

第一章 OpenMC 程序的编译与安装

本章将介绍编译、安装 OpenMC (0.10.0) 的两种方法。第一种方法，简称：conda 方法，安装相对容易。另一种使用源码进行编译与安装，需涉及编译环境的搭建。

本文选择 Ubuntu 18.04 进行程序的安装。因为 linux 系统小巧、功能强大，并且拥有齐全的编译工具，方便开展工作；在程序运行效率上，优势比 Windows 明显，可以节省程序运行时间。另外，linux 发行版很多，可选择自己喜欢的版本，不过本文建议新手选择 Ubuntu 系统，因为它的高级打包工具 (apt) 在安装编译工具时，可以省下不少精力。其他版本如 Centos，由于 yum 下载的 OpenMC 编译工具版本太低使得程序无法正常编译，需要先编译、安装高版本编译工具，对新手十分不方便。

3.1 编译方法介绍

3.1.1 Conda-forge 安装方法

OpenMC 代码保存在 conda-forge，conda-forge 类似一个云盘，当系统拥有 conda 环境，即可安装 OpenMC。Conda 是一个开源软件包管理系统和环境管理系统，它包含在所有版本的 Anaconda。Anaconda 是一个免费开源的 Python 发行版，拥有全面的 Python 库(library)和相关软件，方便用户使用。由于 OpenMC 程序需要依赖 Python 工具和它的模块来处理数据结果，为此需要安装 Anaconda，再来安装 OpenMC。

由于 Anaconda 版本的更新速度较快，新版本在运行 OpenMC 时候，会出现许多问题，比如 OpenMC 最高支持 Python 3.7.1,当安装 Python 3.7.3 或者更高版本时候，Python 可能无法导入 OpenMC 了。另外，Anaconda 版本比较低时，需根据 Python 版本安装 conda-forge 对应 Python 版本的 OpenMC。

本文选择 anaconda3-5.3.1-Linux-x86_64.sh 的 Anaconda 版本。

第一步，下载 Anaconda 软件包：<https://repo.continuum.io/archive>

第二步，按图 3.1 步骤进行 Anaconda 安装，若需设置 conda 环境变量请编辑默认目录下的.bashrc 文件：

```
gedit ~/.bashrc
export PATH="/your_path_to/anaconda3/bin:$PATH"
source ~/.bashrc
```

第三步，添加 conda-forge 频道，并安装 OpenMC：

```
conda config --add channels conda-forge
```

```
conda install openmc
```

如需下载其他版本，可参考如下指令，图 3.2 显示 conda-forge 不同的 OpenMC 版本：

```
conda search openmc --channel conda-forge
```

```
conda install openmc=0.10.0=py37h1b5bbce_1002
```

Installing on Linux

1. Download the installer:

- [Miniconda installer for Linux.](#)
- [Anaconda installer for Linux.](#)

2. In your terminal window, run:

- Miniconda:

```
bash Miniconda3-latest-Linux-x86_64.sh
```

- Anaconda:

```
bash Anaconda3-latest-Linux-x86_64.sh
```

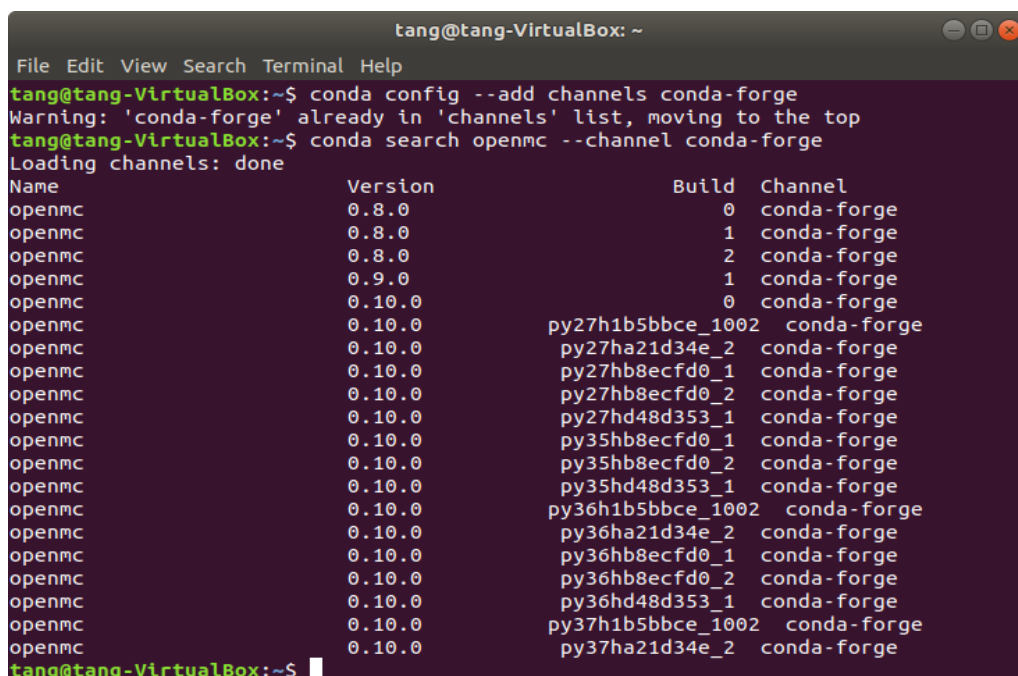
3. Follow the prompts on the installer screens.

If you are unsure about any setting, accept the defaults. You can change them later.

4. To make the changes take effect, close and then re-open your terminal window.

5. Test your installation.

图 3.1. Anaconda 的安装步骤



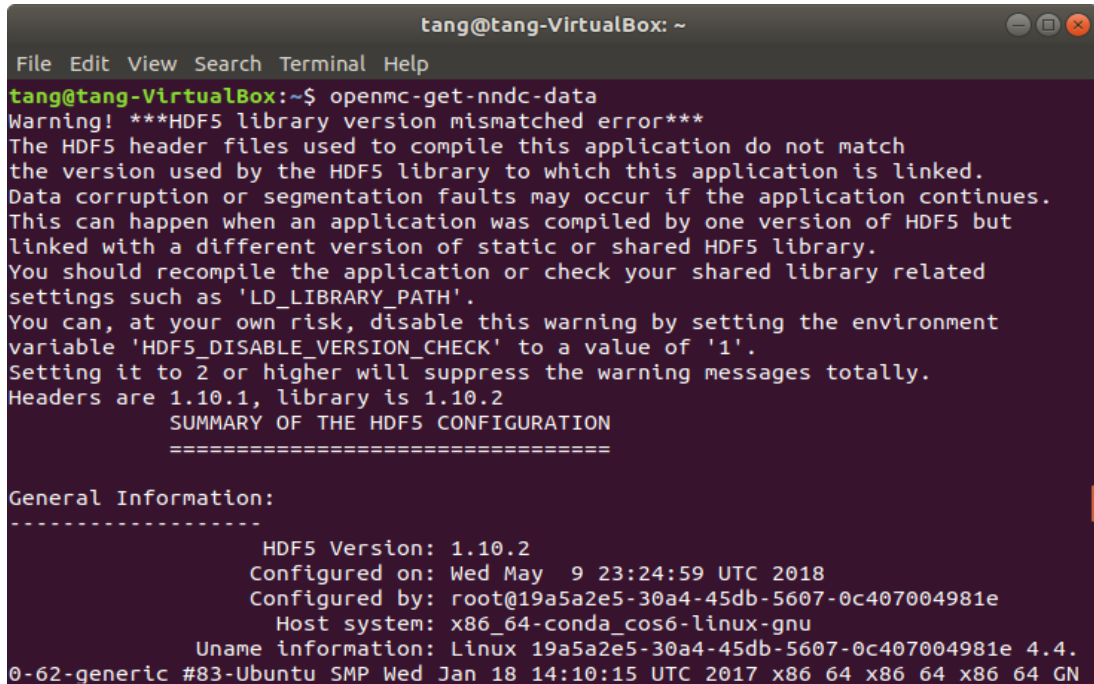
```
tang@tang-VirtualBox: ~
File Edit View Search Terminal Help
tang@tang-VirtualBox:~$ conda config --add channels conda-forge
Warning: 'conda-forge' already in 'channels' list, moving to the top
tang@tang-VirtualBox:~$ conda search openmc --channel conda-forge
Loading channels: done
Name                               Version          Build      Channel
openmc                             0.8.0            0          conda-forge
openmc                             0.8.0            1          conda-forge
openmc                             0.8.0            2          conda-forge
openmc                             0.9.0            1          conda-forge
openmc                             0.10.0           0          conda-forge
openmc                             0.10.0           py27h1b5bbce_1002  conda-forge
openmc                             0.10.0           py27ha21d34e_2     conda-forge
openmc                             0.10.0           py27hb8ecfd0_1     conda-forge
openmc                             0.10.0           py27hb8ecfd0_2     conda-forge
openmc                             0.10.0           py27hd48d353_1     conda-forge
openmc                             0.10.0           py35hb8ecfd0_1     conda-forge
openmc                             0.10.0           py35hb8ecfd0_2     conda-forge
openmc                             0.10.0           py35hd48d353_1     conda-forge
openmc                             0.10.0           py36h1b5bbce_1002  conda-forge
openmc                             0.10.0           py36ha21d34e_2     conda-forge
openmc                             0.10.0           py36hb8ecfd0_1     conda-forge
openmc                             0.10.0           py36hb8ecfd0_2     conda-forge
openmc                             0.10.0           py36hd48d353_1     conda-forge
openmc                             0.10.0           py37h1b5bbce_1002  conda-forge
openmc                             0.10.0           py37ha21d34e_2     conda-forge
tang@tang-VirtualBox:~$
```

图 3.2. 查阅 conda-forge 不同的 OpenMC 版本

3.1.2 源码安装方法

OpenMC 源码编译前，需安装编译工具：fortran、c++、CMake，还有 git 工具。同时，因前面提过 OpenMC 处理数据需借助 Python，因此需要有 Python 环境，即安装 Anaconda。

注意：Anaconda 版本选择 anaconda3-5.1.0-Linux-x86_64.sh，该版本自带 hdf5=1.10.1，可避免图 3.3 出现的错误，如需高版本 hdf5，请按照提示修改源



```
tang@tang-VirtualBox: ~
File Edit View Search Terminal Help
tang@tang-VirtualBox:~$ openmc-get-nndc-data
Warning! ***HDF5 library version mismatched error***
The HDF5 header files used to compile this application do not match
the version used by the HDF5 library to which this application is linked.
Data corruption or segmentation faults may occur if the application continues.
This can happen when an application was compiled by one version of HDF5 but
linked with a different version of static or shared HDF5 library.
You should recompile the application or check your shared library related
settings such as 'LD_LIBRARY_PATH'.
You can, at your own risk, disable this warning by setting the environment
variable 'HDF5_DISABLE_VERSION_CHECK' to a value of '1'.
Setting it to 2 or higher will suppress the warning messages totally.
Headers are 1.10.1, library is 1.10.2
SUMMARY OF THE HDF5 CONFIGURATION
=====

General Information:
-----
HDF5 Version: 1.10.2
Configured on: Wed May 9 23:24:59 UTC 2018
Configured by: root@19a5a2e5-30a4-45db-5607-0c407004981e
Host system: x86_64-conda_cos6-linux-gnu
Uname information: Linux 19a5a2e5-30a4-45db-5607-0c407004981e 4.4.0-62-generic #83-Ubuntu SMP Wed Jan 18 14:10:15 UTC 2017 x86_64 x86_64 x86_64 GN
```

码。

图 3.3. Hdf5 version >1.10.1 的 OpenMC 运行错误

下面进行安装演示：

第一步，安装 anaconda3-5.1.0-Linux-x86_64.sh 的 Anaconda，

第二步，安装 fortran、c++、CMake 和 git：

```
sudo apt install gfortran g++ cmake git
```

第三步，clone 源码，进行编译与安装，图 3.4 显示 OpenMC 已安装成功。注意：OpenMC 官网有稳定版和开发版，二者代码有点出入，本文选择稳定版：

```
git clone https://github.com/mit-crpg/openmc.git
cd openmc                                进入这个目录
pip install .                            安装 OpenMC 的 Python 模块
mkdir build && cd build                  创建并进入 build 目录
cmake ..                                  CMake 搭建编译环境
make                                       编译
```













sudo make install 安装 OpenMC


```
tang@tang-VirtualBox:~/openmc/build$ openmc
ERROR: Settings XML file 'settings.xml' does not exist! In order to run
OpenMC, you first need a set of input files; at a minimum, this includes
settings.xml, geometry.xml, and materials.xml. Please consult the user's
guide at http://openmc.readthedocs.io for further information.
ERROR STOP
tang@tang-VirtualBox:~/openmc/build$
```

图 3.4. OpenMC 程序运行后，报“缺少输入卡错误”

3.2 数据库安装方法

为了能够运行 OpenMC，需拥有横截面数据库（cross sections）。OpenMC 采用 HDF5 格式储存核数据。注意：由于开发人员保罗在 2019 年 1 月份对程序源码做了修改，获取核数据库的脚本被放置 Github 的 openmc-dev/data 仓库里，如图 3.5 所示，若采用源码安装 OpenMC，可考虑下载脚本获取截面库数据。假如拥有 ACE 格式的数据文件，可使用 openmc-ace-to-hdf5 指令其转化

 .gitignore	Update conversion script for ENDF/B-VII.0 data from MCNP	3 months ago
 README.md	added the generate word to readme	3 months ago
 convert_fendl.py	sorted imports	3 months ago
 convert_jeff32.py	removed unused blocksize code	3 months ago
 convert_lib80x.py	Update convert_mcn71.py and convert_lib80x.py scripts	3 months ago
 convert_mcn70.py	Update conversion script for ENDF/B-VII.0 data from MCNP	3 months ago
 convert_mcn71.py	Update convert_mcn71.py and convert_lib80x.py scripts	3 months ago
 convert_nndc71.py	removed unused blocksize code	3 months ago
 convert_tendl.py	sorted imports	3 months ago
 generate_cendl.py	tidied up cendl script and removed accidental .sh script	2 months ago
 generate_endf71.py	sorted imports	3 months ago
 generate_jendl.py	added file size to jendl	3 months ago

 README.md

OpenMC Data Scripts

This repository contains a collection of scripts for generating HDF5 data libraries that can be used with OpenMC. Some of these scripts convert existing ACE libraries (such as those produced by LANL) whereas generate scripts use NUJOY to process ENDF files directly. Note that unless you are interested in making a customized library, you can find pregenerated HDF5 libraries at <https://openmcmcs.anl.gov>.

为.h5 的 hdf5 格式文件。

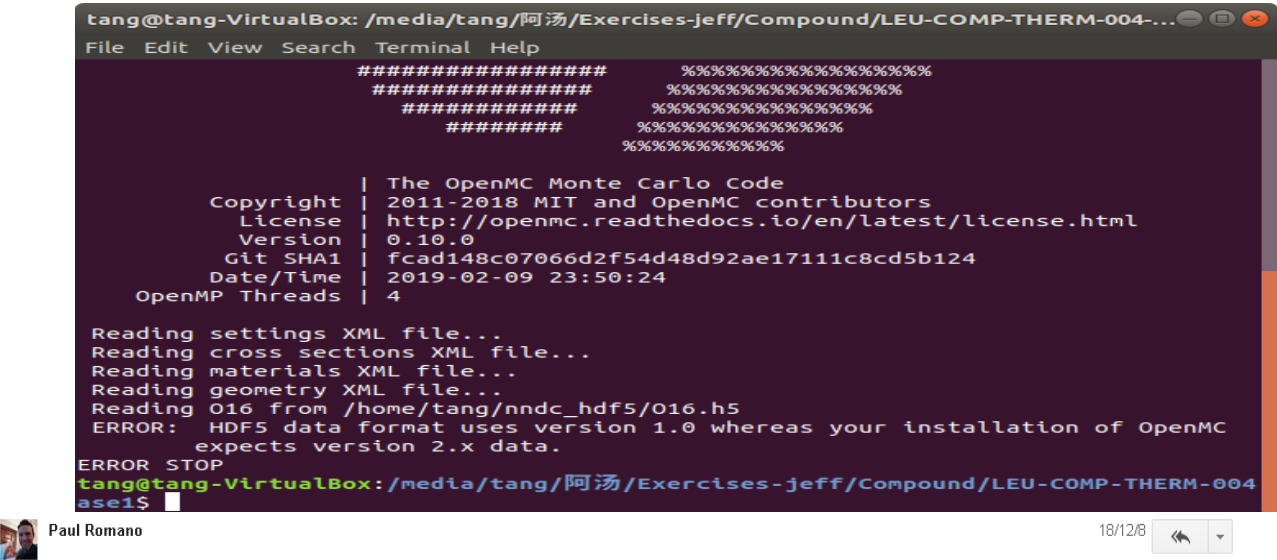
图 3.5. openmc-dev/data 脚本仓库

3.2.1 数据库版本

当核数据的版本与 OpenMC 程序版本不匹配时，可能出现图 3.6 的错误提示：“OpenMC 需要 v2 版的数据，但 HDF5 数据格式是 v1 版”，或者意思相反。这是因为 OpenMC(0.10.0)刚发布时，代码未拥有光子输运的功能，之后开发团队在 Github 上开发新的代码完善这一功能，然而存放在 conda-forg 的 OpenMC 代码，开发团队未进行代码升级。所以 OpenMC（0.10.0）具有两个版

本，不支持光子输运的为 v1 版，支持的为 v2 版。一般从 conda-forg 安装的 (0.10.0) 版本是 v1 版，因此需要 v1 版的核数据；选则源码编译、安装，需要 v2 版核数据。具体原因，请看图 3.7 保罗的解释^[24]。

图 3.6. 核数据版本不匹配



★ 其他收件人: drshi...@gmail.com

按帖子翻译为中文

Hi Jon,

Since our last released version, 0.10.0, the format for data files has changed to accommodate the new photon transport capability in OpenMC and the ability to run coupled neutron-photon calculations. In particular, reactions in the HDF5 files are now marked to indicate whether they are "redundant". Sometimes, it is necessary to include reactions that are otherwise defined as the sum of existing reactions because photon production data may be attached to the redundant summed reaction. A common situation is to see photon production attached to MT=3, which would otherwise not be listed in a file. Thus, with the change in the format, v1 format files can only be used with version 0.10.0 and prior, and v2 files can only be used with the current develop branch and whatever the future release will be. If you want the NNDC data in the new format, here is the tarball:

<https://anl.box.com/s/snrluuy79o2ffpvpng9bsuis5k1811d>

图 3.7. v1 与 v2 版解释

3.2.2 ENDF/B-VII.1 数据库

ENDF / B-VII.1 是由美国的 NNDC^[25] (National Nuclear Data Center) 研究中心制作，该库含有中子反应和热散射库，其中中子反应库仅在 293.6k 的温度下生成，而热散射库生成范围在 19k-296k，因此反应材料温度若大于 320k，则不能使用该截面库。

3.2.2.1 Conda 安装方法

注意：conda-forg 的 OpenMC 为 v1 版，获取 ENDF/B-VII.1 的脚本未删除，因此可直接获取：

openmc-get-nndc-data

该指令会帮自动下载 ENDF/B-VII.1 数据库，然后自动合成 hdf5 格式的文件库，时间可能 1 个小时。若不想浪费时间，可在保罗分享的链接如图 3.8 下载 v1 版的 ENDF/BVII.1 数据库，打开需要代理：

<https://anl.app.box.com/s/a6sw2cep34wlz6b9i9jwiotaqoayxcxt>

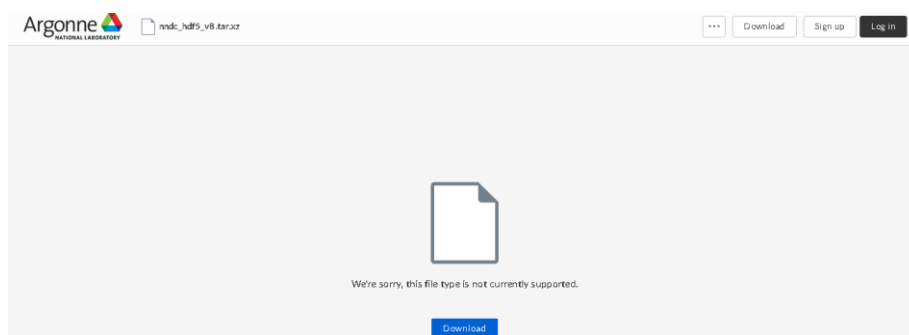


图 3.8. ENDF_v1 版

3.2.2.2 源码安装方法

源码方法安装的 OpenMC 是通常是 v2 版，但也有另外情况。只要在 OpenMC 源码托管仓库的 release 下载（0.10.0）源码包进行编译安装，会得到 v1 版，因为压缩包的代码未更新。由于更新后的代码移除了获取核数据库的脚本（release 源码包未删除），需个人下载获取脚本，仓库位于 Github 的 openmc-dev/data：

```
python convert_nndc71.py
```

可通过以下网站，下载 HDF5 格式的 v2 版 ENDF/B-VII.1 库，下载需要代理：

<https://anl.app.box.com/s/snrluuy79o2fffvpvpng9bsuis5kl811d>

或访问 OpenMC 预合成横截面库网站下载，如图 3.9，下载需要代理：

<https://openmc.mcs.anl.gov/lanl-data-libraries/>

3.2.3 JEFF3.2 数据库

JEFF 库是核能组织 NEA^[26]（The Nuclear Energy Agency）的一个项目，该库提供了 12 种不同温度的中子反应库以及不同温度下的热散射库。

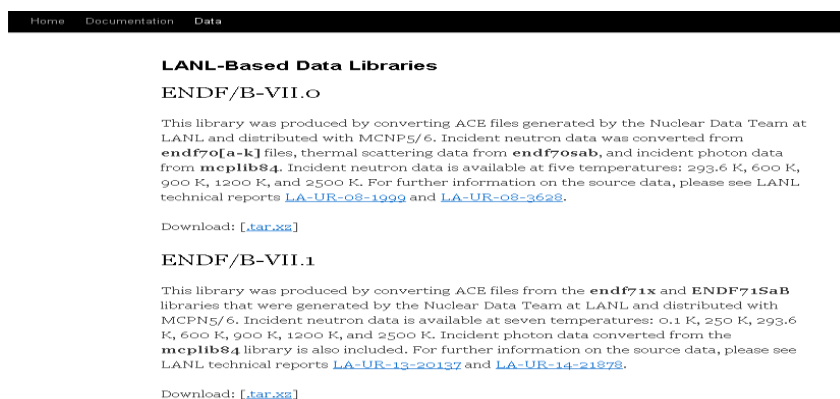


图 3.9. OpenMC 预合成横截面库

注意：整个库的压缩文件高达 9GB，生成 OpenMC 支持的 HDF5 格式文件，系统需要准备大于 40GB 的存储空间。

由于访问 jeff 库的链接被 Great Fire Wall 屏蔽，因此正常无法访问网页，需借助代理。

若采用 conda 方法安装 OpenMC，只需命令行模式，输入：

`openmc-get-jeff-data`

注意：该方法只安装 v1 版核数据。由于数据量大、网络状态不稳定，可能使下载的数据包丢失。解决办法，请访问本文的教程网站，如图 3.10：

<https://github.com/t2015q/OpenMC-Users-Guide>

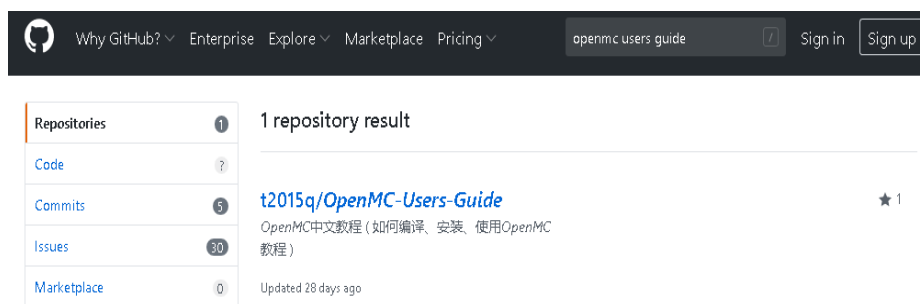


图 3.10. OpenMC-Users-Guide 仓库

需要 v2 版的核数据，只需访问预合成横截面库网站：

<https://openmc.mcs.anl.gov/lanl-data-libraries/>

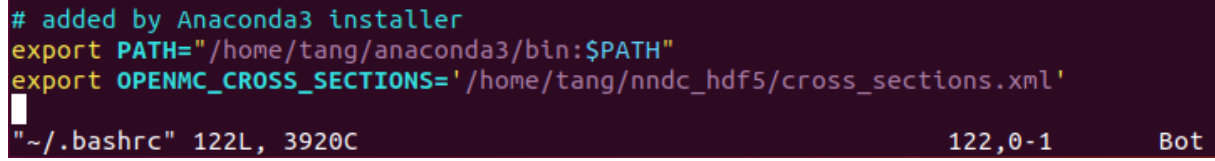
3.2.4 路径设置方法

数据库安装完毕，但 bash 未设置 OPENMC_CROSS_SECTIONS 的环境变量，OpenMC 不知到哪里读取核数据文件，也就无法准确划分天然核素的种类。因此，参考图 3.11(a,b)，vi 编辑器对 .bashrc 文件进行设置：

`vi ~/.bashrc`

`export OPENMC_CROSS_SECTIONS="/ your_path_to/cross_sections.xml"`

`source ~/.bashrc`

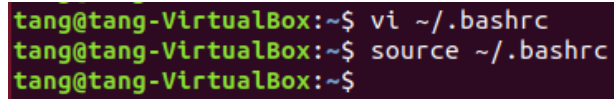


A terminal window showing the content of the `~/.bashrc` file. The text is as follows:

```
# added by Anaconda3 installer
export PATH="/home/tang/anaconda3/bin:$PATH"
export OPENMC_CROSS_SECTIONS='/home/tang/nndc_hdf5/cross_sections.xml'
```

At the bottom of the terminal, there is a status bar with the text: `"~/.bashrc" 122L, 3920C` on the left, `122,0-1` in the center, and `Bot` on the right.

(a) 图 3.11. `.bashrc` 设置横截面环境变量



A terminal window showing the execution of two commands:

```
tang@tang-VirtualBox:~$ vi ~/.bashrc
tang@tang-VirtualBox:~$ source ~/.bashrc
tang@tang-VirtualBox:~$
```

(b) 图 3.11. `vi` 编辑、`source` 执行 `.bashrc` 文件