NeuLAND calibration revisit: Millepede algorithm and error analysis of map2cal calibration

Yanzhao Wang[1]

[1] University of Cologne, Institute for Nuclear Physics, Germany

NeuLAND weekly meeting
05.04.2024

Email: *ywang@ikp.uni-koeln.de*

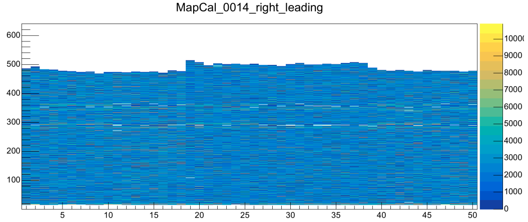
Overview

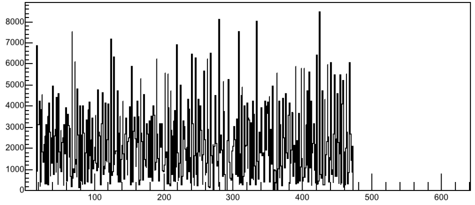
- Statistical error analysis of TDC calibration
- Millepede algorithm principle
- Millepede algorithm application to NeuLAND
- Preliminary results

NeuLAND TDC calibration

TDC calibration relation:

$f(x)$: Channel number $\Longrightarrow$ Time value (ns)





Y projection bin 11

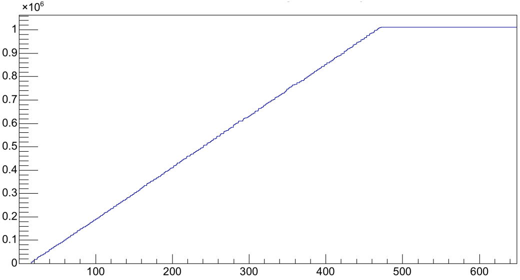
Two time components:

- coarse time ($\pm$ 5 ns)
- fine time ($\pm$ 10 ps)

Fine time Calibration procedures:

1. Collection of of all TDC values
2. Calculation of cumulative distribution of TDC values
3. Scaling of CFD to $0 \sim 5$ ns

CDF:



Convolution of two distributions:

- Uniform distribution: fine time $\sim \mathcal{U}(f(n), f(n+1))$
- Multinomial distribution: $f(n), f(n+1) \sim \mathcal{M}_k(3, p_a, p_b),\ \ p_a = \mathrm{CDF}(n), p_b = \mathrm{PDF}(n)$

Statistical error analysis of TDC calibration

Convolution of two distributions:

- Uniform distribution: fine time $\sim \mathcal{U}(f(n), f(n+1))$
- Multinomial distribution: $f(n), f(n+1) \sim \mathcal{M}_k(3, p_a, p_b),\ \ p_a = \mathrm{CDF}(n), p_b = \mathrm{PDF}(n)$

Mathematical derivation:

Mean:

$$E(X) = \sum_{n_a, n_b} \mathcal{M}(n_a, n_b; 3, p_a, p_b) \cdot \int_{n_a}^{n_b} x f(x)\, dx = \bar{n_a} + \bar{n_b}/2$$

Variance:

$$\begin{aligned}
Var(X) &= E(X^2) - E^2(X) \\
E(X^2) &= \sum_{n_a, n_b} \mathcal{M}(n_a, n_b; 3, p_a, p_b) \cdot \int_{n_a}^{n_b} x^2 f(x)\, dx = \sum_{n_a, n_b} \mathcal{M}(n_a, n_b; 3, p_a, p_b)(n_a^2 n_b + n_a n_b^2 + n_b^2/3) \\
&= Var(N_a) + E^2(N_a) + Cov(N_a, N_b) + E(N_a)E(N_b) + \left(Var(N_a) + E^2(N_b)\right)/3 \\
Var(X) &= N^2 p_b^2/12 + N\left(-(p_a - \frac{1-p_b}{2})^2 + 1/4 - p_b/6 - p_b^2/12\right)
\end{aligned}$$

Validation of the error analysis

Exact solution (no approximation):

$$\delta = \sqrt{\frac{N^2 p_b^2}{12} + N\left(-(p_a - \tfrac{1-p_b}{2})^2 + \tfrac{1}{4} - \tfrac{p_b}{6} - \tfrac{p_b^2}{12}\right)}$$

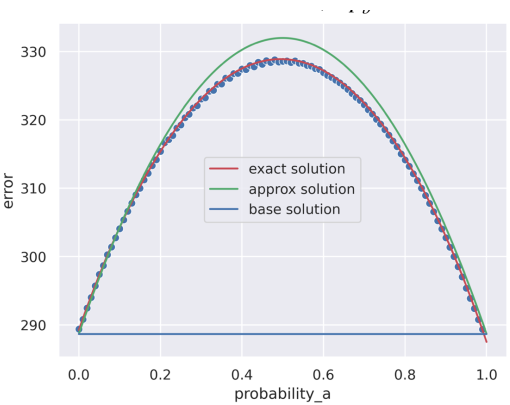
Approximate solution:

$$\delta = \frac{Np_b}{\sqrt{12}} + \frac{\sqrt{3}}{p_b}(-(p_a - \tfrac{1}{2})^2 + \tfrac{1}{4}) \quad (p_b \neq 0)$$

Base solution:

$$\delta = \frac{Np_b}{\sqrt{12}}$$

Comparison to Monte-Carlo data:



$p_b : 0.01$

$N : 100'000$

NeuLAND time and position calibration

Time relation:

$$t = \frac{t_r + t_l}{2} - \frac{L}{2 \cdot C_e} + t_{\text{sync}}$$

Position relation:

$$x = \frac{C_e}{2} \left(t_r - t_l + t_{\text{offset}} \right)$$

Calibration parameters:

- C_e : effective speed of light
- t_{sync} : time synchronization among scintillators
- t_{offset} : time offset between adjacent PMTs

NeuLAND time and position calibration

Time relation:

$$t = \frac{t_r + t_l}{2} - \frac{L}{2 \cdot C_e} + t_{\text{sync}}$$

Position relation:

$$x = \frac{C_e}{2}\left(t_r - t_l + t_{\text{offset}}\right)$$

Calibration parameters:

- C_e : effective speed of light
- t_{sync} : time synchronization among scintillators
- t_{offset} : time offset between adjacent PMTs

Calibration using muon tracks:

Time and position relation for muon tracks:

$$x_\mu = a_x^i \cdot z_\mu + b_x^i \qquad (1)$$

$$y_\mu = a_y^i \cdot z_\mu + b_y^i \qquad (2)$$

$$t_\mu = a_t^i \cdot z_\mu + b_t^i \qquad (3)$$

Calibration parameters for the jth muon track in ith bar:

$$C_e^i,\ t_{\text{sync}}^i,\ t_{\text{offset}}^i,\ a_x^j,\ a_y^j,\ a_t^j,\ b_x^j,\ b_y^j,\ b_t^j$$

Total number of calibration parameters for n muon tracks: $3900 + 6n$

Principle of Millepede algorithm

Linear regression

Residual minimization

$$\partial \sum_{j=1}^{n} \sum_{i=1}^{b(j)} \frac{(\mathcal{Z}_i^j(g_1, ..., g_m, p_1^j, ..., p_l^j))^2}{2(\sigma_i^j)^2} = 0$$

$\mathcal{Z}_i^j$: Residual values (linear) for jth track and ith bar

$g_{1...m}$: m *global parameters*

$p_{1...l}^j$: l *local parameters* for the jth μ track

n : the total number of μ tracks

$b(j)$: the total number of bars for the jth μ track

Linear regression

Residual minimization

$$\partial \sum_{j=1}^{n} \sum_{i=1}^{b(j)} \frac{(\mathcal{Z}_i^j(g_1, \dots, g_m, p_1^j, \dots, p_l^j))^2}{2(\sigma_i^j)^2} = 0$$

$\mathcal{Z}_i^j$: Residual values (linear) for jth track and ith bar

$g_{1\dots m}$: m *global parameters*

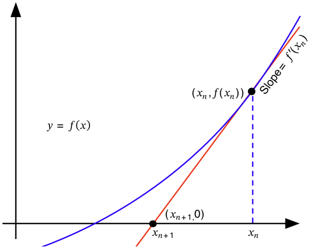
$p_{1\dots l}^j$: l *local parameters* for the jth μ track

n : the total number of μ tracks

$b(j)$: the total number of bars for the jth μ track

Newton's method

- *Single variable:*



$$\delta x = x_{n+1} - x_n$$
$$= -f(x_n)/f'(x_n)$$

- *Multivariable:*

$$\nabla f(\vec{x}_n) \cdot \delta \vec{x} = -f(\vec{x}_n)$$

Residual minimization:

$$\nabla \vec{f}(\vec{x}_n) = \nabla(\nabla \mathcal{F}(g_1, \dots, g_m, p_1^j, \dots, p_l^j))$$
$$= \mathrm{H}\mathcal{F}(g_1, \dots, g_m, p_1^j, \dots, p_l^j))$$

Hessian operation (second derivative matrix)

$$[\mathrm{H}\mathcal{F}(x_1, x_2, \cdots, x_n)]_{ij} = \frac{\partial^2 \mathcal{F}}{\partial x_i \partial x_j} \sim \sum \frac{\partial \mathcal{Z}}{\partial x_i} \frac{\partial \mathcal{Z}}{\partial x_j}$$

Using Newton's method:

$$[\mathrm{H}\mathcal{F}(\boldsymbol{x})]\, \Delta \boldsymbol{x} = -\nabla \mathcal{F}(\boldsymbol{x})$$

$$\color{green}{\leftarrow m \rightarrow}\ \color{red}{\leftarrow \sim n \cdot l \rightarrow}$$

$$\left[\begin{array}{c||c:c:c} \sum_j \mathcal{C}_j & \ldots & \mathcal{G}_j & \ldots \\ \hline\hline \vdots & \ddots & 0 & 0 \\ \hdashline \mathcal{G}_j^T & 0 & \Gamma_j & 0 \\ \hdashline \vdots & 0 & 0 & \ddots \end{array}\right] \cdot \left[\begin{array}{c} \color{red}{\Delta \mathbf{g}} \\ \hline\hline \vdots \\ \hdashline \Delta \mathbf{p}^j \\ \hdashline \vdots \end{array}\right] = - \left[\begin{array}{c} \sum_j \partial_{\mathbf{g}} \mathcal{F}_j \\ \hline\hline \vdots \\ \hdashline \partial_{\mathbf{p}^j} \mathcal{F} \\ \hdashline \vdots \end{array}\right]$$

Principle of Millepede algorithm

Hessian operation (second derivative matrix)

$$[\mathrm{H}\mathcal{F}(x_1, x_2, \cdots, x_n)]_{ij} = \frac{\partial^2 \mathcal{F}}{\partial x_i \partial x_j} \sim \sum \frac{\partial \mathcal{Z}}{\partial x_i} \frac{\partial \mathcal{Z}}{\partial x_j}$$

Using Newton's method:

$$[\mathrm{H}\mathcal{F}(\mathbf{x})]\, \Delta\mathbf{x} = -\nabla \mathcal{F}(\mathbf{x})$$

$$\begin{array}{cc} \overset{\leftarrow\; m \;\rightarrow}{} & \overset{\leftarrow\; \sim n \cdot l \;\rightarrow}{} \end{array}$$

$$\left[\begin{array}{c||c:c:c} \sum_j \mathcal{C}_j & \ldots & \mathcal{G}_j & \ldots \\ \hline\hline \vdots & \ddots & 0 & 0 \\ \hdashline \mathcal{G}_j^T & 0 & \Gamma_j & 0 \\ \hdashline \vdots & 0 & 0 & \ddots \end{array}\right] \cdot \left[\begin{array}{c} \Delta\mathbf{g} \\ \hline\hline \vdots \\ \hdashline \Delta\mathbf{p}^j \\ \hdashline \vdots \end{array}\right] = - \left[\begin{array}{c} \sum_j \partial_{\mathbf{g}} \mathcal{F}_j \\ \hline\hline \vdots \\ \hdashline \partial_{\mathbf{p}^j} \mathcal{F} \\ \hdashline \vdots \end{array}\right]$$

Schur complement method:

$$\tilde{\mathcal{C}} \cdot \Delta\mathbf{g} = \mathcal{D}$$

where

$$\tilde{\mathcal{C}} = \sum_j \mathcal{C}_j + \sum_j \left(-\mathcal{G}_j \Gamma_j^{-1} \mathcal{G}_j^T\right)$$

$$\mathcal{D} = \sum_j \partial_{\mathbf{g}} \mathcal{F}_j - \sum_j \mathcal{G}_j \Gamma_j^{-1} \partial_{\mathbf{p}^j} \mathcal{F}$$

Advantages

- Simultaneous fitting of all parameters
- Computation complexity independent of local parameter size
- No muon track reconstruction

Residual equations for NeuLAND

For horizontal bars

$$0 = b_x + C_e^i t_{\text{offset}}^i/2 + C_e^i \Delta t/2 + a_x z_i \tag{1}$$

$$t_{\text{sum}}/2 = b_t + \frac{L}{2C_e^i} + t_{\text{sync}}^i + a_t z_i \tag{2}$$

$$y_i = b_y + a_y z_i \tag{3}$$

For vertical bars

$$0 = b_y + C_e^i t_{\text{offset}}^i/2 + C_e^i \Delta t/2 + a_y z_i \tag{4}$$

$$t_{\text{sum}}/2 = b_t + \frac{L}{2C_e^i} + t_{\text{sync}}^i + a_t z_i \tag{5}$$

$$y_i = b_x + a_x z_i \tag{6}$$

Residual equations for NeuLAND

For horizontal bars

$$0 = b_x + C_e^i t_{\text{offset}}^i/2 + C_e^i \Delta t/2 + a_x z_i \quad (1)$$

$$t_{\text{sum}}/2 = b_t + \frac{L}{2C_e^i} + t_{\text{sync}}^i + a_t z_i \quad (2)$$

$$y_i = b_y + a_y z_i \quad (3)$$

For vertical bars

$$0 = b_y + C_e^i t_{\text{offset}}^i/2 + C_e^i \Delta t/2 + a_y z_i \quad (4)$$

$$t_{\text{sum}}/2 = b_t + \frac{L}{2C_e^i} + t_{\text{sync}}^i + a_t z_i \quad (5)$$

$$y_i = b_x + a_x z_i \quad (6)$$

Linearization

- Parameter redefinition $C_t^i \equiv C_e^i t_{\text{offset}}^i$ in Eq. (1, 4):

$$\begin{aligned} 0 &= b_x + C_e^i t_{\text{offset}}^i/2 + C_e^i t_{\text{diff}}/2 + a_x z_i \\ &= b_x + C_t^i/2 + C_e^i t_{\text{diff}}/2 + a_x z_i \end{aligned}$$

- Taylor expansion of C_e^i near C_{e0}^i in Eq. (2, 5):

$$t_{\text{sum}}/2 = b_t + \frac{L}{2C_e^i} + t_{\text{sync}}^i + a_t z_i$$

$$\sim b_t + L\left(\frac{1}{C_{e0}^i} - \frac{C_e^i}{2(C_{e0}^i)^2}\right) + t_{\text{sync}}^i + a_t z_i$$

First order derivatives of residual equations

Derivatives of all parameters for ith bar:

Eq.	a_x	a_y	a_t	b_x	b_y	b_t	$t^i_{\text{sync}}(\text{H})$	$t^i_{\text{sync}}(\text{V})$	$C^i_t(\text{H})$	$C^i_t(\text{V})$	$C^i_e(\text{H})$	$C^i_e(\text{V})$	meas	err
(1)	z_i	0	0	1	0	0	0	0	$\frac{1}{2}$	0	$\frac{t_{\text{diff}}}{2}$	0	0	$\frac{C^i_{e0}}{2}\delta t_{\text{diff}}$
(4)	0	z_i	0	0	1	0	0	0	0	$\frac{1}{2}$	0	$\frac{t_{\text{diff}}}{2}$	0	$\frac{C^i_{e0}}{2}\delta t_{\text{diff}}$
(2)	0	0	z_i	0	0	1	1	0	0	0	$-\frac{L}{2(C^i_{e0})^2}$	0	$\frac{t_{\text{sum}}}{2} - \frac{L}{C^i_{e0}}$	$\frac{1}{2}\delta t_{\text{sum}}$
(5)	0	0	z_i	0	0	1	0	1	0	0	0	$-\frac{L}{2(C^i_{e0})^2}$	$\frac{t_{\text{sum}}}{2} - \frac{L}{C^i_{e0}}$	$\frac{1}{2}\delta t_{\text{sum}}$
(3)	0	z_i	0	0	1	0	0	0	0	0	0	0	y_i	w_{bar}
(6)	z_i	0	0	1	0	0	0	0	0	0	0	0	x_i	w_{bar}

Variable names:

t_{sum} : Summation of the adjacent PMT times

t_{diff} : Difference of the adjacent PMT times

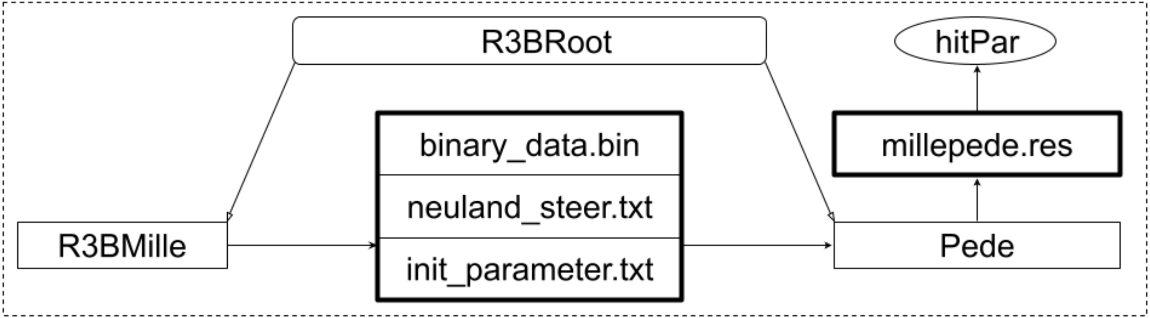
δt_{sum} : Error value of t_{sum}

δt_{diff} : Error value of t_{diff}

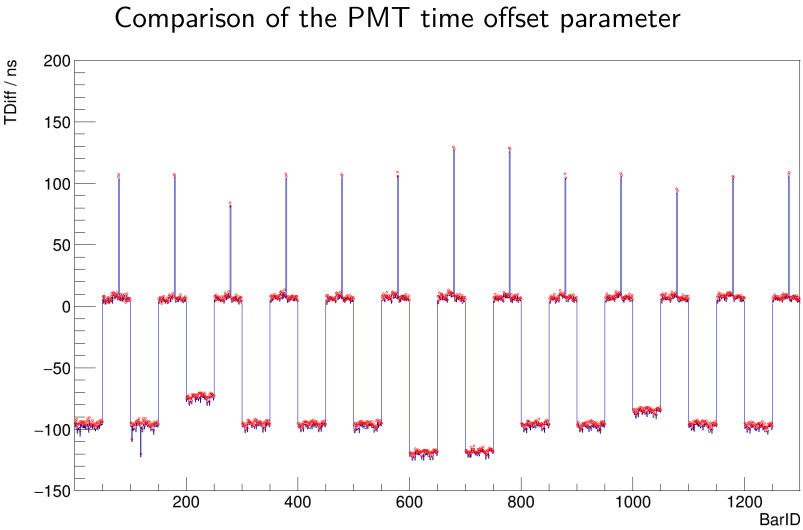
w_{bar} : Width of the ith bar

x_i, y_i, z_i : x, y, z positions of the ith bar

Integration with R3BRoot

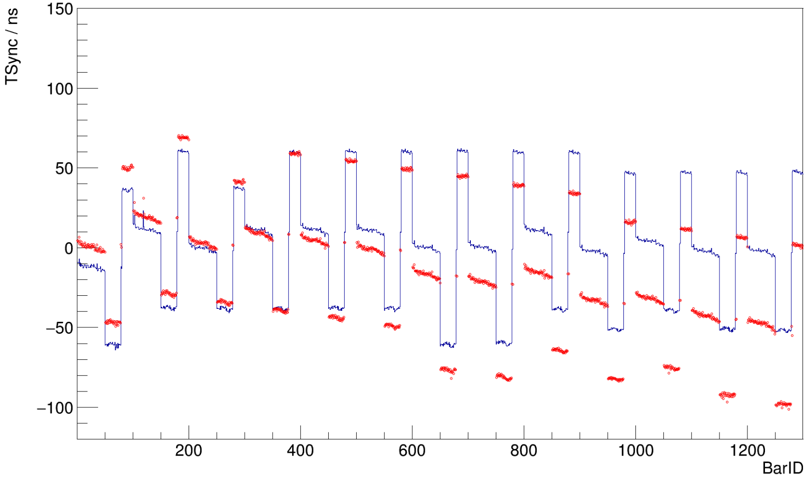


Preliminary results from Millepede algorithm



Preliminary results from Millepede algorithm

Comparison of the scintillator time synchronization



Comparison of the scintillator effective of c