

简单Geant4介绍

北京大学 乔颢

Geant4

维基:Geant4 是由CERN基于C++面向对象技术开发的蒙特卡罗应用软件包, 用于模拟粒子在物质中运输的物理过程。

官网:Geant4 is a toolkit for the simulation of the passage of particles through matter.

Geant4 是 MC 软件。用于 模拟 粒子 与 物质 作用。

Geant4 主要要素

1. MC (Physics List)
2. 物质结构和构成 (Detector Construct)
3. 粒子生成 (Primary Generator)
4. 模拟干预 (常用于数据导出以及简化模拟) (Geant4xxxAction)
5. 数据处理 (ROOT)

Physics List (G4VUserPhysicsList)

PhysicsList 用于描述在粒子和物质相互作用的过程中所需要考虑的物理过程 (G4VProcess)。

Geant4中所有发生的物理过程都由这个类控制是否启用。

Geant4有一些预置的PhysicsList可供使用。

http://geant4.cern.ch/geant4/support/proc_mod_catalog/physics_lists/useCases.shtml

当然也可以自己继承G4VUserPhysicsList来修改决定各种物理过程列表。

Detector Construct (G4VUserDetectorConstruction)

这个类用于描述探测器的结构。

一个探测器部件往往由以下方式所描述：

1. Solid 描述该部件的几何结构(例如, 长方形, 圆柱形, 多面体)
2. LogicalVolume 描述该部件的物理状态(例如, 材料)
3. PhysicsVolume 描述该部件的相对位置(例如, 旋转, 平移)

多个探测器部件最终组成一整个探测器。

Geant4要求探测器部件之间不能有重叠, 不能有越界(子部件超出父部件的范围)

Primary Generator (G4VUserPrimaryGeneratorAction)

用于描述如何产生MC模拟中最初的一套粒子。

一般而言该类描述产生的PrimaryParticles的位置, 能量, 类型, 动量。而后这些粒子与物质发生作用(由PhysicsList指定)以及产生的次级粒子等都由Geant4自动完成。

可以认为是产生粒子源。

Geant4行为 (G4XXXAction)

这是一组类，用于干预MC过程。可以认为是一套Hook。(可以没有)

Geant4提供了4个Action

1. G4UserRunAction
2. G4UserEventAction
3. G4UserStackingAction
4. G4UserTrackingAction
5. G4UserSteppingAction

Run

每一次告诉Geant4程序开始模拟(beamOn命令)开始, 到最后模拟结束的过程。被称作一个Run。

在一个Run中, PhysicsList, DetectorConstruct 都是不可改变的。

在每次Run开始的时候, 探测器结构会被初始化, 同时Geant4会根据PhysicsList载入各个粒子的属性以及其与物质反应的散射截面等等参数。

通常我们运行一次Geant4模拟便会产生且只产生一个Run(非多线程)。所以我们通常会利用Run来进行数据的存储, 它在整个模拟过程中都是可以获取得到的。

Event

每个Run下的构成单位便是Event。

Event由PrimaryGenerator产生最初粒子开始, 到所有的粒子(初级粒子, 次级粒子)都处理完成(停止运动或者跑出边界等)为止。

Event即相当于一次事件。

一个Run中通常会有成百上千个Event。

Stacking

在MC过程中，储存数据的堆栈。

比如说PrimaryParticle和物质发生反应产生一个次级粒子，这时候这个次级粒子的相关信息就会被压入Stacking等待处理。

简单的理解，Event开始时由PrimaryGenerator产生初级粒子，即将这些粒子的相关信息压入Stack。Geant4循环的处理Stack里面的数据，并将处理后的结果重新压入Stack中或者丢弃(该粒子不会发生后续的反应)。当Stack为空时Event也就结束了。

Track / Step

Track保存粒子完成一步模拟后的状态信息。例如此时的能量, 动量, 粒子类型等。

Stack中保存的信息就是各个粒子对他Track。

Step描述粒子从上一个Track到下一个Track的过程。Step会储存该步骤的信息, 例如沉积的能量, 方向等。

Geant4行为 (G4XXXAction)

通过以上的五个Action, 我们可以干预MC模拟的过程, 可以从MC中读取数据等。

读取数据的例子:

StepAction中获取改步沉积的能量以及所在的位置。

干预模拟的例子:

模拟平均自由程的时候Kill掉发生一次碰撞的粒子。

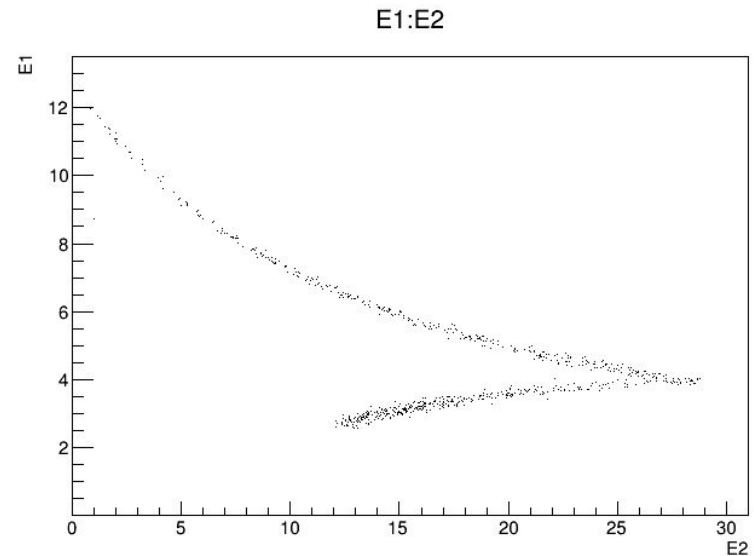
例子

问题

能量为0-50MeV的Alpha粒子穿过两块紧贴的硅板。厚度分别为0.1mm和0.4mm。

Alpha粒子会损失能量。在两块硅板损失的能量分别为 E_1 , E_2 。

研究 E_1 和 E_2 的关系。



准备

复制CMakeLists.txt（描述编译文件的程序）

新建include文件夹

新建src文件夹

复制粘贴以及修改SimpleProgram.cpp（主程序）

修改PhysicsList。

选择使用默认的Shielding物理过程。

在SimpleProgram.cpp指定使用的PhysicsList

```
// Physics list  
G4VModularPhysicsList* physicsList = new Shielding;  
physicsList->SetVerboseLevel(1);  
runManager->SetUserInitialization(physicsList);
```


构建探测器

继承自 `G4VUserDetectorConstructor`.

需要实现虚函数 `Construct()`

最后需要返回一个 `LogicalVolume` 包含所有的探测器。

添加Actions

继承G4VUserActionInitialization。并在其中注册各个Action。

必须注册PrimaryGeneratorActon。用于产生0-50MeV的Alpha粒子。

Run用于初始化TTree, 交换临时数据, 最后保存TTree。

RunAction用于新建Run, 并调用Run的初始化函数等。

EventAction用于TTree->Fill()

StepAction用于得到沉积能量和位置, 判断其属于哪个探测器, 并将数据保存在Run中。

最后在SimpleProgram.cpp制定使用改ActionInitialization。

编译 模拟

——

编译: `mkdir build; cd build; cmake ..; make;`

运行: 需要 `run.mac` 文件来告诉Geant4需要干什么。

`./SimpleProgram run.mac`

分析: 利用Root对产生的.root文件进行分析。

谢谢

Useful Resource

— — —

物理过程手册: [http://geant4.web.cern.](http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/PhysicsReferenceManual/fo/PhysicsReferenceManual.pdf)

[ch/geant4/UserDocumentation/UsersGuides/PhysicsReferenceManual/fo/PhysicsReferenceManual.pdf](http://geant4.web.cern.ch/geant4/UserDocumentation/UsersGuides/PhysicsReferenceManual/fo/PhysicsReferenceManual.pdf)

中文入门介绍:

[http://home.ustc.edu.cn/~chiwang/User%20Guid/Geant4\[1\].5.0%BA%BA%BB%AF%CB%B5%C3%F7%CA%E9.pdf](http://home.ustc.edu.cn/~chiwang/User%20Guid/Geant4[1].5.0%BA%BA%BB%AF%CB%B5%C3%F7%CA%E9.pdf)

常用物理过程:

http://geant4.cern.ch/geant4/support/proc_mod_catalog/physics_lists/useCases.shtml

Useful Resource

几何构建说明:

<http://geant4.web.cern.ch/geant4/G4UsersDocuments/UsersGuides/ForApplicationDeveloper/html/Detector/geomSolids.html>

Track信息: 见中文入门介绍5章

Step信息: 见中文入门介绍5章