores the less important aspects (neutral – neutral or charged – charged merging, multiple splitting/merging)

> The most relevant observables:

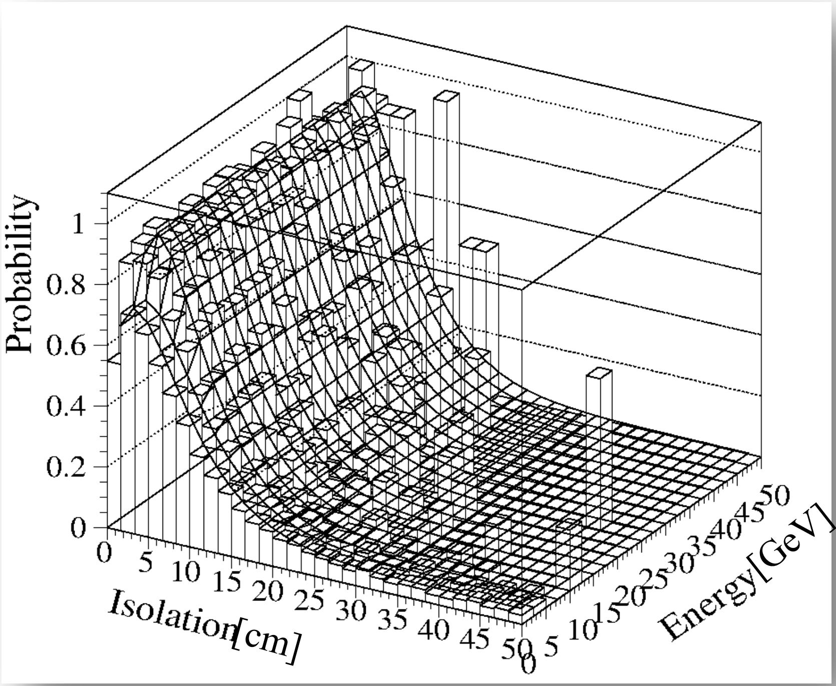
- The cluster energy
- The distance between the cluster and the nearest particle of the "opposite" type **(Isolation)**
- Whether we are dealing with an electro-magnetic or hadronic shower
- Which region of the detector the particle reaches (barrel or end-caps)

Parametrising the Association Errors: Splitting Probability

> The probability that a cluster would split:



Charged hadrons: double counting

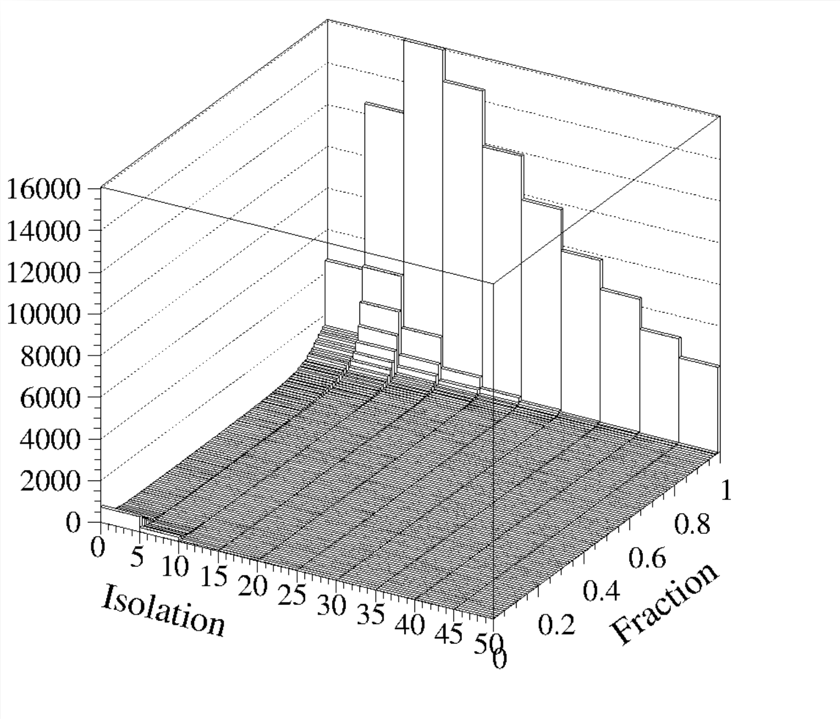


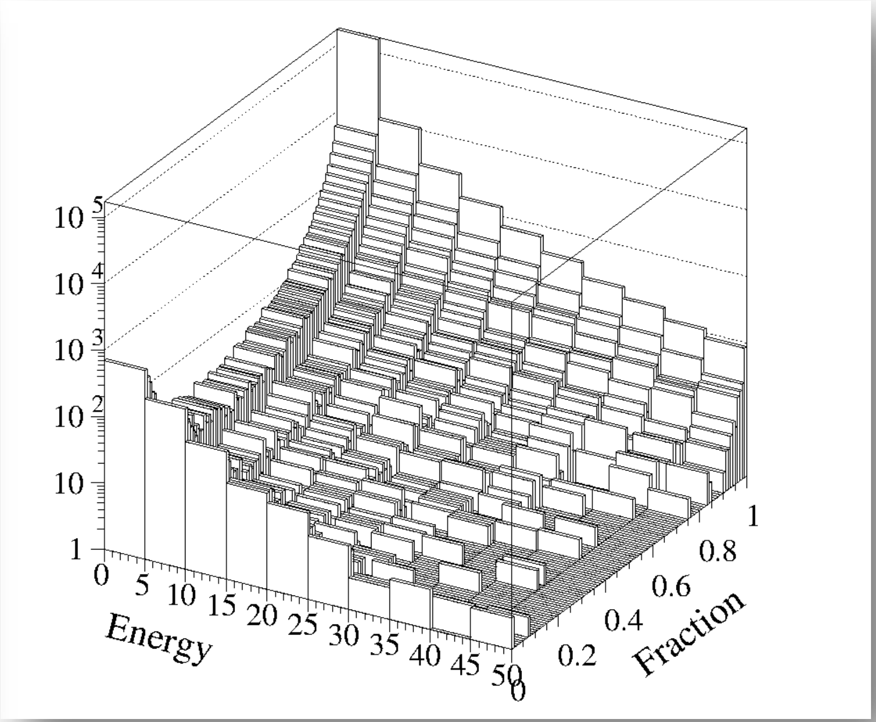
Photons: energy loss

- The splitting probabilty depends strongly on isolation for the EM case
- The dependency on the detector region was 5%

Parametrising the Association Errors

> The probability to split/merge an entire cluster: Photons: energy loss

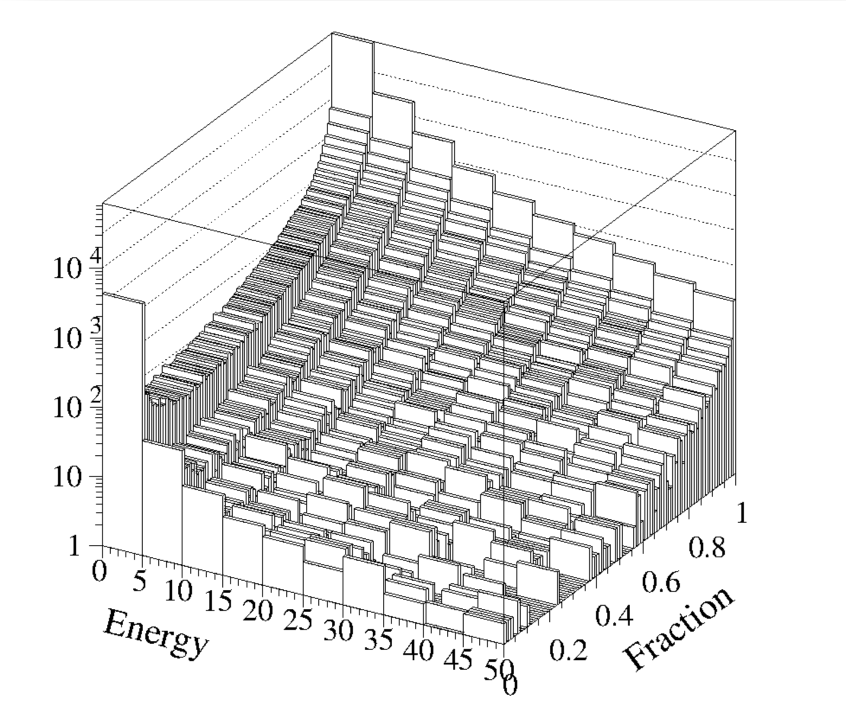




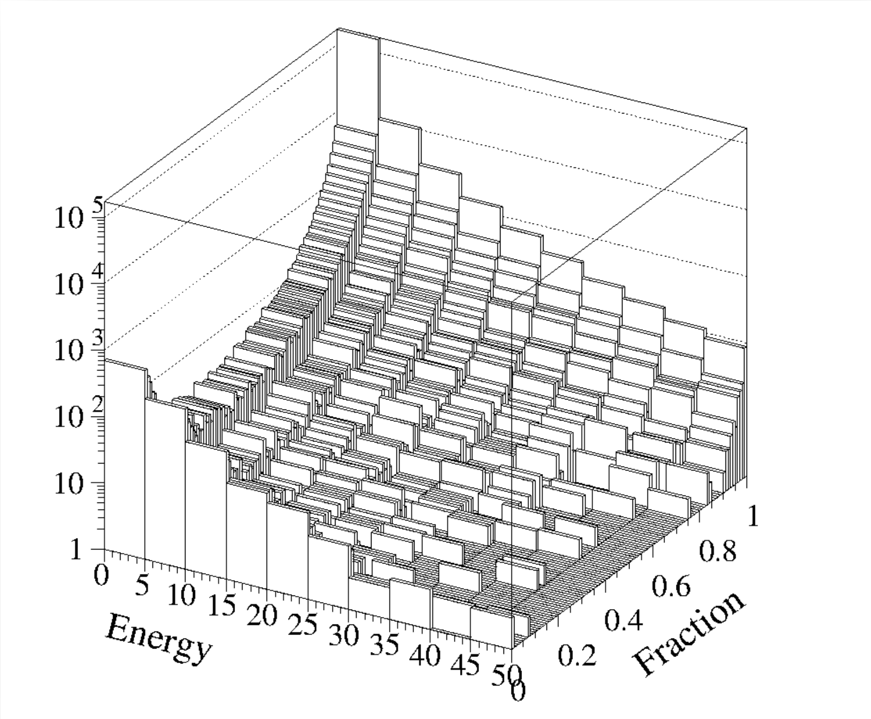
- Depends almost entirely on the cluster‘s energy

Parametrising the Association Errors

> The probability to split/merge an entire cluster:



Charged hadrons: double counting

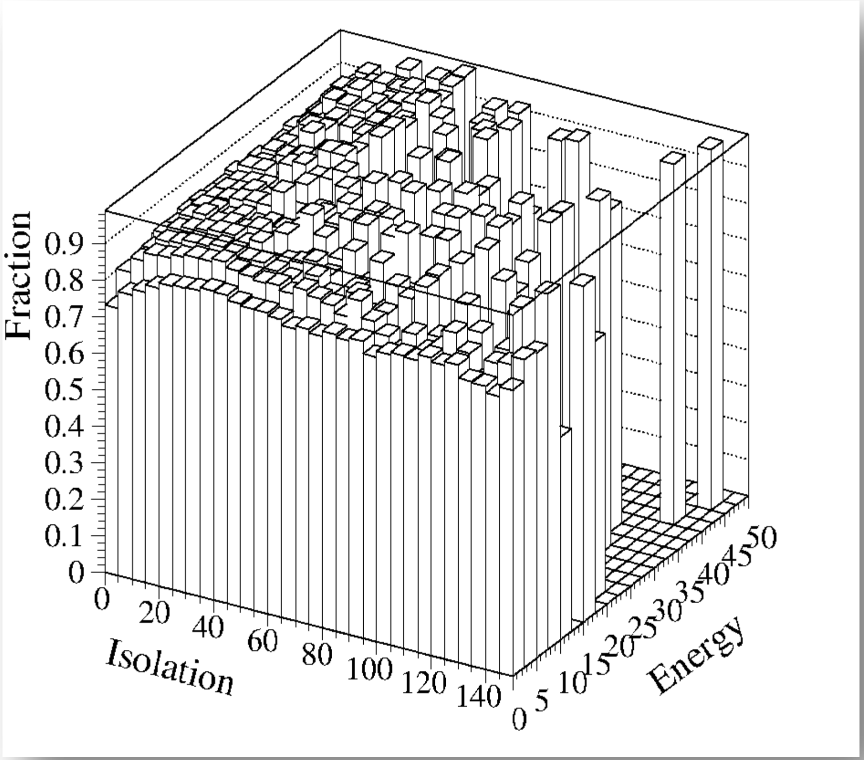


Photons: energy loss

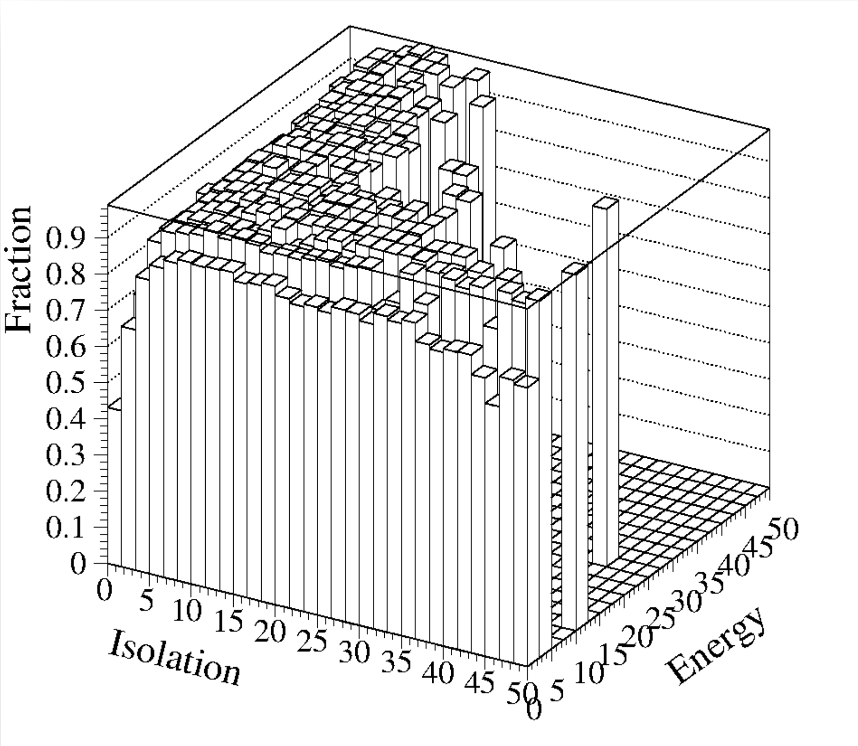
- Depends almost entirely on the cluster's energy

Parametrising the Association Errors

> The average of correctly assigned fraction of the cluster energy:



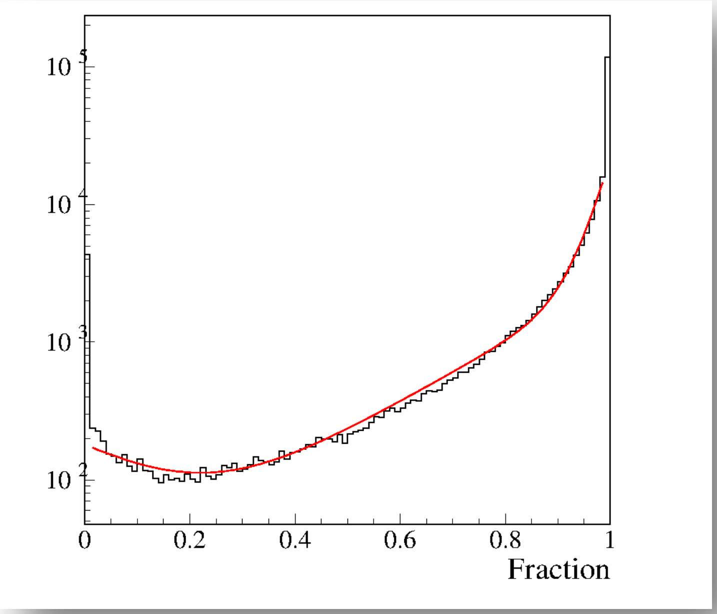
Charged hadrons: double counting



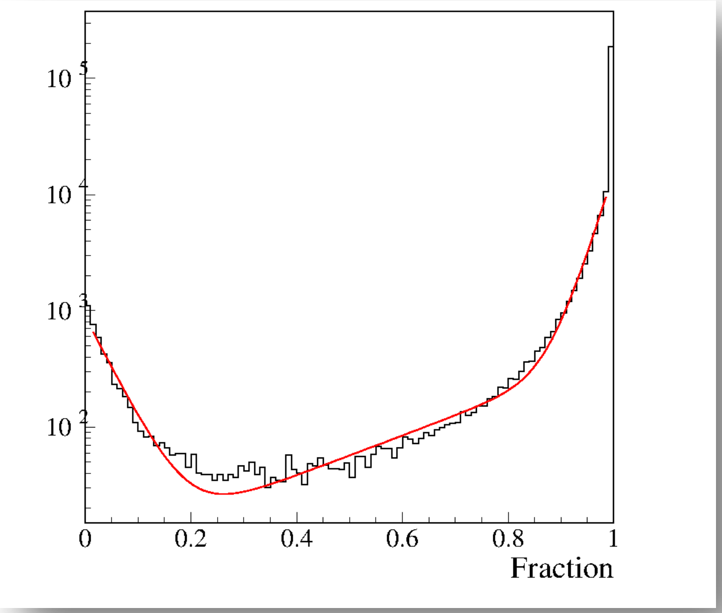
Photons: energy loss

Parametrising the Association Errors

> Fitting the splitting fraction (for an example average fraction):



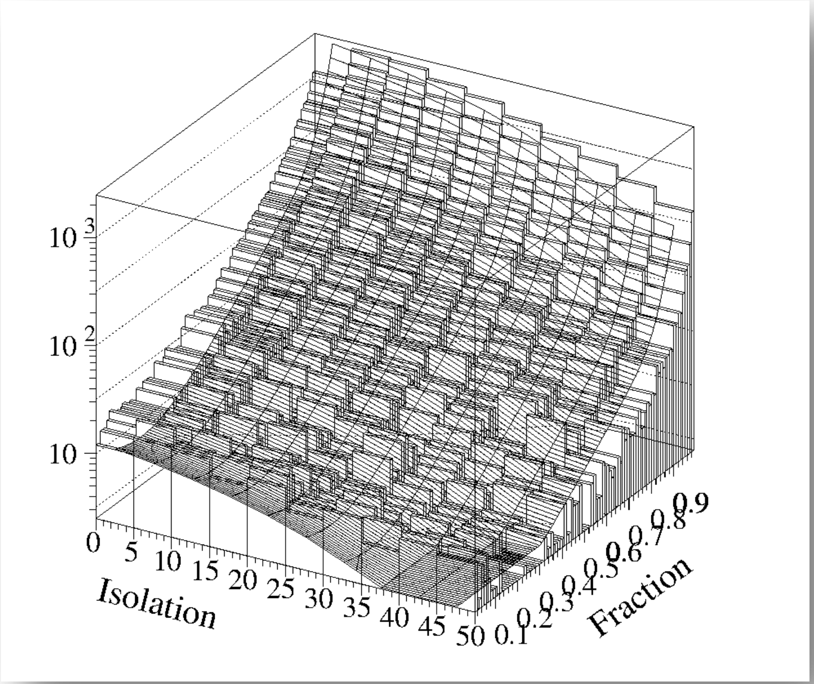
Charged hadrons: double counting



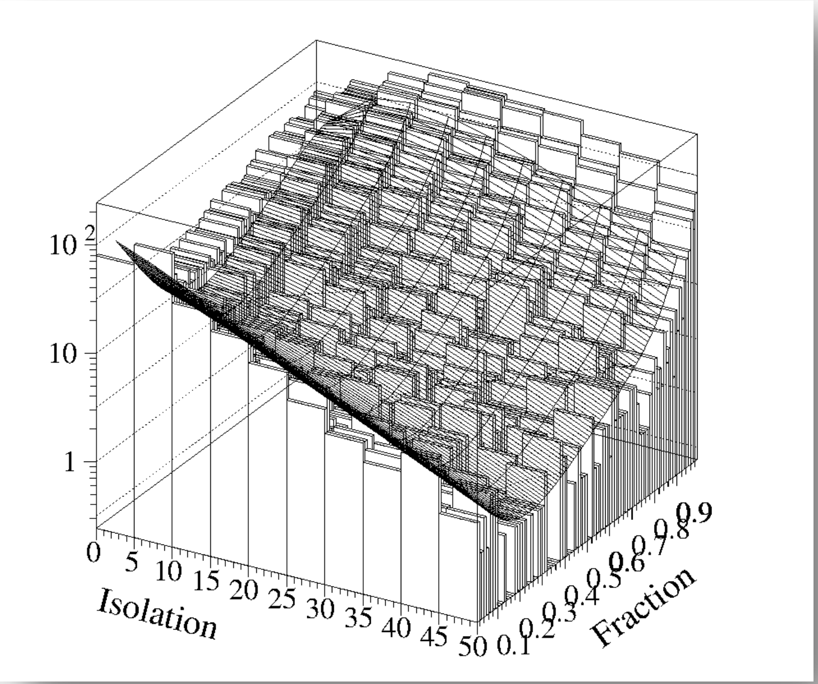
Photons: energy loss

Parametrising the Association Errors

- The probability that only a fraction of the cluster would split:
  - As a function of distance:



Charged hadrons: double counting

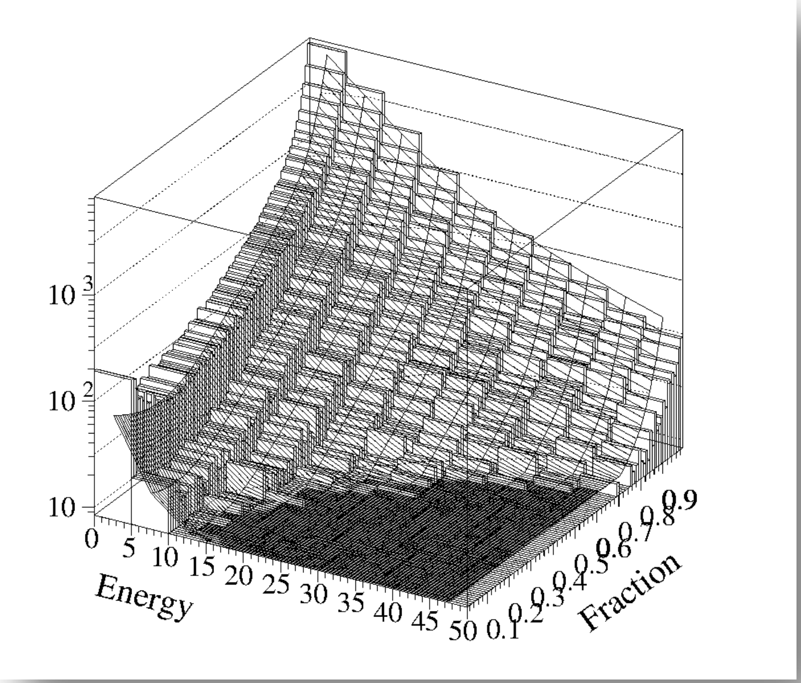


Photons: energy loss

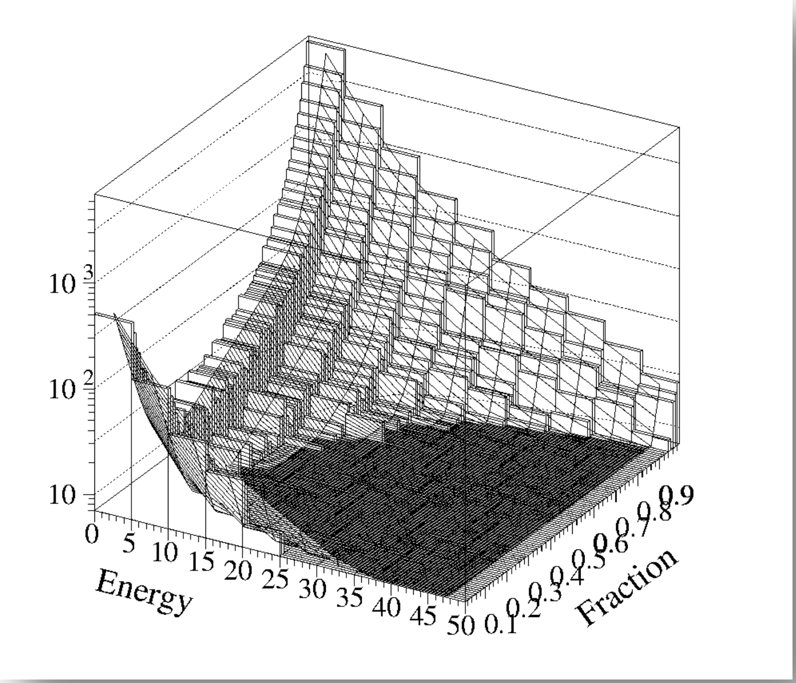
Parametrising the Association Errors

> The probability that only a fraction of the cluster would split:

- As a function of energy:



Charged hadrons: double counting



Photons: energy loss

- The partial splitting probability depends on both energy as well as distance
- The dependence can be expressed in terms of the **average fraction**

Parametrising the Association Errors: Technicalities

- The PFL user routine ZAUPFL is called in ZAUSER
- On first call: Reads and stores parameters
- Clusters are made from simulated showers of each particle in ZAUMKC
- Loop (once) over all particles:
  - Determine shortest distance to neighbour of right type and charge in the same calorimeter
  - Simulate: (ZACCON)
  - No split or complete split or partial split
  - If partial: fraction split off
  - Compare E and p: If too different, try again a few times
  - Remove and push out any empty showers or clusters; adjust pointers

Parametrising the Association Errors

- Parametrisations:
  - The distributions have been fitted with combinations of exponentials and linear functions
  - In total, 28 parameters x 4 cases (EM/hadronic x double-counting/ lost energy) are needed
  - These parameters are fed to SGV as input data files to the user routine that takes care of the particle flow emulation
- Key feature: output data in the same standard format as FullSim (LCIO)
  - Enables direct comparison between SGV – FullSim
  - Allowed the development and fine tuning of the benchmark analyses for the ILC TDR
  - All TDR analyses use mixed FullSim and SGV backgrounds – SGV for high cross section backgrounds , e.g. low P_t $\gamma\gamma$
- Now, compare the fast simulation output to Mokka & PandoraPFA...

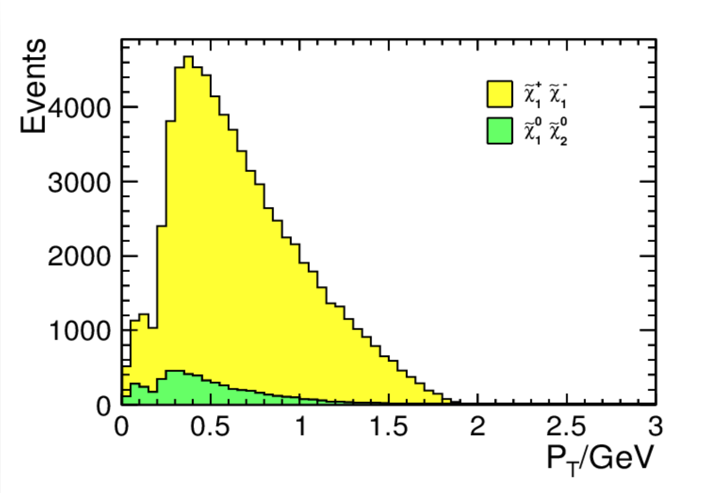
SGV – FullSim Comparisons

Light Higgsinos Scenario

- Three light higgsinos:

Particle	Mass [GeV]
$\tilde{\chi}_1^0$	166.59
$\tilde{\chi}_1^\pm$	167.36
$\tilde{\chi}_2^0$	167.63

- $\tilde{\chi}_1^\pm$ and $\tilde{\chi}_2^0$ are almost mass degenerate
- Mass difference to LSP ($\tilde{\chi}_1^0$) is **smaller than 1 GeV**
- All sleptons are much heavier: $\mathcal{O}(10^3 \mathrm{GeV})$



- 2 escaping LSPs → large amount of missing energy
- Small mass difference to LSP
- Few visible decay products
- Very low P_T

SGV – FullSim Comparisons (Light Higgsinos)

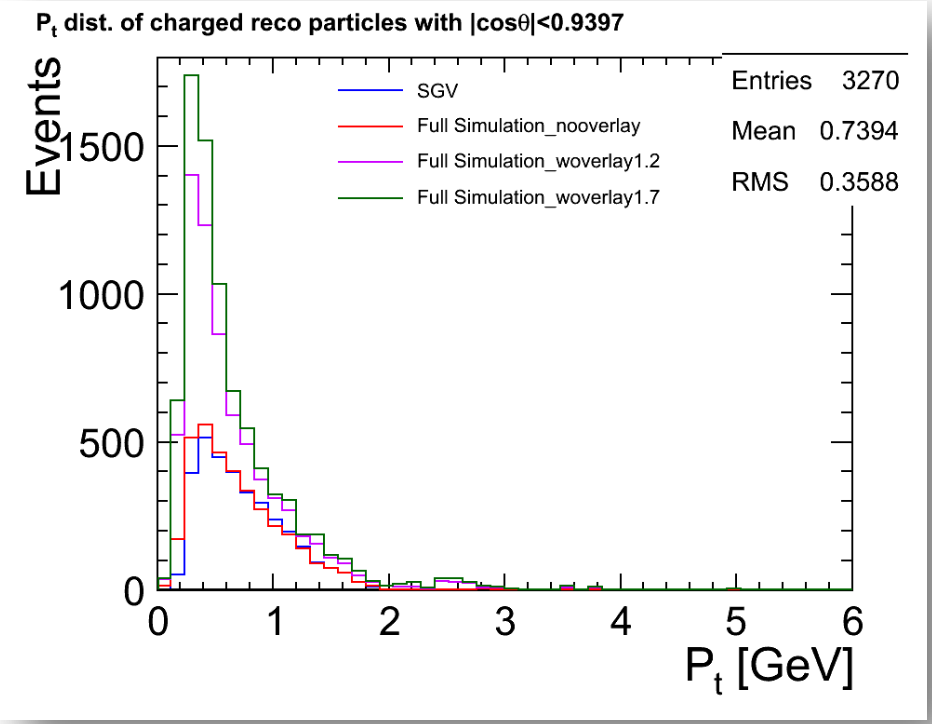
- Comparison performed using:
  2000 events

$$e^+e^- \rightarrow \tilde{\chi}_1^+ \tilde{\chi}_1^- \gamma$$

$$\tilde{\chi}_1^+ \tilde{\chi}_1^- \rightarrow 2\,\tilde{\chi}_1^0\, W^{+*} W^{-*}$$

- Polarization:(e-,e+)=(-0.8,+0.3)
- Ecm = 500 GeV,
- L = 500 fb^{-1}

- All reconstructed particles within a |cos(θ)| < 0.9397 have been considerered
- SGV and FullSim are in very good agreement
- The differences between FullSim and SGV are much smaller than the effect of the low Pt $\gamma\gamma$ background overlay ("pile-up")

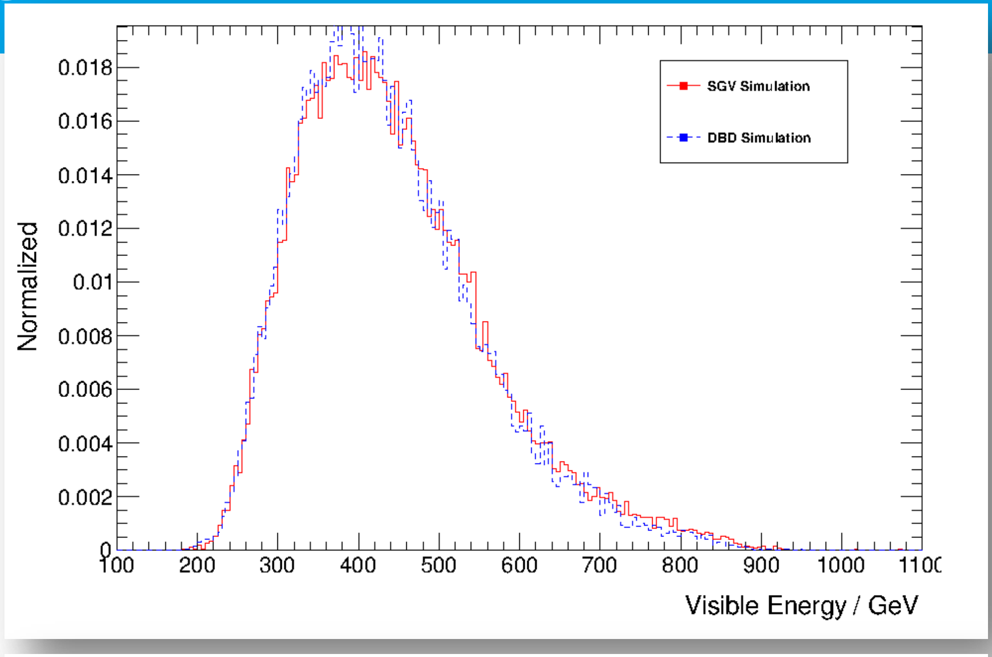


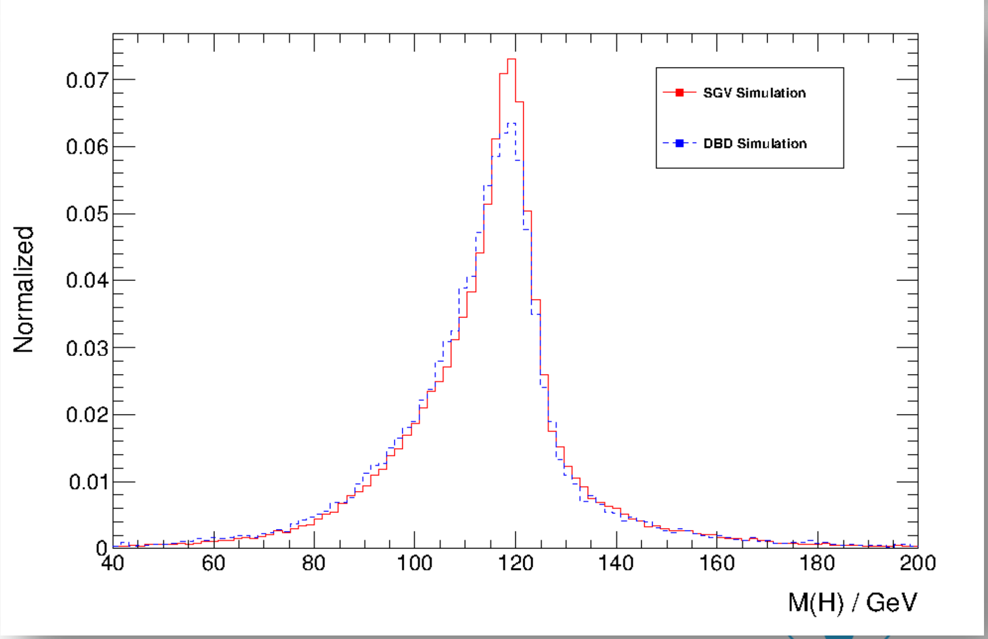
(Hale Sert)

SGV – FullSim Comparisons

- SGV performs very well even in complex anlyses like the **Higgs self-coupling**
- SGV – FullSim comparison:
  - Polarization:(e-,e+)=(-0.8,+0.2)
  - Ecm = 1 TeV,
  - M_H = 120 GeV
  - L = 2 ab^{-1}
- Δλ/λ : 17% SGV; 18 % FullSim

(Junping Tian)

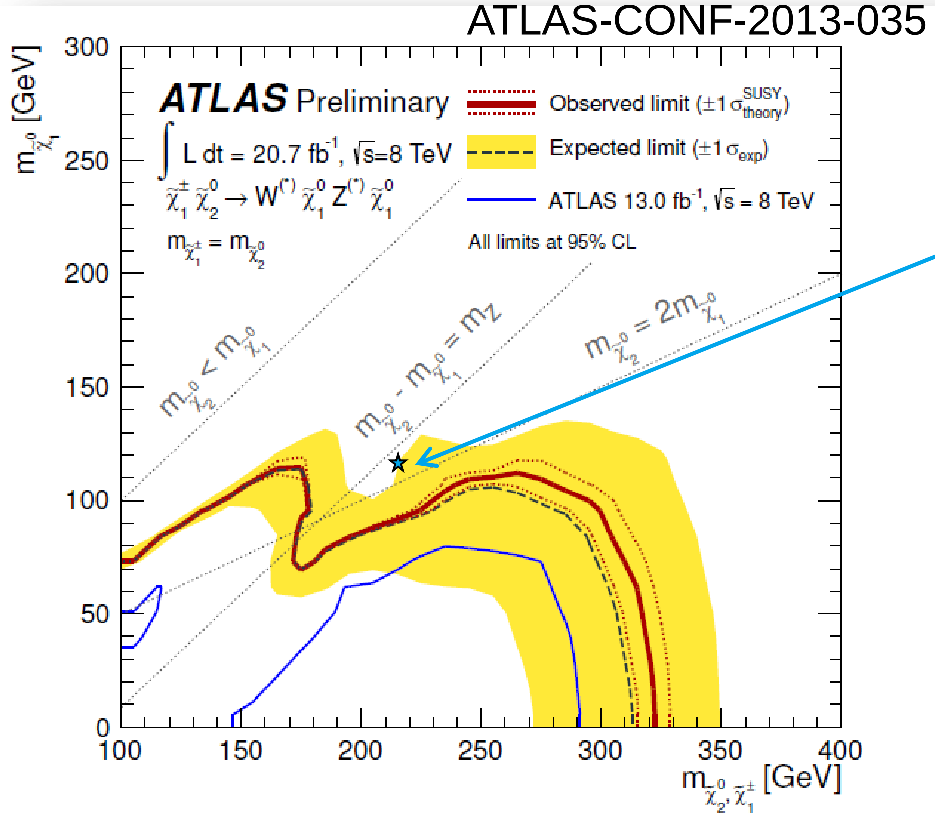




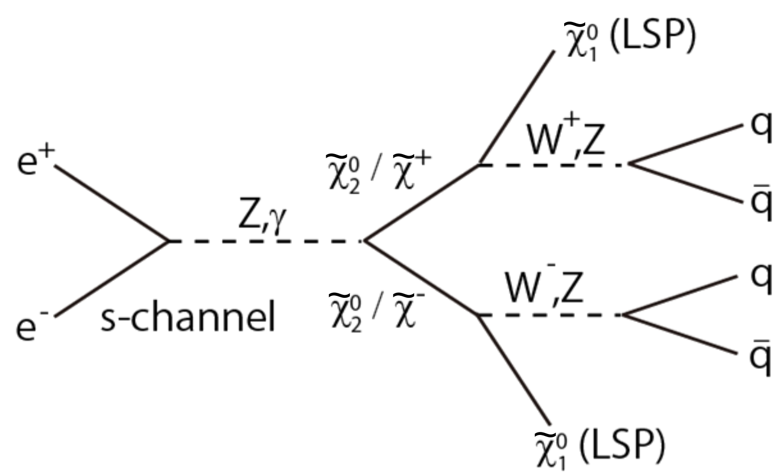
SGV – FullSim Comparisons: $\tilde{\chi}_1^\pm$ and $\tilde{\chi}_2^0$ Pair Production

"Point 5" benchmark : gaugino pair production at ILC

http://arxiv.org/pdf/1006.3396.pdf (ILD LoI)
http://arxiv.org/pdf/0911.0006v1.pdf (SiD LoI)



Particle	Mass [GeV]
$\tilde{\chi}_1^0$	115.7
$\tilde{\chi}_1^\pm$	216.5
$\tilde{\chi}_2^0$	216.7
$\tilde{\chi}_3^0$	380



- **The signal channels have been chosen as a challenge to the Particle Flow.**
- **The analysis is extremely sensitive to jet energy measurements.**

SGV – FullSim Comparisons: $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Pair Production

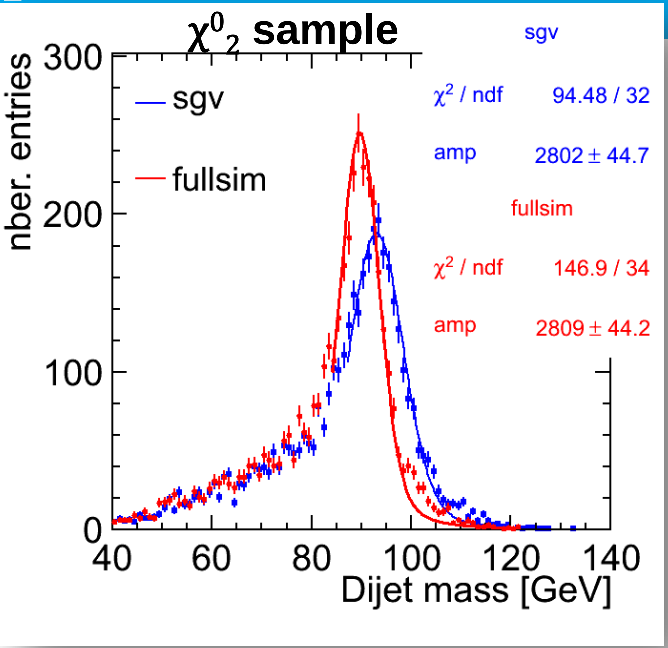
- **Use dijet mass to separate $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ events → measure cross section**
- Dijet [Boson] Mass Comparison – LOI to SGV performed using:

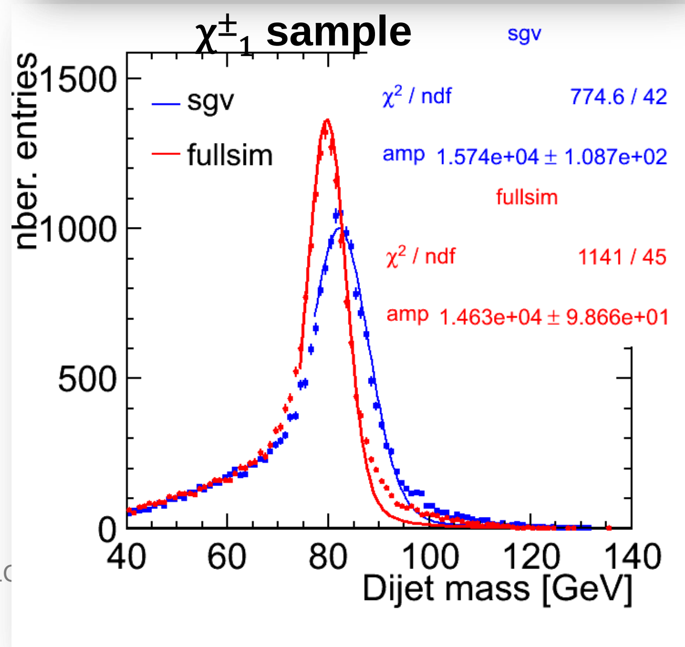
~ 20000 $e^+e^- \rightarrow \tilde{\chi}_1^+\tilde{\chi}_1^- \Rightarrow \tilde{\chi}_1^{\pm} \rightarrow \tilde{\chi}_1^0 W^{\pm}$ events

~ 4000 $e^+e^- \rightarrow \tilde{\chi}_2^0\tilde{\chi}_2^0 \Rightarrow \tilde{\chi}_2^0 \rightarrow \tilde{\chi}_1^0 Z^0$ events

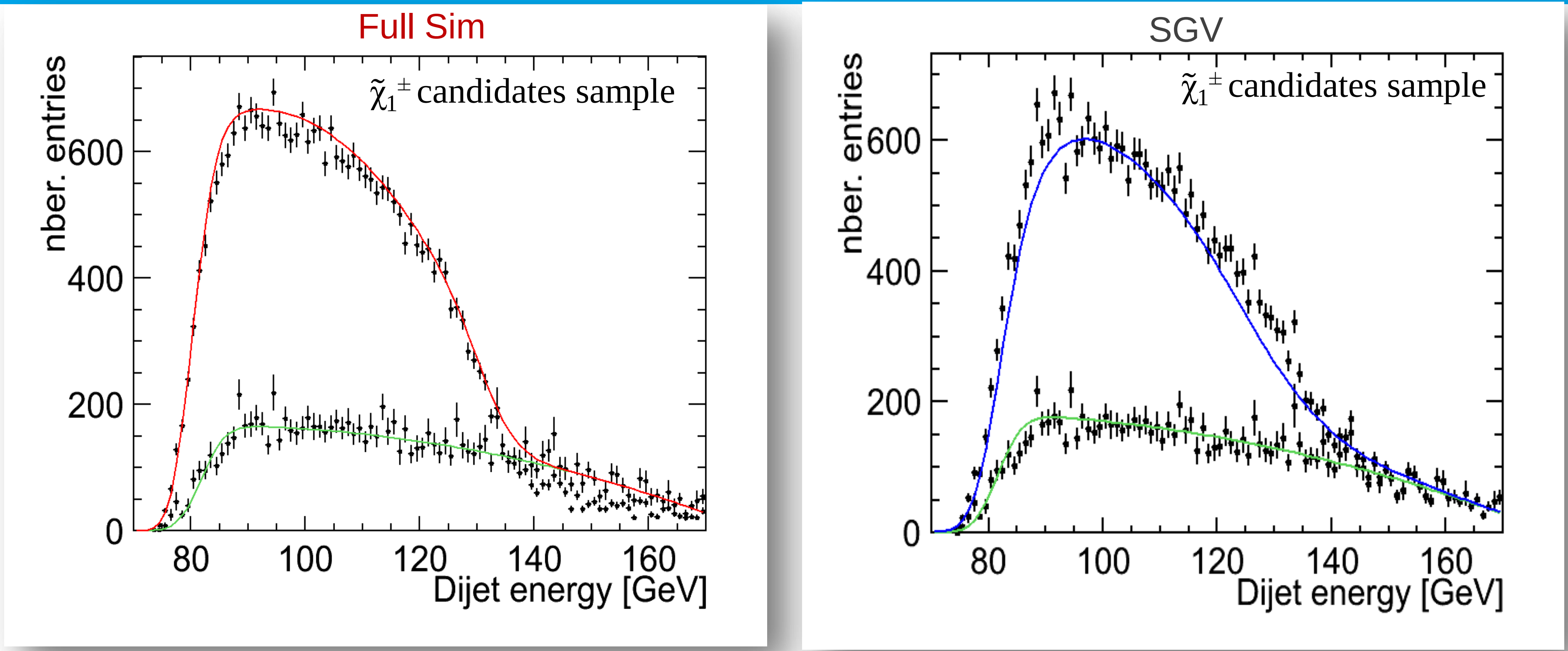
- Polarization:(e-,e+)=(-0.8,+0.3)
- Ecm = 500 GeV,
- L = 500 fb^{-1}
- Force event into 4 jets (Durham)

- The SGV distribution is wider and shifted with ~3 GeV towards higher energies.





SGV – FullSim Comparisons: $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^{0}$ Pair Production

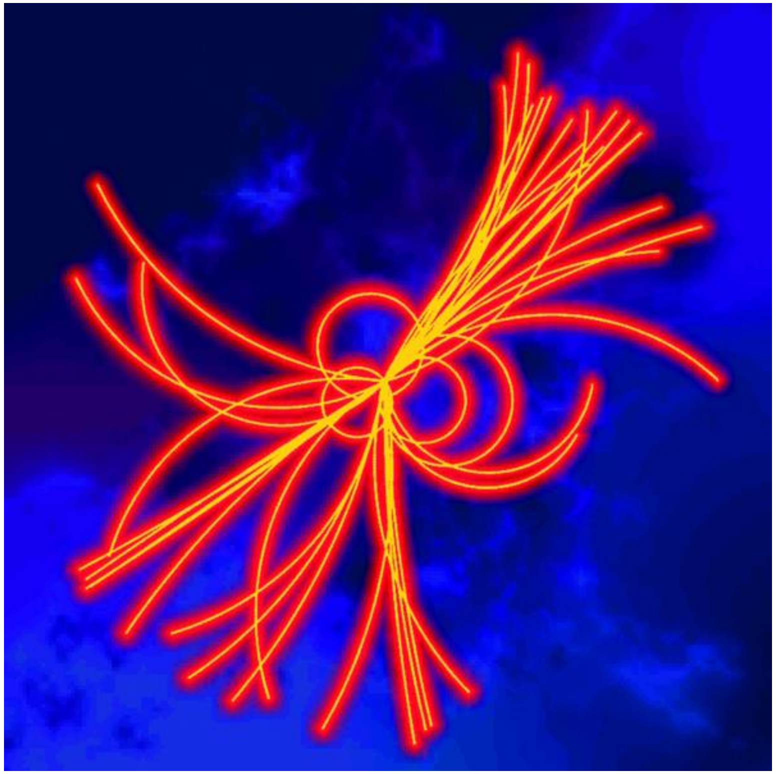


- **Use dijet energy spectrum „end points“ in order to calculate masses**
- **Compare dijet energy SGV - FullSim**
- Force events into 4 jets
- Perform pre-selection and kinematic fit
- Use only best permutation of the kinematic fit
- The SGV distribution is sharper and slightly shifted towards higher energies

Conclusions

- The ILC community has a fast simulation tool that performs very well: **SGV**
- The Particle Flow implementation in SGV has been presented:
  - The most relevant parametrisations are the total / partial cluster splitting probabilities as a function of energy and distance to closest particle
- One key feature: output in the same data format as Full Sim which enables direct comparisons as well as using mixed backgrounds.
- SGV has been used successfully in complex analyses, e.g. the light Higgsinos scenario and the Higgs self coupling study
- Nevertheless, measurements that are extremely sensitive to the jet energy (e.g. the $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Pair Production) show the need to further improve the particle flow parametrisation

Thank You!

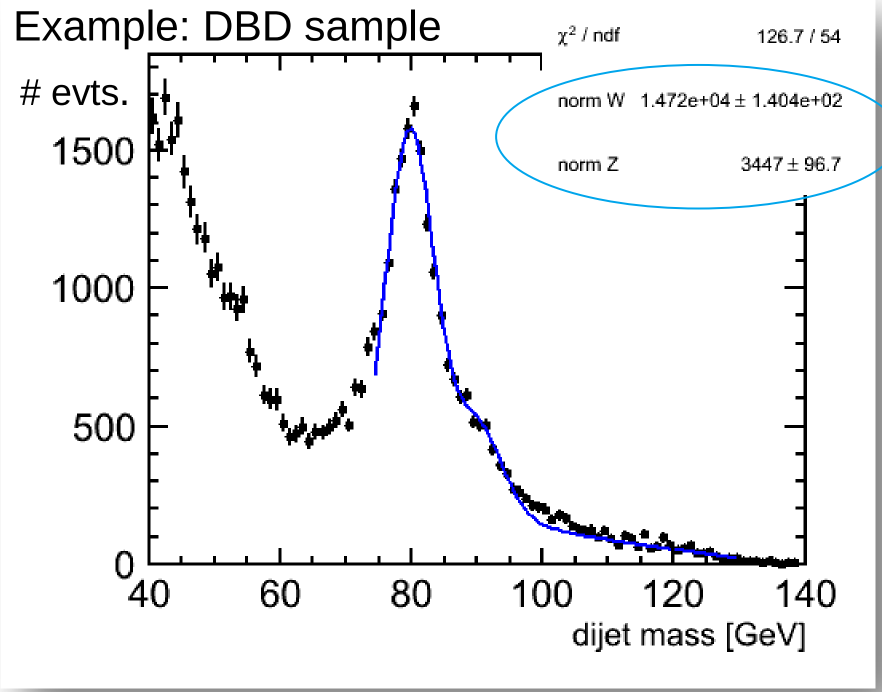


Back up slides

$\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Cross Section Measurement

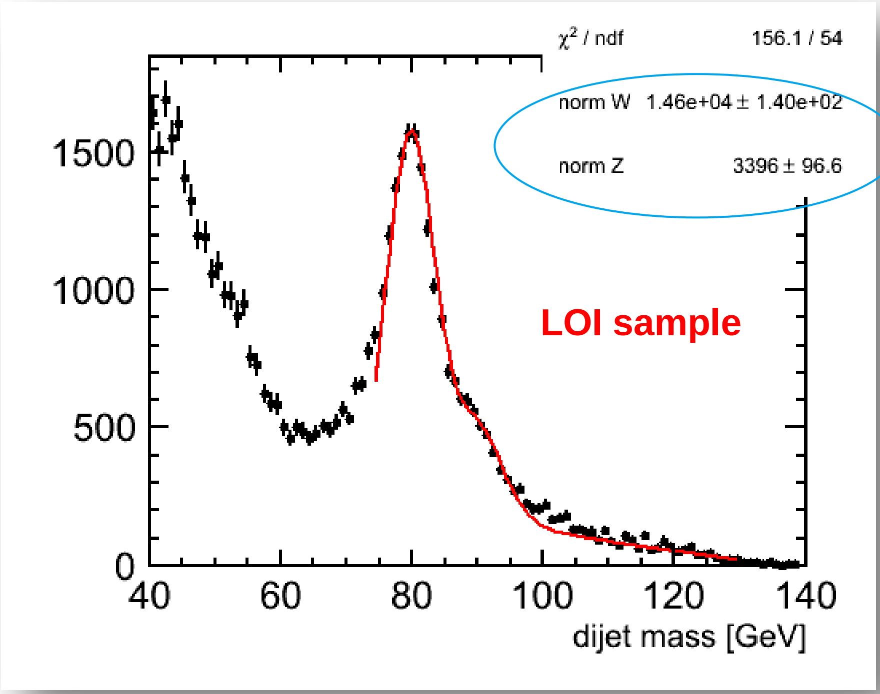
> Separating W and Z pairs candidates:

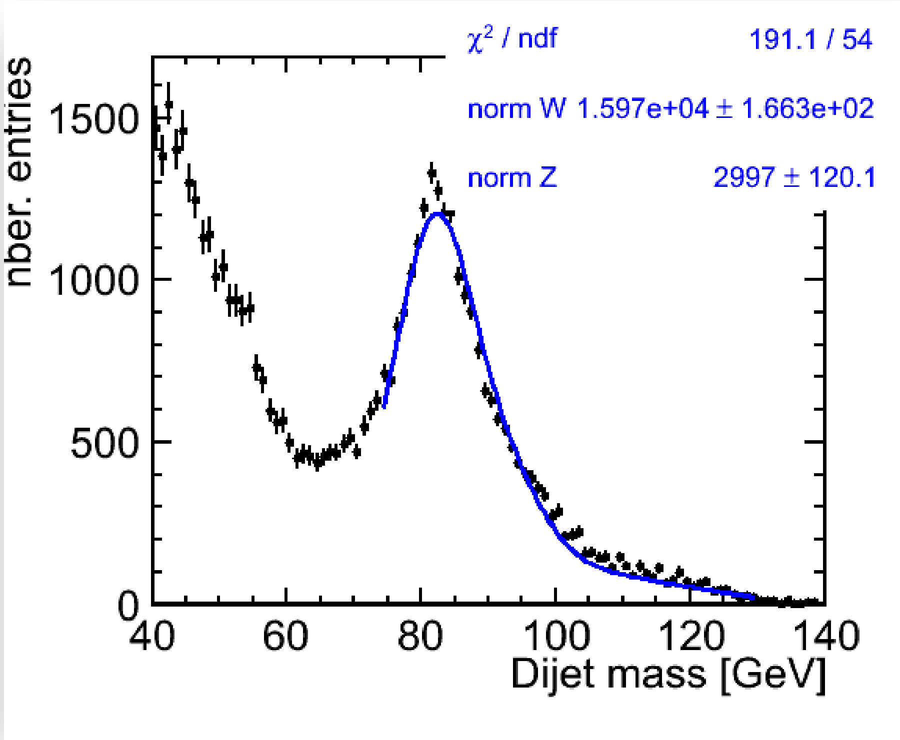
- SM background fitted with polynomial
- Signal distributions fitted with Voigt profile
  - Width (Γ) set to boson's natural width (2.11 GeV for W and 2.5 GeV for Z
  - Voigt $\sigma \simeq 3.5$ GeV detector resolution, deduced from a SM sample. The σ from the signal only sample is in the same ballpark!
- Determine relative W/Z fractions from fit



$\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Cross Section Measurement

- Cross section calculation: determine the amount of W and Z pairs candidates.



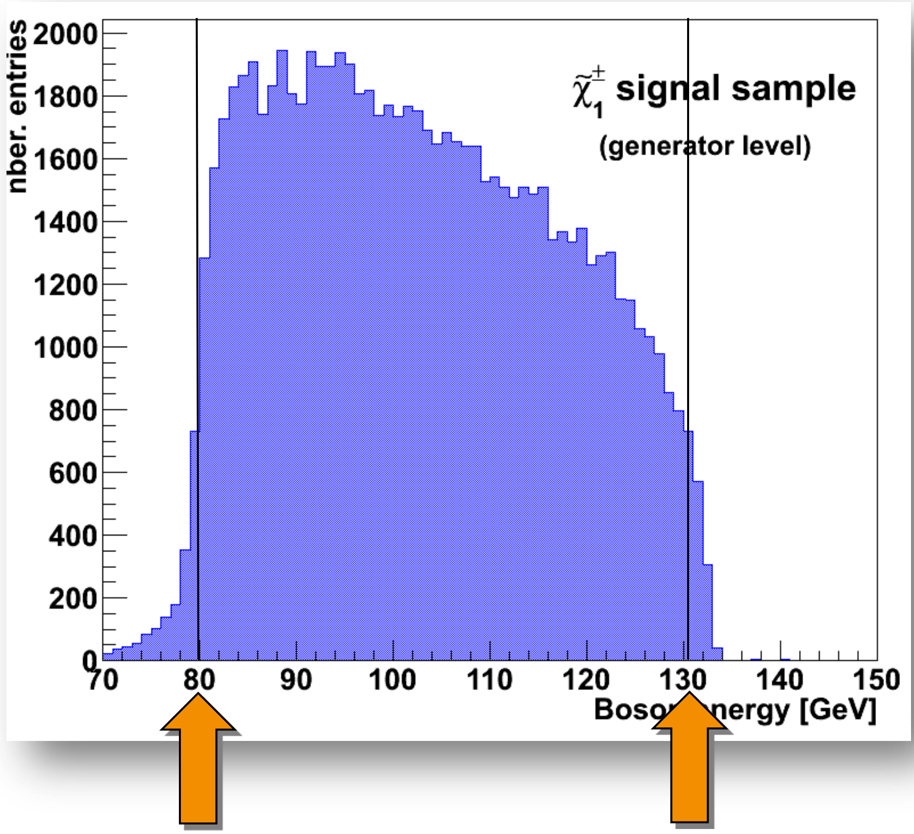


$\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^{0}$ Mass Measurement

- Mass difference to LSP ($\tilde{\chi}_1^0$) is **larger** than M_Z
- Observe the decays of real gauge bosons
- 2 body decay → the edges of the energy spectrum are kinematically determined
- **Use dijet energy spectrum „end points“ in order to calculate masses**

$$\gamma = \frac{E_{beam}}{M_\chi}$$

$$E_{\pm} = \gamma \cdot EV^{*} \pm \gamma \cdot \beta \cdot \sqrt{E_V^{*2} - M_V^{2}}$$

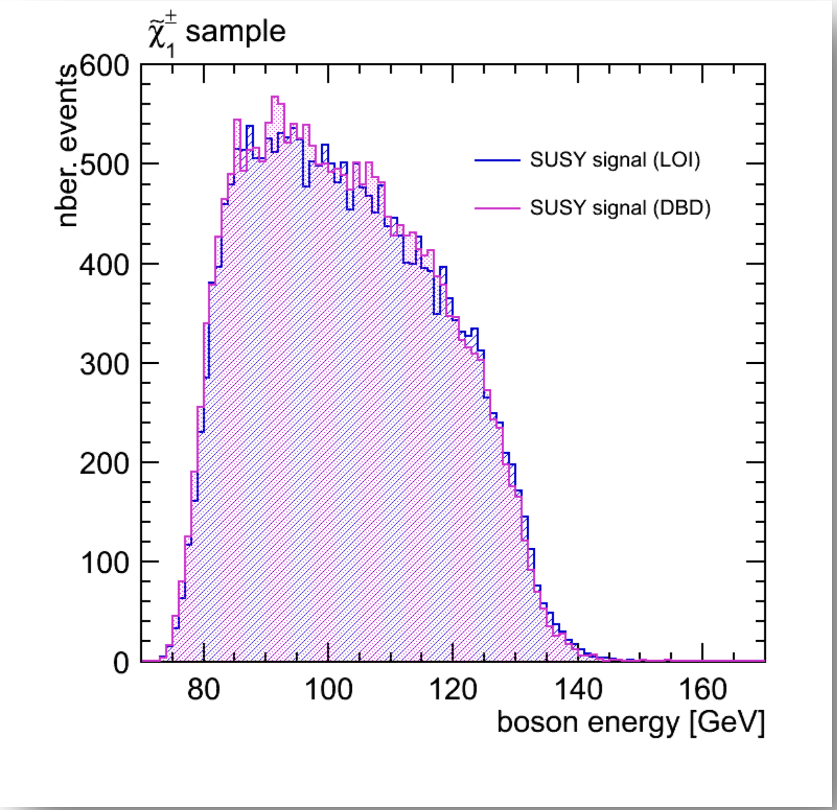


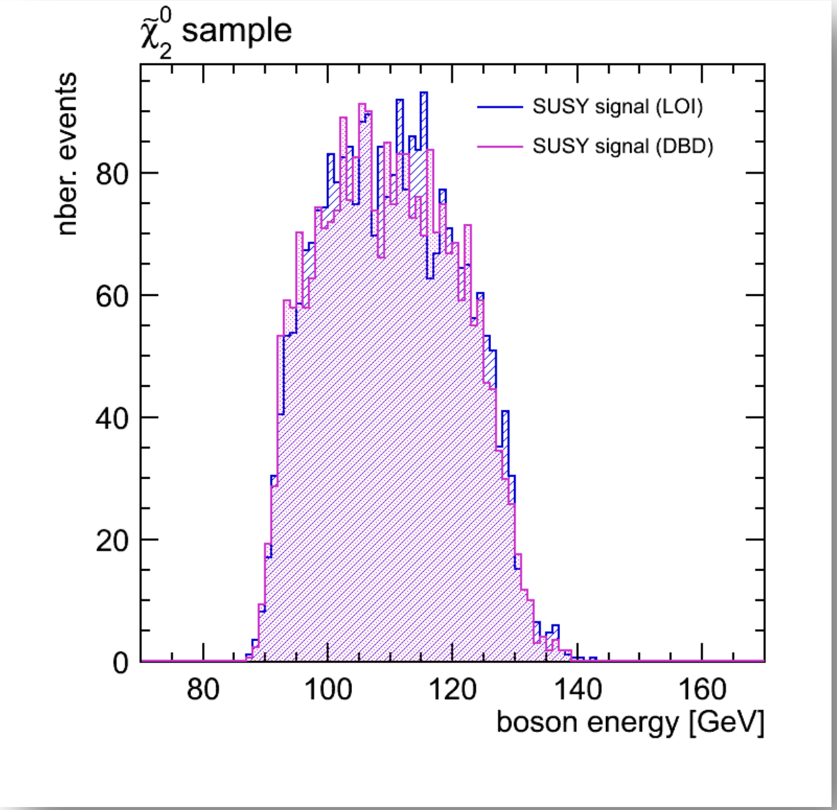
Real edge values [GeV]:

W_{low}	W_{high}	Z_{low}	Z_{high}
80.17	131.53	93.24	129.06

Dijet [Boson] Energy Comparison

> Use dijet energy to measure $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ mass





> The DBD distribution appears slightly narrower and shifted towards lower energies. Nevertheless, the two distributions agree very well.

$\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Signal Sample Further Separation

- Calculate χ^2 with respect to nominal W / Z mass

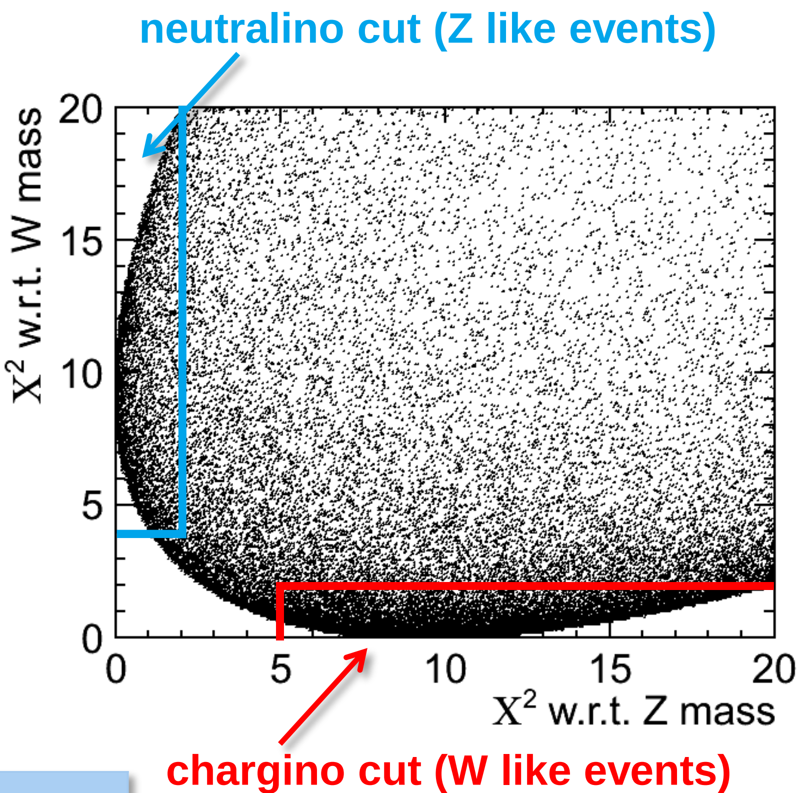
$$\chi^2(m_{j1}, m_{j2}) = \frac{(m_{j1} - m_V)^2 + (m_{j2} - m_V)^2}{\sigma^2}$$

$\Downarrow$

min $\chi^2 \rightarrow \tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ separation

- Downside: lose statistics
  - Cut away 43% of $\tilde{\chi}_1^{\pm}$ surviving events
  - Cut away 68% of $\tilde{\chi}_2^0$ surviving events
- However, after the χ^2 cut, the separation is quite clear:

Obs.	DBD		LOI	
	$\tilde{\chi}_1^{\pm}$	$\tilde{\chi}_2^0$	$\tilde{\chi}_1^{\pm}$	$\tilde{\chi}_2^0$
Efficiency	57%	32%	56%	34%
Purity (total)	63%	35%	62%	35%
Purity (SUSY)	94%	68%	95%	66%



Endpoint Extraction using an FIR Filter

- Finite Impulse Response (FIR) filters are digital filters used in signal processing.
- FIR filters can operate both on discrete as well as continuous values.
- The concept of “finite impulse response“ ↔ **the filter output** is computed as a finite, weighted sum of a finite number of values from the filter input.

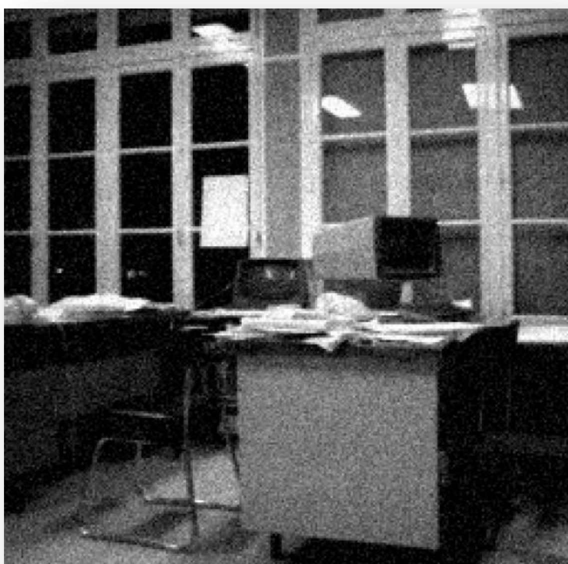
$$y[n] = \sum_{k=-M_1}^{M_2} b_k x[n-k]$$

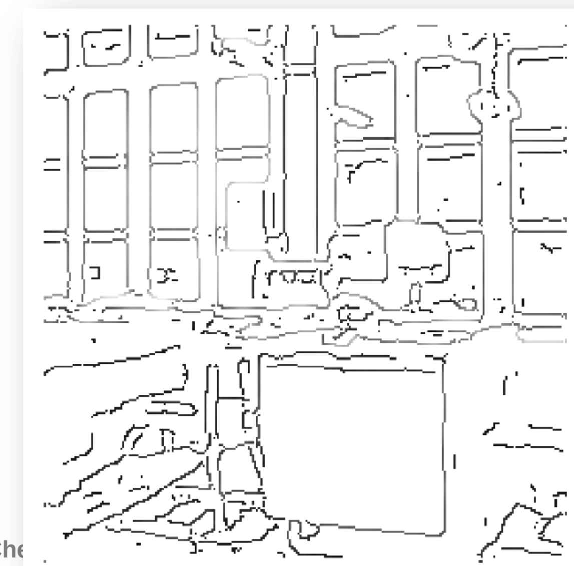
the input signal

the filter coefficients (weights)

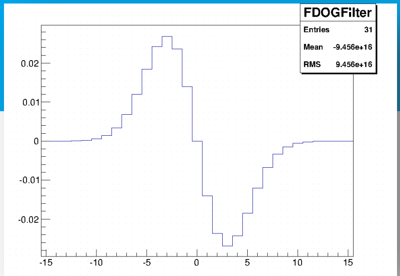
- y is obtained by convolving the input signal with the (finite) weights
- FIR filters are used to detect edges in image processing techniques:

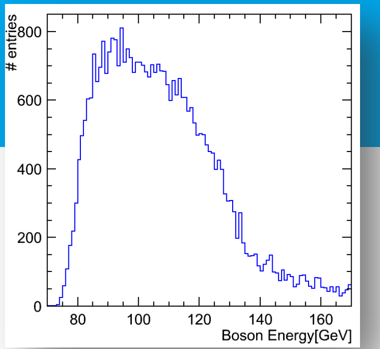
D. Demigny, T. Kamlé





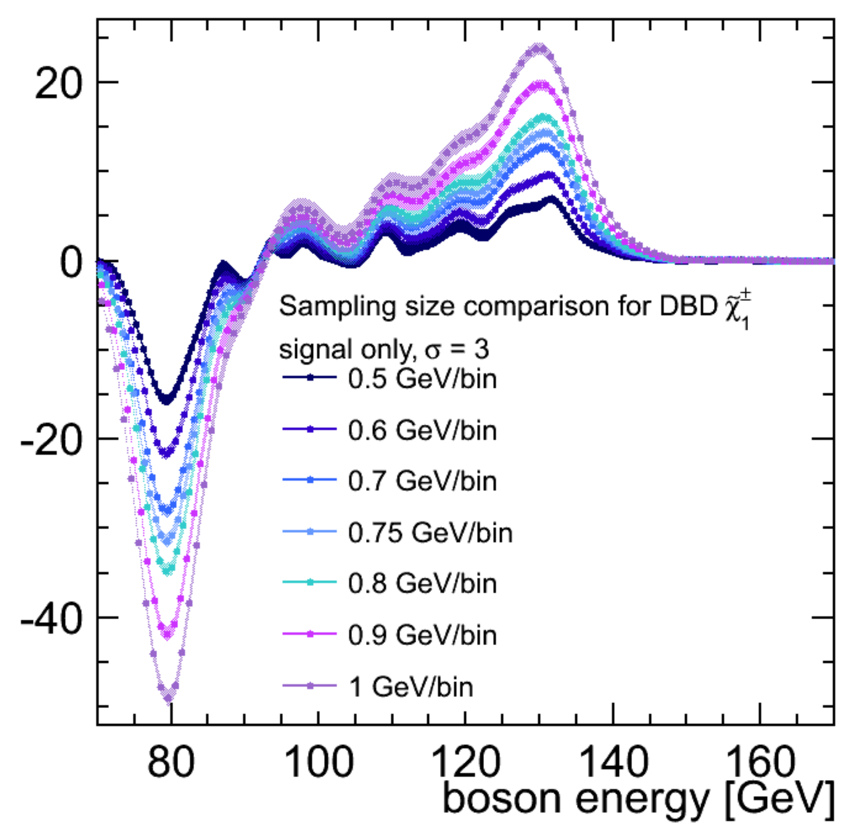
Testing the FDOG Filter

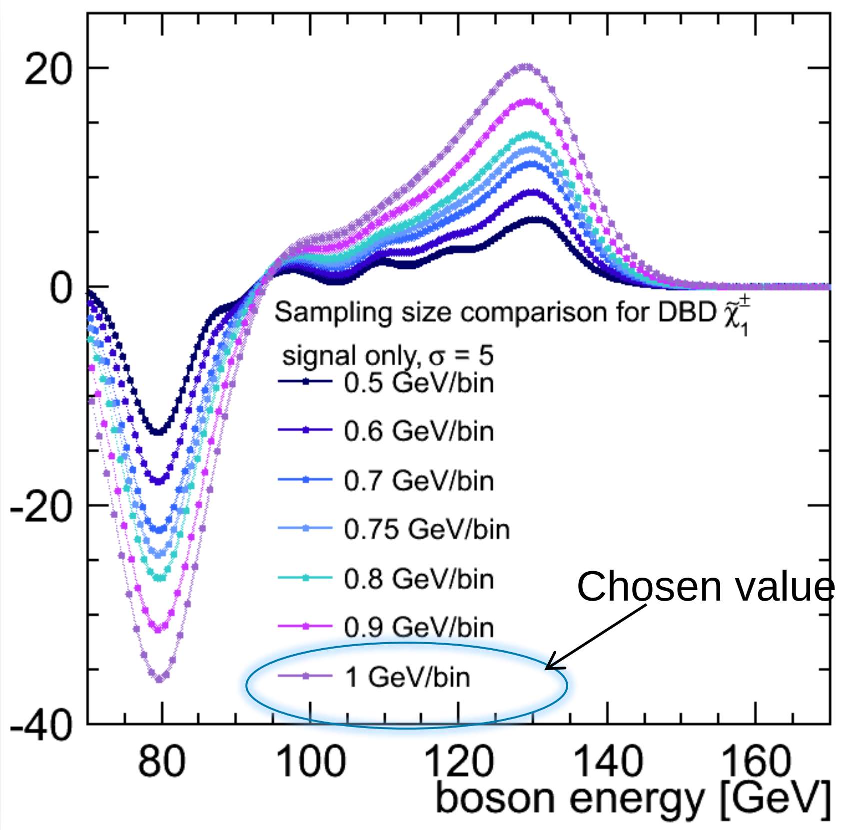




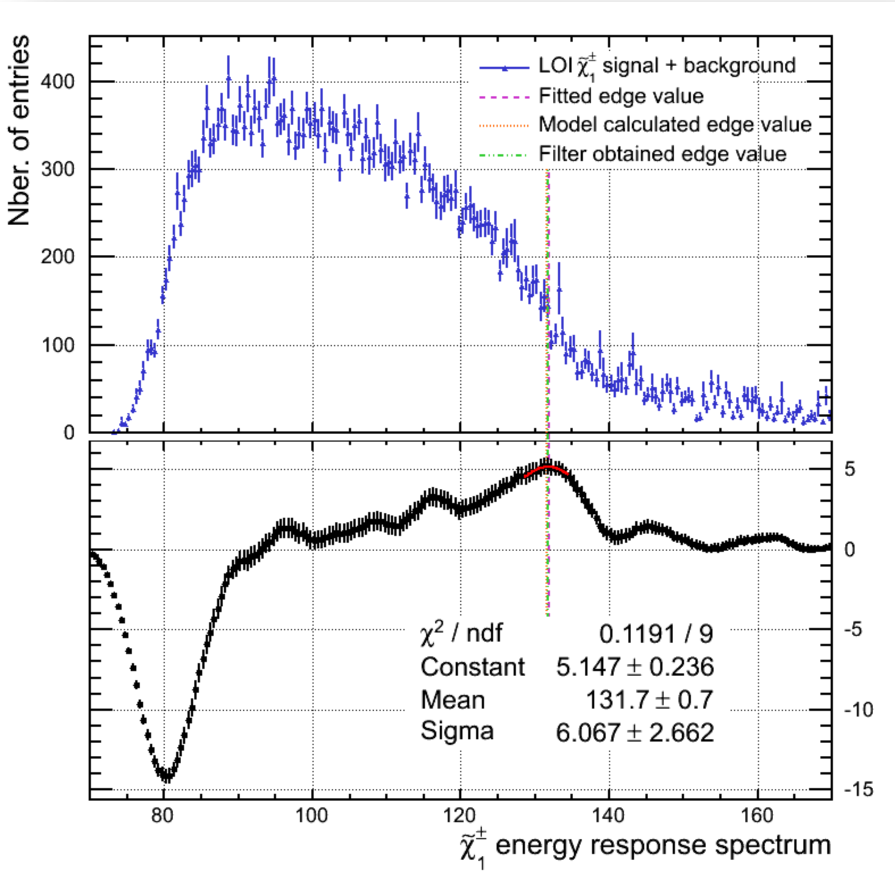
- There are two important filter characteristics that must be optimised: the bin size and the filter size.

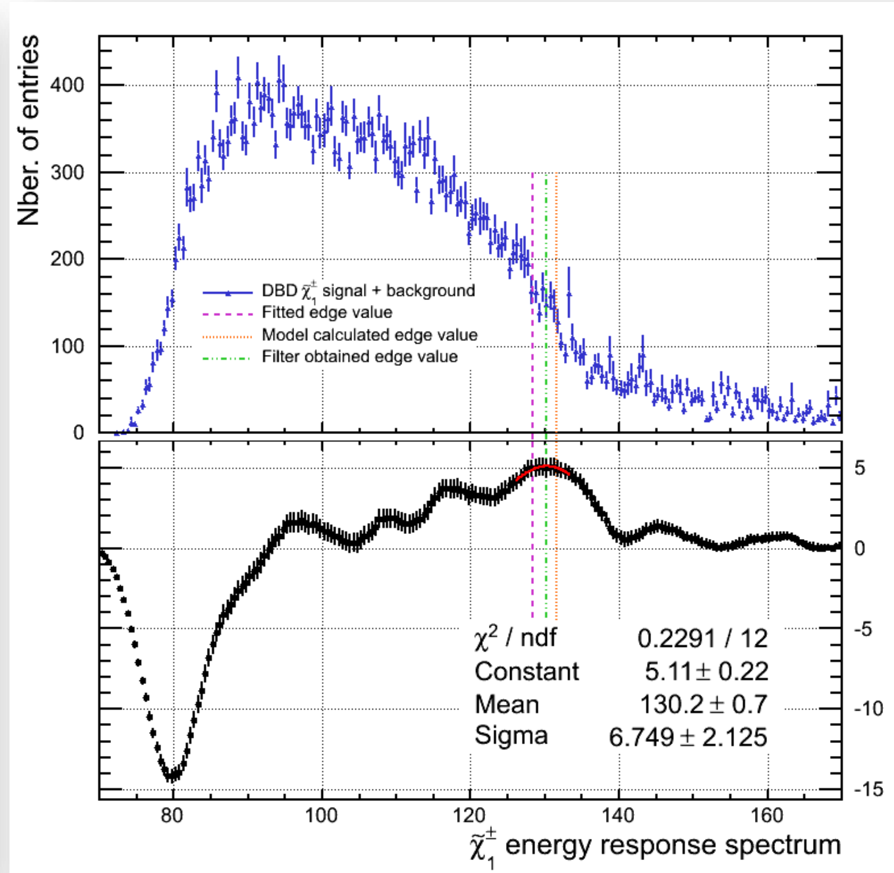
Filter response after applying the FDOG Filter to the $\tilde{\chi}_1^\pm$ energy distribution:





FIR Edge Extraction Comparison – LOI to DBD

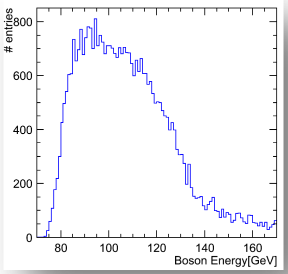


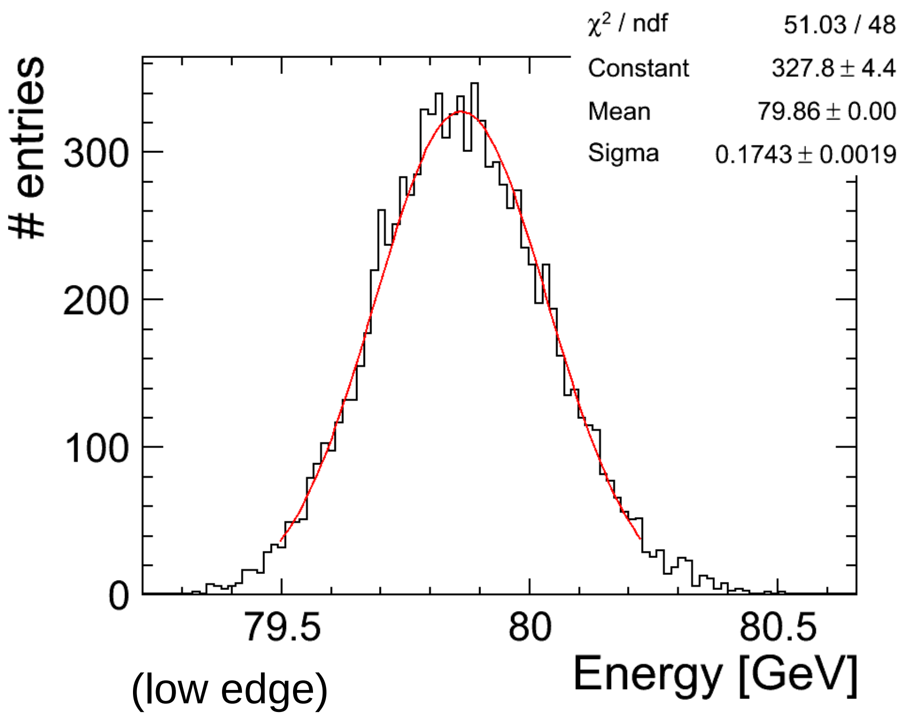


In the **LOI** case: the fitted and filter values are extremely close to the real model value.
In the **DBD** case: the filter value is much closer to the model one than the fitted edge.

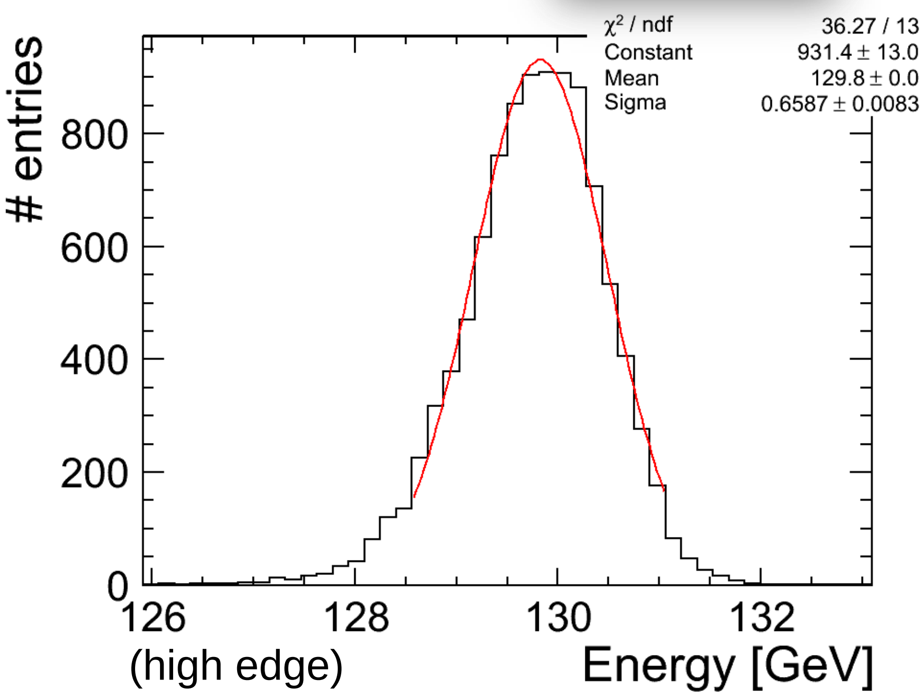
Toy MC for the Filter Edge Extraction

- To estimate the statistical precision of the edge extraction → toy MC
- 10000 $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ energy spectra have been produced
- The FDOG filter was then applied 10000 times
- Example: for the $\tilde{\chi}_1^{\pm}$ case:



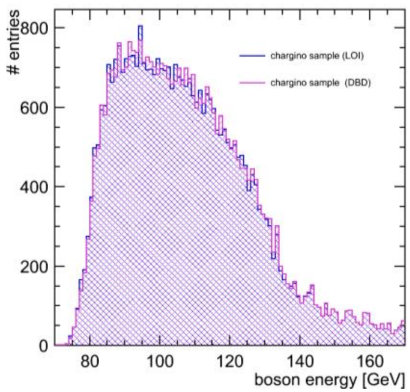


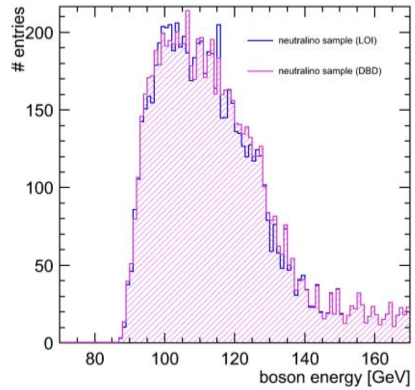
(low edge)



(high edge)

Edge Extraction Comparison





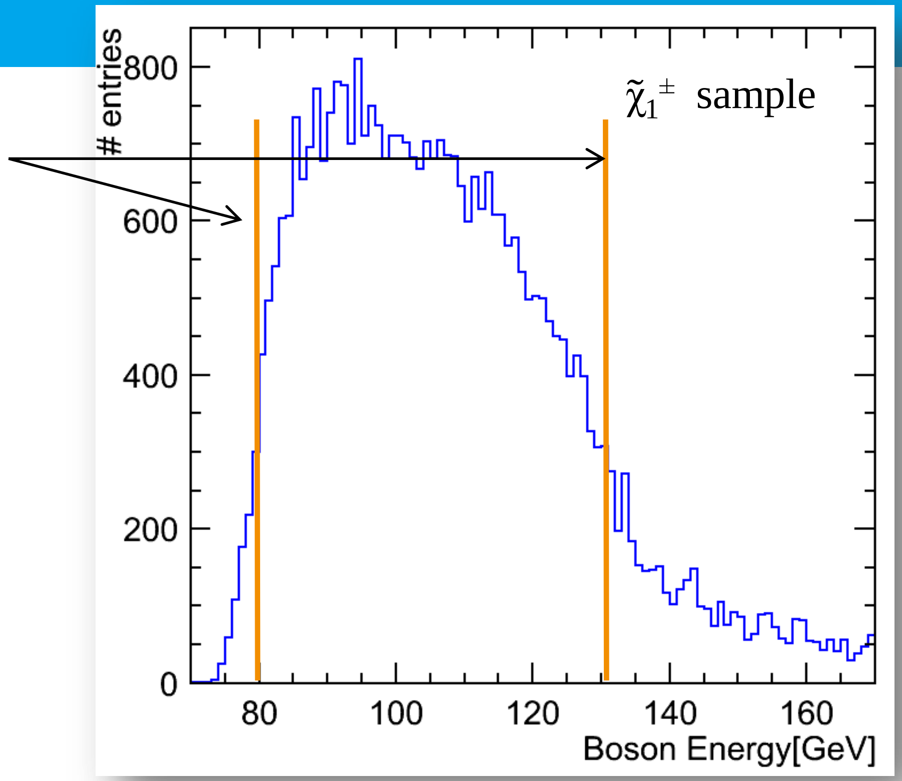
	True	80.17	131.53	93.24	129.06
	Sim.	**Edge W_{low} [GeV]**	**Edge W_{high} [GeV]**	**Edge Z_{low} [GeV]**	**Edge Z_{high}[GeV]**
	LOI	79.7±0.3	131.9±0.9	91.0±0.7	133.6±0.5
	DBD	79.5±1.7	128.3±1.2	91.9±0.8	127.9±0.7
filter	LOI	80.3±0.6	131.7±0.7	91.6±0.7	129.0±0.6
filter	DBD	80.1±0.2	130.2±0.7	91.9±0.2	127.2±0.7

The filter extraction method is preferable:

- it is more stable
- provides smaller uncertainties in determining the edge position.

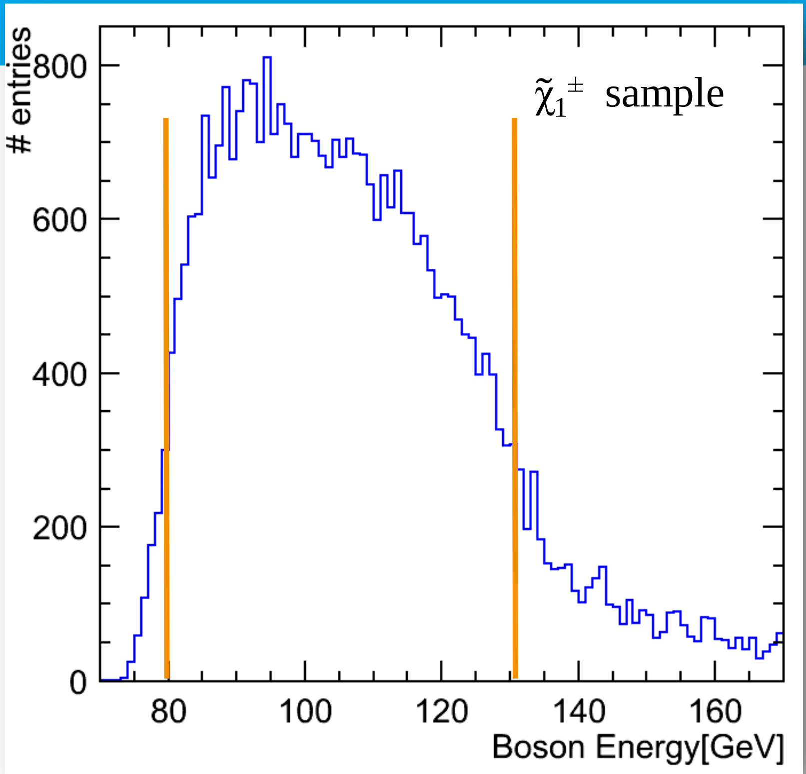
Applying an FIR Filter

- Goal: find edge positions in spectrum
- Strategy: use weighted sums of bin content values to find patterns in distribution



Applying an FIR Filter

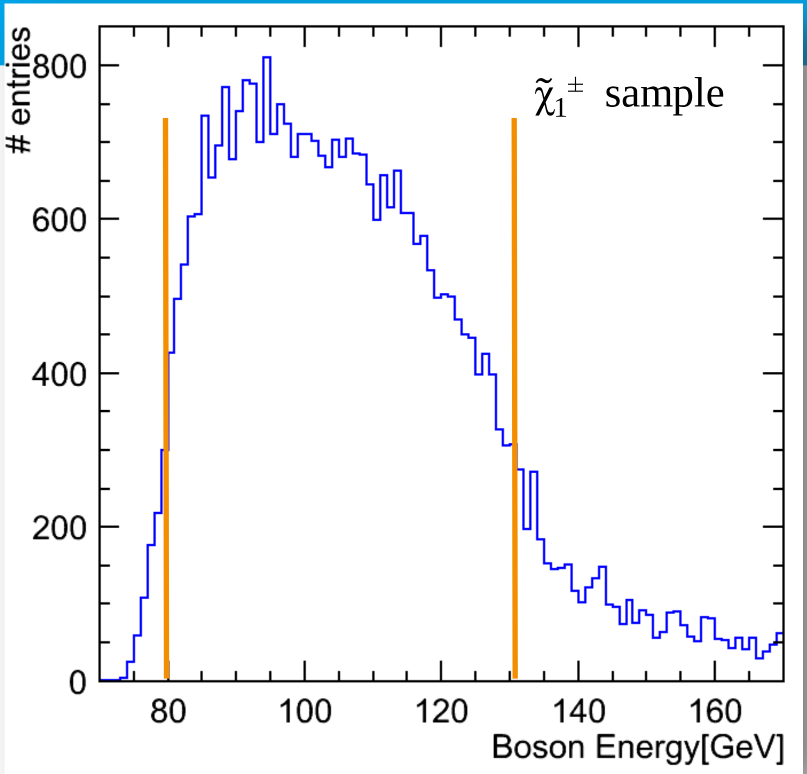
- Goal: find edge positions in spectrum
- Strategy: use weighted sums of bin content values to find patterns in distribution
- Consider the histogram as an array of bin content values



Bin #	1	2	3	...	98	99	100
Entries	0	15	28	...	34	22	4

Applying an FIR Filter

- Goal: find edge positions in spectrum
- Strategy: use weighted sums of bin content values to find patterns in distribution
- Consider the histogram as an array of bin content values
- Consider an array of chosen weights (smaller than the histogram!)
- Create new array of the same size:
  - Each entry in the new array is the weighted sum of the bin content values from the bins surrounding the corresponding bin in the original array.
  - The array is filled using the **same** (finite) weights each time.
- The value of the output depends on the pattern in the neighbourhood of the considered bin and NOT on the position of the bin
- The pattern of weights = kernel
- The filter application = convolution



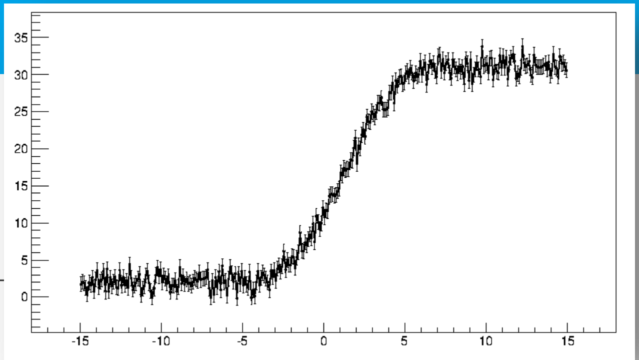
Bin #	1	2	3	...	98	99	100
Entries	0	15	28	...	34	22	4

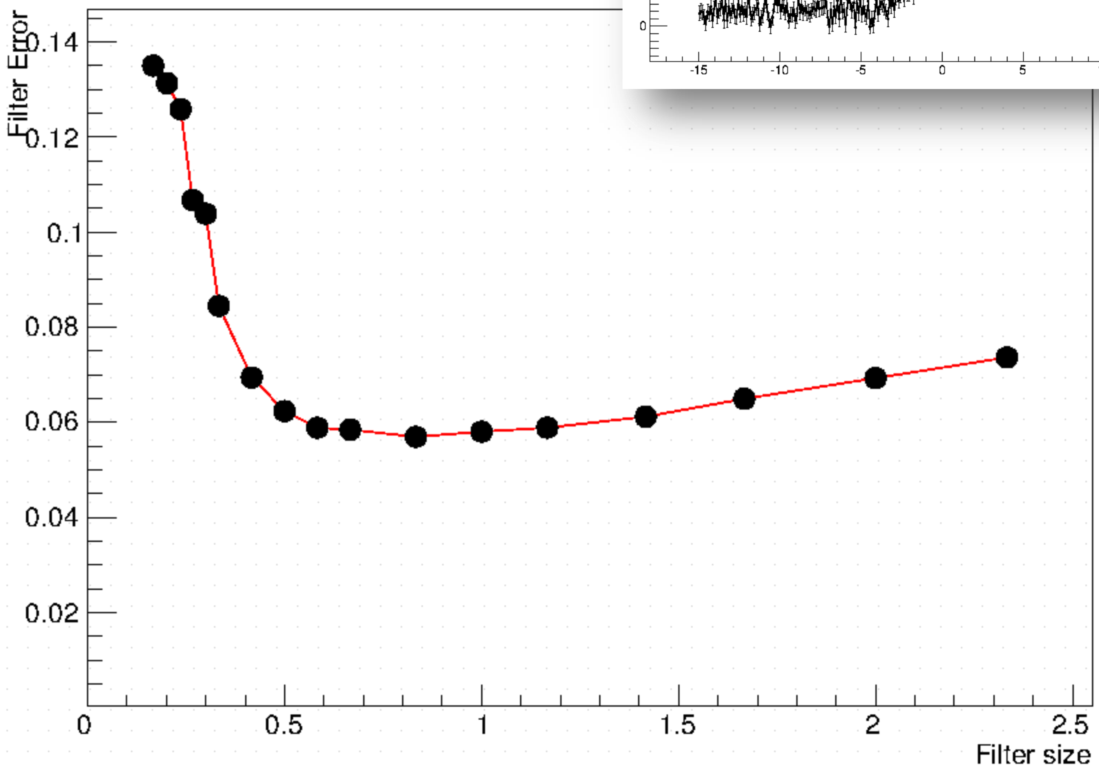
$$\underbrace{w_1 \times 0 + w_2 \times 15 + w_3 \times 28}_{} = \text{val2}$$

Entries	val1	val2	val3	...	val98	val99	val100

Testing the FDOG Filter

Studied the effect of the filter size on a smeared step edge monte carlo data.





S. Caiazza

The FDOG filter does indeed perform best.
The filter size should be comparable to the size of the edge feature.
We chose $\sigma = 5$ bins.

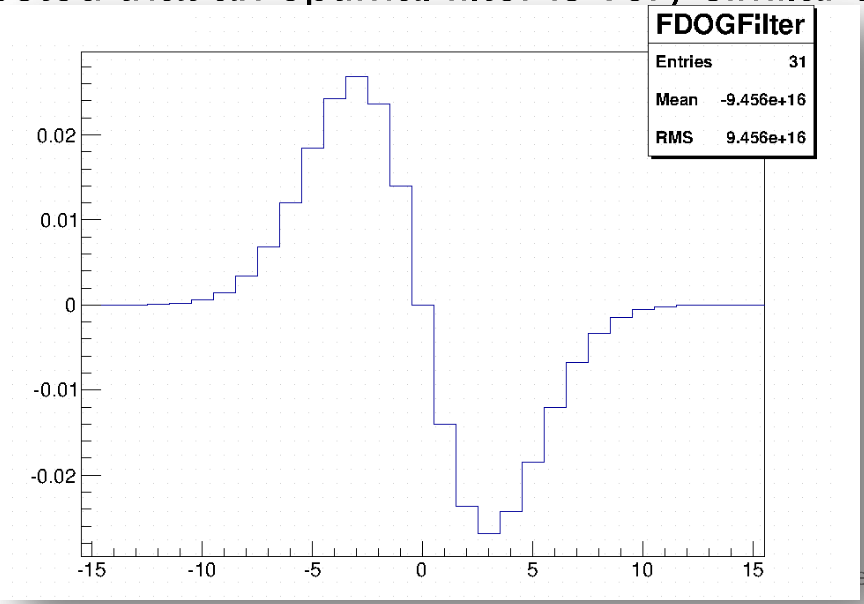
Choosing the Appropriate Filter

- The first derivative as kernel works
- It is however a high pass filter → may be rather noisy
- In order to choose an apropriate filter one can apply the following criteria:

  Canny's criteria: [J. F. Canny. **A computational approach to edge detection.** *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 679-698, 1986]

  - Good detection: probability of obtaining a peak in the response must be high
  - Localisation: standard deviation of the peak position must be small
  - Multiple response minimisation: probability of false postive detection must be small

- Canny has suggested that an optimal filter is very similar to the first derivative of a Gaussian

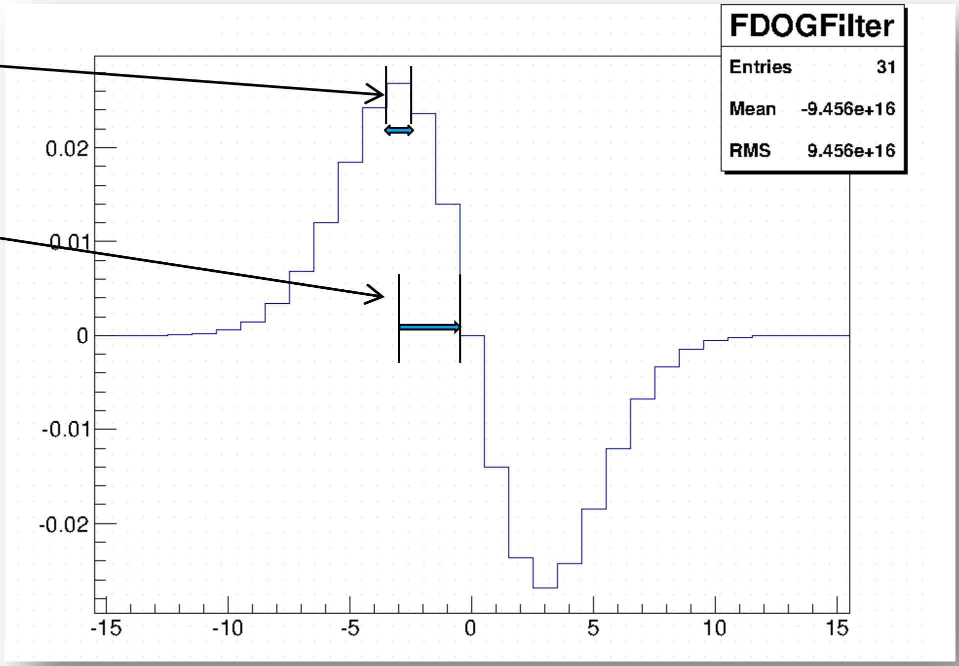


Testing the FDOG Filter

> There are two important filter characteristics that must be optimised:

the bin size

the filter size



It is crucial to strike the right balance between the two:

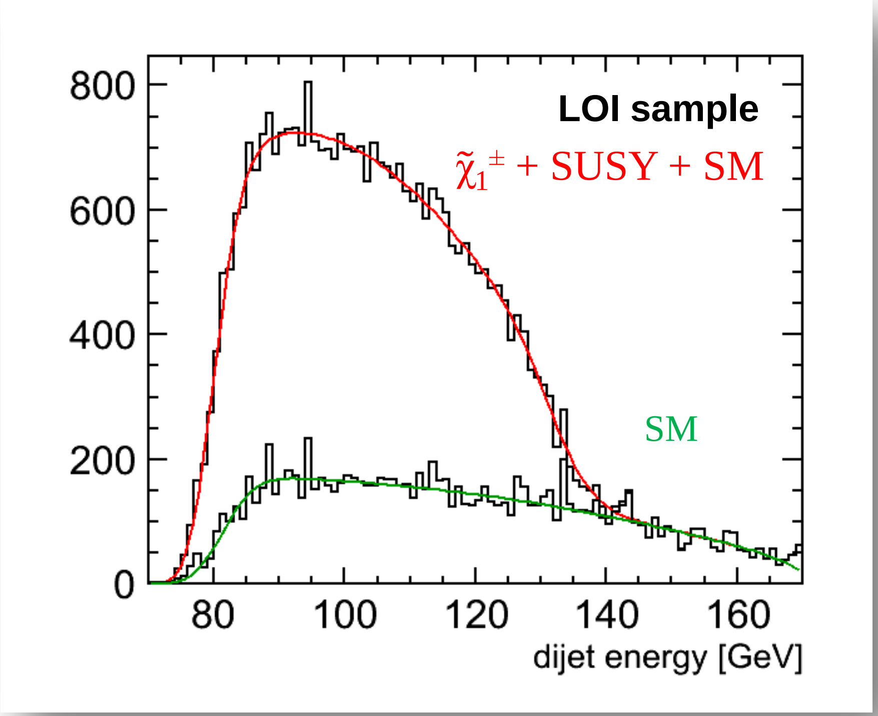
- If the bin size is too small → the filter picks up a lot of statistical fluctuations
- If the filter size is too large → the edge position cannot be localised anymore

A toy MC study is needed to optimise the filter and bin size!

$\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Mass Measurement – "Endpoint" Method

> Fit dijet energy spectrum and obtain edge positions:

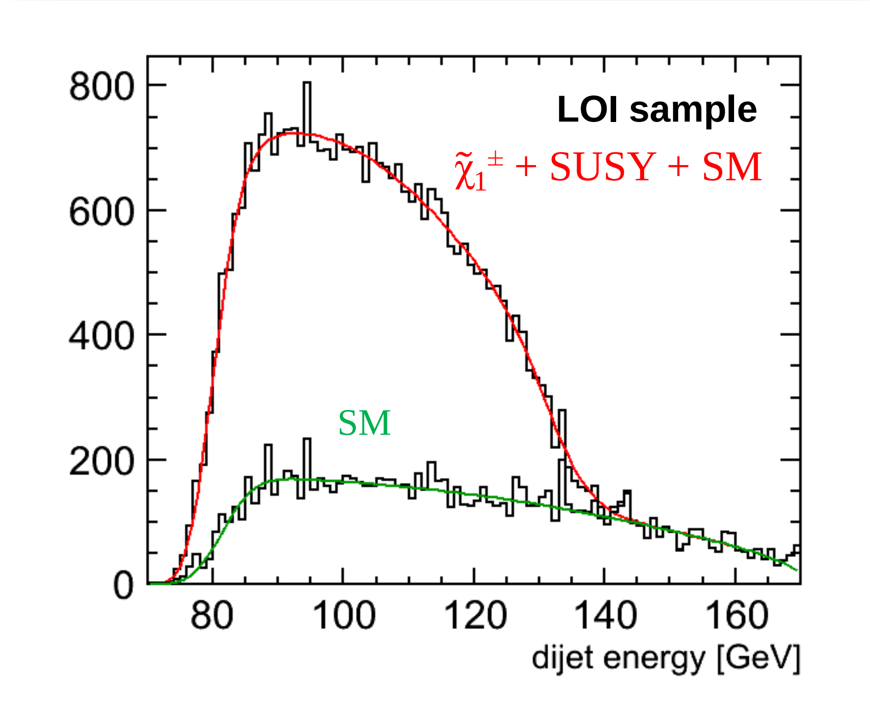
$$f(x; t_{0\_1}, b_{0\_2}, \sigma_{1\_2}, \gamma) = fSM + \int_{t_0}^{t_1} (b_2 t^2 + b_1 t + b_0) V(x - t, \sigma(t), \gamma) dt$$

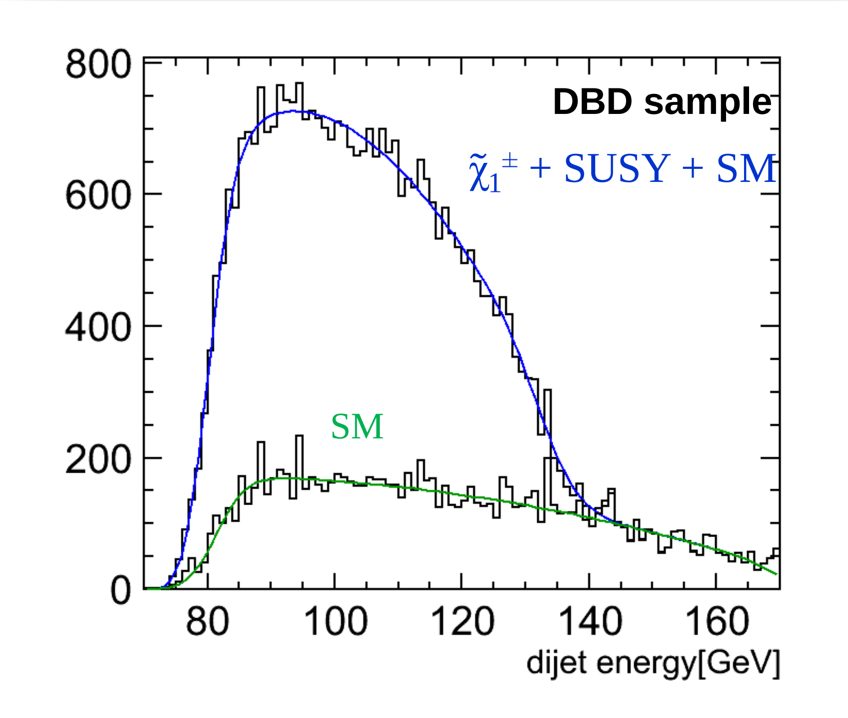


Where:

- The polynomial accounts for the slope of the initial spectrum
- The Voigt function accounts for the detector resolution and gauge boson width

Endpoint Extraction Comparison – LOI to DBD



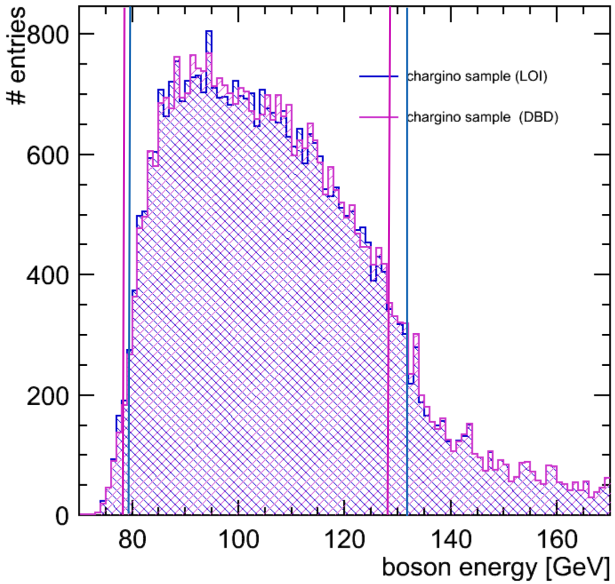


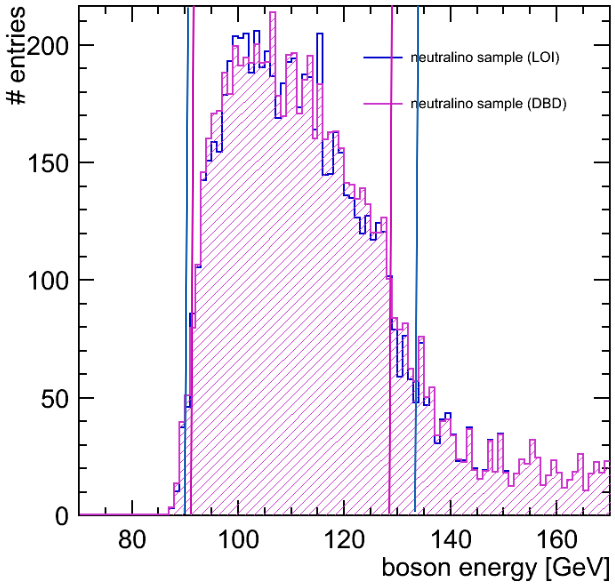
$E_{low} \simeq 79.7 \pm 0.3$ GeV
$E_{high} \simeq 131.9 \pm 0.9$ GeV

$E_{low} \simeq 79.5 \pm 1.7$ GeV
$E_{high} \simeq 128.3 \pm 1.2$ GeV

- The DBD distribution appears slightly shifted towards lower energies. Nevertheless, the two distributions agree very well.

Issues of the „Endpoint Method“





Sim.	Edge W_{low} [GeV]	Edge W_{high} [GeV]	Edge Z_{low} [GeV]	Edge Z_{high}[GeV]
DBD	79.5±1.7	128.3±1.2	91.9±0.8	127.9±0.7
LOI	79.7±0.3	131.9±0.9	91.0±0.7	133.6±0.5

The fitting method appears to be highly dependent on small changes in the fitted distribution → it is clearly NOT appropriate for a comparing the simulation and reconstruction performance.

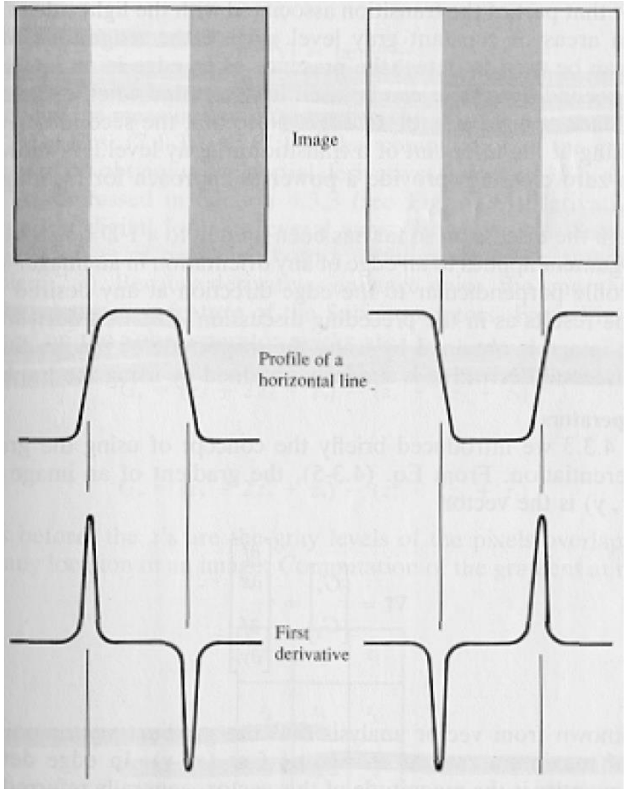
We need to apply a different edge extraction method!

Applying an FIR Filter – Example: the box function

- The changes of a function can be described by the derivative → interpret the histogram as a 1D function
- The points that lie on the edge of the distribution → detected by local maxima and minima of the first derivative

$$f'(x) = \lim_{h \to 0} \frac{f(x+h)-f(x)}{h} \approx f(x+1) - f(x) \quad (h = 1)$$

- The first derivative is approximated by using the **kernel [-1, 0, 1]**



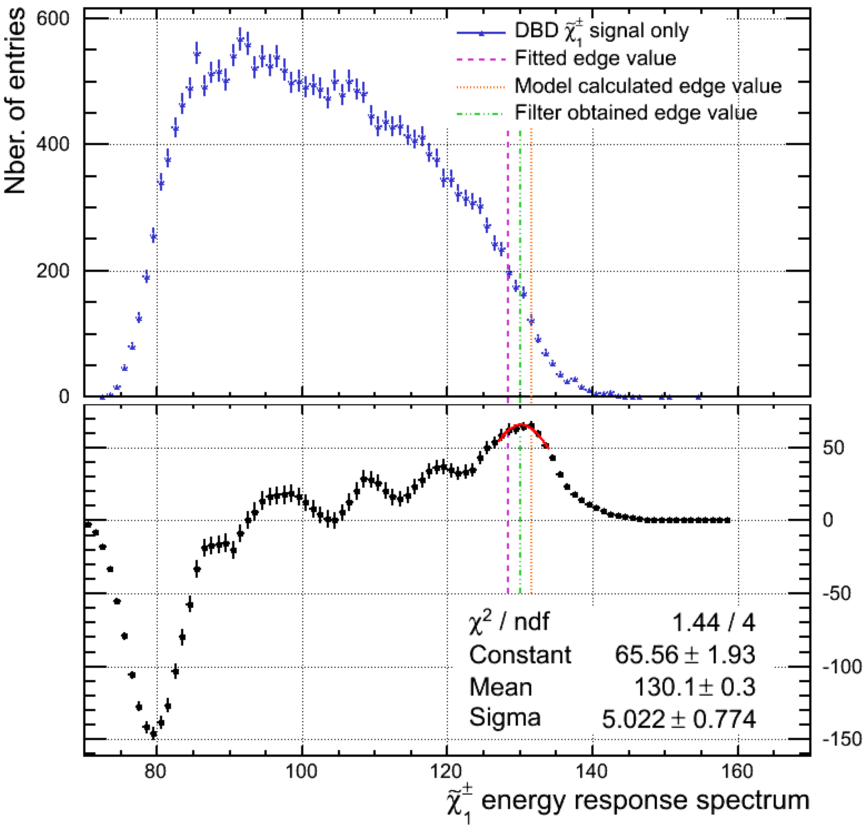
Applying an FIR Filter – Example: the box function

- The changes of a function can be described by the derivative → interpret the histogram as a 1D function
- The points that lie on the edge of the distribution → detected by local maxima and minima of the first derivative

$$f'(x) = \lim_{h\to 0}\frac{f(x+h)-f(x)}{h} \approx f(x+1) - f(x) \quad (h = 1)$$

- The first derivative is approximated by using the **kernel [-1, 0, 1]**
- The kernel is convoluted with the histogram:

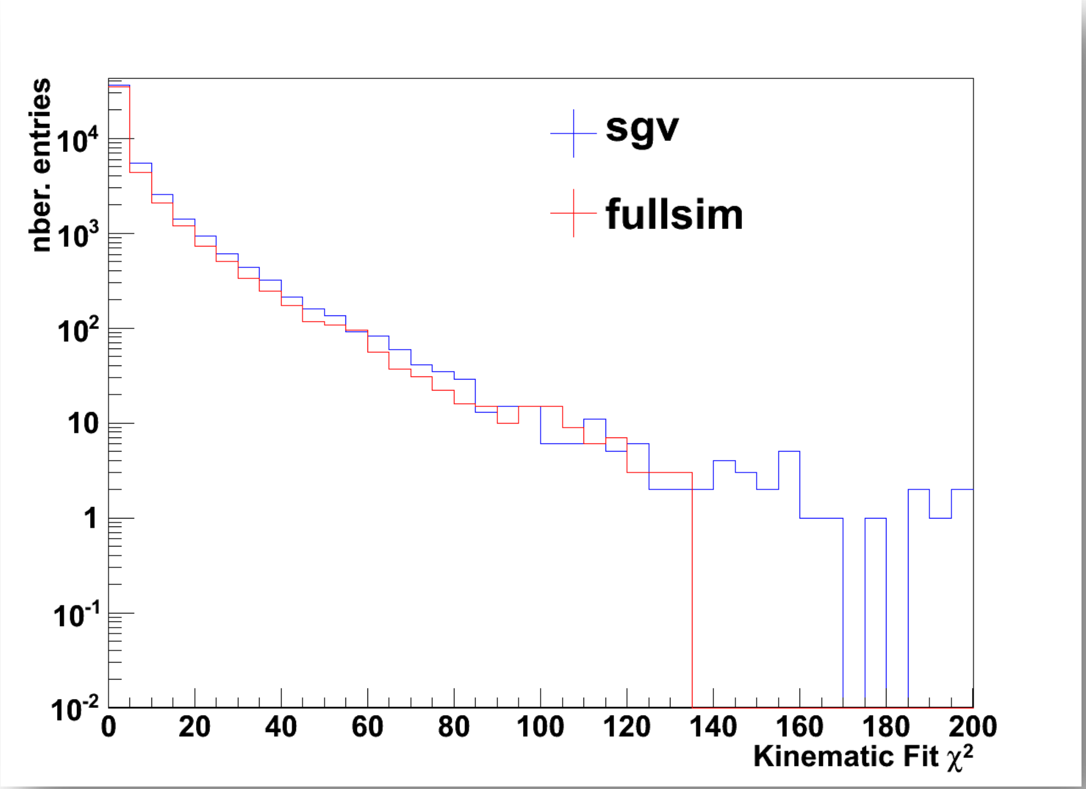
$$response_i = -1 \times bin_{i-1} + 0 \times bin_i + 1 \times bin_{i+1}$$



$\tilde{\chi}_1^\pm$ and $\tilde{\chi}_2^0$ Separation as Study case for Particle Flow

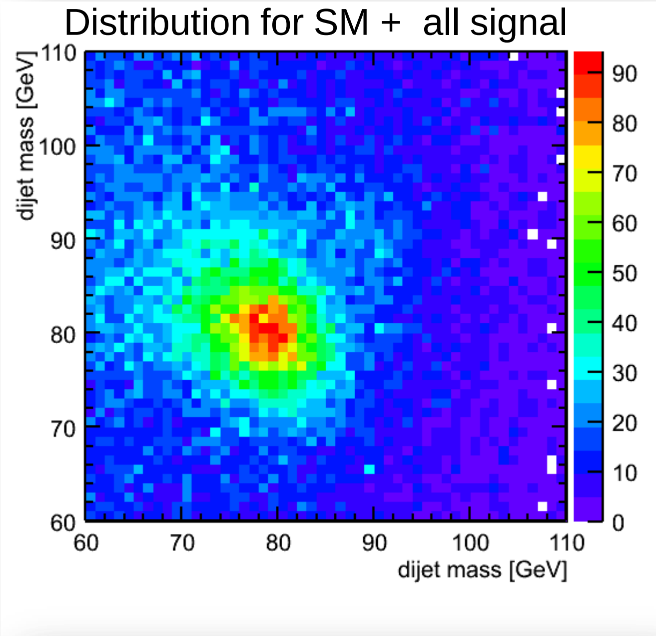
- Signal topolgy:
  4 jets and missing energy
- Event preselection (kinematics, etc.)
- Perform kinematic fit:
  equal mass constraint
  (determine best jet pairing)

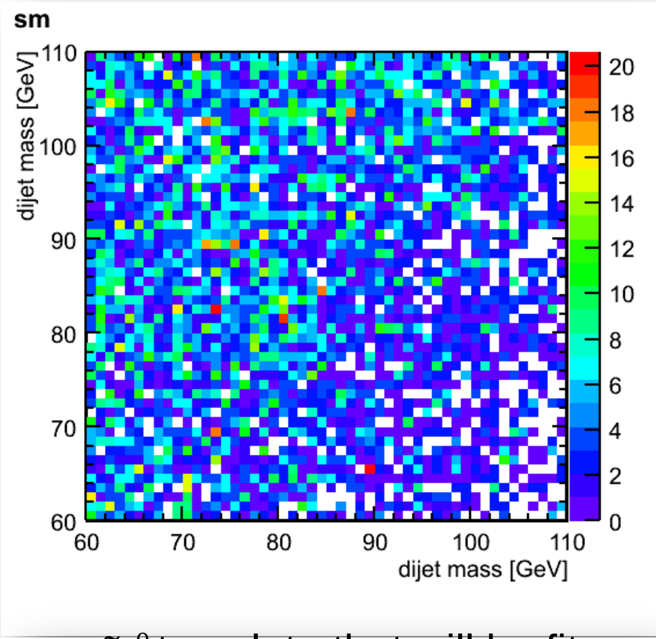
Kinematic fit χ^2



3.2. $\tilde{\chi}_1^{\pm}$ and $\tilde{\chi}_2^0$ Cross Section Measurement
3.2.2. 2D dijet mass fit

Example: DBD sample





Subtract SM

Perform template fit