

阶段一 实验报告

—— Gamma 在液体闪烁体内的能量沉积

致理-数理 1 刘苏青* 2021013371

致理-物 12 卢一鸣[†] 2021013384

致理-数理 1 马腾跃[‡] 2021013389

2023 / 08 / 08

摘 要

本文首先大致概括了大作业阶段一的整体思路，然后详细阐述了 git 上各文件的用法，最后提出大作业实现过程中遇到的问题，并给出相应的解决方法，便于读者快速了解我们组的项目成果。

关键词：光电效应、康普顿散射、相互作用点、液体闪烁体

*liu-sq21@mails.tsinghua.edu.cn

[†]luym21@mails.tsinghua.edu.cn

[‡]mt21@mails.tsinghua.edu.cn

目录

1	整体思路	3
2	实现方式	4
2.1	光电效应	4
2.2	康普顿散射	5
2.3	相互作用点	6
2.4	模拟液闪分子与光子的作用全过程	7
3	小组分工	9
4	成果复现说明	10
5	提升部分	12
5.1	电子沉积点的绘图形式	12
5.2	使用 Bethe 公式精细考虑电子在液闪中的能量沉积	13
5.3	一个根据任一概率密度函数进行抽样的方法	15
5.4	在模拟过程中考虑瑞利散射 [1]	16
5.4.1	散射截面的计算	16
5.4.2	散射角采样	17
5.4.3	加入主程序	17
5.4.4	复现方式	17
5.4.5	模拟结果呈现	18

1 整体思路

大作业阶段一模拟了如下过程：一个初始能量为 1MeV、2MeV、5MeV 或 10MeV、初始顶点为 (0, 0, 0) 的光子，沿初始方向为 x 轴正方向发出，然后根据总截面通过 `interaction.py` 采样相互作用点。在相互作用点处，根据两种效应的截面大小随机选择发生哪种效应，并继续前往下一个相互作用点，直到发生光电效应为止。如果某次发生的是光电效应，则通过 `photoelectric.py` 采样电子出射角并计算电子能量；如果某次发生的是康普顿散射，则通过 `compton.py` 采样散射光子能量并计算电子能量、光子散射角和电子反冲角。

2 实现方式

2.1 光电效应

此 Python 脚本命名为 photoelectric.py，存储在 photoelectric 文件夹中。你可以通过在 BASH 命令行中输入如下指令调用该程序，以绘制入射光子能量为 ENERGY 时光电效应电子出射角的分布，并将结果存储在 OUTPUT-FILE-NAME.pdf 中（注意，这里是对于五种原子都绘制了电子出射角分布）

```
$ python3 ./photoelectric/photoelectric.py -e ENERGY -o OUTPUT-FILE-NAME
```

为避免调用单一函数时需要反复从系统文件中读取原子信息，我们将对于同一类原子 (Z 相同) 可能调用的所有操作函数封装入了一个 photoelectric 类中。类中具体结构如下

- **初始化:** 对应函数 `__init__`。通过调用 `init_cs(Z)`, `init_le_cs(Z)`, `init_high(Z)`, `init_low(Z)` 四个函数将系统文件中所有有关 Z 原子的信息存入该类的默认信息中。这样此后对于计算插值等操作，都可以直接调用类中已有的信息，而不必重新从系统文件中进行读取。
- **截面大小插值:** 对应函数 `interpolation(E_gamma, data)`。利用 `numpy.searchsorted` 函数找到能量 `E_gamma` 在 `data` 区间的前驱后继，以该前驱后继构成的直线在 `E_gamma` 处取值作为插值得到的 `E_gamma` 对应的截面大小。**注意，此函数的需要保证 `E_gamma` 在 `data` 所包含的能量大小区间内。**
- **计算截面:** 对应函数 `get_single_atom_photoelectric_cross_section(E_gamma)` 这里是根据题目要求对光子能量进行分类，最后返回“截面大小插值”的结果。（也即完成“截面大小插值”所需前提）
- **抽样光电子散射角:** 对应函数 `sampleing(E_gamma)`。直接按照题目给定的抽样方法进行了随机抽样。

除该 photoelectric 类外，代码中还实现了绘制入射光子能量为 `E_gamma`、碰撞原子核核电荷数为 Z 时，光电子的散射角分布图像的功能。

- **绘制光电子角分布曲线:** 对应函数 `draw_photoelectric_theta(Z, E_gamma)`。是用于绘制关于核电荷数为 Z 的原子的光电子角分布曲线。这一函数直接调用了 `photoelectric` 类中 `sampleing` 函数 20000 次, 采样出数量足够多的光电子 (在不同散射角度分布的密度不同)。此后运用高斯核密度估计得到光电子散射角分布曲线, 并在 $[0, 180]$ 度中均匀选取 2000 个点, 利用 `matplotlib.pyplot.plot` 函数绘制角分布曲线。

2.2 康普顿散射

此 Python 脚本命名为 `compton.py`, 使用 SciPy 库进行康普顿散射模拟, 使用 `matplotlib` 库来可视化结果。你可以通过在 BASH 命令行中输入如下指令调用脚本绘制图像:

```
$ python3 ./compton/compton.py -e ENERGY -o OUTPUT-FILE-NAME
```

下面逐步介绍脚本的主要组成部分和功能:

- **常量和库:** 导入脚本首先导入所需的库, 包括 `scipy.interpolate`、`scipy.constants`、`random`、`mat`、`numpy`、`matplotlib.pyplot`、`scipy.stats`、`os` 和 `argparse` (用于接收命令行参数)。
- **命令行参数:** 脚本使用 `argparse` 模块来处理命令行参数。它接受两个可选参数: `-e` 用于指定入射光子能量, `-o` 用于提供生成图像的输出文件名。
- **常量和单位:** 脚本使用 `scipy.constants` 模块定义 `electron_mass` 和 `c`, 以计算能量的单位为 MeV。能量单位对于康普顿散射模拟是很重要的。
- **康普顿散射截面函数:** `compton_sigma_e(E0, Z)` 用于计算康普顿散射截面。给定入射光子能量 $E0$ 和原子序数 Z , 函数从预先存在的 `data` 文件中读取特定质子数单个原子在固定入射光子能量时的截面数据, 若输入的入射能量 $E0$ 不存在于文件中, 则进行线性插值 `interpolate` 以计算所提供能量的截面。
- **散射点计算函数:** `deposition_point(E0, theta)` 使用给定的入射光子能量 $E0$ 和散射角度 θ 计算反冲电子的散射角度 θ_e 。

- **散射光子、电子状态采样散射采样函数**: `compton_scattered_photon_sampling(E0)` 用于对散射光子进行能量采样。输入是入射光子能量 E_0 。对于每一个输入的光子入射能量，函数会按照 Klein-Nishina 公式的形式和抽样方法抽样光子出射、入射能量比例 $\epsilon(\frac{E_1}{E_0})$ 和光子的散射角 θ ，并在最后运用能量守恒，调用 `depoisit_point` 函数计算出电子的反冲角度分布 θ_e ，最终返回这三个数值以方便后续进行连续采样与绘图，并为相互作用连续模拟做准备。
- **散射光子能量、角度和反冲电子角度分布函数**: `compton_scattered_photon_ploting(E0)` 可视化散射光子和反冲电子的能量和角度分布。该函数调用散射采样函数迭代地对不同光子入射能量 E_0 下散射光子的能量 E 、 θ （散射角）和 θ_e （反冲电子角度）进行采样（5000 次），并生成了包含 1000 个数据点的线性等间距数组以对最后的计数数据进行高斯核密度估计，绘制出平滑美观的三分式曲线图，反映了散射光子能量、角度和反冲电子角度关于入射光子能量的分布。

2.3 相互作用点

此 Python 脚本命名为 `interaction.py`，通过 `interacted(E_gamma)` 函数实现了相互作用点的采样，并通过主函数中的 `draw_curve()` 画出液闪分子和各组分原子的两种截面随能量的变化，存储在 `cs_curve.pdf` 文件中。

你可以通过在 BASH 命令行中输入如下指令调用程序，以生成液闪分子与各组分原子两种散射截面和入射光子能量的关系图。

```
$ python3 ./interaction.py
```

该脚本具体结构如下：

- **导入数密度**：函数 `extract_density(filename)` 用于从给定的 `gdm1` 文件中提取元素的数密度信息，并以字典形式返回每个元素及其对应的数密度。
- **导入质子数**：函数 `Z_number(filename)` 用于从给定的 `gdm1` 文件中提取元素的质子数 Z ，并以字典形式返回每个元素及其对应的质子数。

- **模拟光子在液体闪烁体中的相互作用：**函数 `interacted(E_gamma)` 模拟了光子在液体闪烁体中与原子发生相互作用的过程。根据输入的光子能量 E_γ ，返回一个布尔值来表示是进行康普顿散射还是光电效应，以及距离下一次反应的位置和与哪个原子发生的反应。
- **绘制两种截面随入射光子能量的变化曲线：**函数 `draw_curve()` 绘制了液体闪烁体的康普顿散射截面和光电效应截面随入射光子能量的变化曲线。同时，它还返回了计算所得的液闪分子的康普顿效应截面和光电效应截面，以及入射光子能量 E_γ 取样点的信息。

2.4 模拟液闪分子与光子的作用全过程

我们在 `main.py` 中实现了模拟一个光子射入液闪后与液闪分子发生相互作用全过程的功能。可以通过如下代码调用该程序，会在当前路径得到题目要求的 `deposit.h5` 文件 (模拟 SIMULATION-TIME 个能量为 ENERGY 光子射入的事件)。

```
$ python3 ./main.py -e ENERGY -n_simulation SIMULATION-TIME
```

我们将任意时刻的光子/电子都视作对象，编写了 `main_atom` 类。具体结构如下：

- **初始化：**对应函数 `__init__`。在类中存储了方向 `Direction`(3 维 numpy array)，能量 `Energy`(float)，位置 `Position`(3 维 numpy array) 三个变量
- **移动粒子：**对应函数 `next_atom(theta, phi, distance, new_energy)`。表示产生一个新的 `main_atom` 类对象 (意味产生一个新粒子)。新粒子的**随体坐标轴**是先绕**原粒子随体坐标轴**的 Y 轴顺时针旋转 `theta` 角，再绕**原粒子随体坐标轴**的 Z 轴顺时针旋转 `phi` 角。新粒子的位置是原粒子的位置沿**新粒子随体坐标轴**的 Z 轴正方向移动 `distance` 距离得到。新粒子的能量是 `new_energy`。

很容易发现我们需要编写一个向量绕定轴旋转的功能，我们将这个功能集成在了 `./tool/vector_tool.py` 中。

- **向量绕定轴旋转：**对应代码 `./tool/vector_tool.py` 中 `rotate_vector(vector, axis, angle)` 函数。该函数将返回三维向量 `vector` 绕三维向量 `axis` 轴正方向，顺时针旋转 `angle` 角度后的向量

模拟光子入射发生的反应全过程在主函数中完成。

- **模拟所有反应过程：**在 `main.py` 的主函数中。根据调用 `interaction.py` 中的 `interacted(E_gamma)` 函数的返回值选择此后是模拟光电效应还是康普顿散射过程。二者后续处理大致相似：采样电子散射角、能量 → 计算光子散射角、能量（若有）→ 利用 `next_atom` 函数处理出新对象 → 继续调用 `interacted` 函数采样新一轮反应（若不是光电效应）

3 小组分工

- **刘苏青**: 查找瑞利散射资料并录入其数据, 编写 `interaction.py`、`interaction_all_extend.py` 程序, 整理仓库, 并编辑写作实验报告的主要框架及对应部分。
- **卢一鸣**: 编写 `photoelectric.py`、`main.py`、`main_all_extend.py`、`main_Rayleigh.py` 程序, 整理仓库, 并编辑写作实验报告的对应部分。
- **马腾跃**: 编写 `compton.py`、`Rayleigh.py` 散射截面部分、`main_Bethe.py` 和 `Makefile` 文件, 编写并优化画图程序 `analysis.py`, 整理仓库, 并编辑写作实验报告的对应部分。

4 成果复现说明

为确保整个流程的完整性、可复现性和一次性，我们使用 GNU make 来组织所有的程序文件。执行特定任务需要直接在命令行中输入对应的命令，比如 make all 来执行所有任务，或者 make pe_theta_1 来执行单个任务（模拟光电效应）。提供的参数 energy、n_simulation 可以在 Makefile 顶部中指定。

Makefile 各指令具体调用方法如下：

1. make all **或** make：执行所有任务，包括运行所有脚本、生成所有结果和可视化任务。
2. make pe_theta_\$(energy)：运行 photoelectric.py 脚本模拟光电效应，生成电子出射角分布文件 pe_theta_\$(energy).pdf。你需要在 Makefile 顶部更改光子能量 energy。
3. make ce_photo_\$(energy)：运行 compton.py 脚本模拟康普顿效应，生成散射光子能量分布和散射角分布，生成散射光子能量分布和散射角分布、电子出射角分布文件 ce_photo_\$(energy).pdf。这同样需要在 Makefile 顶部更改光子能量 energy。

4. make cs_curve.pdf：运行 interaction.py 脚本生成截面曲线 pdf 文档。
5. make Rayleigh.pdf：运行 Rayleigh.py 脚本生成瑞利散射截面曲线 pdf 文档。
6. make deposit.h5、make deposit_Bethe.h5 **和** make deposit_Rayleigh.h5：

分别运行 main.py 或 main_Bethe.py 或 main_all_extend 模拟文件，生成模拟结果。

这三个文件依次代表了” **仅考虑基本要求的模拟结果**”、” **考虑了电子的 Bethe 公式的模拟结果**”、” **考虑了电子的 Bethe 公式以及光子的瑞利散射的模拟结果**”

这些需要你在 Makefile 顶部更改光子能量 energy 和该种入射光子能量下的模拟次数 n_simulation。

7. make deposit(__Bethe).html **和** make deposit(__Bethe)_dissymmetry.png：根据模拟结果（精细考虑或不精细考虑电子与液闪分子相互作用）生成的 h5

文件, 运行 analysis.py 文件, 生成沉积能量的分布的可视化 html 和 png 图片。

8. make deposit_all_extend.html 和 make deposit_all_extend_dissymmetry.png: 根据模拟结果 (考虑了所有扩展结果: Bethe 公式以及瑞利散射) 生成的 h5 文件, 运行 analysis.py 文件, 生成沉积能量的分布的可视化 html 和 png 图片。

9. make clean: 执行清理任务, 删除所有的输出文件。

5 提升部分

5.1 电子沉积点的绘图形式

我们在 `analysis.py` 中实现了绘制出不同情形下电子能量沉积点的空间分布的功能。可以通过如下代码调用该程序，会在当前路径得到空间分布点文件（稍后对文件格式进行说明）。

```
# 沉积能量点分布
python3 ./analysis.py -i deposit.h5
```

在绘制电子沉积点的空间分布时，我们最初选用了给出文件中要求的 pdf 格式。但是，如下图所示，电子沉积点的数量较多，在三维图形中，许多细节由于点过于拥挤被掩盖掉了。同时，3D 散点图在平面上展示的角度是单一的，完全无法看出空间分布的规律。

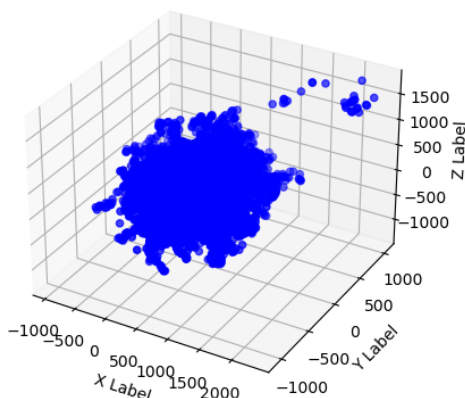


Figure 1: 效果很不好的三维散点图

在查阅相关资料后，我们选择了创建 html 格式的可交互式三维散点图，并调整了散点的不透明度为 0.2，以达到较好的观察与呈现效果。但是，从采样的过程中我们可以得知，三点关于 x 轴应当是旋转对称分布的。也就是说，沿 x 轴的剖面上的信息可能会更有价值。为了更直观、更准确地表现出相互作用点的分布密度，我们采用了二维分层设色“等高线”图来展现 x-z 平面上的信息。图中，像素的颜色代表了这一区域相互作用点分布的密度（归一化）。这样，结合三维可交互式散点图与二维密度分布图，我们能对相互作用点的分布有着更为清晰的认知。为了使边缘清晰、分布直观，我们对相

互作用点的分布再一次采用了高斯核密度估计。

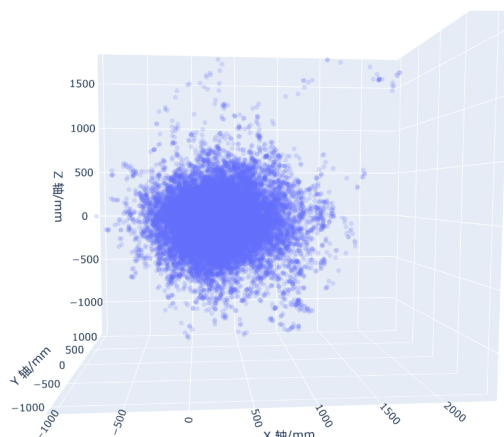


Figure 2: 可交互的三维散点图演示（入射光子能量：1MeV）

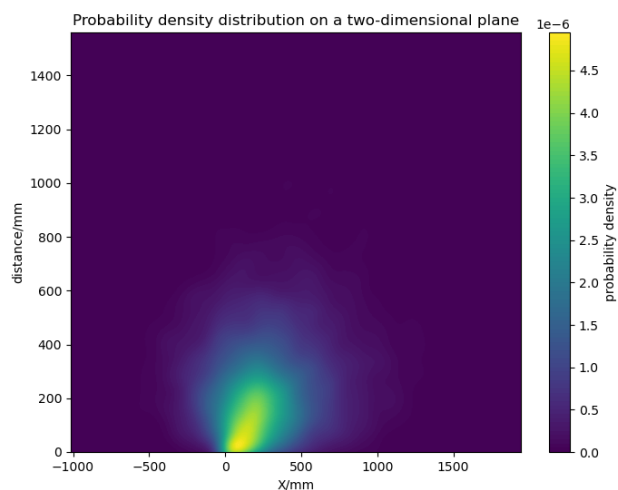


Figure 3: 二维消对称电子沉积点分布图（入射光子能量：1MeV）

5.2 使用 Bethe 公式精细考虑电子在液闪中的能量沉积

我们在 analysis.py 中实现了绘制出不同情形下电子能量沉积点的空间分布的功能。通过如下代码调用该程序，会在当前路径得到精细考虑电子与液闪分子相互作用的空间分布点文件。

```
# 考虑Bethe公式的沉积能量点分布
python3 ./analysis.py -i deposit_Bethe.h5
```

不同出射能量的电子，在液闪分子中的运动距离显然应该不同。在上面的部分中将其均视为运动 2mm，是非常粗糙的一种估计。因此，需要引入 Bethe 公式，定量考虑运动电子在液闪分子中的电离和辐射能损。

- 电离能损（低能为主）：

$$\left(-\frac{dE}{dx}\right)_{ion} = \left(\frac{1}{4\pi\epsilon_0}\right)^2 \cdot \frac{2\pi e^4 N Z}{m_e v^2} \times \left[\ln \frac{m_e v^2 E}{2I^2(1-\beta^2)} - (\ln 2)(2\sqrt{1-\beta^2} - 1 + \beta^2) + (1-\beta^2) + \frac{1}{8}(1-\sqrt{1-\beta^2})^2 \right]$$

- 辐射能损（高能为主）：

$$\left(-\frac{dE}{dx}\right)_{rad} = \left(\frac{1}{4\pi\epsilon_0}\right)^2 \cdot \frac{NEZ(Z+1)e^4}{137m_e c^4} \cdot \left(4 \ln \frac{2E}{m_e c^2} - \frac{4}{3}\right)$$

由于低能和高能的界限并不清晰，在程序的每一步中我们都考虑了两种能损。要注意的是，为了避免歧义，我们在公式中全部使用了国际单位制，并在输出运动距离时重新将米转换为了毫米。程序的具体实现方式如下：

1. 将 main.py 中的 “2” 更换为传入电子出射能量的函数 Bethe_x(elec_energy)

2. 实现函数 Bethe_x(E)：

- 数值准备：对于公式中的 N，即靶粒子数密度，我们采用了所有原子的数密度之和，这样才能够更合理地反映电子与所有粒子的相互作用；对于公式中的 Z，即质子数，我们假设每个步长中选取唯一的 Z 值，并按照各种原子在液闪物质中的数密度作为权重进行随机选择，以较好地模拟出电子与各元素原子之间的相互作用过程。综合权衡精度与计算速度，我们暂时选取一步的步长为 0.001mm。
- calculate_ion_loss(E, step, Z): 这个函数用于计算电离损失，参数 E 代表电子能量，step 是每一步迭代的间距，Z 是目标原子的原子序数。在这个函数中，首先计算洛伦兹因子 (γ)，确定电子的速度 (v)，然后利用 Bethe 公式（电离能量损失公式）计算每一步的能量损失。
- calculate_rad_loss(E, step, Z): 这个函数用于计算辐射损失，参数与上述相同。这个函数主要应用于高能量电子的情况，使用了 Bethe-Heitler 公式。
- 在主循环部分，程序首先确定了本次迭代过程中电子碰撞的原子种类 (chosen_element)。然后，计算相应的能量损失，并从电子总能量中减去这部分能量。同时，更新电子穿过的距离。循环会持续进行直到电子的能量低于特定阈值（如 10^{-30}J ）。

在精细考虑电子在液闪分子中的沉积过程之后，我们又进行了图像的绘制。为方便对比，这里只展示消除对称性之后的剖面图像：

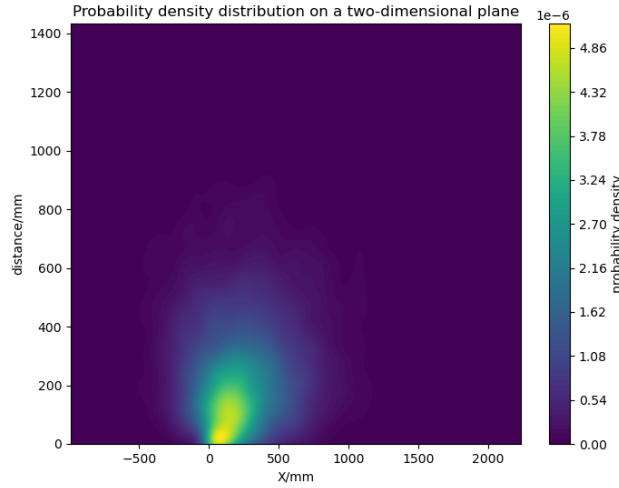


Figure 4: 精细考虑后的消对称电子沉积点分布图（入射光子能量：1MeV）

从图 4 中能够直观的看出，与图 3 相比，电子沉积点向 z (y) 方向的扩散距离比不考虑该效应时比较接近。并且注意到，在密度较高的区域，沉积点分布更为集中。这与 h5 文件中存储的数据相符合，即电子能量较高时电子的运动距离明显大于 2mm，而在电子能量降低为 0.01MeV 或更低之后运动距离便远远小于 2mm。

5.3 一个根据任一概率密度函数进行抽样的方法

在考虑瑞利散射的问题中，需要解决如何根据任一个已知的概率密度函数进行抽样的问题，我们将方法列写如下。

记随机变量 x 服从分布函数为 $F(x)$ ，概率密度函数为 $P(x)$ ，满足 $\frac{dF}{dx} = P$ 。考虑这个函数分布在 $[x_{min}, x_{max}]$ 中，选取若干个散点满足

$$x_1 = x_{min} < x_2 < \cdots < x_{N-1} < x_N = x_{max}$$

此后我们令 $\xi_i = F(x_i)$ 。根据概率论知识，若我们能够知道函数 $F^{-1}(\xi_i) = x_i$ 在任意 ξ 处的取值，我们就可以通过向 F^{-1} 中填入一个均匀分布 ξ ，得到一个满足分布函数为 $F(x)$ 的随机变量 x

于是要解决的问题就是如何得到任意 ξ 处， $F^{-1}(\xi)$ 的值。一个显然的想法是线性插值，但这会造成较大的误差。比较好的做法是使用 rational interpolation。

对于任一个 ξ ，显然我们能找到唯一的一个 i ，满足 $\xi_i \leq \xi < \xi_{i+1}$ 。我们进行如下拟合。

$$F^{-1}(\xi) = x_i + \frac{1 + a_i + b_i}{1 + a_i\eta + b_i\eta^2}(x_{i+1} - x_i)$$

其中 $\eta \equiv \frac{\xi - \xi_i}{\xi_{i+1} - \xi_i}$ 。根据一系列关于 a_i, b_i 的边界条件限制，可以推导出他们的取值如下

$$b_i = 1 - \left(\frac{\xi_{i+1} - \xi_i}{x_{i+1} - x_i} \right)^2 \frac{1}{p(x_{i+1})p(x_i)} \quad (1)$$

$$a_i = \frac{\xi_{i+1} - \xi_i}{x_{i+1} - x_i} \frac{1}{p(x_i)} - b_i - 1 \quad (2)$$

其中 $p(x_i)$ 是 x 服从的概率密度函数在 x_i 处的取值。则进一步我们可以直接推出对于均匀分布抽样得到的 ξ ，其对应的 x 为

$$x = x_i + \frac{(1 + a_i + b_i)\eta}{1 + a_i\eta + b_i\eta^2}(x_{i+1} - x_i)$$

这个算法名叫 RITA (Rational Inverse Transform with Aliasing) algorithm。

5.4 在模拟过程中考虑瑞利散射 [1]

5.4.1 散射截面的计算

此 Python 脚本命名为 Rayleigh.py，使用 SciPy 库进行瑞利散射模拟，使用 matplotlib 库来可视化结果。你可以通过在 BASH 命令行中输入如下指令调用脚本绘制瑞利散射截面图像并保存为 Rayleigh.pdf:

```
$ python3 ./Rayleigh/Rayleigh.py E
```

瑞利散射的单个原子总散射截面计算公式 [1] 为:

$$\sigma_{Ra} = \int \frac{d\sigma_{Ra}}{d\Omega} d\Omega = \pi r_e^2 \int (1 + \cos^2 \theta) [F(x, Z)]^2 d(\cos \theta)$$

其中， r_e 为电子经典半径， $F(x, Z)$ 称为原子散射因子，为一无量纲量。 $x = \frac{\sin(\frac{\theta}{2})}{\lambda}$ ，其中， λ 为入射光的波长，单位为 \AA 。对不同质子数 Z 的原子， F 与 x 的关系表见 data/Rayleigh/F-x-{Z}.dat[2]。对于不在表中的数据，我们采用线性插值来进行计算(越界则 1111111111)。具体实现细节与 2.2 康普顿散射部分非常相似，这里就不再赘述。

对于积分的计算，我们采用数值求积法，将 -1 到 1 划分为 100 个区间，分别近似求得积分函数大小后再乘以区间宽度 0.02，并相加得到近似计算结果。

5.4.2 散射角采样

光子散射角 θ 的余弦值 $\cos \theta$ 满足如下概率密度函数

$$p_{Ra}(\cos \theta) = \frac{1 + \cos^2 \theta}{2} [F(x, Z)]^2$$

记 $\kappa \equiv \frac{E_c}{m_e c^2}$, 则 x 有上界

$$x_{max} = 20.6074 \times 2\kappa$$

我们记 $\pi(x^2) \equiv [F(x, Z)]^2$, 则 $x^2 \equiv t$ 的取值是 $[0, x_{max}^2]$ 。可以将 $\pi(t)$ 视作一个关于 t 的概率密度函数 (需要进行归一化)。因此, 抽样过程如下

- 使用 RITA algorithm, 根据 $\pi(t)$ 抽样一个随机数 $t \in [0, x_{max}^2]$

- 令

$$\cos \theta = 1 - \frac{1}{2} \frac{t}{(20.6074\kappa)^2}$$

- 生成一个 $[0,1]$ 均匀的随机数 ξ
- 如果 $\xi > \frac{1+\cos^2 \theta}{2}$, 返回第一步
- 否则, 就抽样出了一个合法的 $\cos \theta$

5.4.3 加入主程序

事实上非常简单, 但为了便于助教对比, 我们重新编写了 `main_all_extend.py` 以及 `interaction_all_extend.py` 代码。

在 `interacted` 函数中, 我们将瑞利散射的截面也计入了总截面中, 并根据三类截面大小抽样得到本次相互作用的原子及相互作用方式。

在主函数中, 我们新增了一项判断瑞利散射的过程。在这个过程中实际上只需要将光子转向, 这可以通过 2.4 中提到的 `main_atom` 类轻松实现

5.4.4 复现方式

我们也将这一部分加入了 Makefile。可以通过 `make` 指令来绘制各原子和液闪分子瑞利散射截面和入射光子能量的关系图, 以及进行考虑瑞利散射的全相互作用过程模拟。具体内容见 4 成果复现说明部分。

5.4.5 模拟结果呈现

在考虑瑞利散射并精细考虑电子与液闪分子的相互作用之后，我们又进行了图像的绘制。为方便对比，这里只展示消除对称性之后的剖面图像：

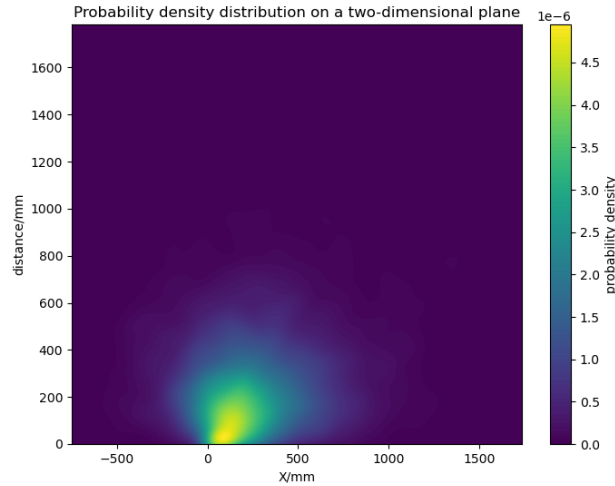


Figure 5: 精细考虑并加入瑞利散射后的消对称电子沉积点分布图 (入射光子能量: 1MeV)

从图 5 中能够直观的看出，与图 4 相比，电子向 y 、 z 方向的扩散距离比不考虑瑞利散射时更长一些，集中区域较分散一些。这与散射起到的作用相符合，即光子能量较低时光子的散射角度比较大，导致光子到达的空间范围比较大。事实上，这一部分的随机性很大，我们多次运行了程序，发现可扩散到的范围非常大。

参考文献

- [1] PENELOPE-2006: A Code System for Monte Carlo Simulation of Electron and Photon Transport Workshop Proceedings, 2006.
- [2] JH Hubbell, WJ Veigele, EA Briggs, RT Brown, DT Cromer, and RJ Howerton. Atomic form factors, incoherent scattering functions, and photon scattering cross sections. Journal of Physical and Chemical Reference Data, 4:471, 1975.