

Particle-Physics-Toolkit (MadGraph5_aMC@NLO + Pythia8 + Delphes) 安装指南

费映帆 2030713006 物理学系

一 前言与安装准备

本教程将介绍如何安装您所需要的三件套，以及基本的相关知识。

- 本教程需要访问 [GitHub](#)。
- 如果在中途遇到任何问题，都可以联系我。如果您希望自己寻找解决方式，请使用 **Google**（推荐）或者 **Bing** 搜索您遇到的错误信息。为了您的身体健康，请不要使用百度或 **CSDN**（您可以搜到以后浏览其中的内容，但质量参差不齐），更不要为解决这些问题付任何人**任何费用**，因为以下提到的所有软件都可以免费使用，其中绝大部分是开源的。
- 本教程默认您已经掌握了基本的电脑使用技巧，包括但不限于开机、关机等。但防止安装和使用中的意外（包括心情变坏乃至键盘、显示器的损毁），强烈建议您在上手安装之前浏览《[计算机科学没有教的速成课](#)》，它将在很短的时间内带领您进入命令行的奇妙世界。
- 本教程中的所有指令都包含提示符。提示符很有用，它可以让您知道您应该在哪里输入指令，其形式为`$`，`#`或者`>`，提示了您应该在什么环境中输入该命令。如果提示符为`$`，则说明您应该在您的物理机命令行工具下输入该命令。如果提示符为`#`，则说明您应该在 **Docker** 容器中输入该命令。如果提示符为`>`，则说明您应该在启动 **MadGraph5_aMC@NLO** 后在其中输入。您如果遵照本教程的指示，在复制时**不需要**复制提示符。
- 如果您使用 **Docker** 安装（如后文所列），请注意本镜像基于 **Ubuntu 22.04**，若您遇到任何问题，搜索时请注意其提供的解决方式中 **Ubuntu** 版本应至少是 **Ubuntu 22**¹。
- 本教程在操作不当的情况下可能会导致您宝贵文件的丢失（潜在原因包括不小心抹掉您的硬盘、删除您的文件和操作不当导致系统内核的损坏）。所以我强烈建议您在上手操作前先**备份自己的重要数据**。每个人是自己数据的第一责任人，本教程不会为您的数据丢失负责。

二 我已经看完了前面的，开大！

1 通过 **Docker** 安装，全平台通用

您今天很幸运！我已经帮您准备好了所有的安装步骤，包括其中最让人头疼的部分，只需要简简单单运行一行命令就可以完成安装！

Docker 是一个虚拟化软件，可以在您的电脑上创建一个虚拟的 **Linux** 环境，这样您就可以在 **Windows**、**MacOS**、**Linux** 上使用三件套了。注意如果您选择在 **Windows** 或者 **MacOS** 上使用 **Docker**，不需要额外安装 **Linux**，因为我们的 **Docker** 镜像自带 **Ubuntu 22.04**。

¹Ubuntu 系统自 **Ubuntu 16** 以来出现了许多变化，其中涉及到了包括包管理器和磁盘分配的许多改变，所以如果遇到任何与之前使用（如果您使用过的话）不符的情况，请不要惊慌，因为您遇到的问题应该是很常见的。

那么代价是什么呢，古尔丹？如果您是 **MacOS** 或者 **Linux** 系统，那么恭喜您，您只需要接受这些小小的优惠就好了。但如果您是 **Windows** 用户，您可能需要做一些额外的设置，以防止 **Docker** 做出魔幻操作吞吃您电脑所有的宝贵空间。

首先您需要通过 [Docker 官网](#) 下载 **Docker**。

下载并安装好之后，打开 **MacOS** 终端 (`terminal.app`) 或者 **Windows** 命令行（需要管理员身份运行）(`cmd.exe`)，执行

```
$ docker run -it ivanfei/particle-physics-toolkit /bin/bash
```

其中，`ivanfei/particle-physics-toolkit` 是我在 **Docker Hub** 上发布的镜像，您可以在 [Docker Hub](#) 上找到它²。

显然此时您的电脑上并没有这个镜像，所以在保证联网的前提下它会进行下载，并自动开始运行。之后您可以在 **Docker** 图形界面中看到这个镜像以及基于这个镜像的容器，并可以方便地开关和管理。

在容器以镜像为基础运行后，您就可以访问到一个全新的 **Ubuntu 22.04** 系统了，除了三件套之外，您还可以直接将它当作一个 **Ubuntu** 系统来使用，安装其他软件等。

但如果您留心了刚刚那条命令，可能会注意到它只是启动了一个容器，并没有做空间映射。这意味着您在删除容器后，您对容器做的修改将会**全部消失**。为了防止悲剧的发生，建议您浏览[入门教程](#)以熟悉基本的操作以保存您珍贵的文件。

2 （可选，但不建议）在您电脑原生环境安装

2.1 Windows

什么？您想在 **Windows** 上使用三件套？

很遗憾由于软件原因您不能这么做（也许可以？如果喜欢折腾的话可以尝试自行编译），但是您可以安装 **WSL** (**Windows Subsystem for Linux**) 来安装 **Linux** 系统，然后在 **Linux** 系统上安装三件套。对于虚拟机有许多完善的商业化软件，比如 **VMware Workstation** 和 **VirtualBox**，您可以安装他们获得比较用户友好的体验。但是 **Docker**（的其中一个版本）在 **Windows** 上本质上就是 **WSL**，所以不妨考虑一下[章节 1](#)？

如果您很偏爱双系统，那么您可以考虑安装 **Ubuntu** 和 **Windows** 双系统，然后在 **Ubuntu** 系统上安装三件套。好处是您可以在没有什么性能损失的情况下使用 **Linux** 系统，坏处则是每次切换系统您都需要重新启动，并且如果您是初学者，分配磁盘可能存在潜在风险，如不小心抹掉您的 **Windows** 系统。

所以我强烈建议如果您对自己的操作不是很有信心，或不知道您每一步做了什么，在操作前请先备份您的数据。说实话，我建议您在任何时候都应该养成备份的习惯，因为数据无价。相对于硬件损坏，其导致的文件丢失更让人难过，不要问为什么我这么明白:(。而这可以通过您小小地操作极大程度地避免。您可以选择拷贝到外置储存中，也可以使用各种各样的云服务，具体形式取决于您

²您可能浏览过我之前写的安装教程，在那里我指示的镜像是 `ivanfei/particle-physics-pack`。这个镜像目前已经被停用了，新镜像修复了之前占用过大空间的问题，相当于一个更精简更完善的版本。

的预算与习惯。

假设您已经解决了磁盘分配等一系列双系统问题或者您安装好了虚拟机后，可以直接参考章节 2.3。

2.2 MacOS

我在此很高兴的通知您，MacOS 作为一类 Unix 系统，可以以相对原生的方式运行三件套。

安装 Xcode 首先，您需要安装 Xcode，在 App Store 中就可以找到。

安装 Xcode 命令行工具(xcode-select) 编译三件套需要很多的依赖，比如 GCC 和 GFORTRAN，其中一些是由 Xcode 命令行工具提供的（或者如果您足够头铁，也可以在编译过程中查看依赖缺失的错误，并自行亲自一个个安装依赖）。

您只需要在 Mac 终端（terminal.app）里执行

```
$ xcode-select --install
```

安装过程中您可能需要同意苹果公司的条款，如果命令行中出现句子问您是否接受某条款，您可以放心同意³。

安装 Homebrew Homebrew 是一个十分强大的 MacOS 包管理器。您可以把包管理器理解为一个监管应用的工具，可以列出所有您安装的大小程序，用以更新、卸载或安装它们。

为了安装 Homebrew，您需要在终端里执行（可能会比较慢）

```
$ /bin/bash -c "$(curl -fsSL
↳ https://raw.githubusercontent.com/Homebrew/installHEAD/install.sh)"
```

安装结束了以后执行以确认安装成功

```
$ brew doctor
```

（可选）为了加快下载速度可以将 Homebrew 换成清华源：

```
$ git -C "$(brew --repo)" remote set-url origin
↳ https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/brew.git
$ git -C "$(brew --repo homebrew/core)" remote set-url origin
↳ https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-core.git
```

³如果此时您看一下您的磁盘空间，您会发现 Xcode 占了好大一部分。您可能会奇怪为什么需要这么大的一个软件，或者为您失去的空间感到心疼。其中的原因，抛开她很烂、做的很大不谈，是因为她包含了很多软件需要的工具（编译器之类的，这里面有很多的历史渊源），而这是获得这些工具最简单的办法。如果您之后（不得不）用 Xcode 开发苹果公司自己的软件，那它会变的远比现在更加臃肿。所以相信我，不必感到心疼，珍惜现在只有 10G 大小的 Xcode 吧。

```
$ git -C "$(brew --repo homebrew/cask)" remote set-url origin
  https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-cask.git
$ echo 'export
  HOMEBREW_BOTTLE_DOMAIN=https://mirrors.tuna.tsinghua.edu.cn/homebrew-bottles'
  >> ~/.zshrc
$ source ~/.zshrc
$ brew update
```

安装 Conda Conda 是一个包管理器，可以帮助您安装三件套的依赖。您可以在 [Miniconda 官网](#) 或者 [Anaconda 官网](#) 下载安装包，安装包的选择取决于您的 MacOS 版本。除了三件套以外，Conda 还是一个非常强大的 Python 包管理器，可以一站式解决诸如“我 Python 装哪里去了”、“我 Python 版本怎么混起来了”、“我 Python 装了哪些包”等等问题，您可以浏览官网的介绍，这里不多赘述。

安装 ROOT Delphes 的运行依赖 ROOT（按照以往同学的经验，这步往往是最困难的一步）。参考 [ROOT 官网](#)，您可以通过 Conda 安装 ROOT。即在命令行中输入

```
$ conda config --set channel_priority strict
$ conda create -c conda-forge --name <my-environment> root
```

其中 <my-environment> 是您喜欢的环境名称，可以自行指定。如果您不知道该指定什么，可以指定为 root-env。

经过一段时间的等待，您只需每次使用前在命令行中输入

```
$ conda activate <my-environment>
```

就可以开始使用 ROOT 了。

安装 MadGraph5_aMC@NLO 您已经度过了其中最困难的部分，休息一下吧！MadGraph 提供预编译版本（*DJ Airhorn Sound Effect*），所以您只需要在 [MadGraph5_aMC@NLO 官网](#) 下载压缩包（tar.gz 结尾，建议版本 3.4.2），解压到您喜欢的位置就可以了。

安装 Pythia8 以及 Delphes 这步可能会出现一些问题，但是您不用担心，我们会一步步解决。在先启用 ROOT 环境后，您只需在命令行里进入 MadGraph 的解压目录，然后输入

```
$ ./bin/mg5_aMC
```

就可以启动 MadGraph 了。您的终端可能会要求您安装 six 作为依赖，您只需要同意即可。

接着您会看到经典的欢迎界面，并且您的命令提示符会变成

```
MG5_aMC>
```

代表您已经在使用 MadGraph 了。您可以随时通过摁下 `Ctrl + D`（注意是 `Ctrl` 而非 `Command`）退出 MadGraph。

MadGraph 采用 Python 作为脚本语言，故您只需要输入

```
> install pythia8
```

和

```
> install Delphes
```

来安装 Pythia8 与 Delphes 即可。这两个过程可能会花费一些时间。

您可能会在此期间遇到编译错误。（待续）

假设您的电脑机魂足够强大，您不会遇到任何问题，那么恭喜您！您已经完成了所有的安装过程，可以开始您的快乐粒子物理了！

MacOS FAQ（建议浏览一遍）（也待续，对不起）

- 我在 launch 过程中出现编译错误，怎么办？

2.3 Linux（待续）(Ubuntu 22.04 Jammy Jellyfish)

如果您选择以此方式安装，恭喜您选择了一个最为平坦同时也最为坎坷的路。

本部分基于 Ubuntu 22.04⁴，如果有偏好也可以选择其他发行版，但需要您自行参考其他文档。

⁴有趣的事实：每一版 Ubuntu 都按照一个形容词以及一个小动物的名称组成，并且首字母一致。版本号也不遵守通常的规则，而是采用年. 月份的规则，为了加以区分使用了 22.04 而非 22.4。