R&D of the Energy Calibration for the SiD EM Calorimeter based on Machine Learning

2019/11/28

M. Iwasaki\textsuperscript{ABCD}, Y. Naka\textsuperscript{A}, J. Brau\textsuperscript{E}, M. Breidenbach\textsuperscript{F}, K. Morikawa\textsuperscript{A}, H. Nagahara\textsuperscript{D}, Y. Nakashima\textsuperscript{D}, A. L. Steinhebel\textsuperscript{E}, J. F. Strube\textsuperscript{G}, N. Takemura\textsuperscript{D}

Osaka-City U.\textsuperscript{A}, NITEPB, Osaka U. RCNP\textsuperscript{C}, Osaka U. IDS\textsuperscript{D}, U. Oregon\textsuperscript{E}, SLAC\textsuperscript{F}, PNNL\textsuperscript{G}
We have developed the energy calibration method using **machine learning** for the ILC SiD EM Calorimeter (ECAL).

Machine Learning based on the Neural Network (NN) is expected to improve calibration performance over the simple reconstruction method.

In this talk, we report the status of the R&D.
SiD EM Calorimeter (ECAL)

- 30 Layer Si + W sampling calorimeter
- \( \sim 26X_0 \) in total
- Energy resolution (design value) \( (17/\sqrt{E} + 1)\% \)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inner radius of ECL barrel</td>
<td>1.27 m</td>
</tr>
<tr>
<td>Maximum z of barrel</td>
<td>1.76 m</td>
</tr>
<tr>
<td>Longitudinal profile</td>
<td>20 layers ( \times 0.64 X_0 )</td>
</tr>
<tr>
<td></td>
<td>10 layers ( \times 1.30 X_0 )</td>
</tr>
<tr>
<td>EM energy resolution (design value)</td>
<td>( (17/\sqrt{E} + 1)% )</td>
</tr>
<tr>
<td>Readout gap</td>
<td>1.25 mm (or less)</td>
</tr>
<tr>
<td>Effective Molire radius (R)</td>
<td>14 mm</td>
</tr>
</tbody>
</table>

Hexagon Si sensors

1024 pixels per Si sensor
Pixel size = 13mm\(^2\)

ILC TDR, vol.4, Page 89  
[arXiv:1306.6329][physics.ins-det]
Energy Calibration of the Calorimeter

In this study, we try the SiD EM calorimeter (ECAL) energy calibration using Deep Neural Network

Simple Energy Reconstruction “Simple Recon”

\[ E = \alpha \times (\sum E_{i \leq 20} + 2 \sum E_{i > 20}) \]

Since last 10 layers have double thick Tungsten
Energy Calibration of the Calorimeter

In this study, we try the SiD EM calorimeter (ECAL) energy calibration using Deep Neural Network

Several problems on the “Simple Recon” energy calibration

1. Nonlinear detector response
   (due to shower leakage, detector geometry etc..)

2. Different detector response for e and γ (particle-species dependence)
Energy Calibration of the Calorimeter

In this study, we try the SiD EM calorimeter (ECAL) energy calibration using **Deep Neural Network**.

Several problems on the “Simple Recon” energy calibration:

1. **Nonlinear** detector response
   (due to shower leakage, detector geometry etc.)
2. Different detector response for **e** and **γ** (**particle-species** dependence)

![Energy Resolution Graph]

- **Energy Resolution**
- **ΔE(%) × √E**
- **Particle Energy(GeV)**

- Simple Recon

<table>
<thead>
<tr>
<th>Particle</th>
<th>Energy Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Photon</td>
<td></td>
</tr>
<tr>
<td>Electron</td>
<td></td>
</tr>
</tbody>
</table>
Energy Calibration of the Calorimeter

In this study, we try the SiD EM calorimeter (ECAL) energy calibration using **Deep Neural Network**

Several problems on the “Simple Recon” energy calibration

3. **Angular dependence** due to the detector geometry

![Energy Resolution in \( \phi \)-bin](image)

\[ \Delta E(\%) \times \sqrt{E} \]

- E=50GeV

- Simple Recon

- photon

- electron
Energy Calibration using DNN

Neural Network can express the non-linear response

We use DNN regression to obtain the cluster energy

ECAL Data

- Sum of E hit, r, φ, ...

DNN Input

DNN output (Cluster Energy)
Input parameters for DNN

Energy sum in a layer

Distance btw IP and 1st hit

# of hits in a cluster

Δφ btw 1st and last hit

E = 2, 5, 10, 20, 50, 100 GeV
**Results: Energy calibration with DNN**

*Preliminary*

**Photon Energy Resolution** $\Delta E(\%) \times \sqrt{E}$

- Using DNN, we get better resolution for both photon and electron in wide energy region.
Results: Energy calibration with DNN

**Preliminary**

Using DNN, we get better resolution for both photon and electron. There still exist angular dependence.
What kind of data do we input?

There are several reports that **Low-level feature data** (pre-processing data) provides better DNN performance than **High-level feature data** (physics-inspired engineered).

- **Low-level data**: Hit data (position, Energy)
- **Middle-level data**: Layer data (CM position, Sum of E hit in layer)
- **High-level data**: Cluster data (CM position, Sum of E hit, ...)

In this study, we also try **Middle-level data** for the DNN input.
### Results: E calib. with Middle-level data

**Preliminary**

<table>
<thead>
<tr>
<th>Photon Energy Resolution $\Delta E(%) \times \sqrt{E}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Graph showing photon energy resolution]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Electron Energy Resolution $\Delta E(%) \times \sqrt{E}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Graph showing electron energy resolution]</td>
</tr>
</tbody>
</table>

Using DNN + Middle-level data we get better resolution for both photon and electron.
Results: E calib. with Middle-level data

Preliminary

Photon Energy Resolution in $\phi$-bin

Electron Energy Resolution in $\phi$-bin

- E calibration with DNN + Middle-level data gives better Energy resolution for photon.
- Little difference btw Middle-level and High-level data for electron.
- There still exists angular dependence.
As Input parameters for DNN, we also try
Middle-level + High-level data

- Low-level data
  - Hit data (position, Energy)
- Middle-level data
  - Layer data (CM position, Sum of E hit in layer)
- High-level data
  - Cluster data (CM position, Sum of E hit, ...)
E calib. with Middle + High level data

Preliminary

Photon Energy Resolution $\Delta E(\%) \times \sqrt{E}$

- We construct E calibration with DNN using both Middle and High-level data  →  Still under optimization
E calib. with Middle + High level data

Preliminary

Photon Energy Resolution in $\phi$-bin

Electron Energy Resolution in $\phi$-bin

- We construct E calibration with DNN using both Middle and High-level data $\rightarrow$ Still under optimization
- We still have $\phi$-dependence

Further studies are on going

DNN optimization, DNN with low-level data??
Summary

We have developed the Energy Calibration using DNN for SiD ECAL

- We get better Energy resolution using DNN
- As input parameters for the DNN, we try High-level data, Middle-level data, and High + Middle-level data
  \[\rightarrow\text{Middle-level data gives better performance}\]
- We still have angular-dependence

Further studies are on going
DNN optimization, DNN with low-level data
SiD ECAL

Figure II-4.1
Overall mechanical layout of the ECAL.

ILC TDR, vol.4,
Page 89

arXiv:1306.6329
[physics.ins-det]
Figure II-4.2
Drawing of a silicon sensor for the ECAL.
The sensors are segmented into 1024 13 mm$^2$ pixels.

Table II-4.2
Parameters of baseline silicon-tungsten ECAL and the MAPS option.

<table>
<thead>
<tr>
<th></th>
<th>Baseline</th>
<th>MAPS option</th>
</tr>
</thead>
<tbody>
<tr>
<td>pixel size</td>
<td>13 mm$^2$</td>
<td>50× 50 μm similar</td>
</tr>
<tr>
<td>readout gap</td>
<td>1.25 mm</td>
<td></td>
</tr>
<tr>
<td>(incl. 0.32 mm thick Si sensors)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>effective Molière radius</td>
<td>14 mm</td>
<td>14 mm</td>
</tr>
<tr>
<td>pixels per silicon sensor</td>
<td>1024</td>
<td>1 × 10$^6$</td>
</tr>
<tr>
<td>channels per KPiX chip</td>
<td>1024</td>
<td>-</td>
</tr>
<tr>
<td>dynamic range</td>
<td>~0.1 to 2500 MIPs</td>
<td>1 MIP</td>
</tr>
<tr>
<td>heat load</td>
<td>20 mW per sensor</td>
<td>20 mW per sensor</td>
</tr>
</tbody>
</table>
Sampling

Design based on sampling 30 times:
20 layers of 0.64 $X_0$
followed by 10 layers of 1.30 $X_0$

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>inner radius of ECAL barrel</td>
<td>1.27 m</td>
</tr>
<tr>
<td>maximum z of barrel</td>
<td>1.76 m</td>
</tr>
</tbody>
</table>
| longitudinal profile                          | 20 layers $\times$ 0.64 $X_0$
                                            | 10 layers $\times$ 1.30 $X_0$ |
| EM energy resolution                          | $0.17/\sqrt{E} \oplus 1\%$ |
| readout gap                                   | 1.25 mm (or less)      |
| effective Molière radius ($R$)                | 14 mm                  |

Table II-4.1
Nominal parameters of the silicon-tungsten ECAL for SiD.

1024 channels read out by single KPiX chip
NN with Middle-level data

For more precise energy calibration, we try energy calibration using NN with Middle-level data

Low-level data  Hit data (x, y, z, Energy, layer)
Middle-level data  Layer data (CM x,y,z, Energy sum for a layer)
High-level data  Cluster data (CM r,φ, Energy sum, .. for a cluster)
### ECAL calibration with NN: Parameters

Keras + TensorFlow

<table>
<thead>
<tr>
<th>Input</th>
<th>High-level data</th>
<th>Middle-level data</th>
<th>High + Middle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activation</td>
<td></td>
<td>Relu</td>
<td></td>
</tr>
<tr>
<td>Optimization</td>
<td></td>
<td>Adam</td>
<td></td>
</tr>
<tr>
<td>Loss</td>
<td></td>
<td>Mean Squared Error</td>
<td></td>
</tr>
<tr>
<td># hidden layers</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>#epoch</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>Training</td>
<td>400,000</td>
<td>300,000</td>
<td>300,000</td>
</tr>
<tr>
<td>Test</td>
<td>100,000</td>
<td>100,000</td>
<td>100,000</td>
</tr>
<tr>
<td>Validation</td>
<td>100,000</td>
<td>100,000</td>
<td>100,000</td>
</tr>
</tbody>
</table>
Training results for NN

High-level data (Cluster data)
Keras+TensorFlow

Input params.: Sum of E_hit in a cluster, CM r, φ for cluster, ... 7 params
Activation: Relu
Optimization: Adam
Loss: Mean Squared Error
# of hidden layers: 3 layers
# of Epoch: 30
Training data: 400,000 events
Test data: 100,000 events
Val. data: 100,000 events
Training results for NN

Middle-level data (Layer data)

Keras+TensorFlow

Input params. : Sum of E_hit in a layer, CM x,y,z for layer
Activation : Relu
Optimization : Adam
Loss : Mean Squared Error

# of hidden layers : 3 layers
# of Epoch : 30
Training data : 300,000 events
Test data : 100,000 events
Val. data : 100,000 events

LOSS (Mean-Squared-Error)
Training results for NN

High-level + Middle-level data

Keras+TensorFlow

Input params. : High-level + Middle-level data
Activation : Relu
Optimization : Adam
Loss : Mean Squared Error
# of hidden layers : 4 layers
# of Epoch : 30
Training data : 300,000 events
Test data : 100,000 events
Val. data : 100,000 events