# Reconstruction

Maarten de Jong

Reconstruction $\equiv$ Fit of model parameters to data

# Introduction

1. Precision of estimation of parameters increases with more information
2. For a given set of data (i.e. amount of information),
   maximum likelihood yields smallest variance of parameters
3. Science output of KM3NeT relies on reconstruction

# Introduction

1. Precision of estimation of parameters increases with more information

2. For a given set of data (i.e. amount of information),

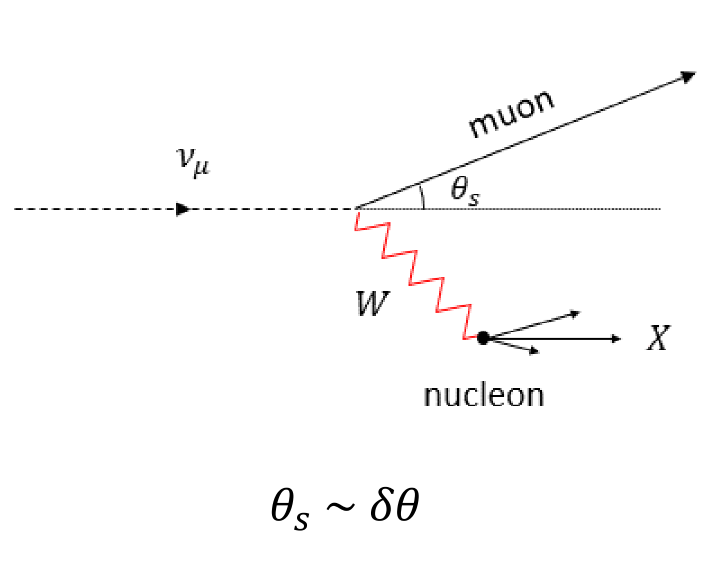
   maximum likelihood yields smallest variance of parameters

3. Science output of KM3NeT relies on reconstruction

PDF describing data

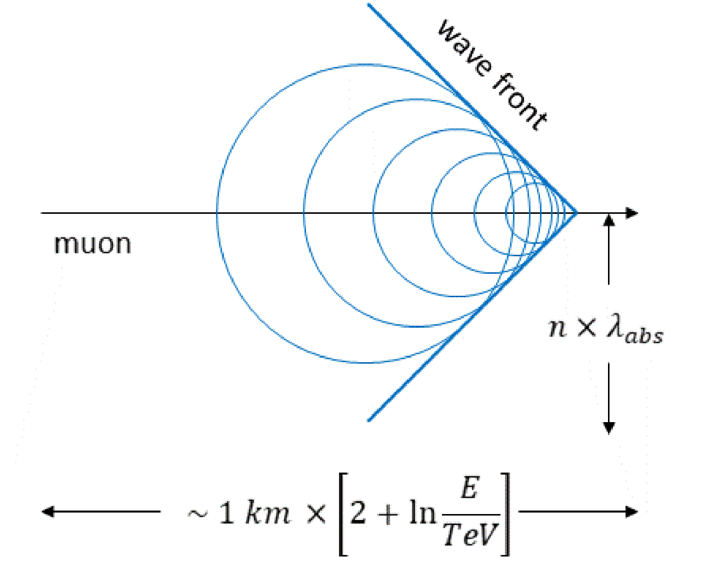
# “golden” channel

neutrino interaction




+

Cherenkov light




=

1. Target mass

   overcome small cross section

2. Transparency

   sparse sensor array

3. Muon range

   good angular resolution

# Me & muon reconstruction

- *Muoproduction of J/$\psi$-mesons and the gluon distribution in nucleons,*[¶]

  PhD thesis, M. de Jong

- Track reconstruction for the Chorus experiment[§]

- Track reconstruction for KM3NeT/ARCA and KM3NeT/ORCA

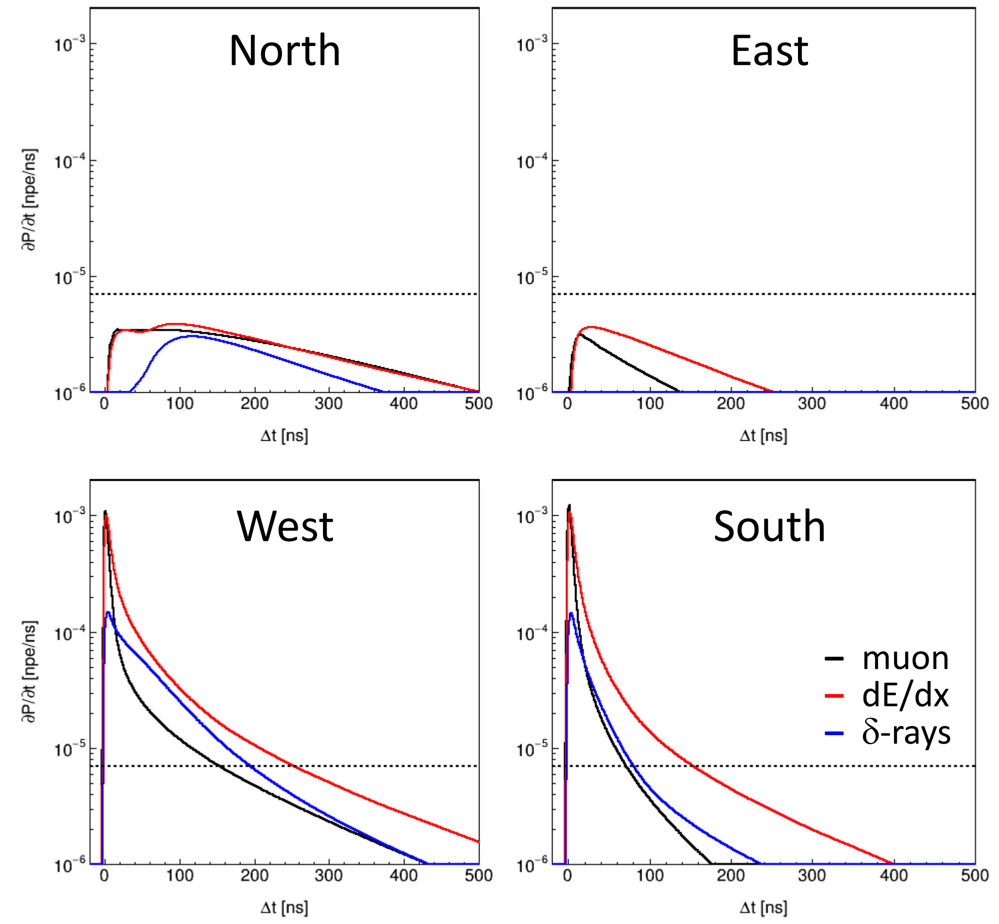
---

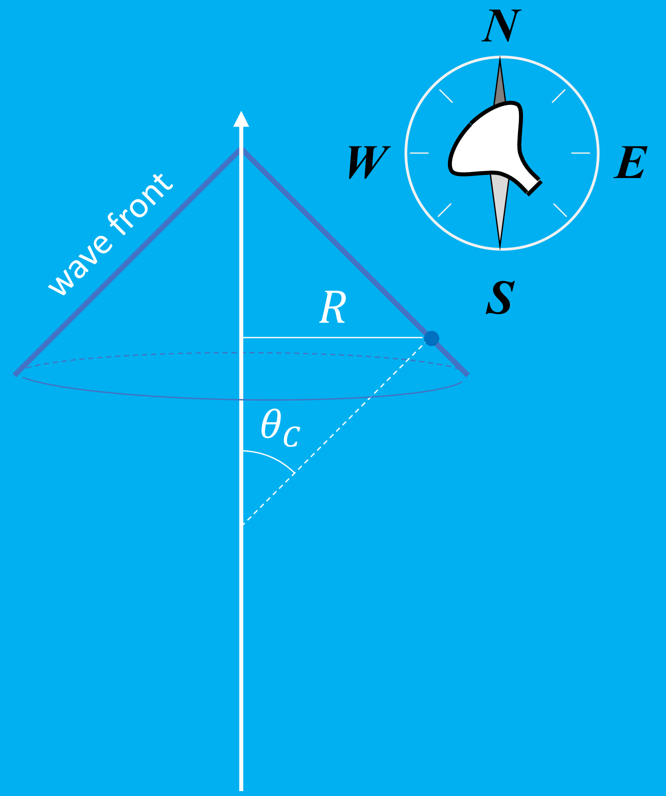
¶ Track reconstruction near muon beam to cover small angle scattering.

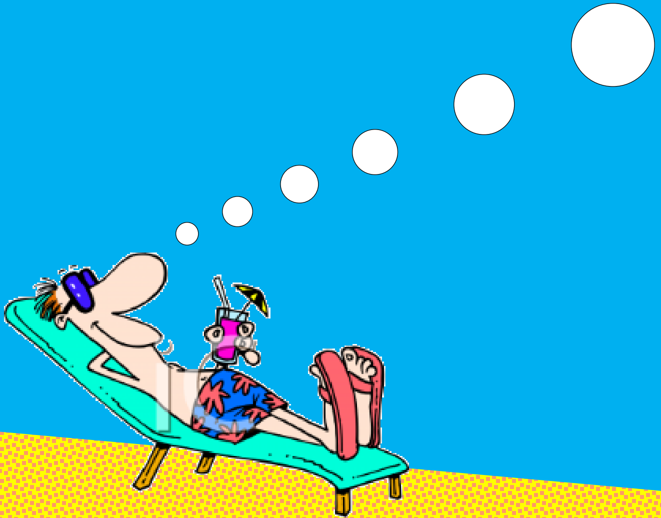
§ Track reconstruction incorporating multiple-Coulomb scattering in co-variance matrix (20% more events in physics target).

# Model

# PDF of Cherenkov light

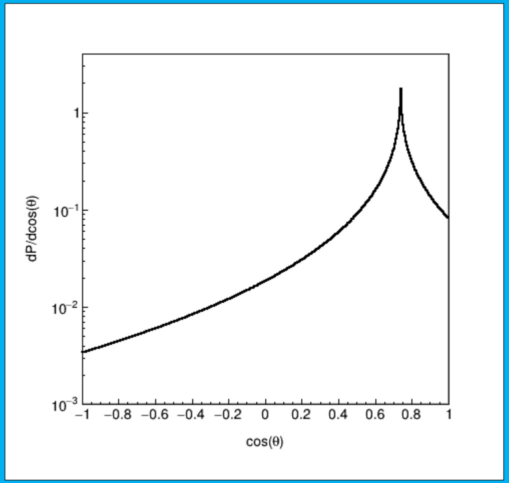


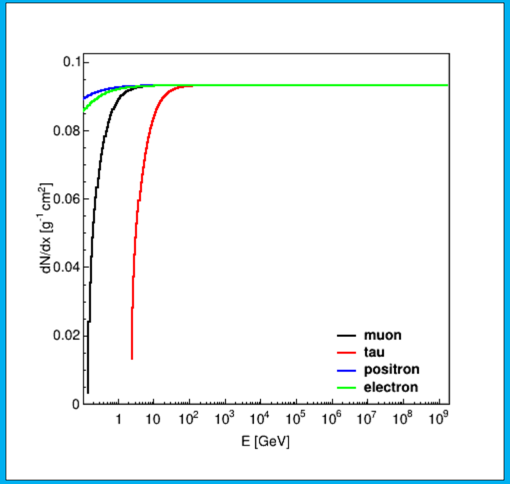



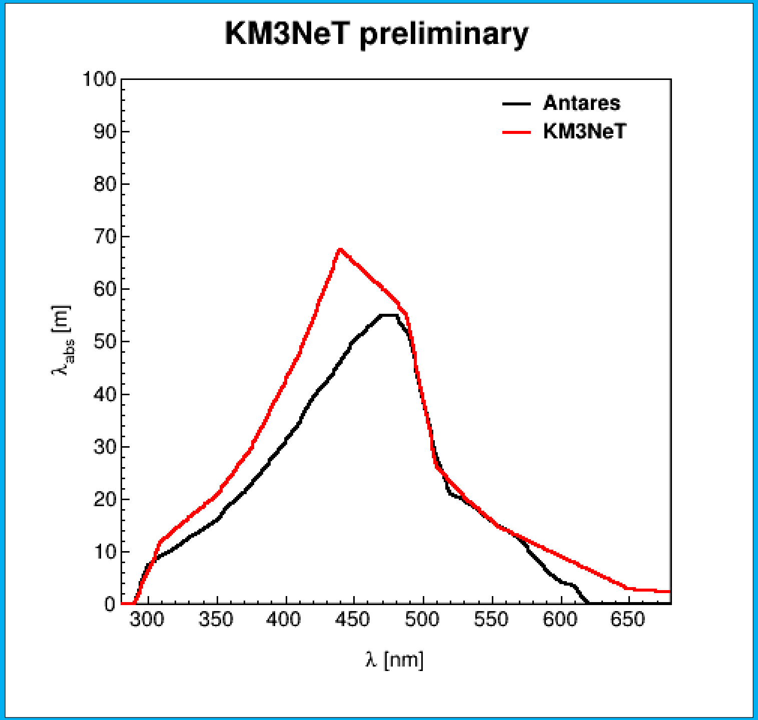
# Light generation

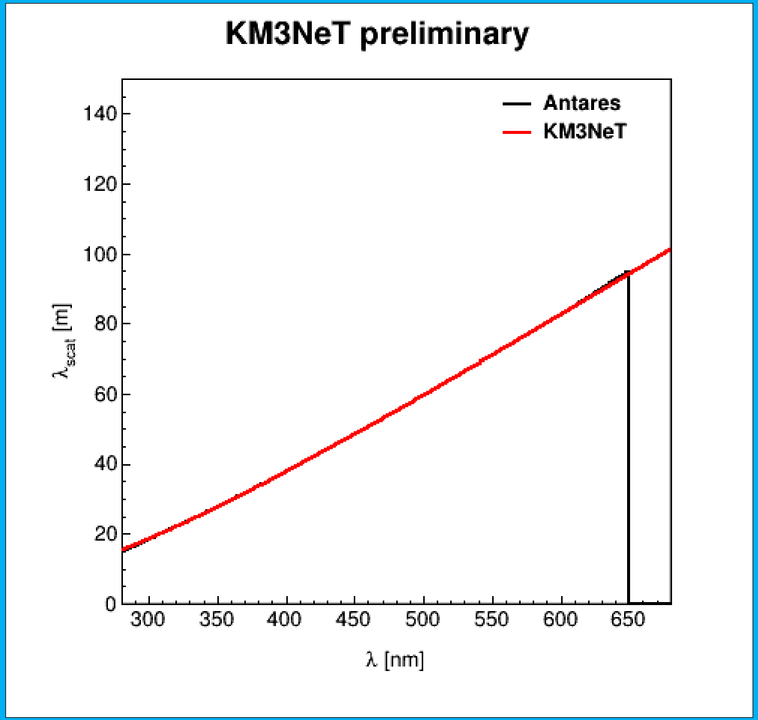
- Cherenkov formula: $$\frac{\partial^2 N}{\partial x \partial \lambda} = \frac{10^9}{\lambda^2} \times 2\pi \times \alpha_{EM} \times (1 - n^{-2})$$
- Muon energy loss: $$-\frac{dE}{dx} = a + bE$$
- EM particle shower: $$1\ \mathrm{GeV} \ \equiv\ 4.73\ m$$
- $\delta$-rays:

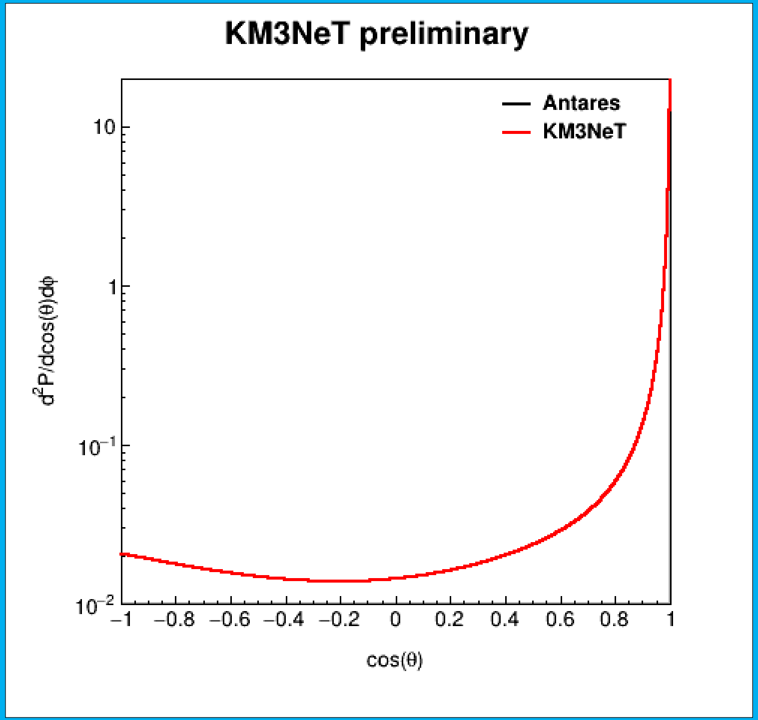


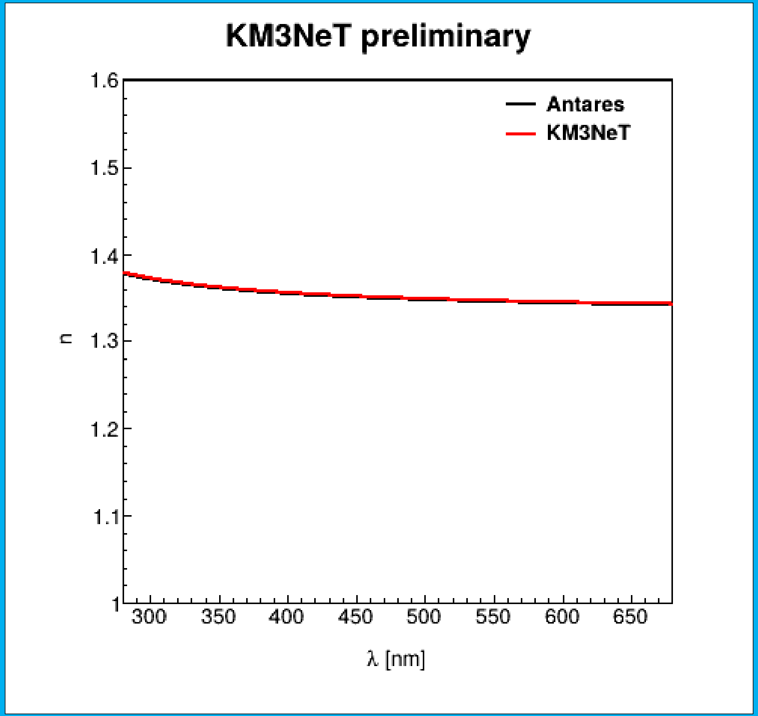
# Light propagation (1/2)



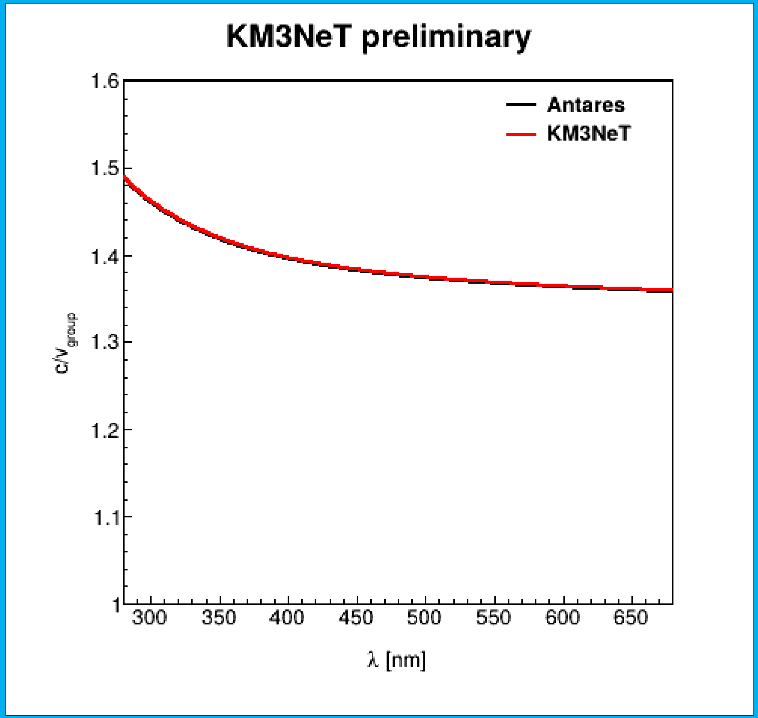



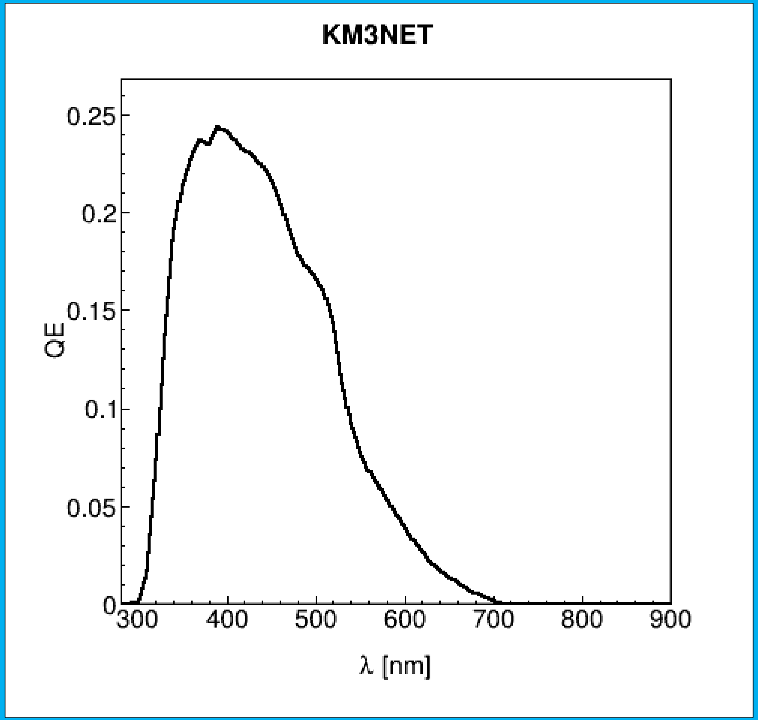
# Light propagation (2/2)

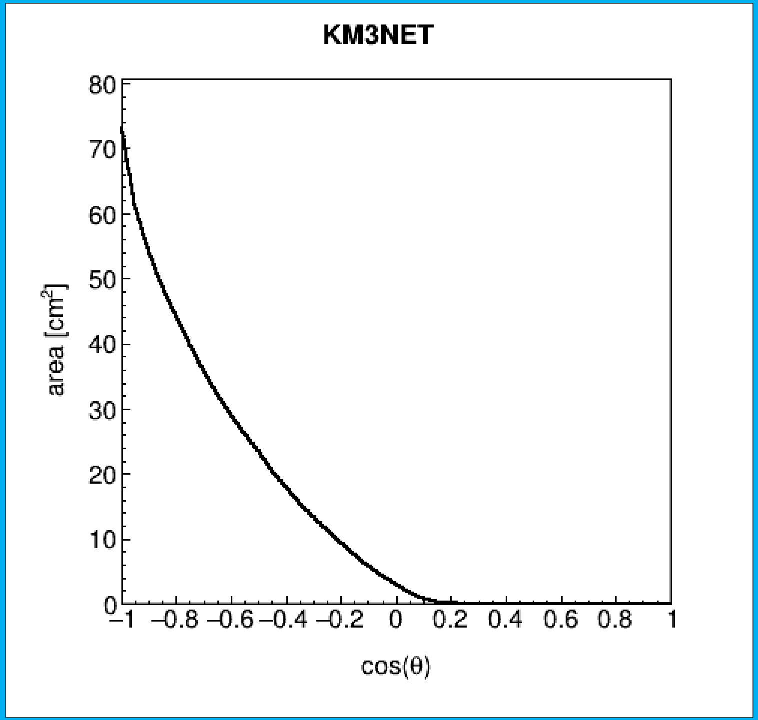


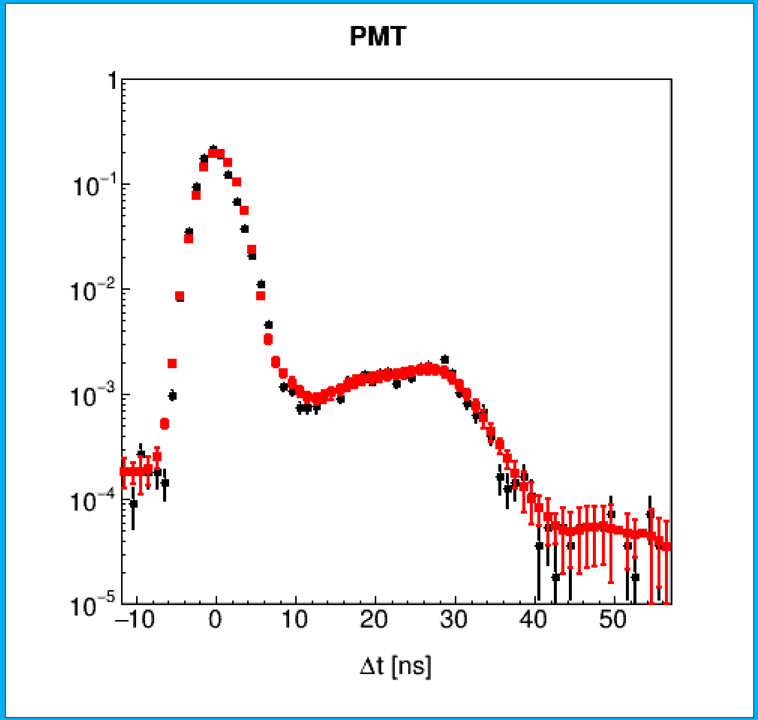

$$\Rightarrow \quad \frac{\partial n}{\partial \lambda} \neq 0 \quad \Rightarrow$$



# Light detection



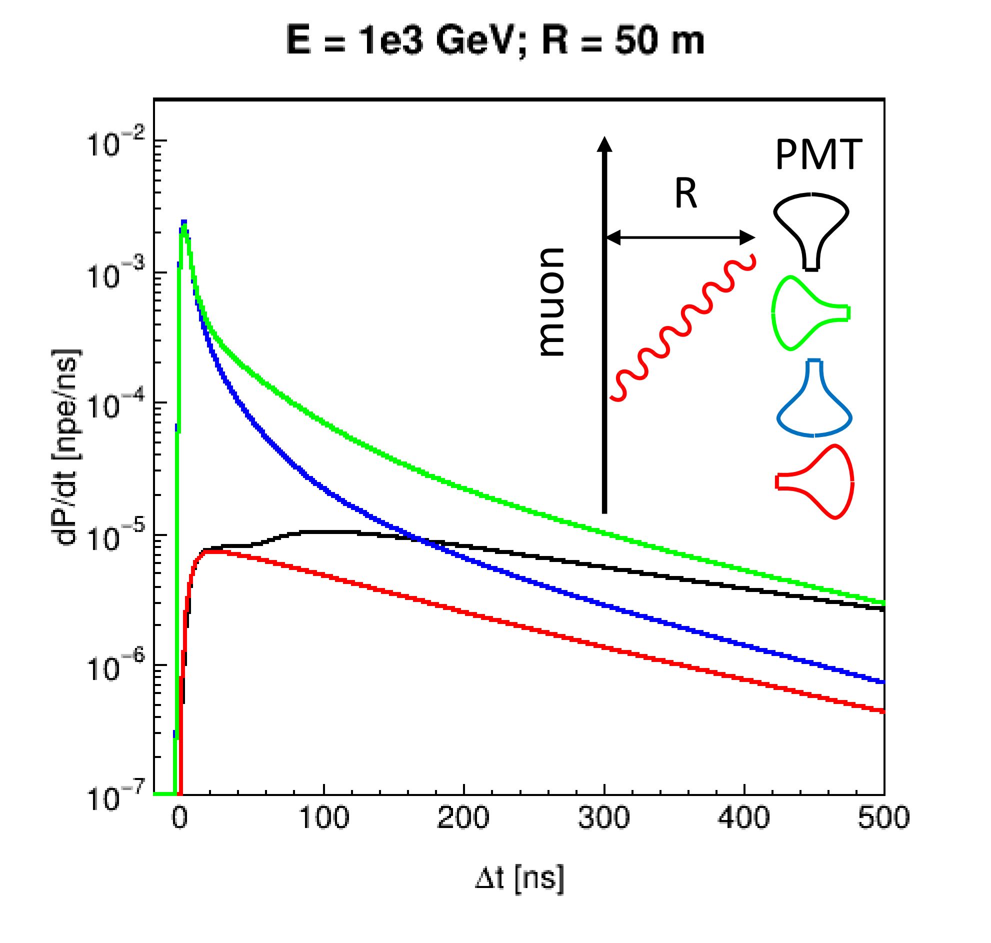

***a posteriori*** **correction**

# PDF (1/2)

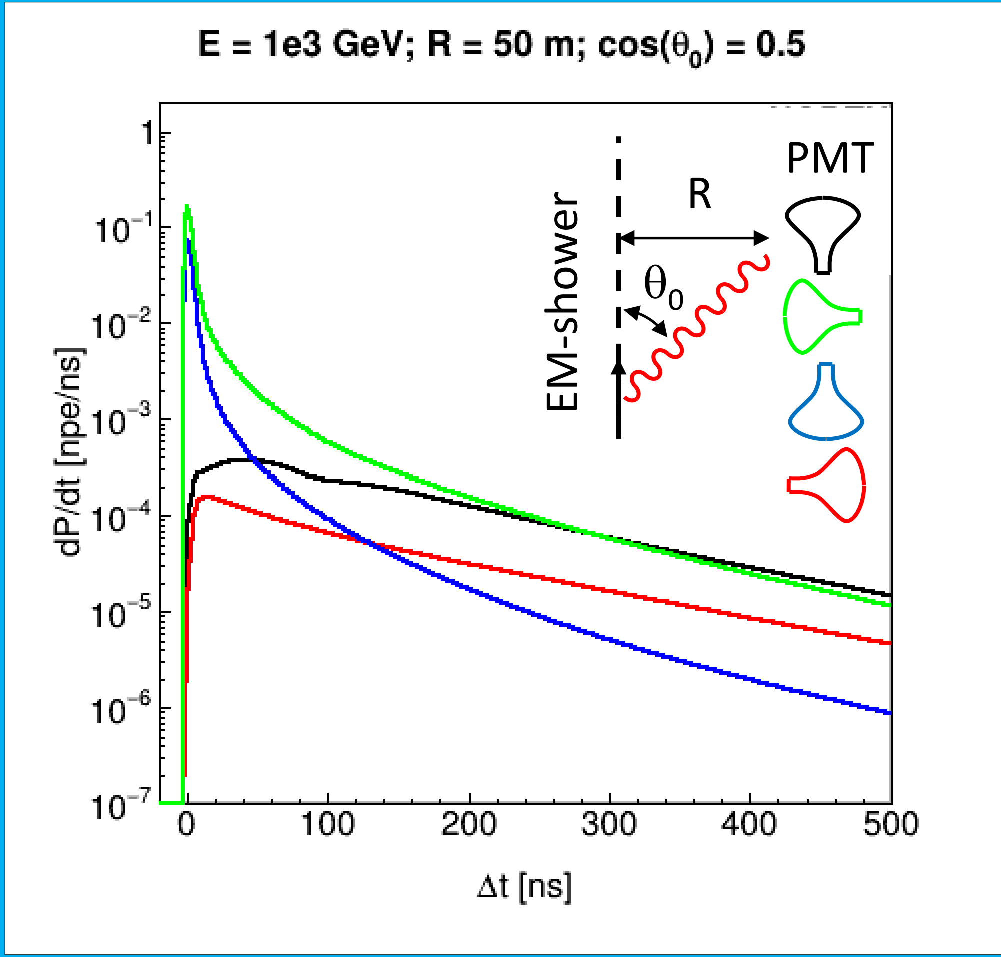
- PDF of arrival time of Cherenkov light involves numerical integration
  - usually done with Monte Carlo technique
  - can efficiently be calculated from first principles when scattering length is much larger than absorption length (see documentation, Maarten and Emanuel)

$$\frac{d\mathcal{P}}{dt} = \iiint d\lambda\, dz\, d\phi_0 \,\frac{1}{2\pi} \underbrace{\frac{dN}{dx}}_{\frac{\partial N_s}{\partial u}} \frac{1}{\lambda_s} \underbrace{\left(\frac{\partial t}{\partial u}\right)^{-1}}_{\frac{1}{1-\cos\theta_s}} \varepsilon(\cos\theta_{\oslash})\, QE(\lambda)\, e^{-d/\lambda_{att}} \frac{dP_s}{d\Omega_s} d\Omega$$

# PDF (2/2)



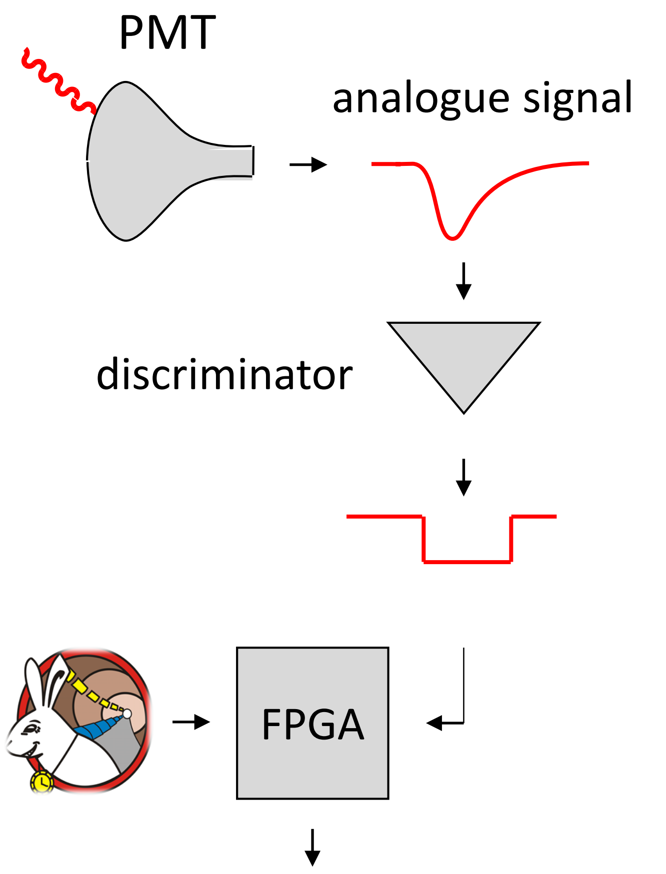
E = 1e3 GeV; R = 50 m
dP/dt [npe/ns]
Δt [ns]
muon
R
PMT

E = 1e3 GeV; R = 50 m; cos(θ0) = 0.5
dP/dt [npe/ns]
Δt [ns]
EM-shower
R
θ0
PMT

# data

# Off-shore signal processing



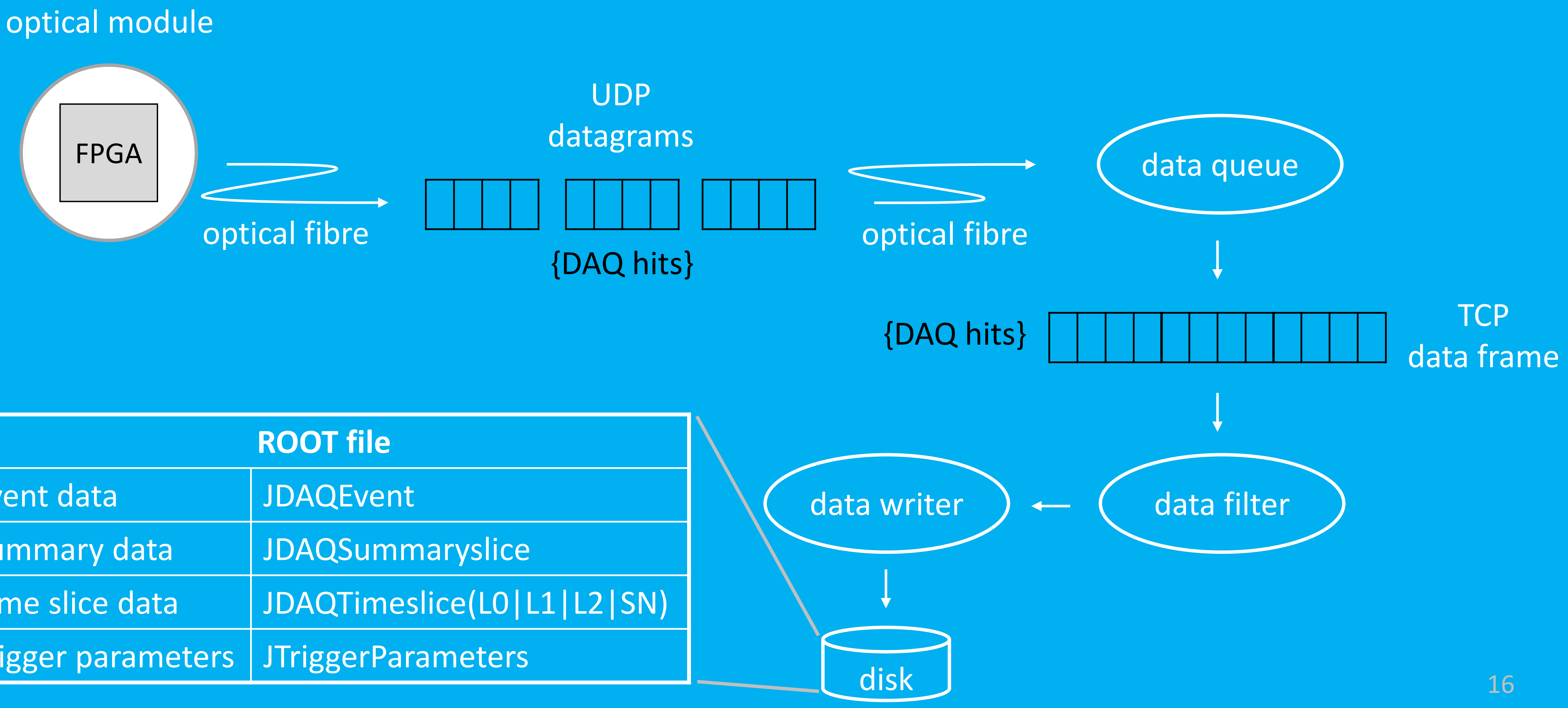

- time (of leading edge)
- time-over-threshold

| 3” PMT | |
|---|---|
| QE | 0.25 |
| TTS | 1.5 ns |
| gain | $10^6$ |
| gain spread | 0.3 |

| discriminator | |
|---|---|
| threshold | 0.20 p.e. |

| DAQ hit | | |
|---|---|---|
| PMT | | 1B |
| time | 1 ns | 4B |
| time-over-threshold | 1 ns | 1B |

# On-shore signal processing




| ROOT file | |
|---|---|
| Event data | JDAQEvent |
| Summary data | JDAQSummaryslice |
| Time slice data | JDAQTimeslice(L0\|L1\|L2\|SN) |
| Trigger parameters | JTriggerParameters |

# Data processing (1/3)

- Level-0 (L0)
  - L0 hit = calibrated DAQ hit (time as well as position and orientation of PMT)
- Level-1 (L1)
  - L1 hit = coincidences of two (or more) L0 hits in the same optical module within a predefined time window (usually around 10 ns)
- Level-2 (L2)
  - L2 hit = L1 hit with additional constraints (e.g. space angle between PMT axes)
- Supernova (SN)
  - SN hit = L1 hit with additional constraints (e.g. multiplicity of L0 hits)

See documentation.

# Data processing (2/3)

- Time slice
  - container of all DAQ hits according level-n selection
- Summary data
  - rate information per PMT (1B)
  - PMT status information (1b)
    - high-rate veto, FIFO (almost) full, etc.
- Trigger parameters
  - configuration of real-time data filter

```
// histogram binning
static int              JDAQRate::getN();
static const double*    JDAQRate::getData();

// histogram filling
double JDAQSummaryFrame::getRate(const int tdc);
double JDAQSummaryFrame::getWeight(const int tdc);
```

See documentation.

# Data processing (3/3)

- Event
  - cluster of hits
    - normal: cluster of causally related L2 hits
    - mixed: cluster of one L2 hit with causally related L0 hits

- Note that
  - different triggers can be applied to the same data → trigger mask
  - same event can be triggered multiple times → number of overlays
  - cluster algorithm is known in literature as "*clique*" → also useful in analysis
  - you can trigger on anything (incl. external signals) → augment science scope

See documentation.

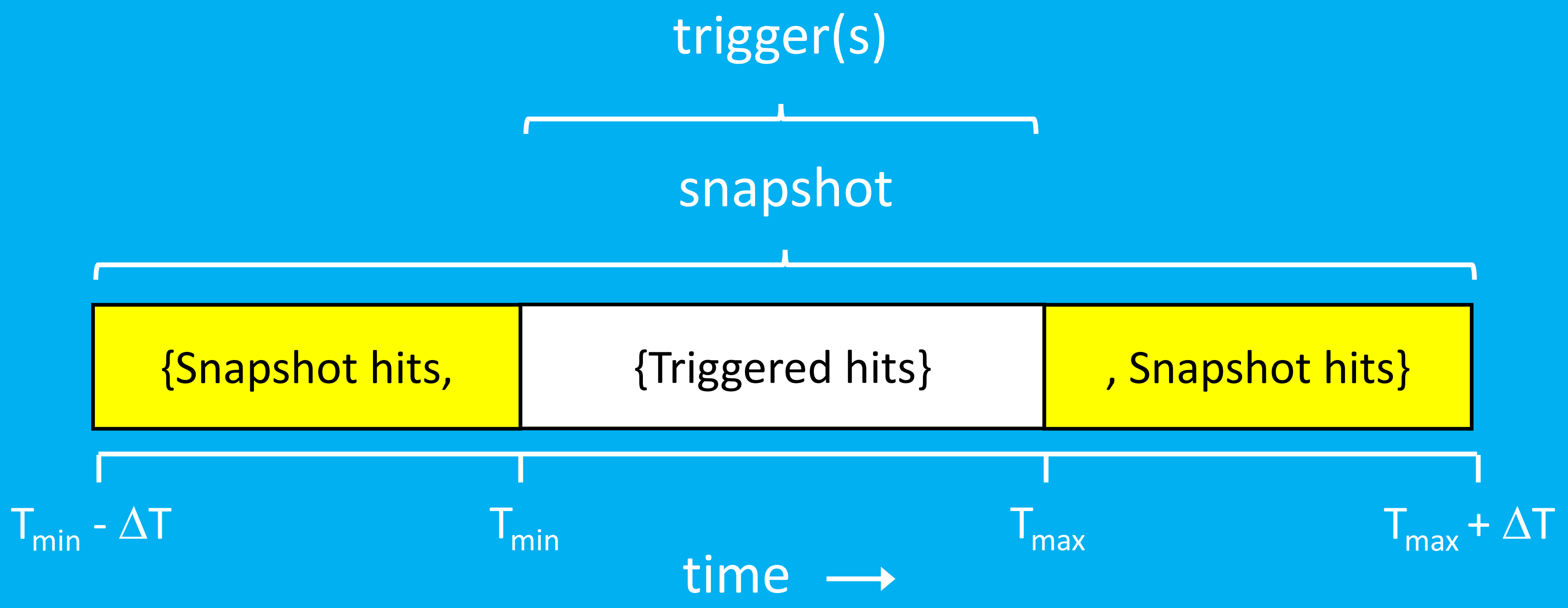
Event

=

Minimal set of causally related hits

a.k.a. triggered hits

# Event data




snapshot¶ = <u>All</u> raw data between $[T_{min} - \Delta T, T_{max} + \Delta T]$

¶ $\Delta T$ = $nD/c$ (where D corresponds to size of detector)

# Reconstruction (1a/3)

Arrival time¶ $$t_j \;\cong\; t_0 + c\left(\Delta z + \frac{R}{\tan\theta_C}\right)$$ muon

$$t_j \;\cong\; t_0 + c(nD)$$ EM-shower

→ problem is non-linear

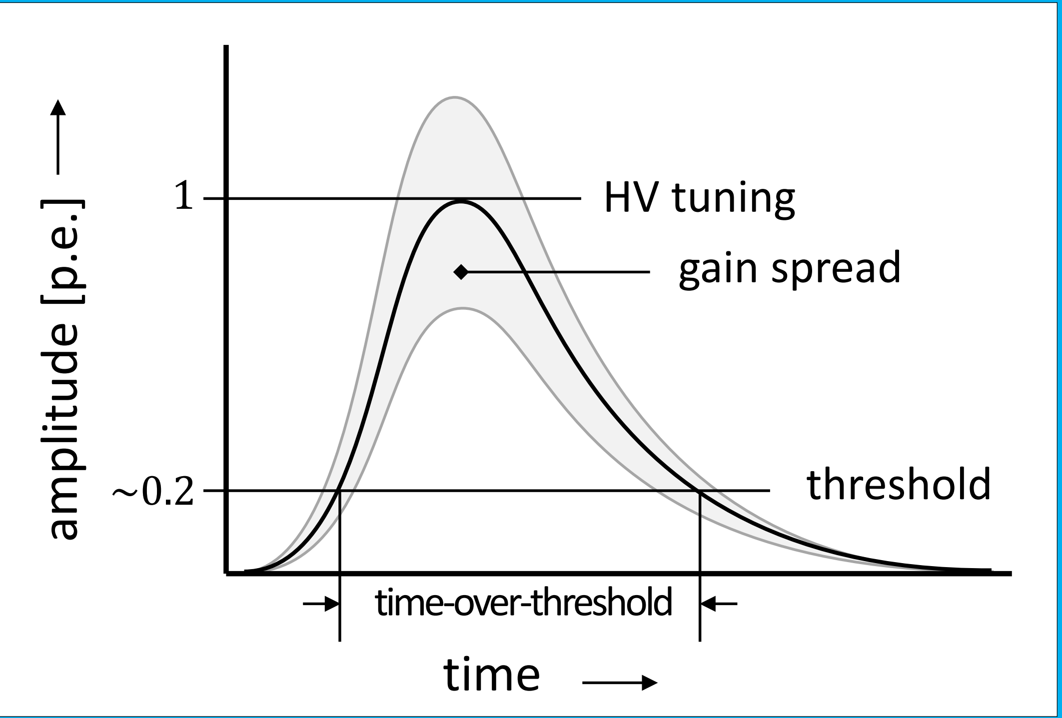
✓ in absence of light scattering, problem can partially be linearized when pairs of hits are considered (e.g. pairs of L1-hits)

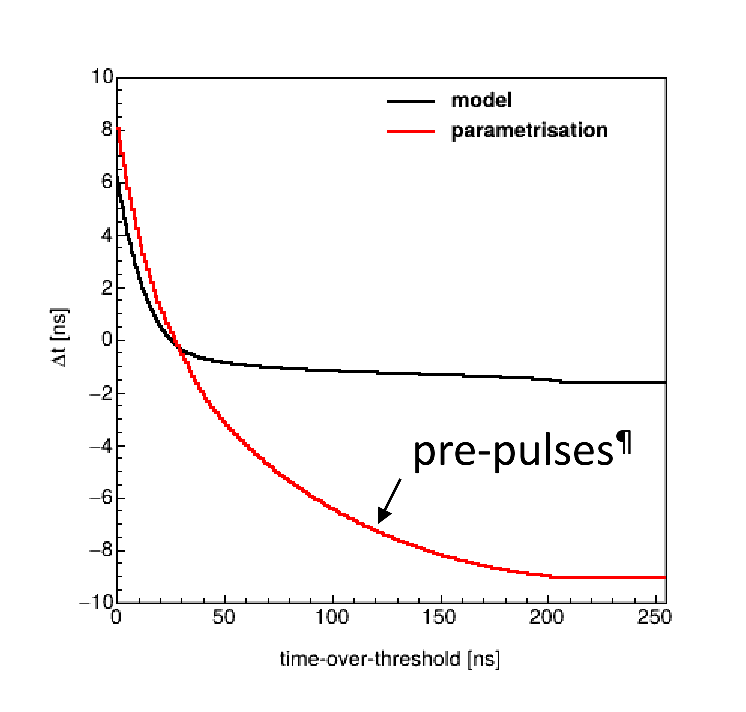
- $(x_0, y_0, t_0)$ of muon in a given direction (see documentation)
- $(x_0, y_0, z_0, t_0)$ of EM-shower (see documentation)

¶ Arrival time ≡ shortest optical path.

# Reconstruction (1b/3)

PMT analogue pulse




¶ See slide 11.

# Reconstruction (2/3)

1. calibration
   - DAQ hit → calibrated time (and time-over-threshold)
   - module identifier and PMT → PMT position and orientation
2. determination of start values of model parameters (a.k.a. prefit)
   - linear fit(s) of (simplified) model to biased selection of data
3. determination of intermediate values of model parameters
   - M-estimator fit of (simplified) model to biased selection of data
4. determination of final values of model parameters
   - maximum likelihood fit using PDF of model to unbiased selection of data¶

¶ Selection of hits should be unbiased with respect to PDF.

# Reconstruction (3/3)

- Model parameters
  - orthogonalisation (following rotation of primary axis to z-axis)
    - muon trajectory $\left(x_0, y_0, t_0, T_x, T_y [, E \;[, B_y]]\right)$ → next slides
    - shower $\left(x_0, y_0, z_0, t_0, T_x, T_y [, E \;[, B_y]]\right)$ → Víctor; J. Seneca, T. van Eden
- PDF
  - contains as-much-as-possible knowledge of the detector response to particles
    - Cherenkov light production
    - light absorption, scattering and dispersion
    - PMT QE, angular acceptance and TTS
    - electronics (discriminator, TDC, etc.)
    - background rates, high-rate veto, etc.

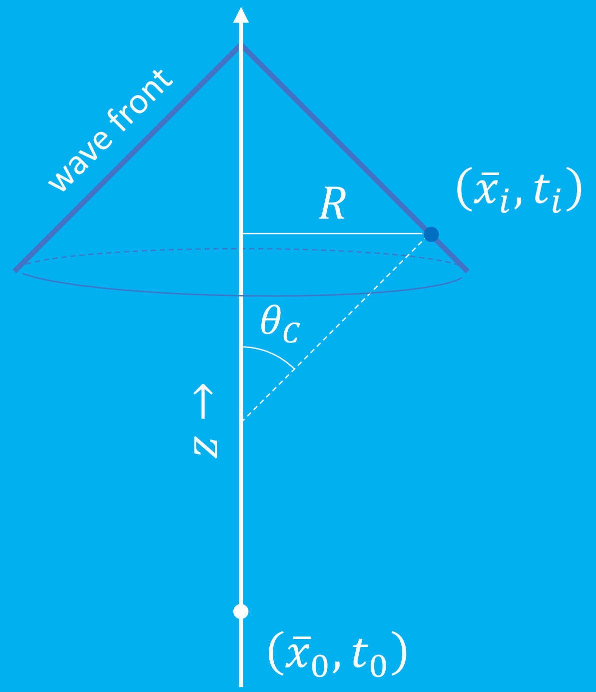
# Start value problem

$$5D = \underset{\text{position-time}}{3D} \otimes \underset{\text{direction}}{2D}$$

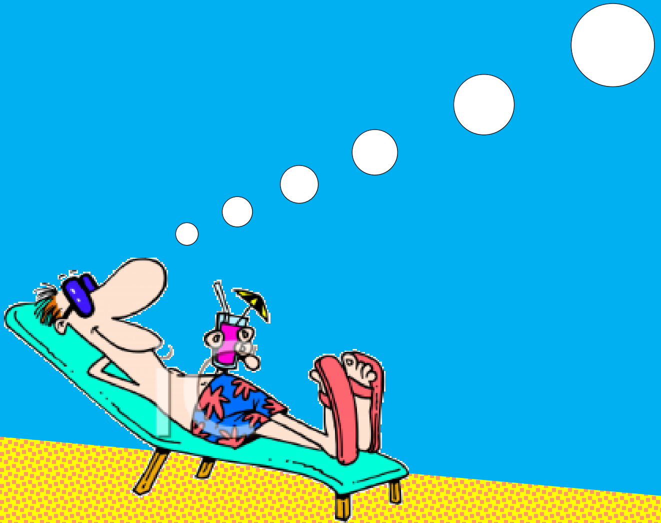
linear fit $(\boldsymbol{x_0}, \boldsymbol{y_0}, \boldsymbol{t_0})$ scan $(\theta, \varphi)$

$$\bar{t}_j^2 - \bar{t}_i^2 - 2(\bar{t}_j - \bar{t}_i)\,\boldsymbol{\bar{t}_0} \;=\; x_j^2 - x_i^2 - (x_j - x_i)\,\boldsymbol{x_0} \;+\; y_j^2 - y_i^2 - (y_j - y_i)\,\boldsymbol{y_0}$$

$$\bar{t}_i \equiv \frac{ct_i}{\tan\theta_C} - \frac{z_i}{\tan\theta_C} \qquad \bar{t}_0 \equiv \frac{ct_0}{\tan\theta_C}$$




$$ct_i \;\cong\; ct_0 + \Delta z + R\tan\theta_C$$



# Linear fit (1a/4)

$$y = H\,\theta$$

$$(n \times 1) \qquad\qquad (n \times k)\ (k \times 1)$$

# Linear fit (1b/4)

$$\underset{(k\times 1)}{\hat{\theta}} = (\underset{(k\times n)}{H^T}\,\underset{(n\times n)}{V^{-1}}\,\underset{(n\times k)}{H})^{-1}\underset{(k\times n)}{H^T}\,\underset{(n\times n)}{V^{-1}}\,\underset{(n\times 1)}{y}$$

$$V = \begin{pmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_n^2 \end{pmatrix} \Leftarrow V_{ij} \mathrel{+}= \delta\alpha^2 \frac{\partial y_i}{\partial \alpha} \frac{\partial y_j}{\partial \alpha}$$

# Linear fit (2/4)

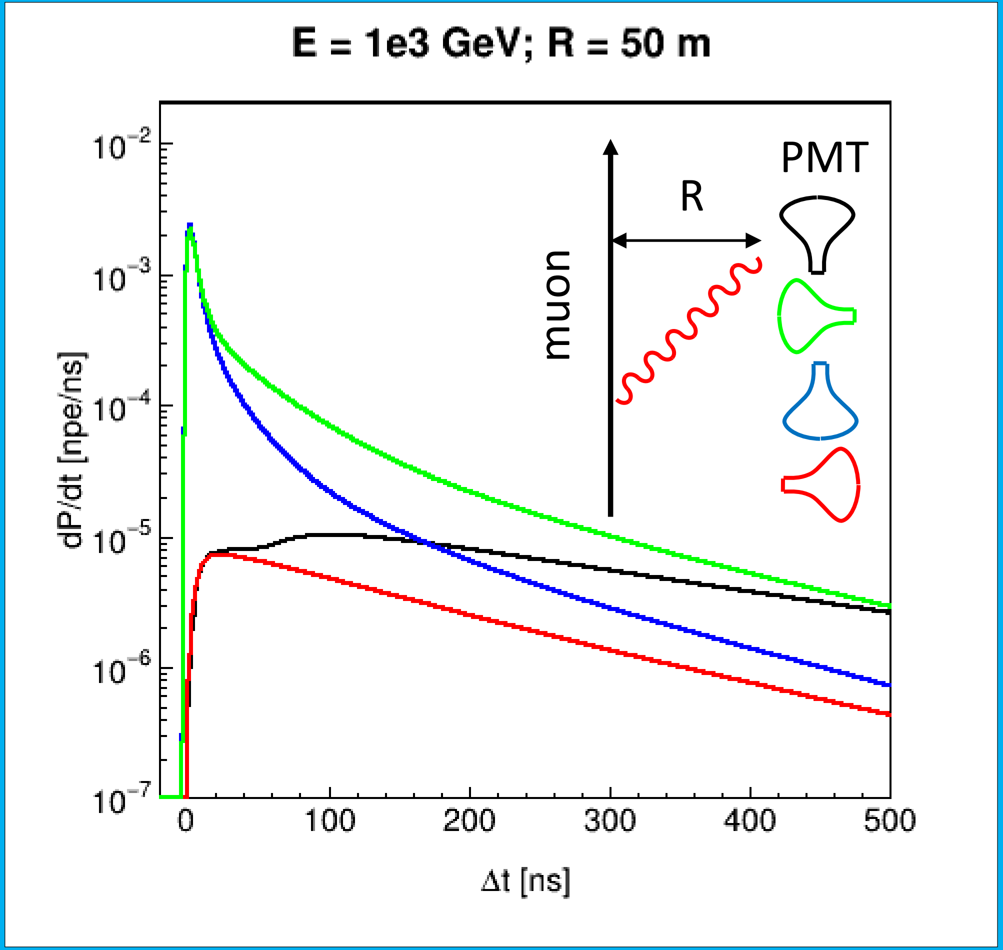
```
JOmega3D omega (parameters.gridAngle_deg * JMATH::PI/180.0);


for (JOmega3D_t::const_iterator dir = omega.begin(); dir != omega.end(); ++dir) {

      const JRotation3D R(*dir);

      for (buffer_type::iterator i = data.begin(); i != data.end(); ++i) {
            i->rotate(R);
      }

      buffer_type::iterator __end = clusterizeWeight(data.begin(), data.end(), match1D);

      …
}
```

# Linear fit (3/4)

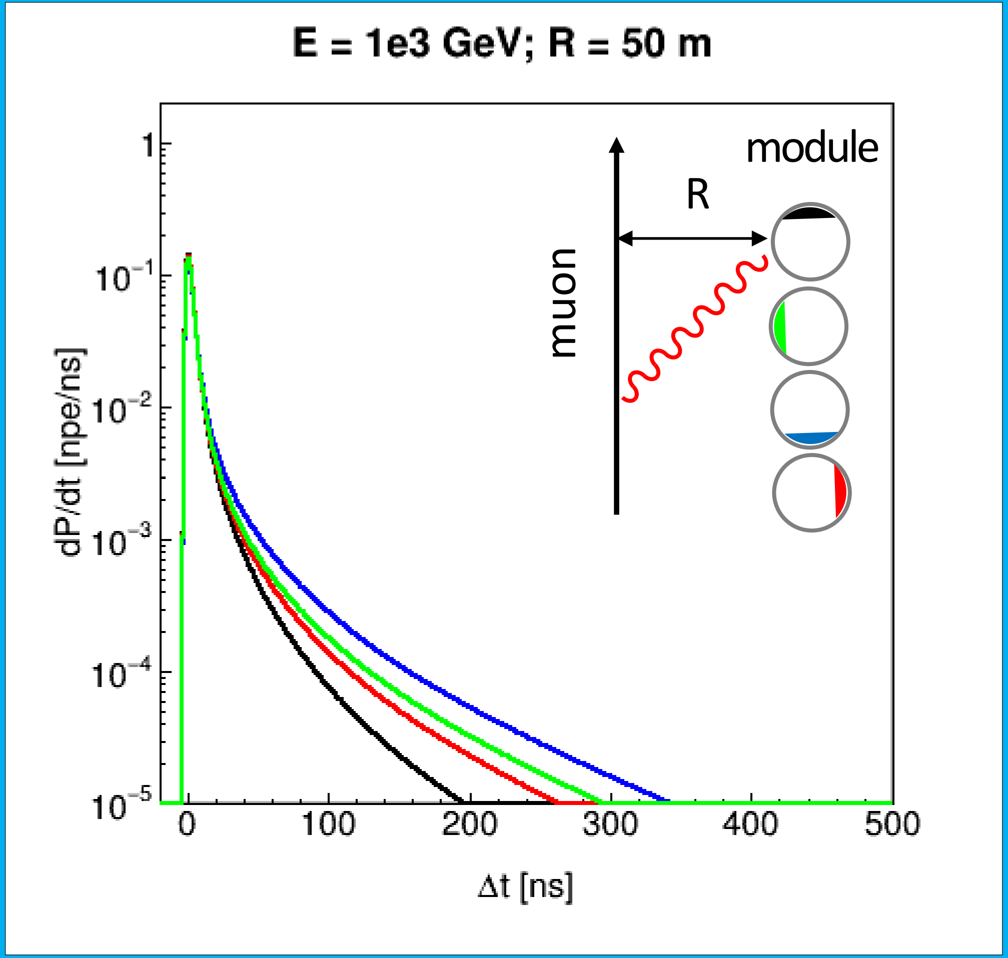
```
JMatrixNZ   V;
JVectorNZ   Y;

for (JOmega3D_t::const_iterator dir = omega.begin(); dir != omega.end(); ++dir) {

    …

    JEstimator<JLine1Z> tz(data.begin(), __end);

    V.set(tz, data.begin(), __end, gridAngle_deg, sigma_ns);
    Y.set(tz, data.begin(), __end);

    V.invert();

    double y = getChi2(Y, V);

    …
}
```

# Linear fit (4a/4)

**L0**

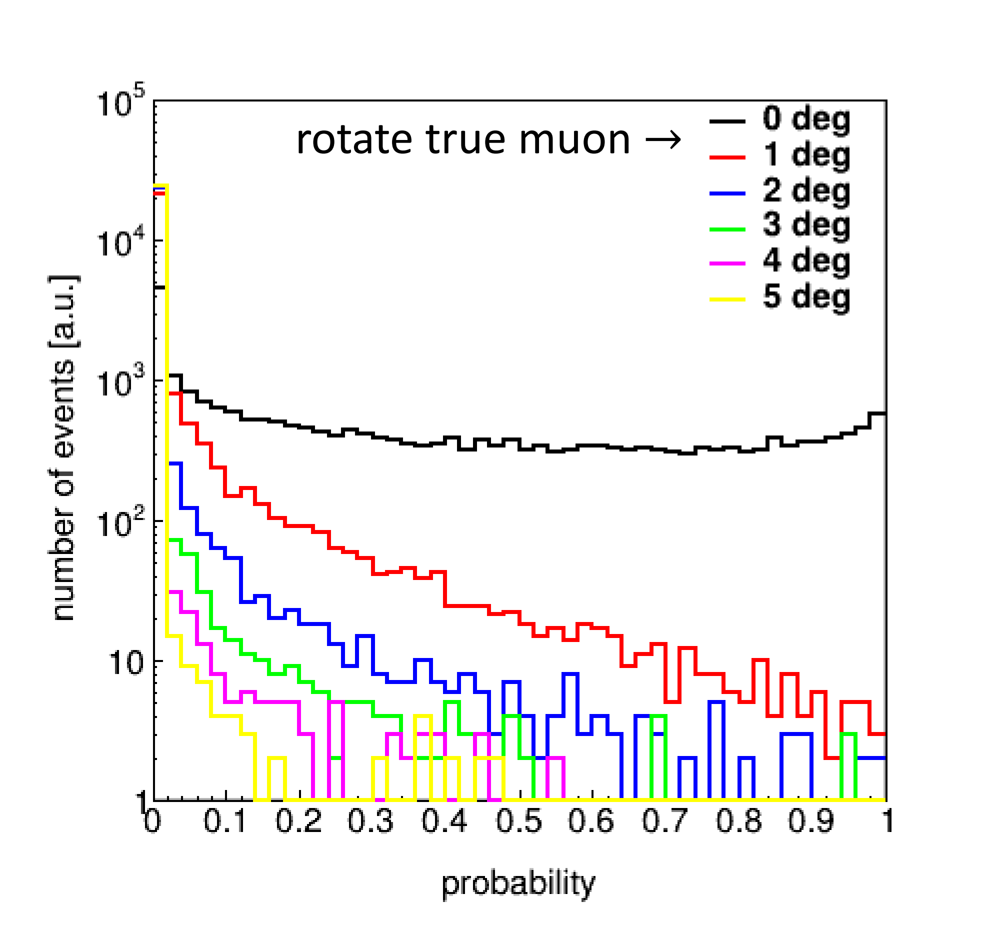



**L1**

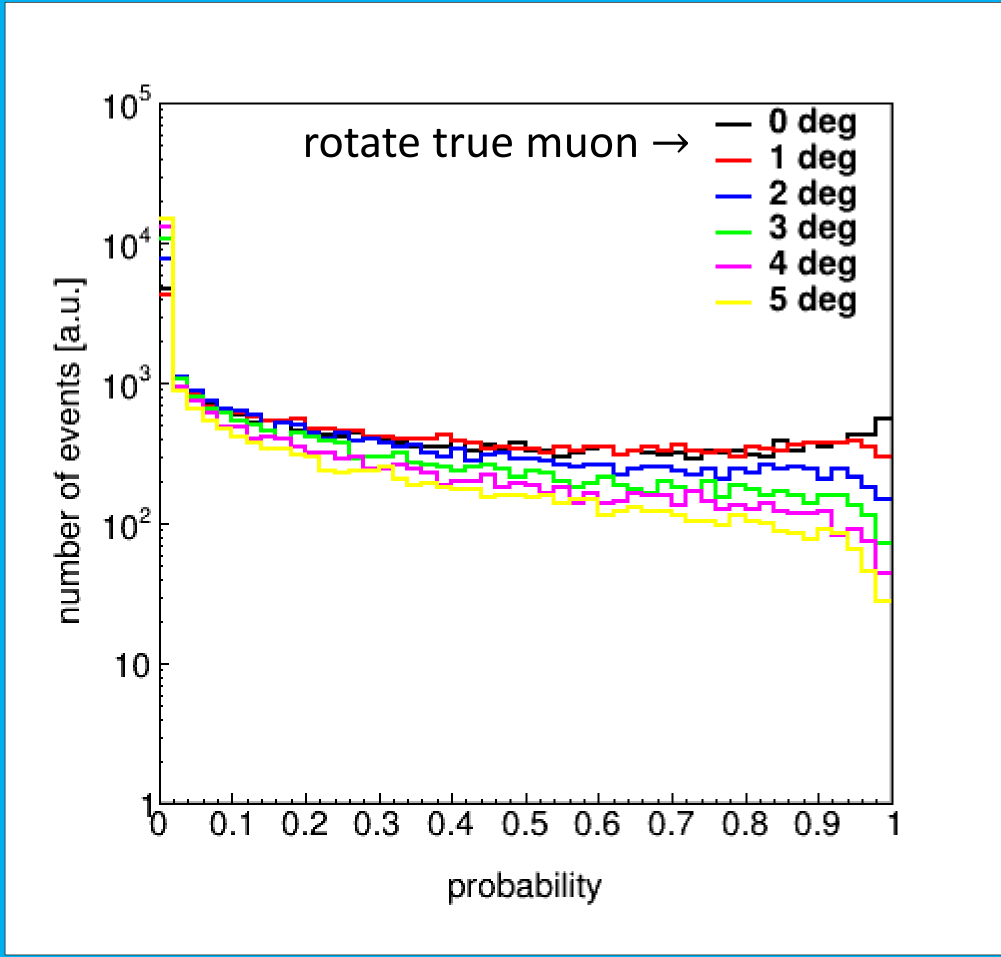


# Linear fit (4b/4)

**w/o co-variances**




**with co-variances**



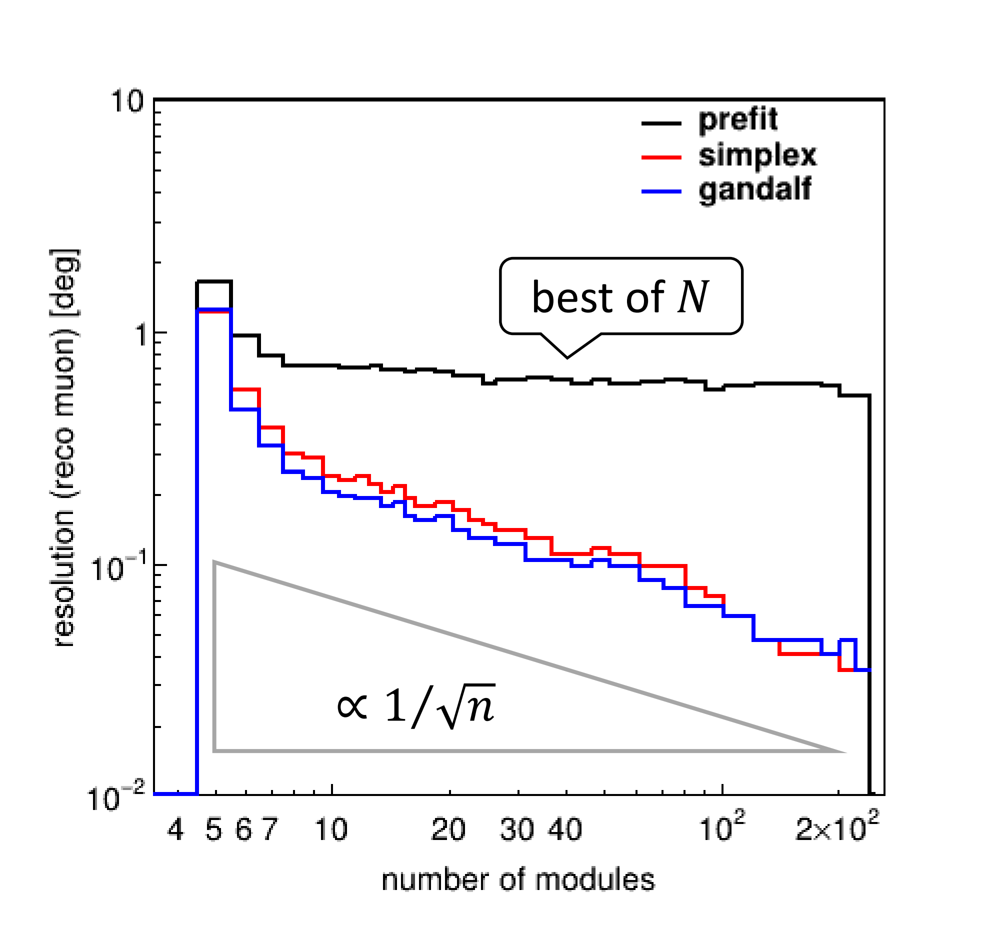
# Final fit (1/3)

```
for (JEvt::const_iterator track = in.begin(); track != in.end(); ++track) {

    const JRotation3D   R (getDirection (*track));
    const JLine1Z       tz(getPosition  (*track).rotate(R), track->getT());

    const JModel<JLine1Z> match(tz, roadWidth_m, JRegressor_t::T_ns, Z_m);

    for (buffer_type::const_iterator i = data.begin(); i != data.end(); ++i) {

        JHitW0 hit(*i, summary.getRate(i->getPMTIdentifier()));

        hit.rotate(R);

        if (match(hit)) {
            buffer.push_back(hit);
        }
    }
    …
}
```

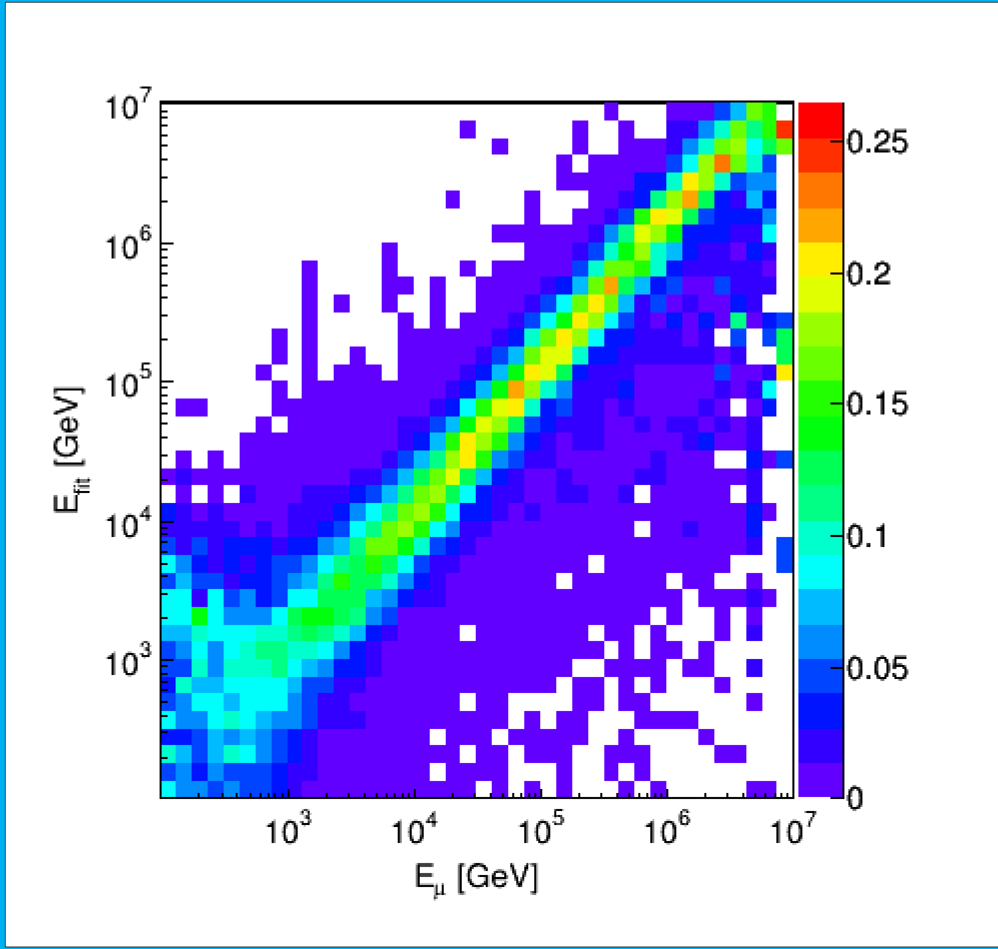
# Final fit (2/3)

```
JRegressor<JLine3Z, JGandalf> regressor(pdf_file, parameters.TTS_ns);

for (JEvt::const_iterator track = in.begin(); track != in.end(); ++track) {
      …
      double chi2 = regressor(tz, buffer.begin(), buffer.end());
      JTrack3D tb(regressor.value);
      tb.rotate_back(R);
      …
}
```

# Final fit (3/3)



prefit
simplex
gandalf
best of $N$
$\propto 1/\sqrt{n}$
resolution (reco muon) [deg]
number of modules

$E_{fit}$ [GeV]
$E_{\mu}$ [GeV]

# Food for thoughts

- How well describes the model our data?
- $T_x$ and $T_y$ are orthogonal parameters ($\Omega \leq 2\pi$)
- ORCA: implement simultaneous fit of muon track + vertex shower?
- ARCA: implement explicitly stochastic energy losses of muon?
- Machine learning?