



中国科学院高能物理研究所

Institute of High Energy Physics Chinese Academy of Sciences



散裂中子源科学中心

Spallation Neutron Source Science Center

白光中子源双束团解谱

Unfolding method for Double bunch on WNS

汇 报 人： 郭罗坤

导 师： 唐靖宇、易晗

日 期： 2026.07.14



目录

01 背景介绍

02 算法简介

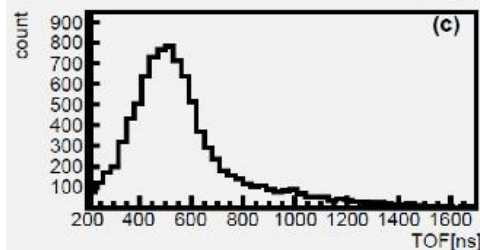
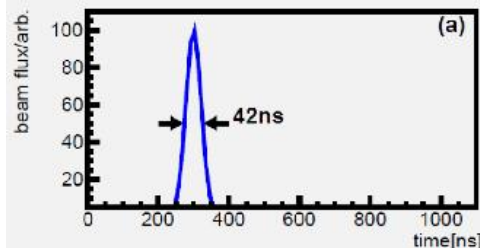
03 程序使用说明

04 注意事项

01 背景介绍

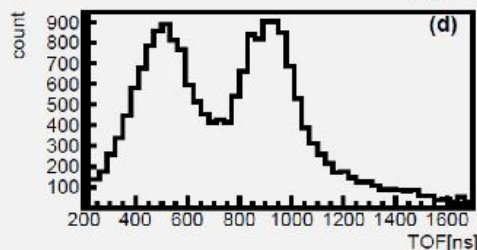
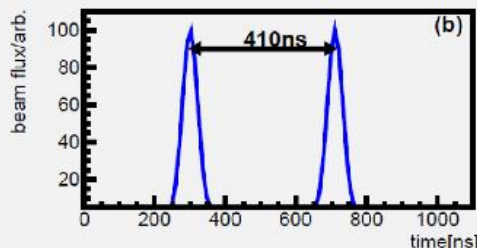
- 白光中子源双束团解谱**升级研究**
- 背景：CSNS加速器正常运行模式为双束团模式，两个束团之间的时间间隔为410ns，两个束团产生的中子TOF谱会进行叠加，对中子**飞行时间和能量测量**带来影响。

单束团质子束



单束团中子
TOF谱

双束团质子束



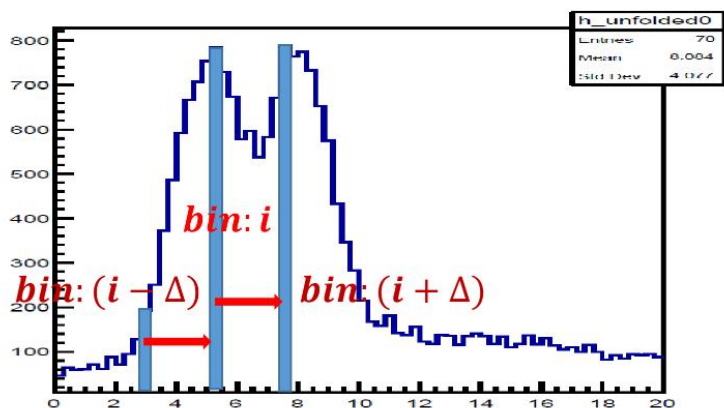
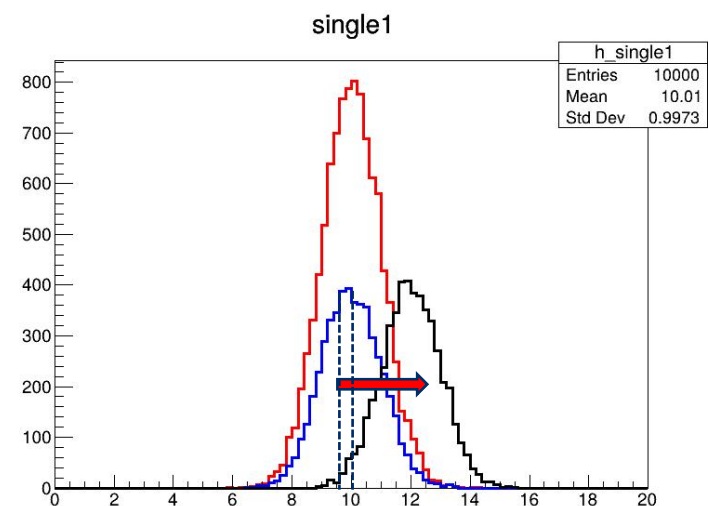
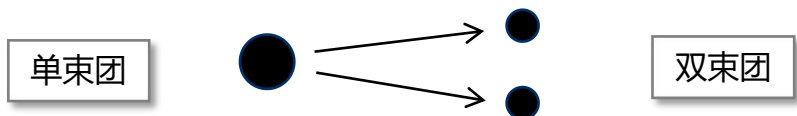
双束团中子
TOF谱

中子能量	飞行时间	能量分辨率(%)
1.0MeV	6.01	7.53
104.5keV	18.15	2.57
11.5keV	55.44	0.98
1.085keV	175.77	0.50
114.5eV	540.79	0.40
12.5eV	1664.83	0.43
1.275eV	5128.92	0.51

对高能中子能量测量带来严重影响

背景介绍

白光中子源双束团解谱升级研究



延迟时间: $Delay = 410ns$

道宽: w

延迟时间引起的延迟道数: $\Delta = Delay/w$

对于第 i 个bin的数据组成: 来自前第 $i - \Delta$ 道一半的数据; 第 i 道剩余的一半数据

$$E_i = 0.5 * C_i + 0.5 * C_{i-\Delta}$$

$$E_j = \sum_i R_{ji} C_i$$

$$E = RC$$

?

$$C = R^{-1}E$$

R_{ji} 是真实值 C_i 对测量值 E_j 的贡献, 同时也对应响应矩阵的矩阵元

- 白光中子源双束团解谱**升级研究**
- 基于贝叶斯条件概率原理，通过构造响应矩阵进行解谱

假设 C_i 是第 i 个bin的真实值， E_j 是第 j 个bin的测量值

$$P(C_i|E_j) = \frac{P(E_j|C_i)P(C_i)}{\sum_{k=1}^n P(E_j|C_k)P(C_k)}$$

$$C_i = \sum_{j=1}^n P(C_i|E_j) * E_j = \sum_{j=1}^n E_j * \frac{P(E_j|C_i)P(C_i)}{\sum_{k=1}^n P(E_j|C_k)P(C_k)}$$

$$C_i^{m+1} = \sum_{j=1}^n P(C_i|E_j) * E_j^m = \sum_{j=1}^n E_j^m * \frac{P(E_j|C_i)P(C_i)^m}{\sum_{k=1}^n P(E_j|C_k)P(C_k)^m}$$

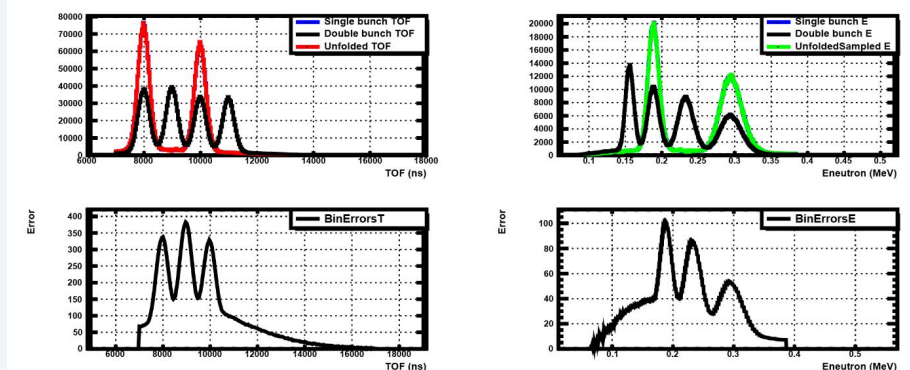
$$E = RC$$

$$R = \begin{bmatrix} 0.5 & & & & \\ \dots & & & & \\ 0.5 & & & & \\ 0.5 & \dots & R_{ji} & & \\ & 0.5 & \dots & & \\ & 0.5 & \dots & R_{jk} & \\ & & 0.5 & \dots & \\ & & & \dots & \\ & & & & 0.5 & \dots & 0.5 \end{bmatrix}$$

R: 响应矩阵

△是固定值，因此响应矩阵形成了一条主对角线和一条次对角线

解谱效果展示



具备功能:

- 线性TOF/能量解谱
- 二维解谱
- 全谱、局部谱

痛点: 不能解对数谱，解能谱需输入对应TOF谱，解谱需要进行代码操作易用性低

- 白光中子源双束团解谱**升级研究**
- 算法改进

$$C_i^{m+1} = \sum_{j=1}^n P(C_i|E_j) * E_j^m = \sum_{j=1}^n E_j^m * \frac{P(E_j|C_i)P(C_i)^m}{\sum_{k=1}^n P(E_j|C_k)P(C_k)^m}$$

解谱的关键在于得到 $P(E_j|C_i)$ 的值，即得到响应矩阵

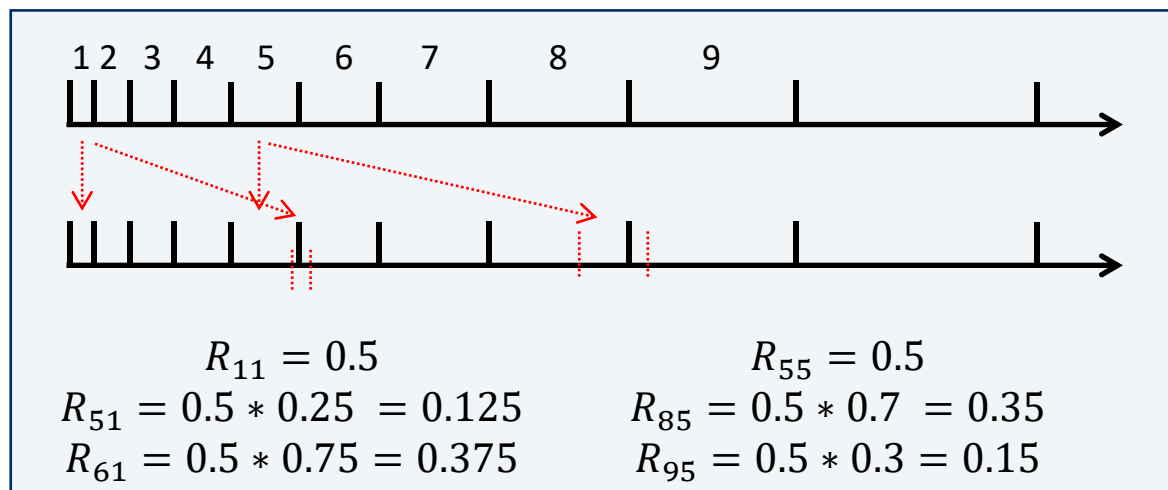
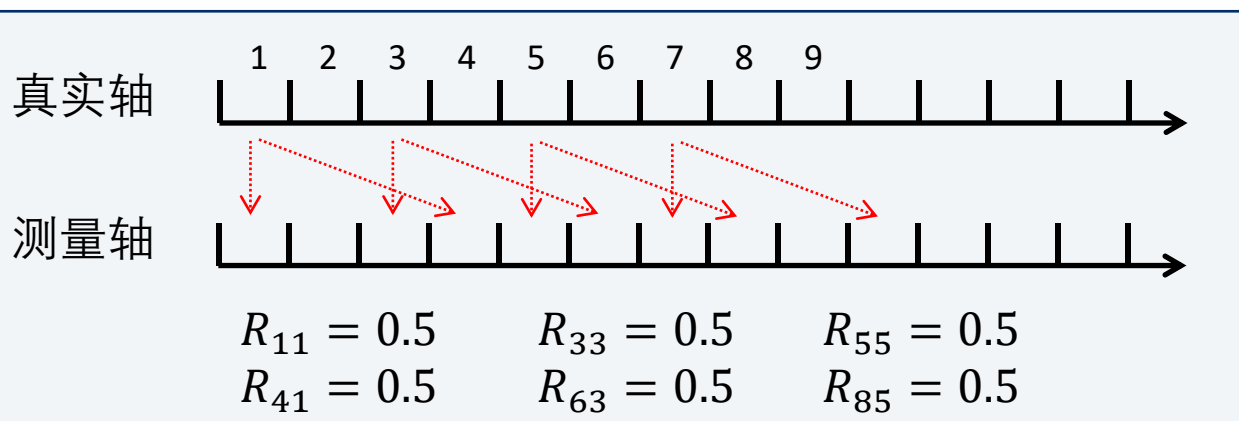
$$\begin{pmatrix} E_1 \\ E_2 \\ E_3 \\ \vdots \\ E_n \end{pmatrix} = \begin{pmatrix} 0.5 & & & & \\ & \ddots & & & \\ & & \ddots & & \\ 0.5 & & & 0.5 & \\ & & \ddots & & \ddots \\ & & & 0.5 & \dots & 0.5 \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \\ C_3 \\ \vdots \\ C_n \end{pmatrix}$$

线性谱的响应矩阵

?

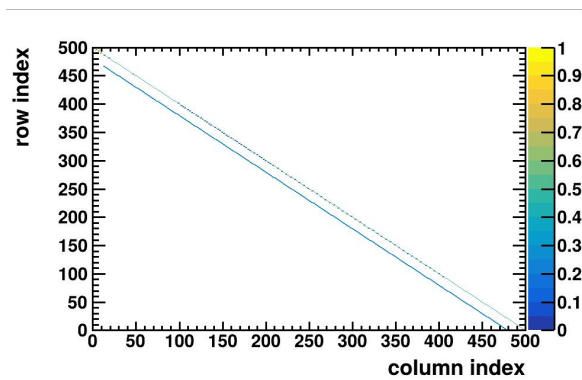
并不是两条对角线

对数谱的响应矩阵



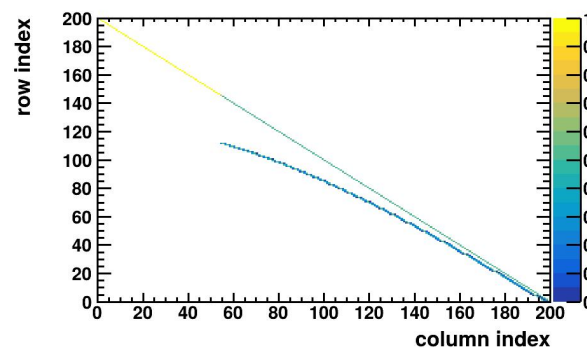
- 白光中子源双束团解谱**升级研究**
- 编写响应矩阵计算代码，得到了线性TOF/能量谱，对数TOF/能量谱响应矩阵，形状如下图所示

线性谱

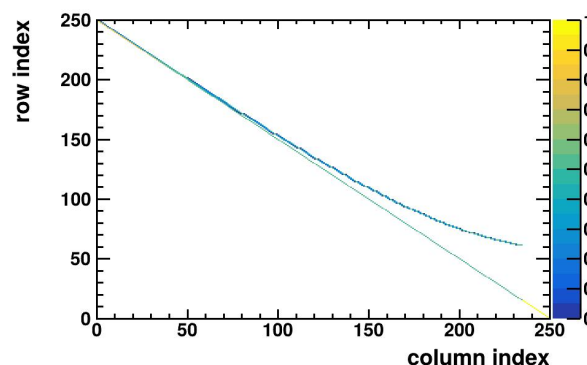
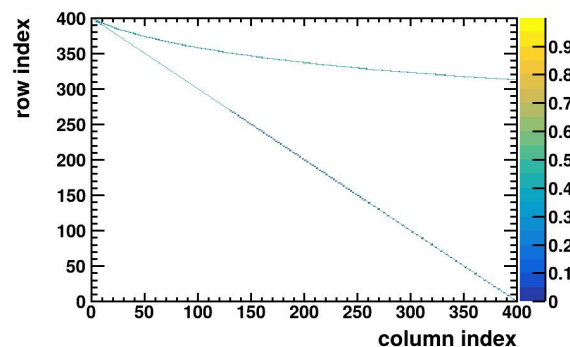


TOF

对数谱



能量



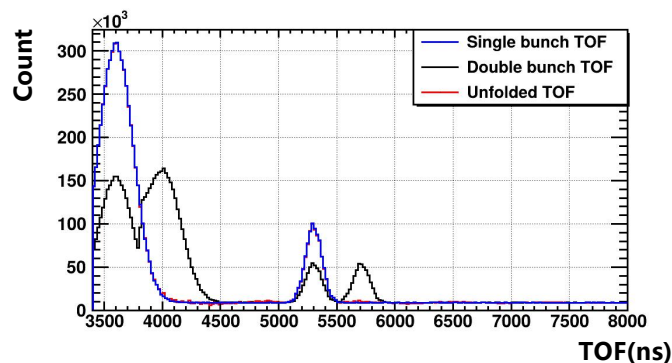
代码实现部分

```
DoubleBunchUnfolder::BuildResponseMatrixT();  
DoubleBunchUnfolder::BuildResponseMatrixE();
```

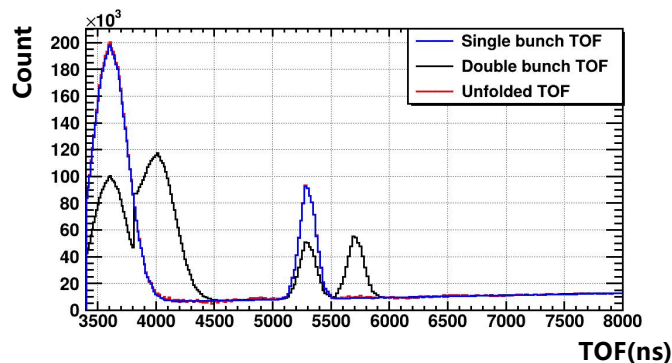
```
// --- case 1: 完全在 TerminalT 左边 ---  
if (high <= TerminalT) {  
    FillMatrixElementT(ResponseMatrix, i, low, high, 1.0);  
    continue;  
}  
// --- case 2: 完全在 TerminalT 右边 ---  
if (low >= TerminalT) {  
  
    // (1) 不偏移部分  
    FillMatrixElementT(ResponseMatrix, i, low, high, 1.0 - Rate);  
  
    // (2) 偏移部分  
    FillMatrixElementT(ResponseMatrix, i, low + Delay, high + Delay, Rate);  
  
    continue;  
}  
// --- case 3: bin 被 TerminalT 切开 ---  
double frac_left = (TerminalT - low) / width;  
double frac_right = (high - TerminalT) / width;  
  
// 左半: 完全不偏移  
FillMatrixElementT(ResponseMatrix, i, low, TerminalT, frac_left);  
  
// 右半: 部分偏移  
FillMatrixElementT(ResponseMatrix, i, TerminalT, high, frac_right * (1.0 - Rate));  
FillMatrixElementT(ResponseMatrix, i, TerminalT + Delay, high + Delay, frac_right * Rate);
```

结果展示

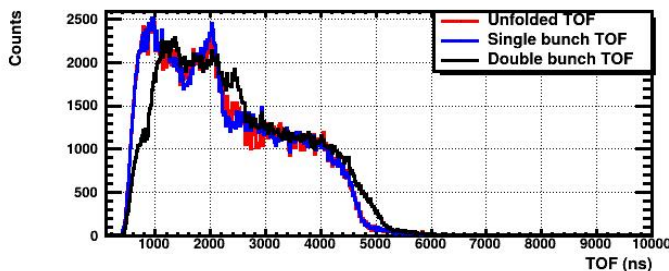
TOF谱解谱结果展示



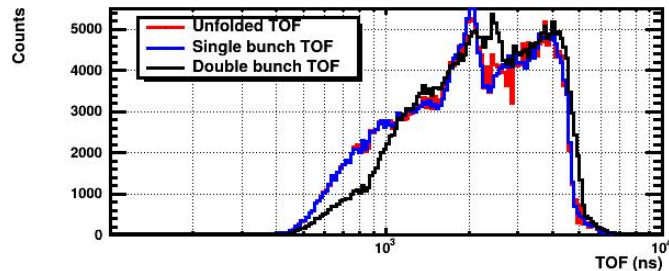
TOF线性谱模拟数据，迭代300次结果



TOF对数谱模拟数据，迭代300次结果

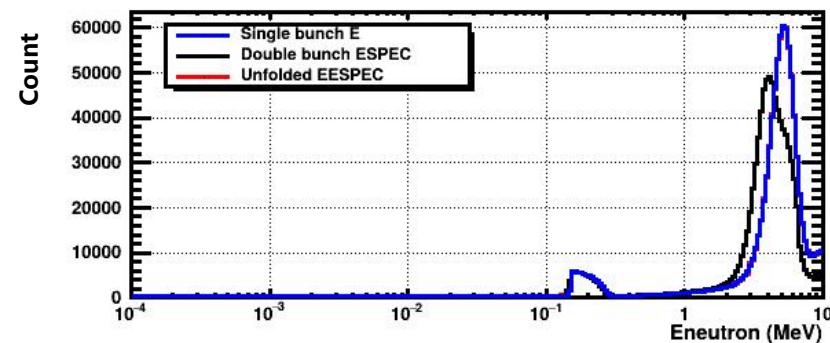


U238裂变截面TOF线性谱实验数据，迭代100次结果

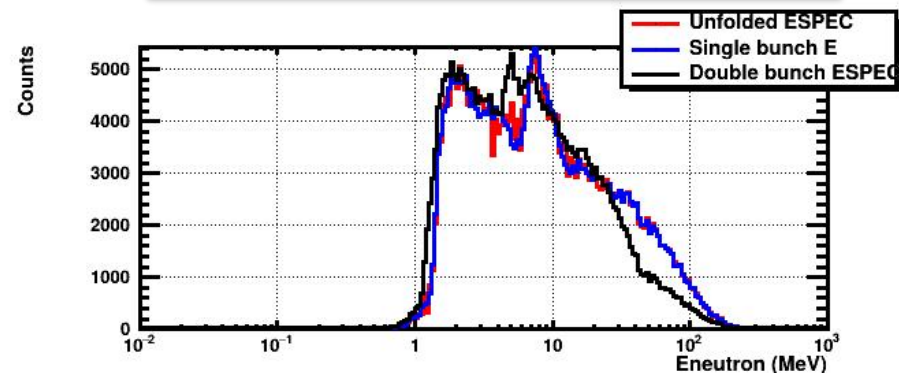


U238裂变截面TOF对数谱实验数据，迭代100次结果

能量谱解谱结果展示



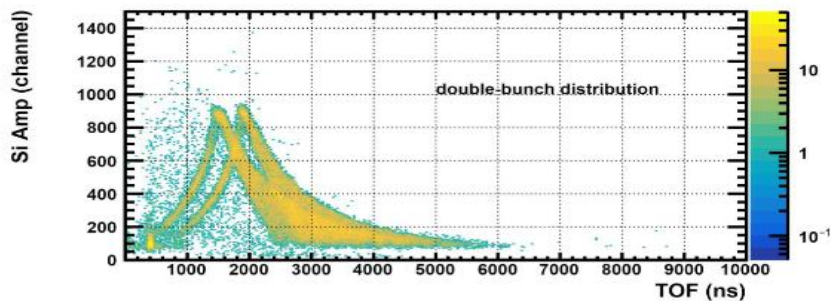
能量对数谱模拟数据，迭代300次结果



U238裂变截面能量对数谱实验数据，迭代100次结果

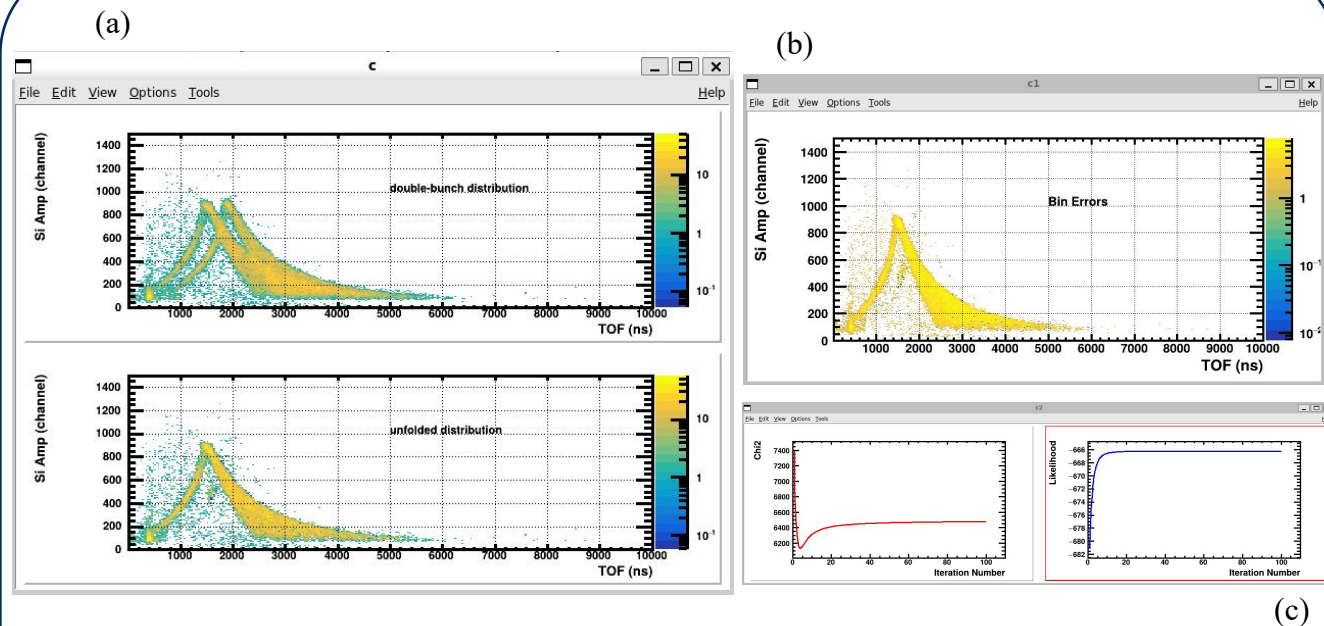
结果展示

- 二维直方图中，横向（TOF或者能量维度）受到双束团影响，纵向不受影响；
- 对每一行单独进行一维解谱



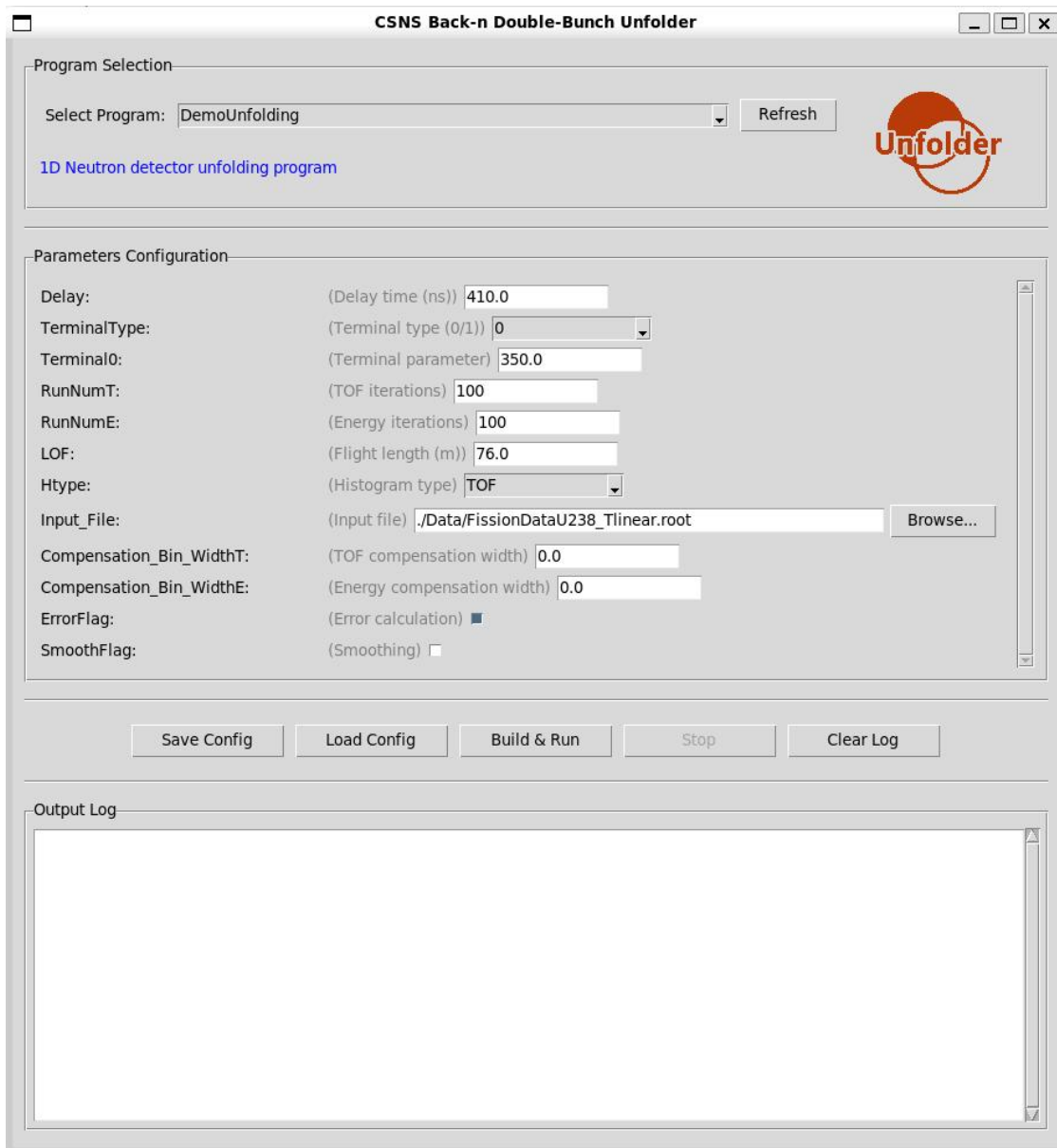
En1	En2	En3			Enn
.....					
E21	E22	E23			E2n
E11	E12	E13			E1n

能量谱解谱结果展示



特点：基于一维解谱程序开发，因此同样支持TOF/能量谱解谱，同时支持任意分Bin等功能

解谱程序运行演示



开发了通用解谱程序包 (C++)
开发了可视化交互界面 (Python)

具备功能:

- 线性TOF/能量谱, 对数TOF/能量谱
- 二维解谱
- 全谱、局部谱

参数说明

Delay: 延迟时间

TerminalType: 横轴起始点类型

Terminal0: 横轴测量起始点

RunNumT: 解谱迭代次数 (时间)

RumNumE: 解谱迭代次数 (能量)

LOF: 飞行距离

Htype: 解谱类型

Input_File: 待解谱文件

Compensation_Bin_WidthT: 补偿时间道宽

Compensation_Bin_WidthE: 补偿能量道宽

ErrorFlag: 误差计算开关

SmoothFlag: 平滑计算开关

save_config: 保存配置

load_config: 加载配置

Build&Run: 一键编译运行

Stop: 停止计算按钮

Clear_log: 清除日志

默认单位

时间: ns

能量: MeV

距离: m

03 程序使用说明

gitlab平台: <https://code.ihep.ac.cn/yih/doublebunchunfolder>

环境要求: gcc/g++, root, cmake, python, python3-tk

目前开发到V4.0版本, 具备功能:

- 线性TOF/能量谱, 对数 (自定义) TOF/能量谱
- 一维谱/二维谱
- 全谱、局部谱
- 可视化

DoubleBunchUnfolder

V4.0

doublebunchunfolder

+

历史

查找文件

编辑

代码

项目信息

60 次提交

1 个分支

0 个标签

131.1 MB 项目存储

自述文件

启用Auto DevOps

添加LICENSE

添加更新日志

添加贡献信息

添加 Kubernetes 集群

添加 Wiki

配置集成

创建于 March 19, 2026

界面参数名称优化

guoluokun@mail.lustc.edu.cn 编辑于 1天前

3b33d2e2

名称	最后提交	最后更新
Data	remove FissionDataU238_linear.root	1星期前
config	界面参数名称优化	1天前
include	代码修改	3星期前
source	代码修改	3星期前
.gitignore	update output	1星期前
CMakeLists.txt	update output	1星期前
DemoUnfolding.C	界面参数名称优化	1天前
DemoUnfolding2D.C	界面参数名称优化	1天前
README.md	add README.md	5天前
logo.png	update logo path	1星期前
program_gui.py	界面参数名称优化	1天前

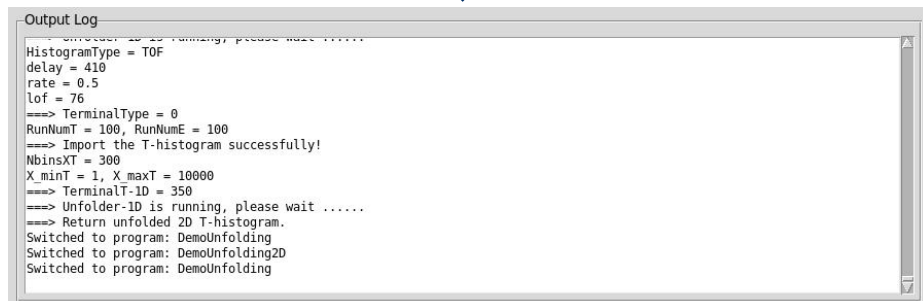
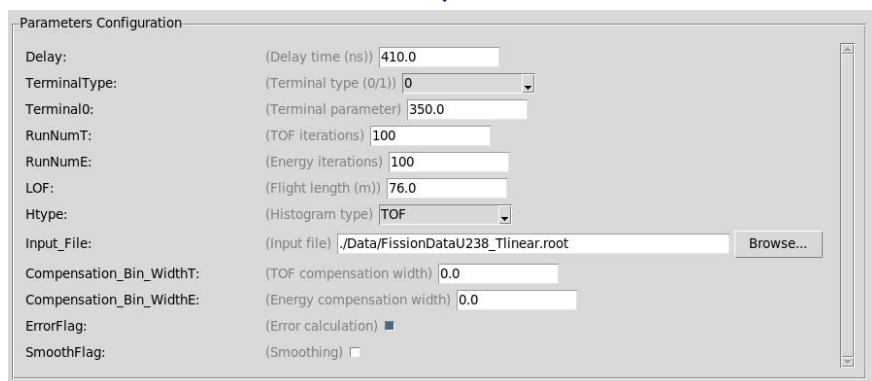
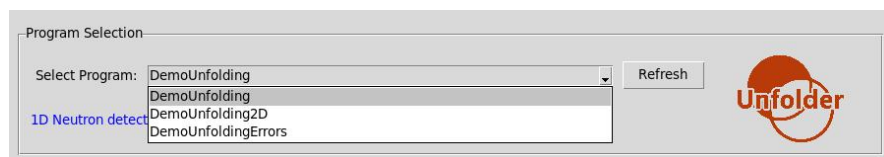
快速开始脚本

可以创建一个 setup.sh 脚本快速配置:

```
#!/bin/bash
echo "=== 克隆项目 ==="
git clone <your-gitlab-repo-url>
cd <project-directory>

echo "=== 配置环境 ==="
source /path/to/root/bin/thisroot.sh
sudo apt-get update
sudo apt-get install python3 python3-pip
sudo apt-get install python3-tk
sudo apt-get install cmake
sudo apt-get install build-essential

echo "=== 启动 GUI ==="
python3 program_gui.py
```



- 选择一维、二维解谱程序，误差分析演示(暂未实现)
- refresh按钮可使全部参数恢复为默认值

- Delay: 延迟时间，默认410ns。
- TerminalType: 横轴起始点类型。TerminalType = 0 横轴从靠近时间零点端开始；TerminalType = 1 横轴从双束团之后（410ns 之后）的任一点开始。
- Terminal0: 横轴测量起始点，TerminalType = 0时配置
- RunNumT: 解谱迭代次数（时间）
- RunNumE: 解谱迭代次数（能量）
- LOF: 中子飞行距离
- Htype: 解谱类型，时间谱(TOF)，能量谱(ESPEC)
- Input_File: 待解谱文件，.root数据格式
- Compensation_Bin_WidthT: 补偿时间道宽设置，TerminalType = 1 时配置。
- Compensation_Bin_WidthE: 补偿能量道宽设置，TerminalType = 1 时配置。
- ErrorFlag: 误差计算开关（默认开启）
- SmoothFlag: 平滑计算开关（默认关闭，不推荐开启）

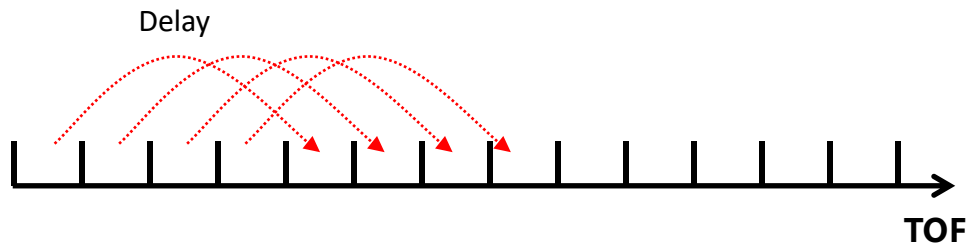
- Save Config 保存配置，在 config 文件夹输出unfolding_config.json 文件，下次打开可视化程序时自动加载该配置，或发送给其他人使用。
- Load Config 加载.json 配置文件，配置文件放置在 config 文件夹下
- Build&Run 一键编译运行（cmake, make, ./DemoUnfolding）
- Stop 程序停止按钮。程序运行期间停止运行；程序运行结束一键清图
- Clear Log 清空日志输出窗口

调试信息，配置信息输出

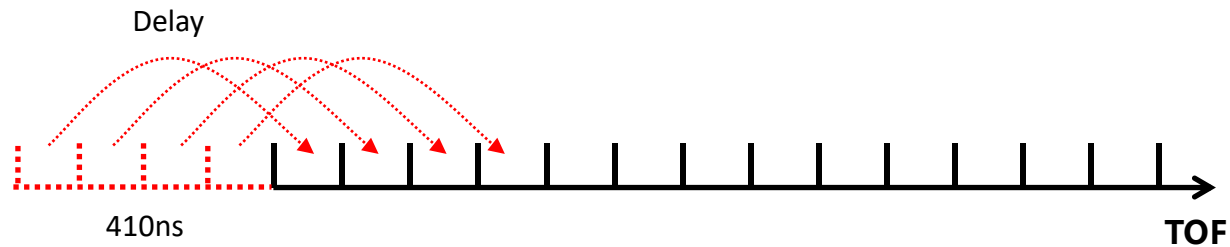
首次运行使用默认参数，直接点Build&Run

04 注意事项

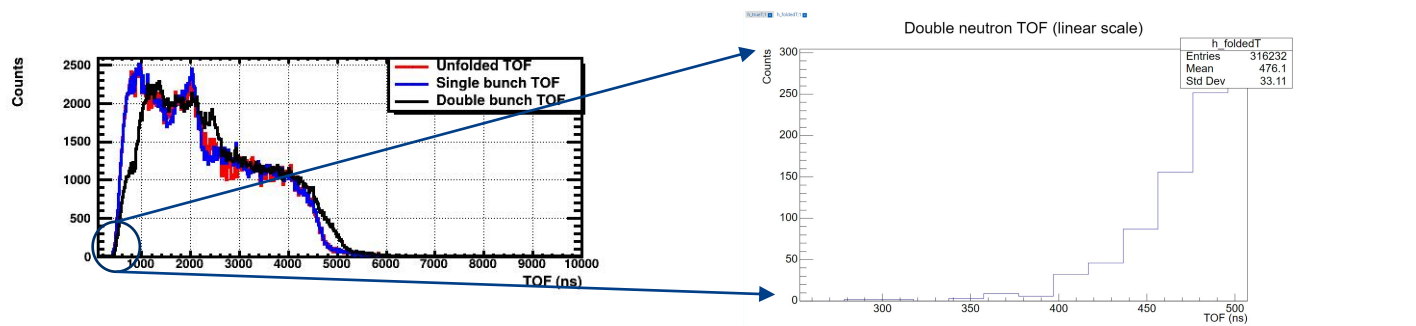
• TerminalType的设置



- TerminalType = 0, TOF谱形的起点是测量轴的任意一点。
- 需要给出Terminal0参数, 即真正的直方图横轴测量起始点
- 这种模式下解谱算法为精确算法, **推荐使用**

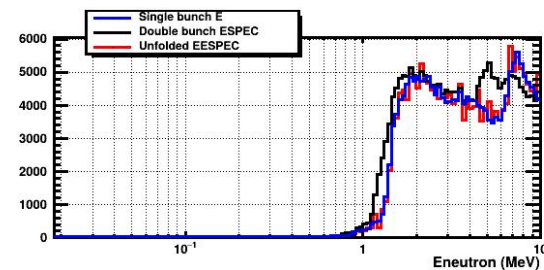
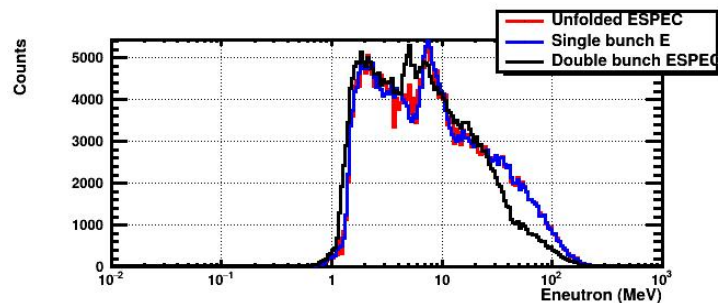


- TerminalType= 1, TOF谱形的起点超出了测量轴。
- 可选择设置 Compensation_Bin_WidthT参数以设置拓展轴的道宽, 不设置则程序会自动处理
- 这种模式下边缘处因数据缺失, 可能会出现失真, **不推荐使用**

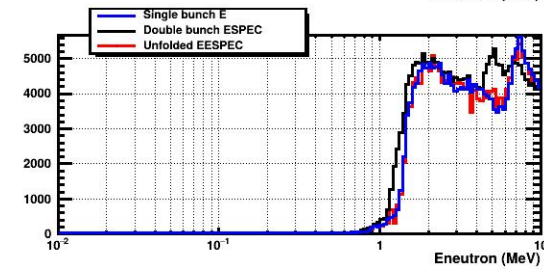


Terminal0可设置为350ns

Terminal0可设置为4/500MeV



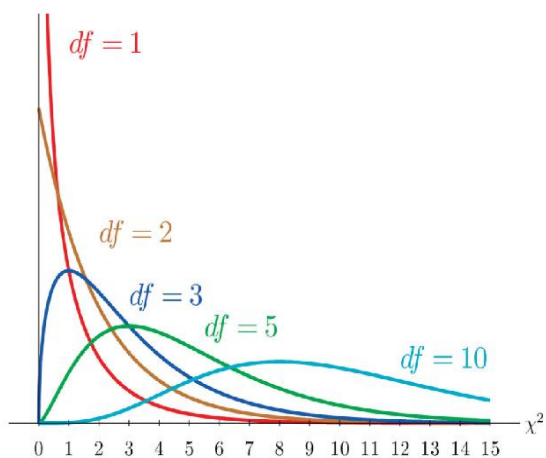
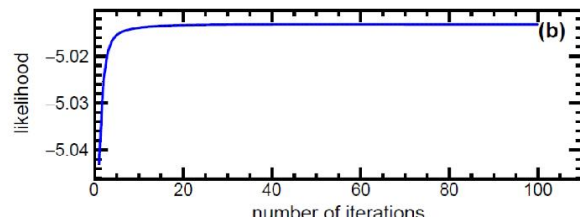
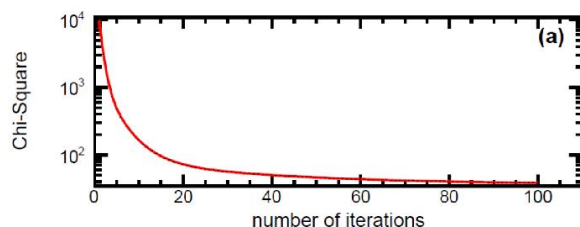
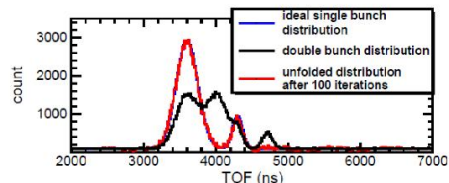
道宽0.6MeV



道宽0.3MeV

04 已注意事项

迭代次数与收敛判据



DF	P										
	0.995	0.975	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10
300	240.663	243.912	320.397	331.789	341.395	349.874	352.425	359.306	366.844	375.369	381.425
350	285.606	290.094	372.051	384.306	394.626	403.723	406.457	414.474	421.900	431.017	437.466
400	330.503	336.482	423.550	436.649	447.632	457.305	460.211	468.724	476.606	486.274	493.132
450	376.483	383.118	475.035	488.849	500.456	510.670	513.736	522.717	531.028	541.212	548.432
500	422.303	429.936	526.401	540.930	553.127	563.852	567.070	576.493	585.207	595.882	603.446
550	468.328	476.510	577.701	592.909	605.667	616.878	620.241	630.084	639.183	650.324	658.215
600	514.529	523.019	628.943	644.800	658.094	669.769	673.270	683.516	692.582	704.568	712.771
650	560.885	569.145	680.134	696.614	710.421	722.542	726.176	736.807	746.625	758.639	767.141
700	607.380	615.577	728.280	745.359	760.601	772.211	775.972	786.974	796.131	810.566	821.347
750	653.997	662.033	776.386	793.043	808.822	821.785	825.870	836.029	845.514	860.308	871.404
800	700.725	708.513	823.513	840.611	856.911	870.275	874.479	884.984	894.786	910.991	923.329
850	747.554	755.099	869.402	886.149	902.249	916.337	920.269	930.808	940.848	957.957	970.534
900	794.475	801.756	915.409	932.782	948.504	962.263	966.031	976.630	986.274	1004.974	1018.026
950	841.480	848.477	961.478	978.478	993.872	1007.211	1011.651	1022.134	1031.831	1050.918	1063.918
1000	888.564	895.257	1007.431	1024.224	1039.269	1052.331	1056.969	1067.669	1076.568	1105.118	1118.517

- 似然值收敛较快
- χ^2 判断收敛程度，不是越小越好，过小可能造成振荡

- 卡方表现的是收敛的置信水平，具体临界值带有一定的“主观性”，需要根据具体情况和统计数据来决定，表示根据真实抽样得到的统计分布相对真实分布的涨落的预期
- 卡方函数查找<http://chisquaretable.net/>
- 对于左侧的解谱（df=300，300个bin）当卡方值小于240时，表示得到的结果已经过于接近单数团分布，当统计量较小时，可能会造成解谱结果的振荡。

• 其他注意事项

- root 数据文件中直方图类型须为 TH1F* h (二维为 TH2F*) 类型, 名称须为 h_foldedT/h_foldedE (二维为 h_foldedT/E2D) , TOF 直方图横坐标的时间单位必须为 ns, 能谱直方图的横坐标单位必须为 MeV;
- 直方图的道宽不宜过窄, 道宽太窄时每道的统计误差太大, 在迭代过程中会将统计误差放大, 得到一些不真实的谱形结构;
- 若输入的直方图统计量很大 (每道的计数均在 10^5 以上) , 那么可将迭代次数设置得比较大 (100 以上) , 保证迭代的谱形能充分收敛到单束团谱形; 若输入的直方图统计量较小 (每道计数在 10^3 以下) , 那么迭代次数不宜过多 (30 左右即可) , 迭代次数过多会将统计误差放大造成输出谱形失真。具体的迭代次数可根据收敛过程判断;
- 当前不推荐启用平滑计算功能 (SmoothFlag = false)



中国科学院高能物理研究所
Institute of High Energy Physics Chinese Academy of Sciences



散裂中子源科学中心
Spallation Neutron Source Science Center

敬请各位专家批评指正
谢谢

2026.07.14