目录

[摘要 II](#_Toc23698)

[引言 II](#_Toc17070)

[A. 天体视半径的变化规律 III](#_Toc28594)

[B. 遮挡率 III](#_Toc11595)

[第一章 月球对宇宙线的遮挡作用 1](#_Toc16000)

[1.1 月球基本参数的变化 1](#_Toc9488)

[1.2 月球磁场分析 1](#_Toc31300)

[第二章 行星对宇宙线的遮挡作用 1](#_Toc6576)

[2.1 月球视半径、天顶角和遮挡率随时间的变化 2](#_Toc28995)

[2.2 太阳系行星大小分布 2](#_Toc20532)

[2.3 行星参数分析 3](#_Toc11480)

[2.4 各表之间的联系图 3](#_Toc27442)

[2.36 天王星参数分析 4](#_Toc12610)

[第三章 小行星及彗星对宇宙线的遮挡作用分析 5](#_Toc8685)

[3.1 小行星的分类和活动规律 5](#_Toc29894)

[3.2 小行星的参数分析 5](#_Toc1374)

[3.6 读者信息管理模块 8](#_Toc4385)

[3.7 图书订购管理模块 10](#_Toc27236)

[3.8 图书借阅管理模块 13](#_Toc2405)

[3.9 系统维护模块 17](#_Toc2746)

[第四章 系统测试 21](#_Toc4001)

[4.1 测试项目 21](#_Toc14951)

[4.2 测试用例 21](#_Toc20757)

[结束语 23](#_Toc15799)

[参考文献 24](#_Toc17601)

# 摘要

宇宙线是来自宇宙的高能粒子[1]，是天体过程的一种间接表现，对宇宙线的研究都对我们了解天体的形成和演化有着至关重要的作用。但是宇宙线并不是总能顺利地到达地球并被人类观测到，影响宇宙线到达人类可观测区域内的诸多因素中，天体的遮挡作用占其中很大的比重。

关键词：宇宙线，月影，遮挡率

# 引言

一般来说，宇宙线是来自宇宙的高能粒子[1]，也是人类能获得的来自太阳系外的直接样本[2]。每秒钟每平方米的面积上大约有1000多个粒子轰击到大气层上[3]。而且宇宙线是天体过程的一种直观的表现，对宇宙线的研究工作可以追溯到1929年. 每秒钟每平方米的面积上约有1000个宇宙线粒子轰击到地球大气层上[1].但是宇宙线并不是总能顺利地到达地球并被人类观测到，影响宇宙线到达人类可观测区域内的诸多因素中，天体的遮挡作用占其中很大的比重。

在这当中，日影和月影由于对宇宙线的遮挡作用比较明显，因此自人们开始注意到天体对宇宙线的遮挡作用时，日影和月影变成了被首要研究的对象，于是我们通常可以通过对月影和日影的对比，来估计不同天体对宇宙线遮挡作用的大小。

月影和日影已经经过了天文学家的多次讨论。本文的目的是仿照月影作用的分析和计算，对太阳系中的行星进行分析，通过将分析结果和月影作用进行对比，指出相对于月影，太阳系中的行星对宇宙线的能否被忽略。

在一些工作开始之前，我们先介绍一些概念性的知识。

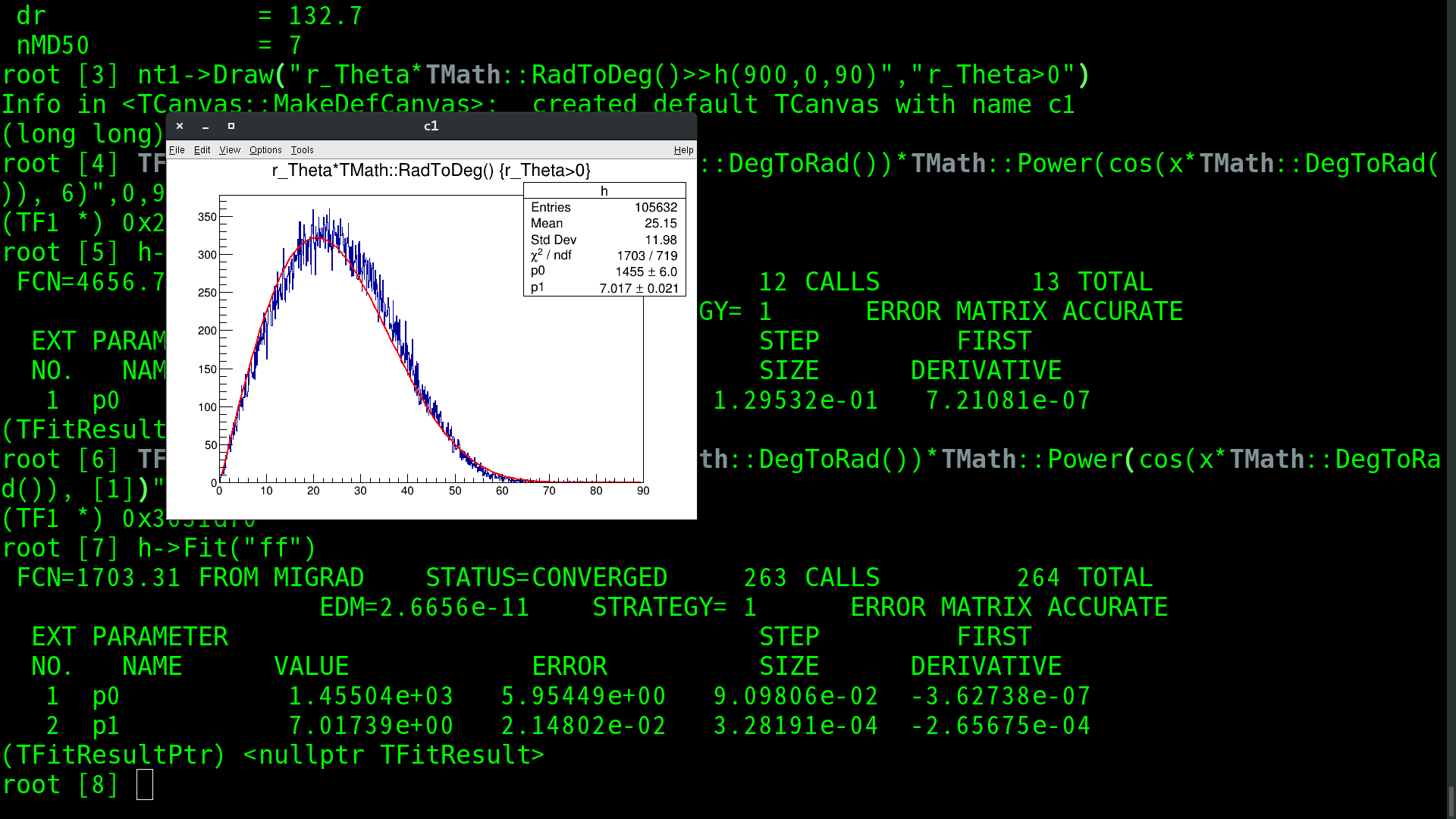
## 天体视半径的变化规律

天体的视半径是指观测者的眼睛到天体中心的连线与观测者的眼睛到天体切线所形成的夹角度数（本文为了能和[2]中保持统一，在对月球的分析阶段使用角度制，其他地方统一使用弧度制）.但是由于天体运动产生影响，天体的视半径往往是一个动态的值，在以下分析中，本文认为在100s的时间内，天体的视半径是保持不变的（具体原因由以下分析得到）.

## 遮挡率

定义某个天体对宇宙线的遮挡率

其中，r为天体的视半径，为测量时间隔（本文认为在此时间间隔内的天体视半径是定值）， 为探测阵列的探测效率. （为了简化计算，本文对于探测率的计算并没有考虑阵列的探测面积）. 的形式，由以下拟合过程得到：



图a 探测效率和月球天顶角的拟合过程

可以看到拟合给出的指数为.为了简化之后的计算过程，这里取指数值为7，可以得到探测效率的解析式为

因此，我们可以得到遮挡率为：

## PyEphem程序包

本文的一切天文计算、数据处理均通过PyEphem进行，PyEphem是python的一个程序包，对于一般的天文计算，该程序包拥有很高的精度。它的主要功能包括：

# 第一章 月球对宇宙线的遮挡作用

## 1.1 月球基本参数及其变化

月球是地球唯一的卫星，也是离我们最近的天体，图1.1是月球视半径变化规律，在大约280个儒略日期间，筛选条件为天顶角小于40°，观测站坐标为90.52°E，30.11°N，海拔4300米，月球的最大视半径为0.29°，视半径均值为0.26°。

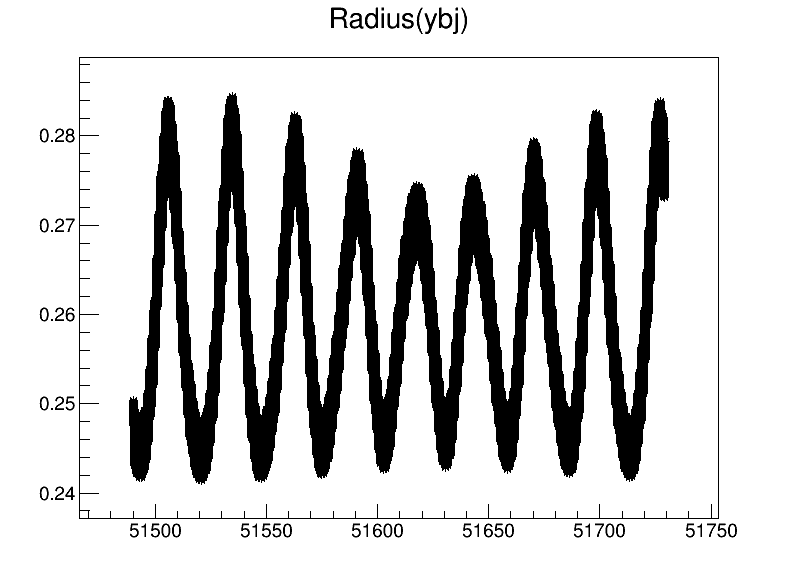


图1.1 月球视半径随时间变化规律

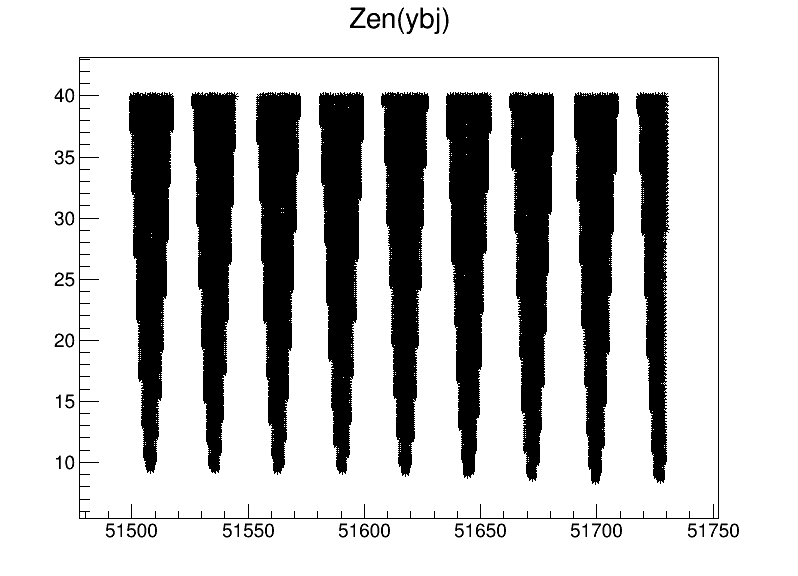


图1.2 月球天顶角随时间的变化

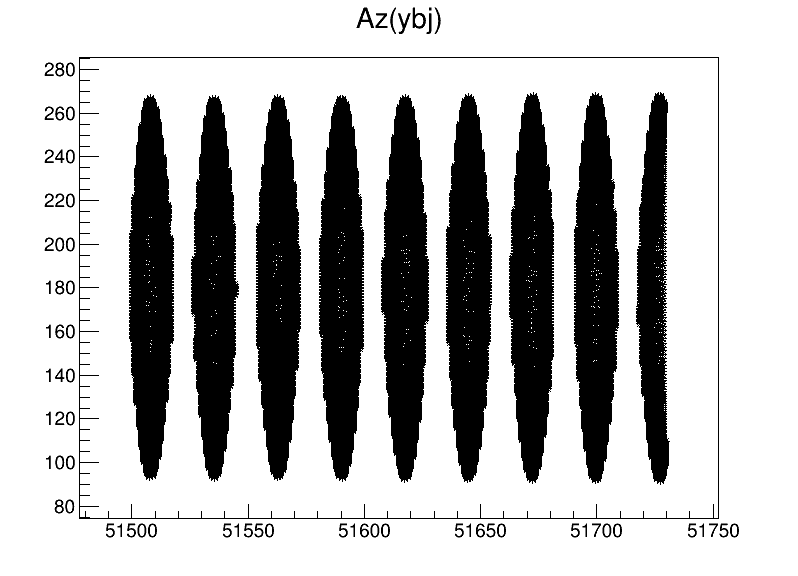


图1.3 月球方位角随时间的变化

从以上图表我们可以看出，月球的视半径、天顶角和方位角都是根据时间变化而变的动态值.

## 1.2 月球遮挡率分析

根据1.1中的参数变化，我们可以得到一个衡量天体遮挡宇宙线程度的标准, 经过计算，我们可以得到，月球对宇宙线的遮挡率的值为207.120064025 。

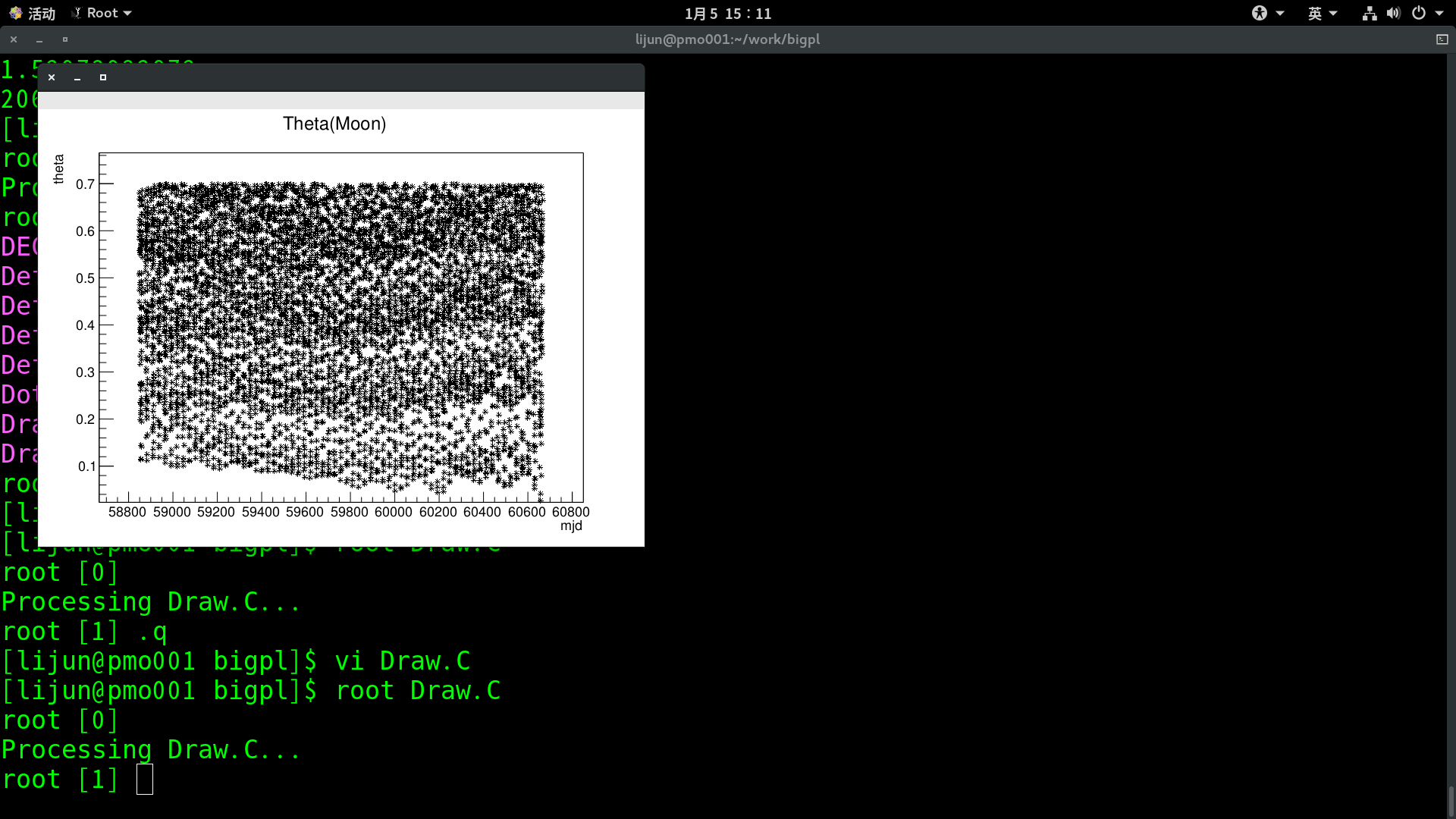
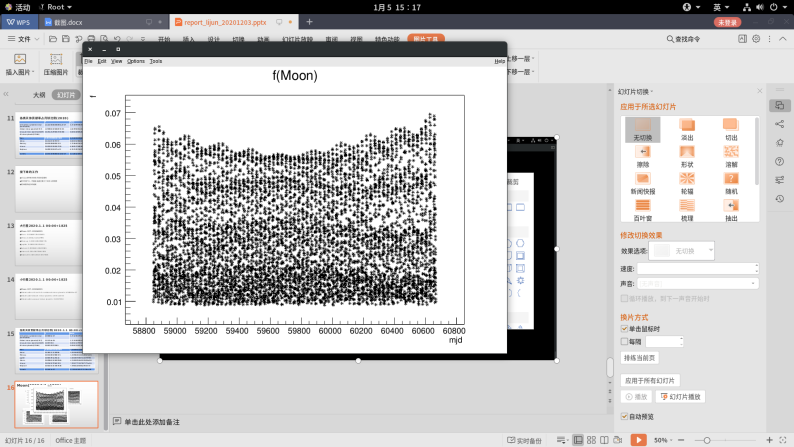
# 第二章 行星对宇宙线的遮挡作用

根据以上分析，我们可以知道月亮对宇宙线的遮挡情况。用同样的方法，更改数据为2020年1月1日之后五年的数据（）我们可以得到太阳系其他行星的视半径、天顶角、以及其他参数的变化情况。

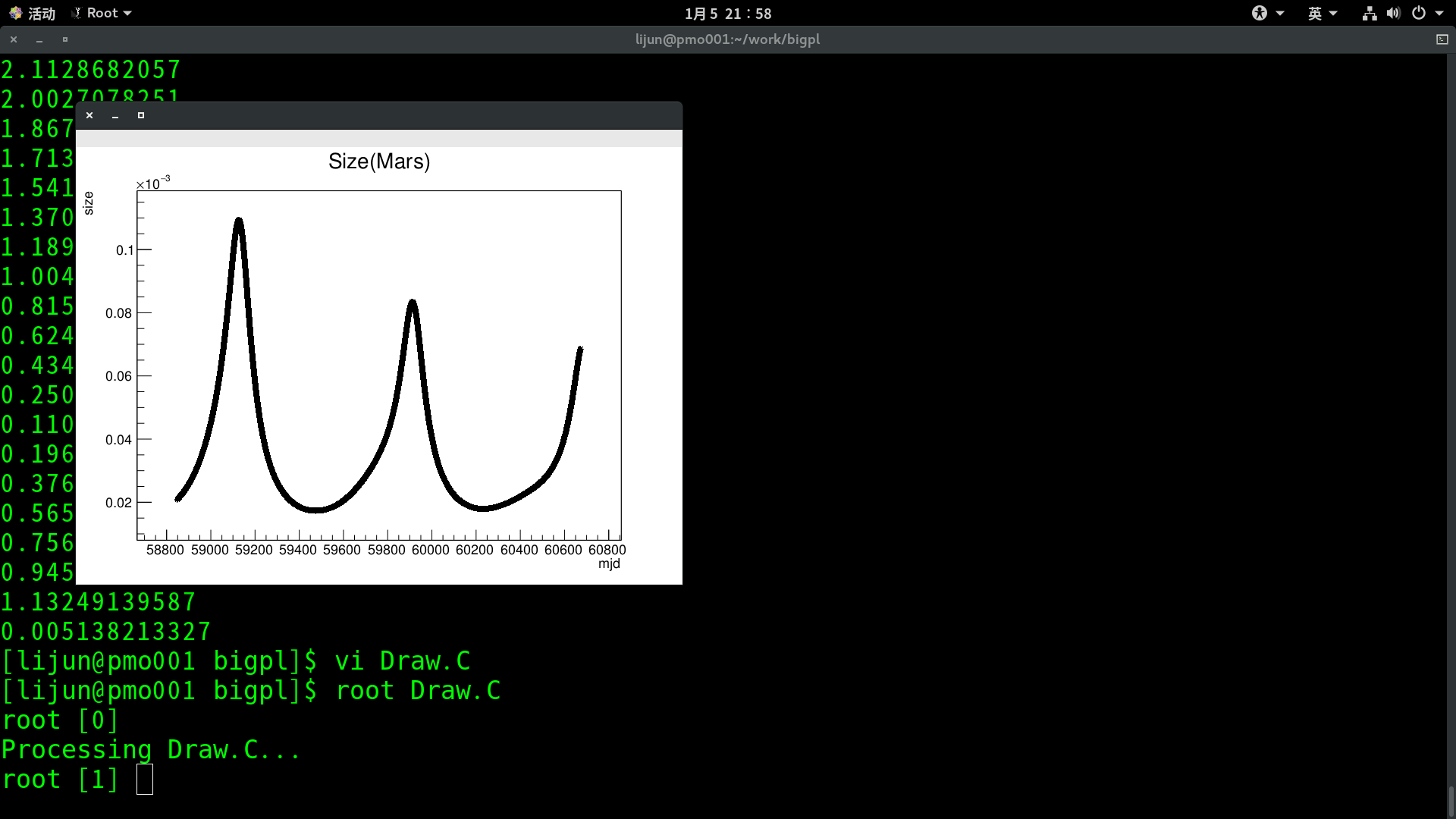
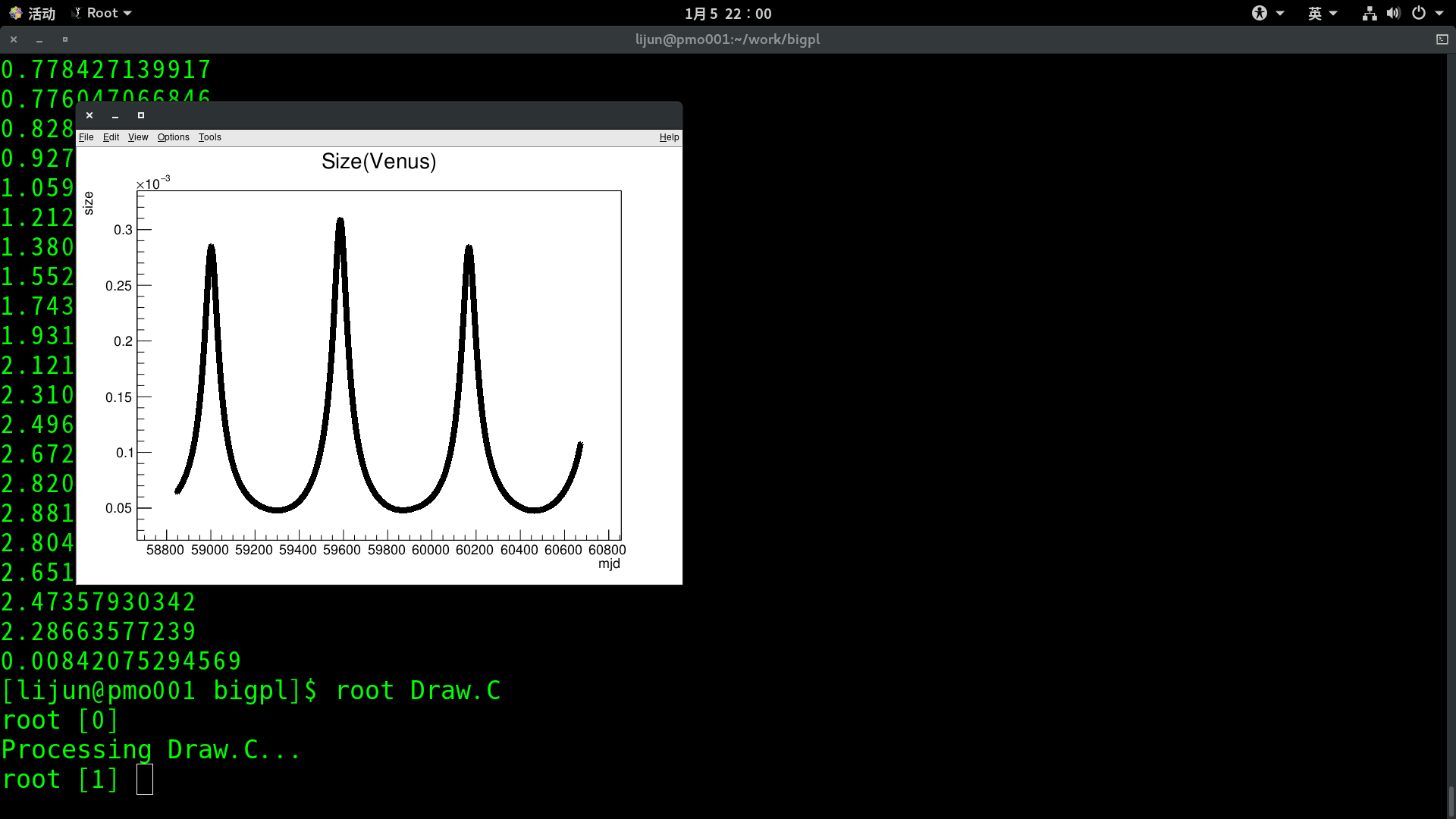
## 2.1 月球视半径、天顶角和遮挡率随时间的变化

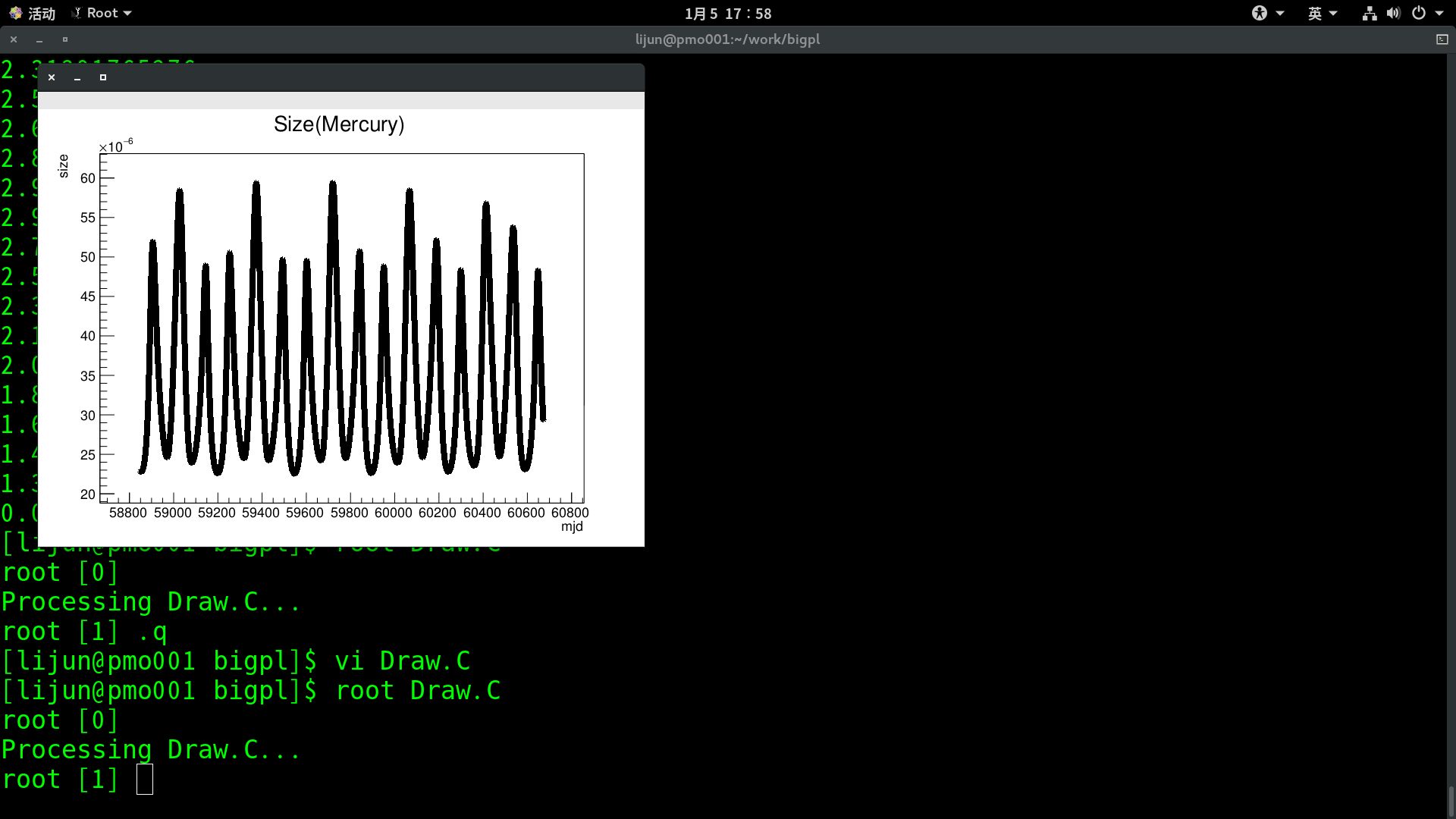
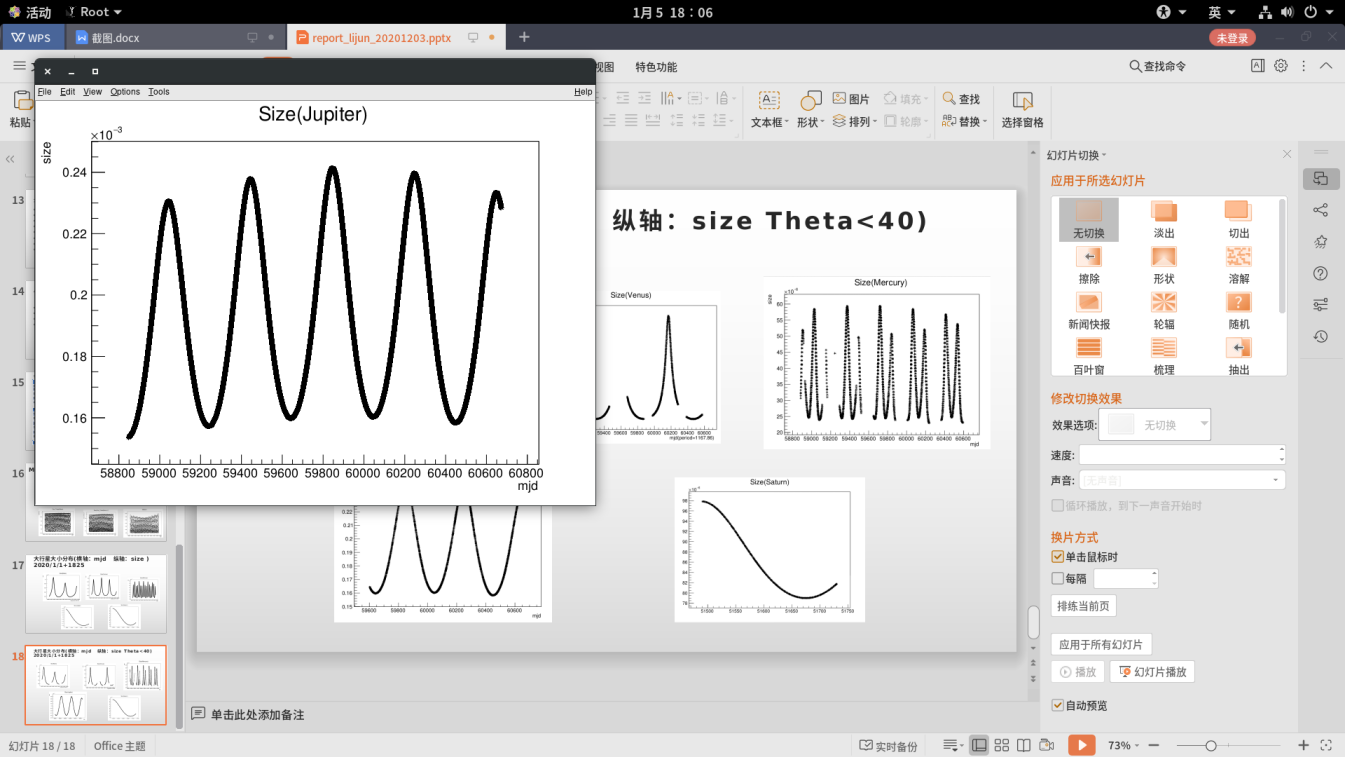


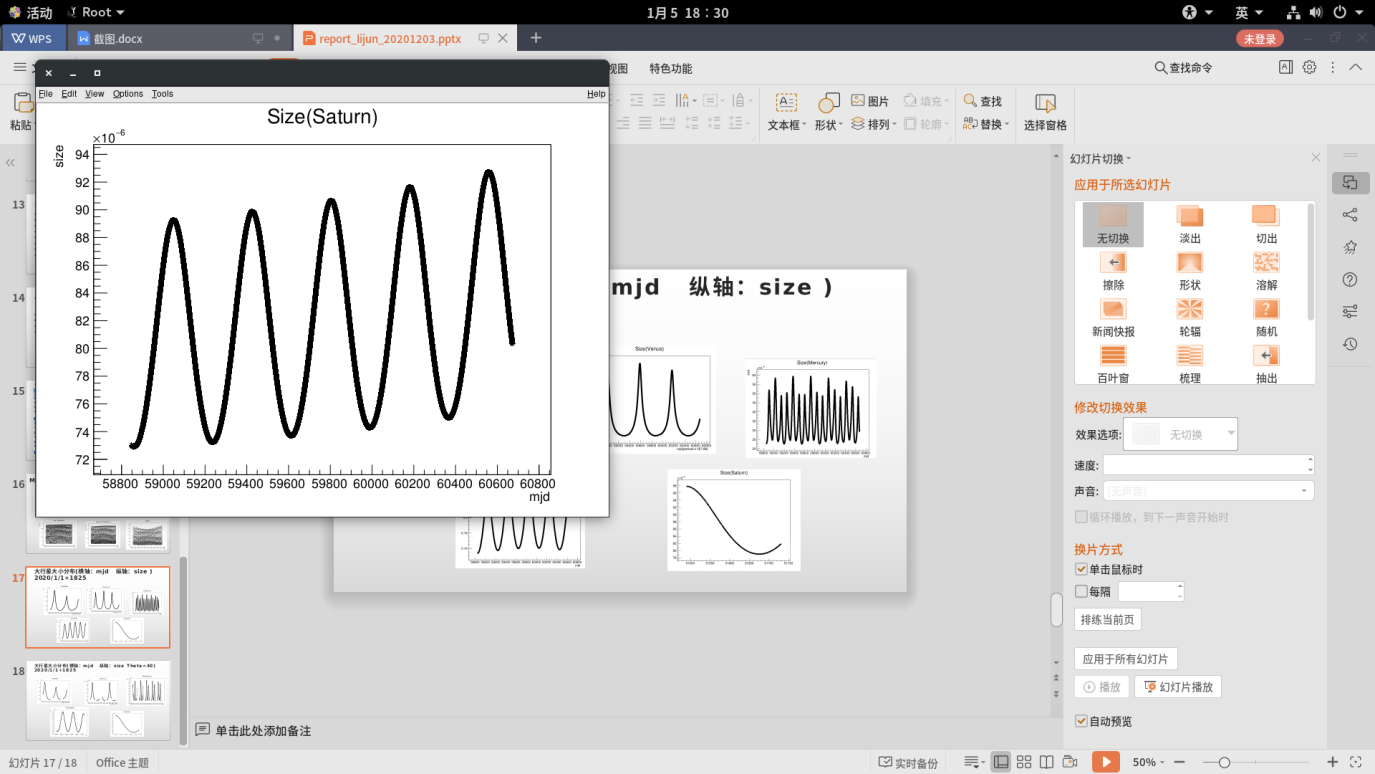
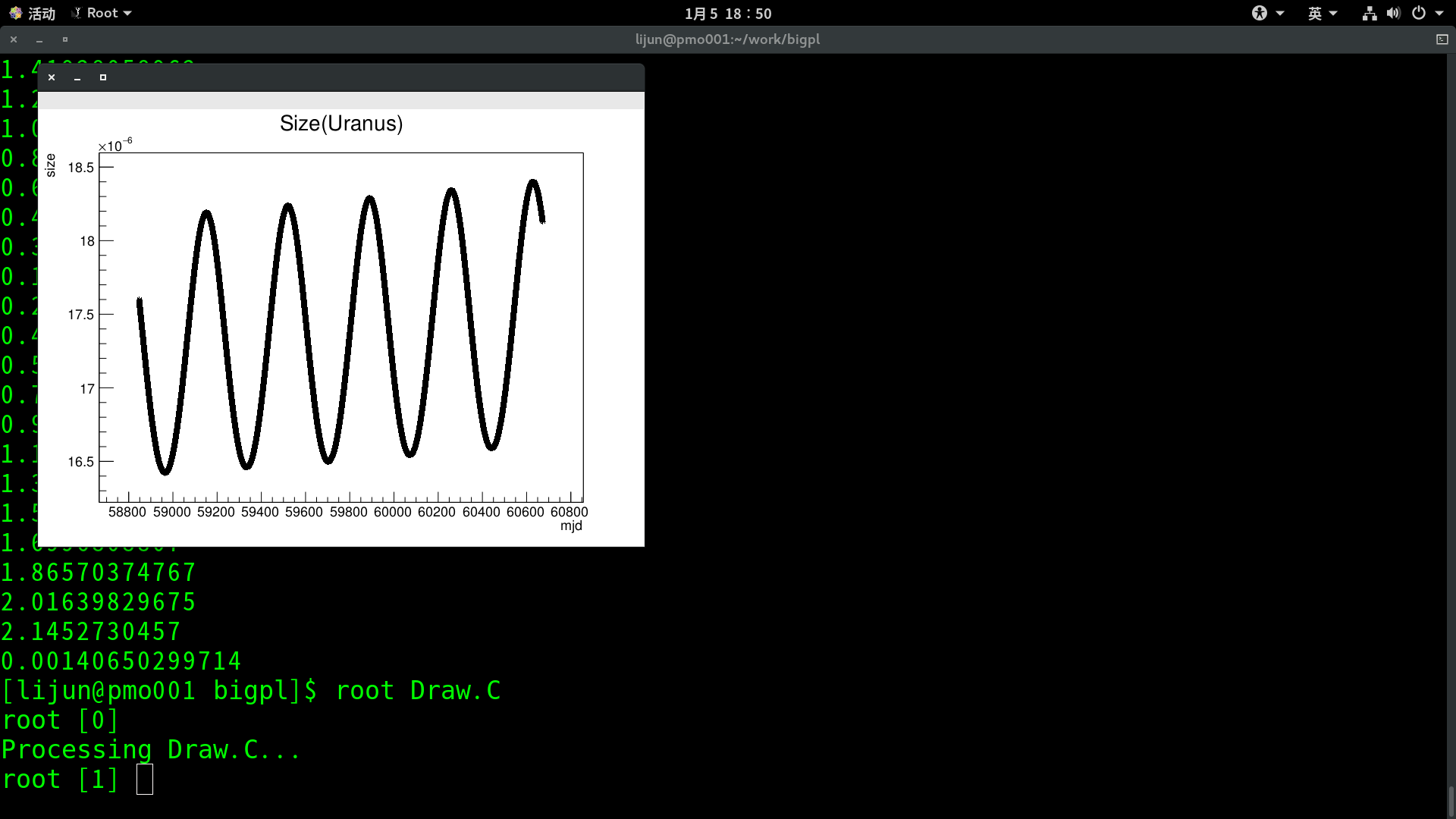
图2.1 月球方位角随时间的变化

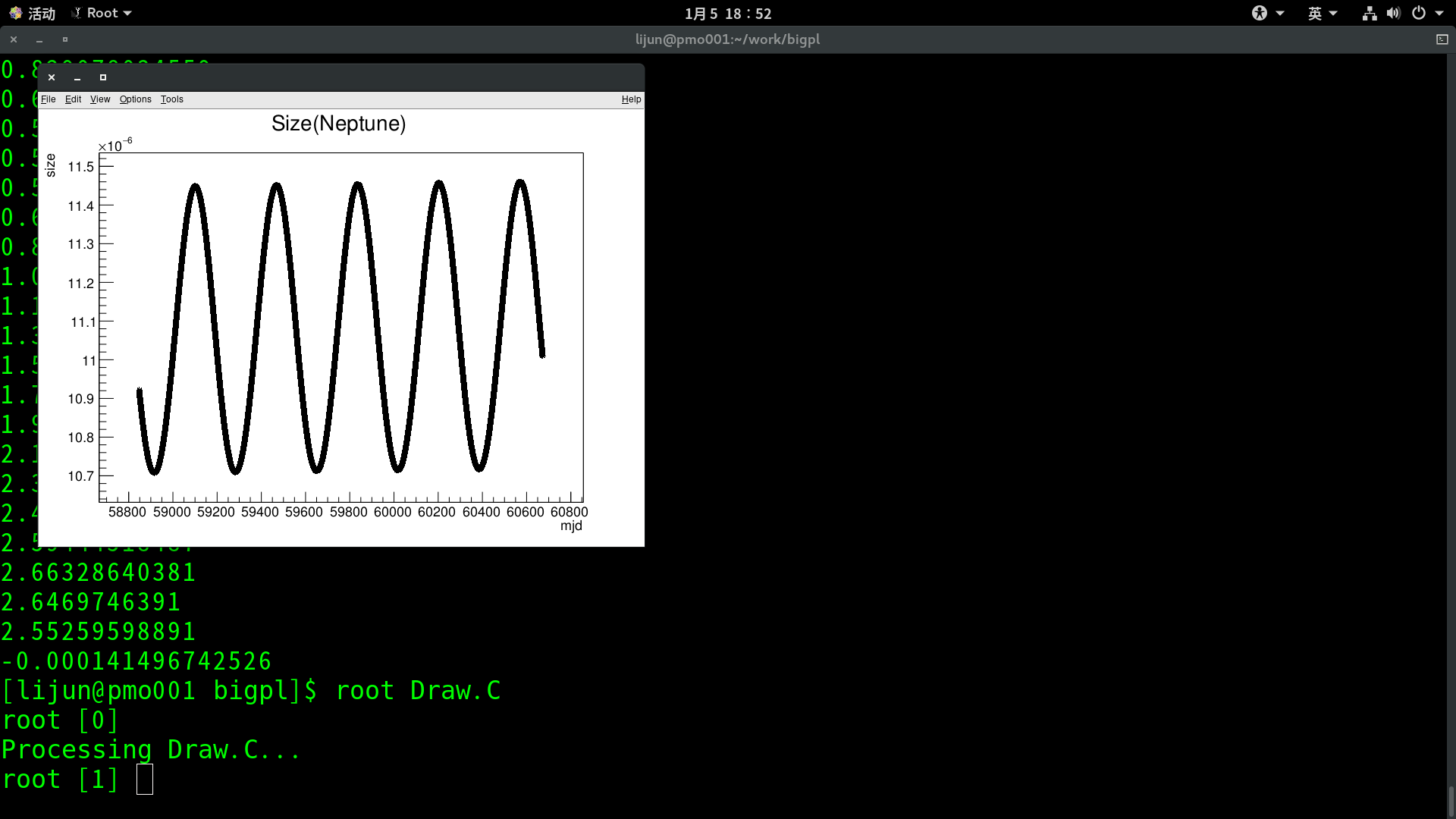
 

## 2.2 太阳系行星大小分布

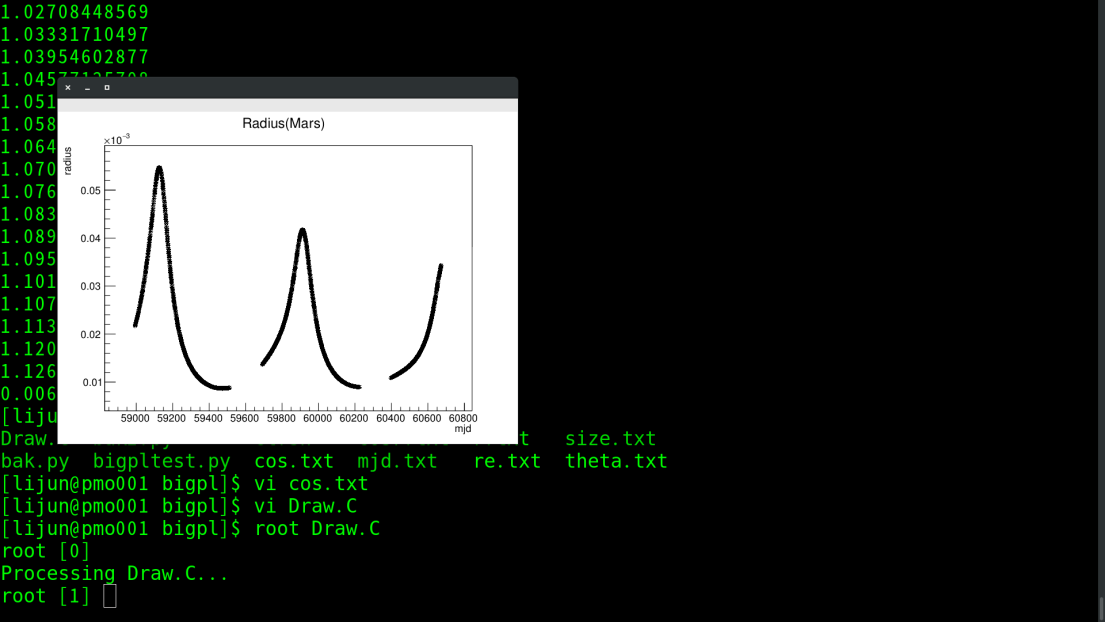
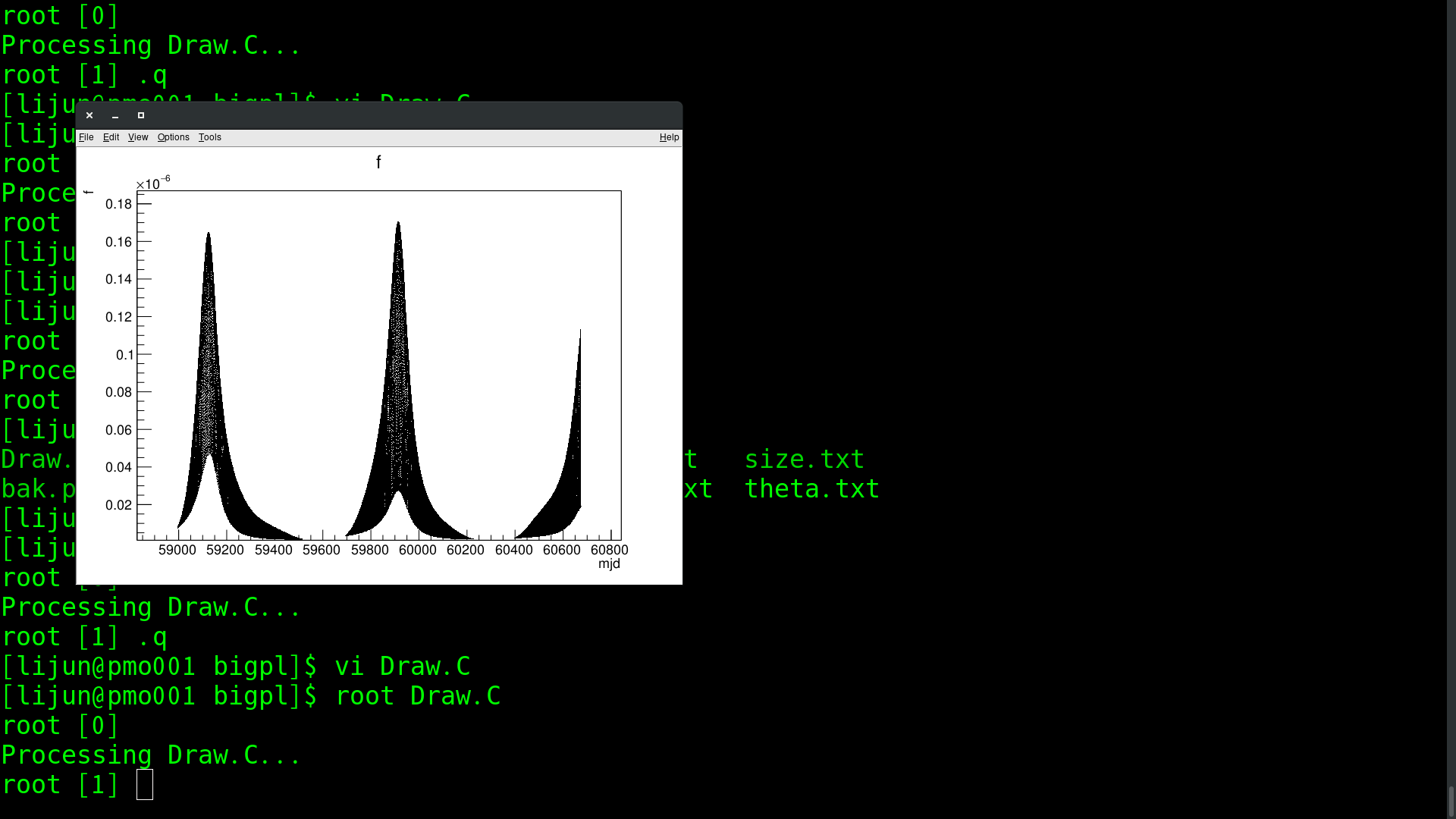
 



从以上结果可以看出，行星的视半径是一个随着时间变化的动态值，而且具有一定的周期性.

## 2.3 行星参数分析

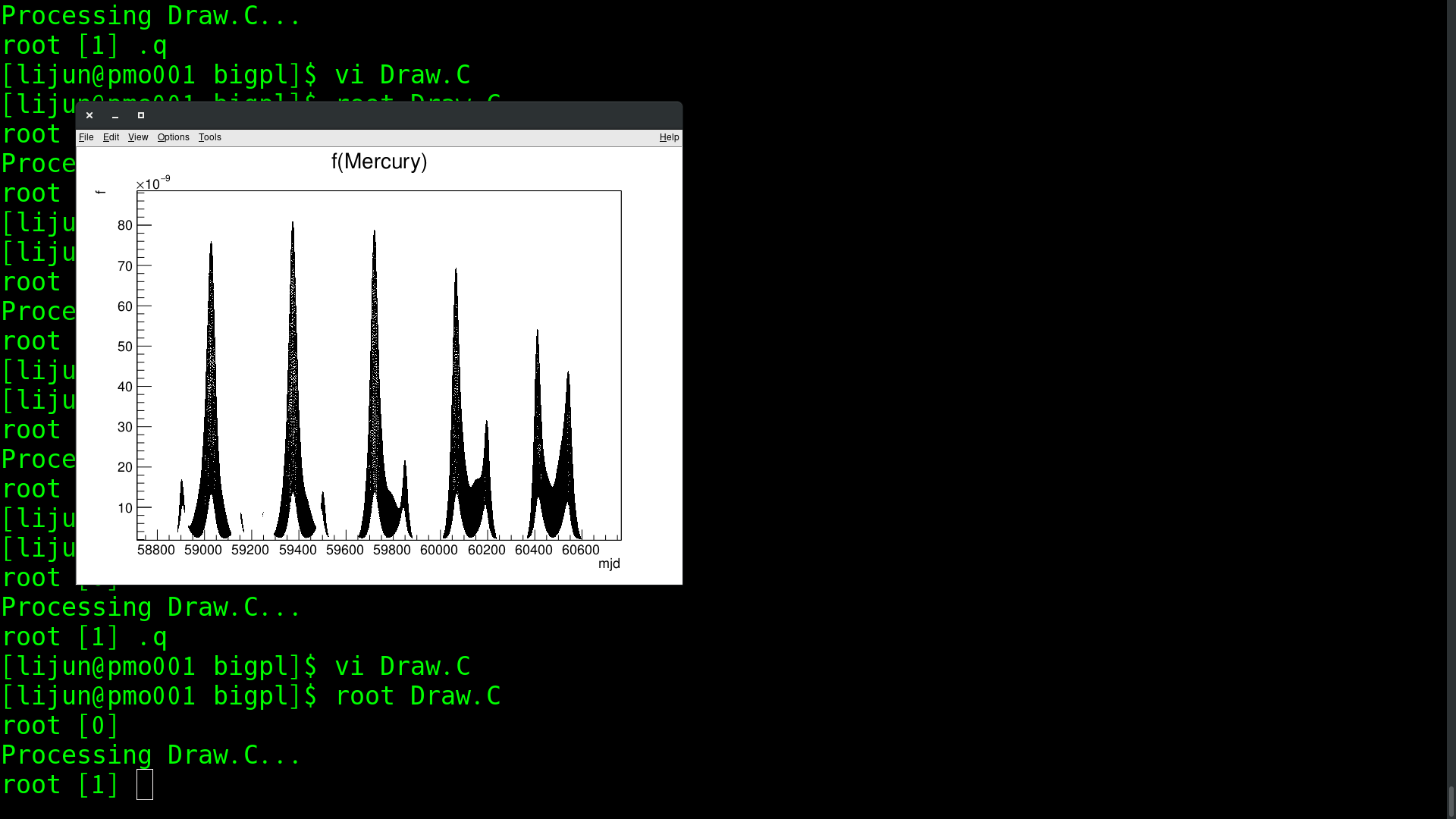
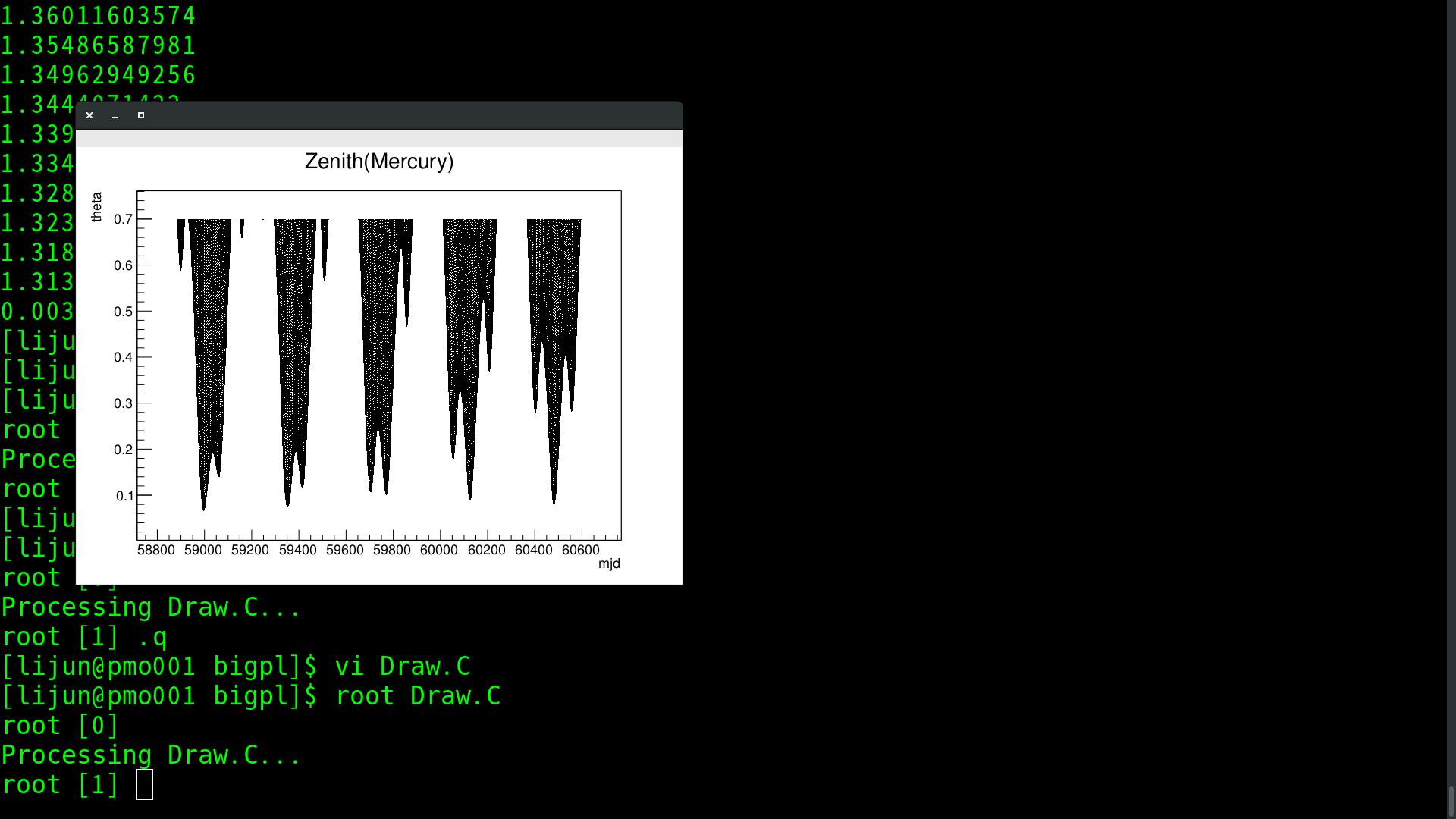
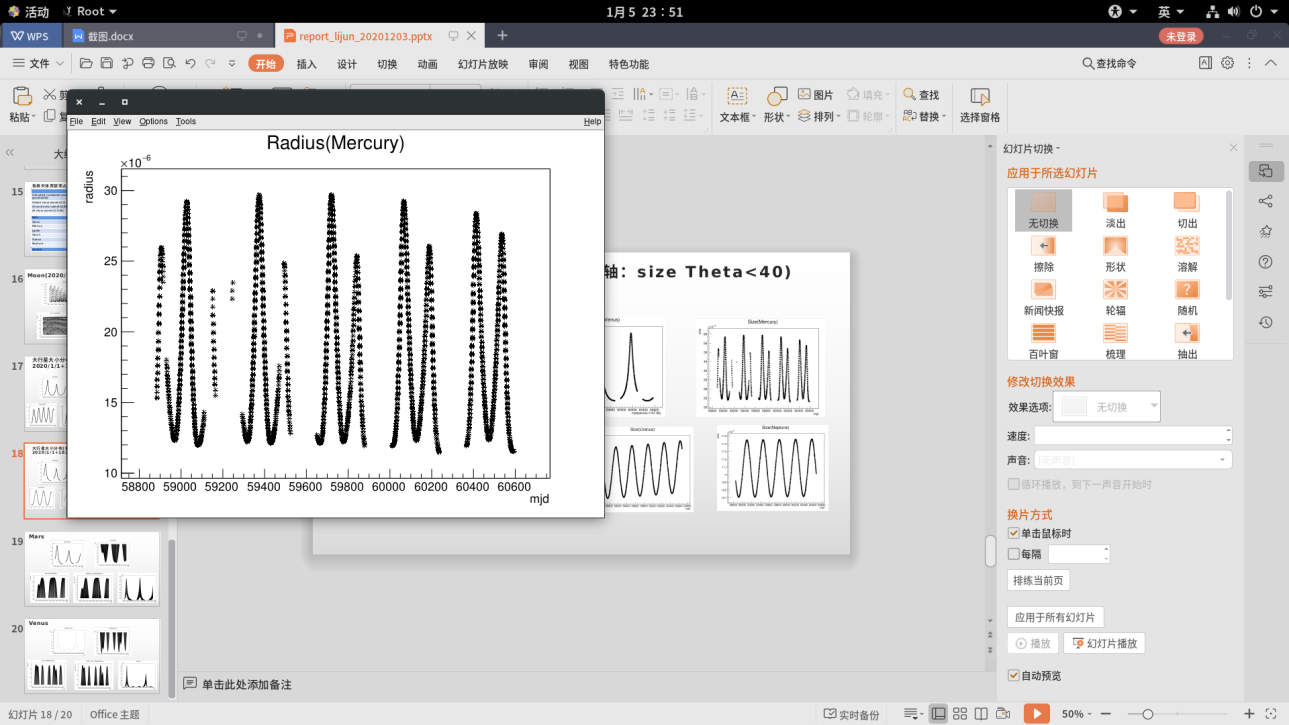
#### 2.31 火星参数分析



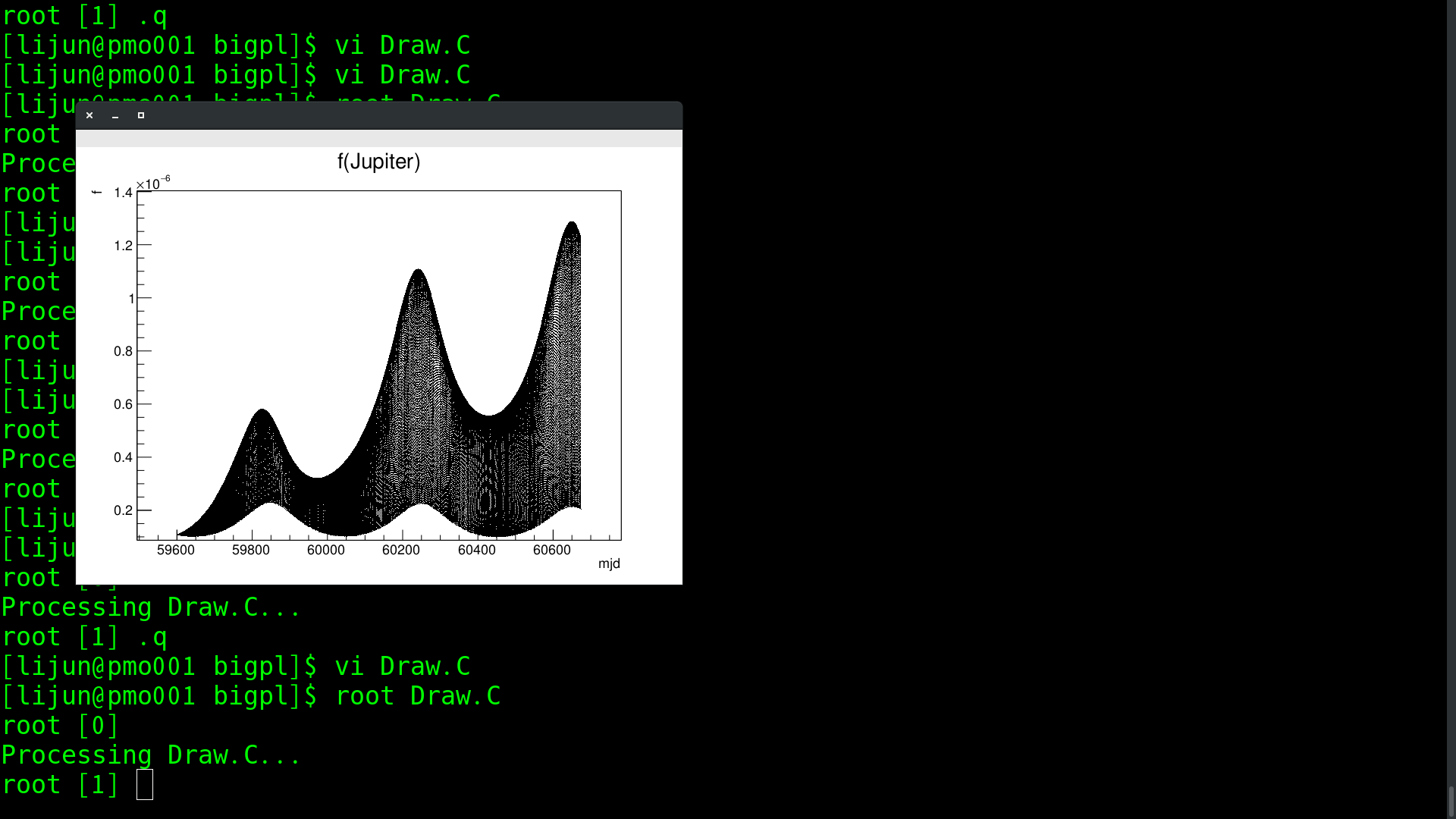
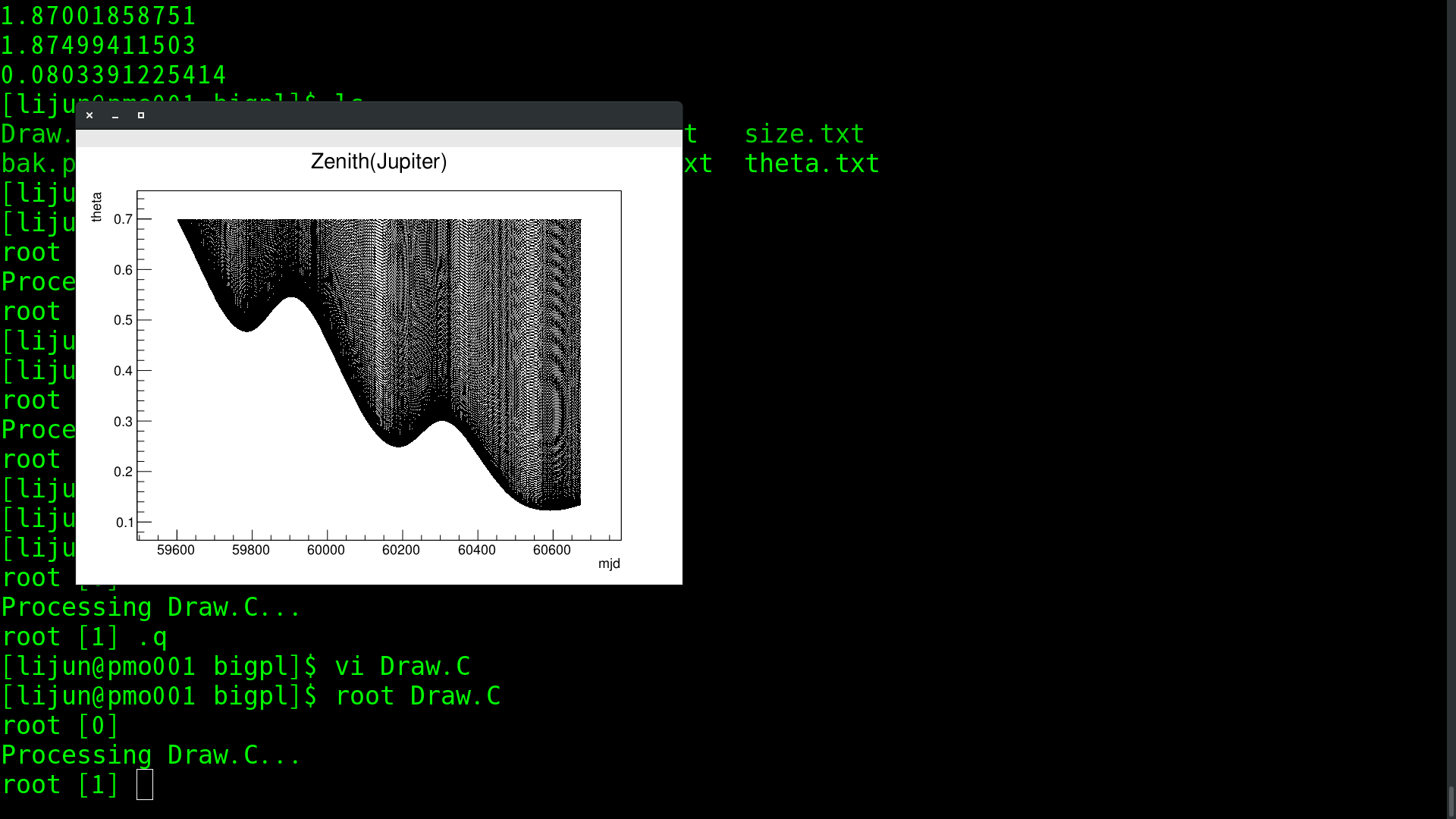
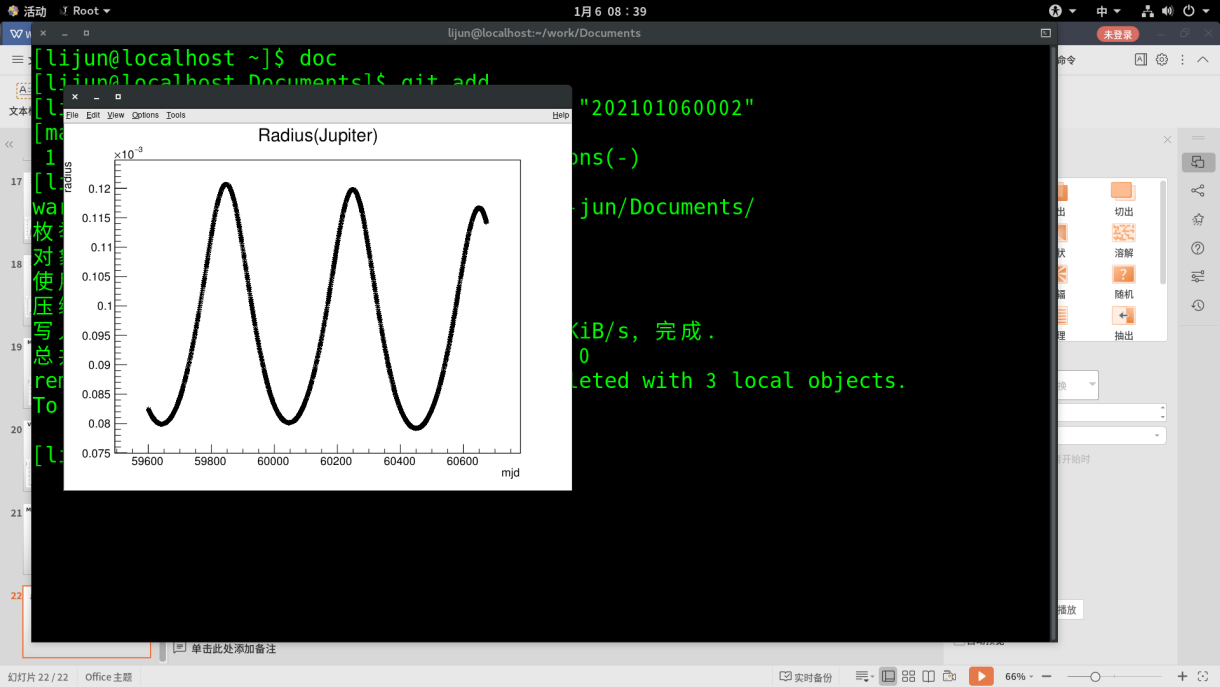
## 2.4 各表之间的联系图

#### 2.32 金星参数分析

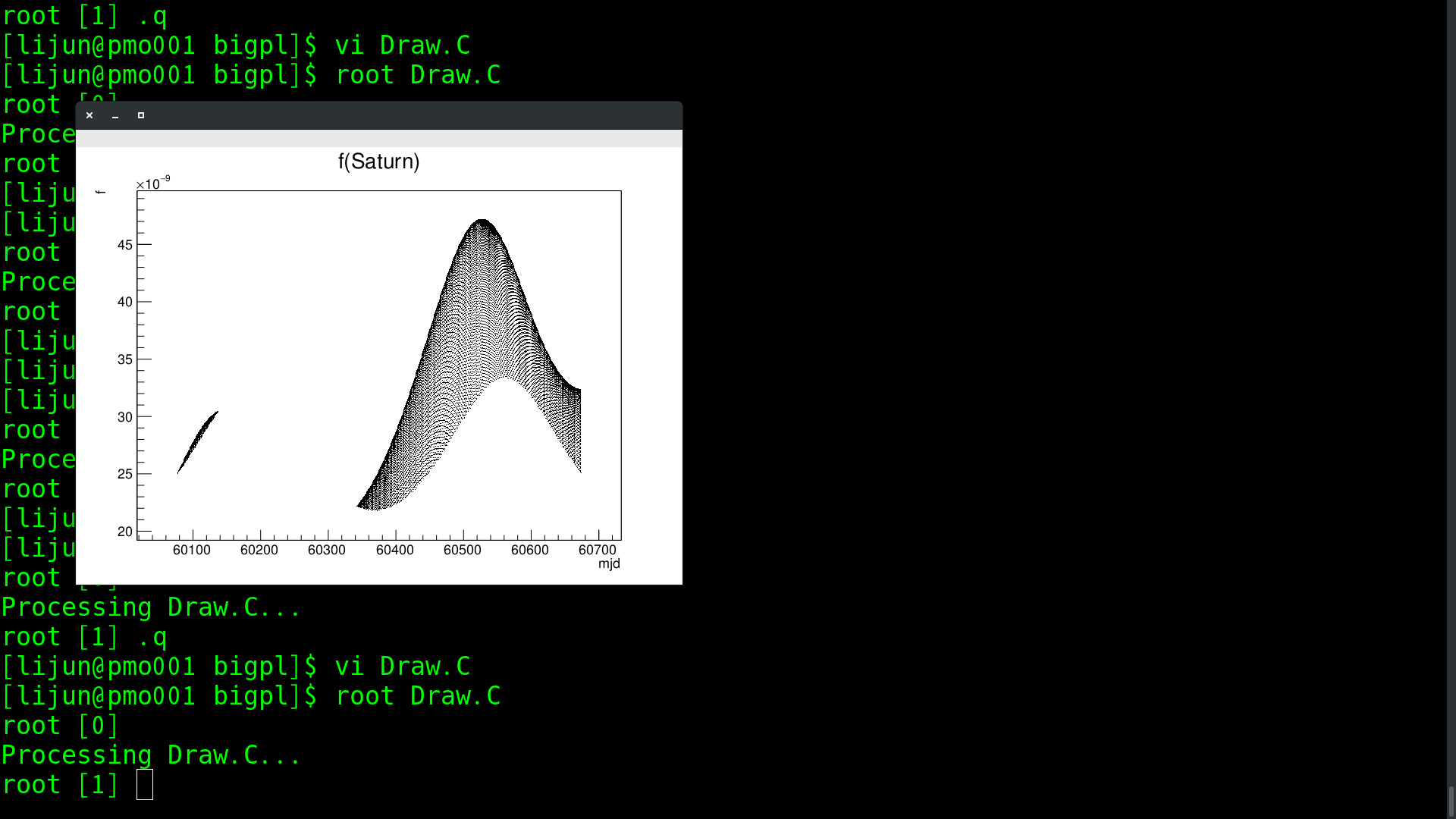
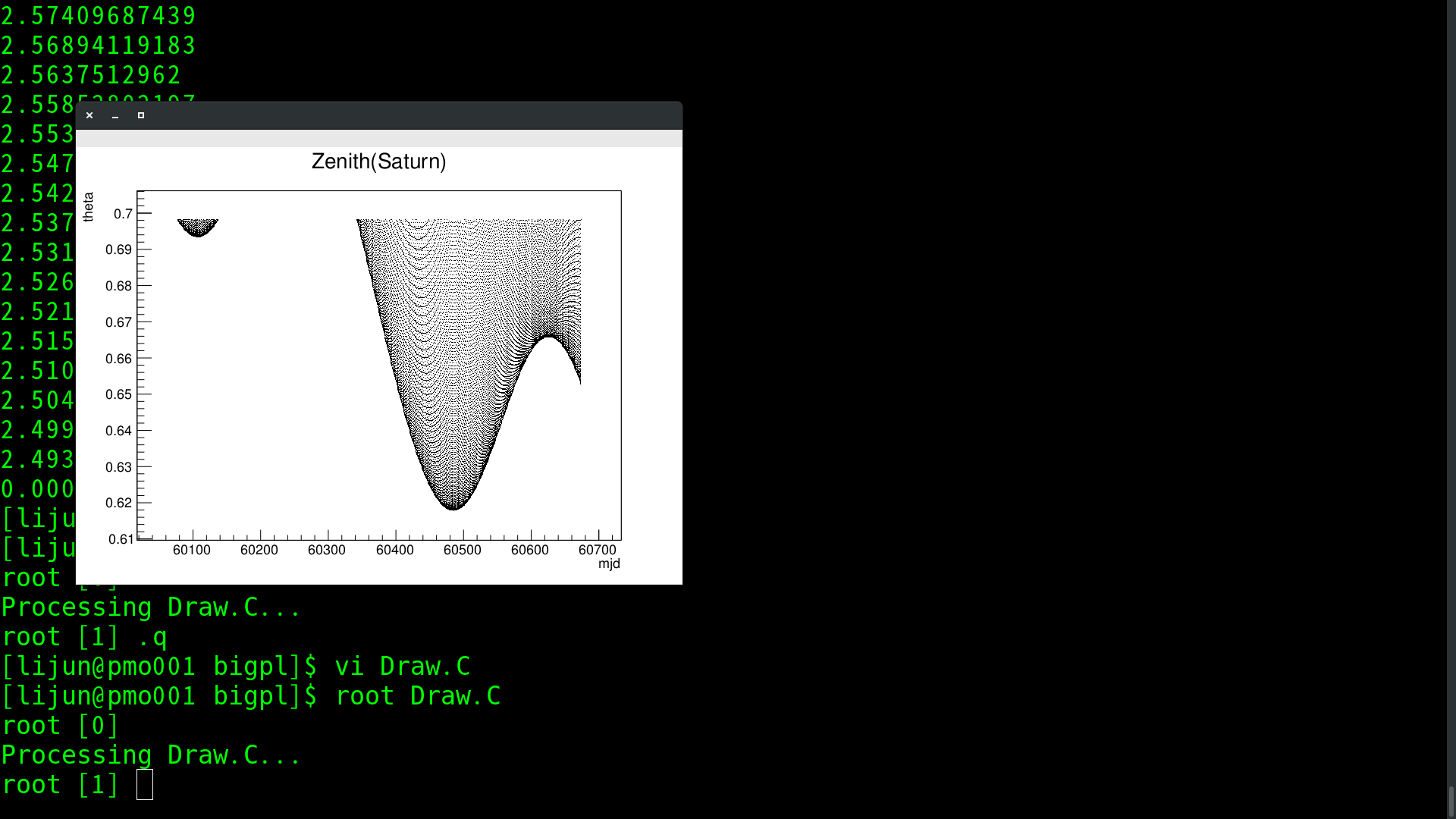
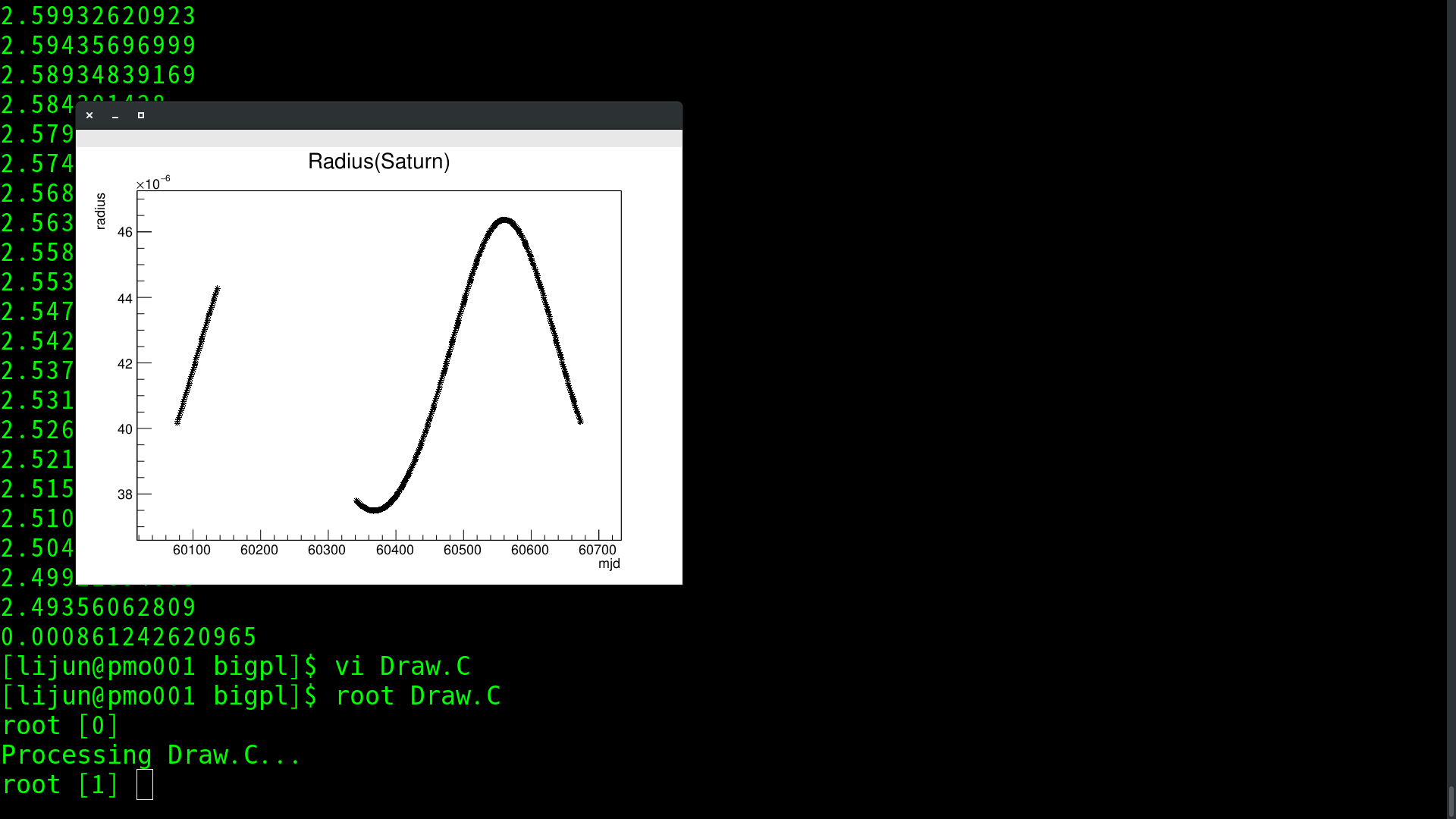
2.33 水星参数分析



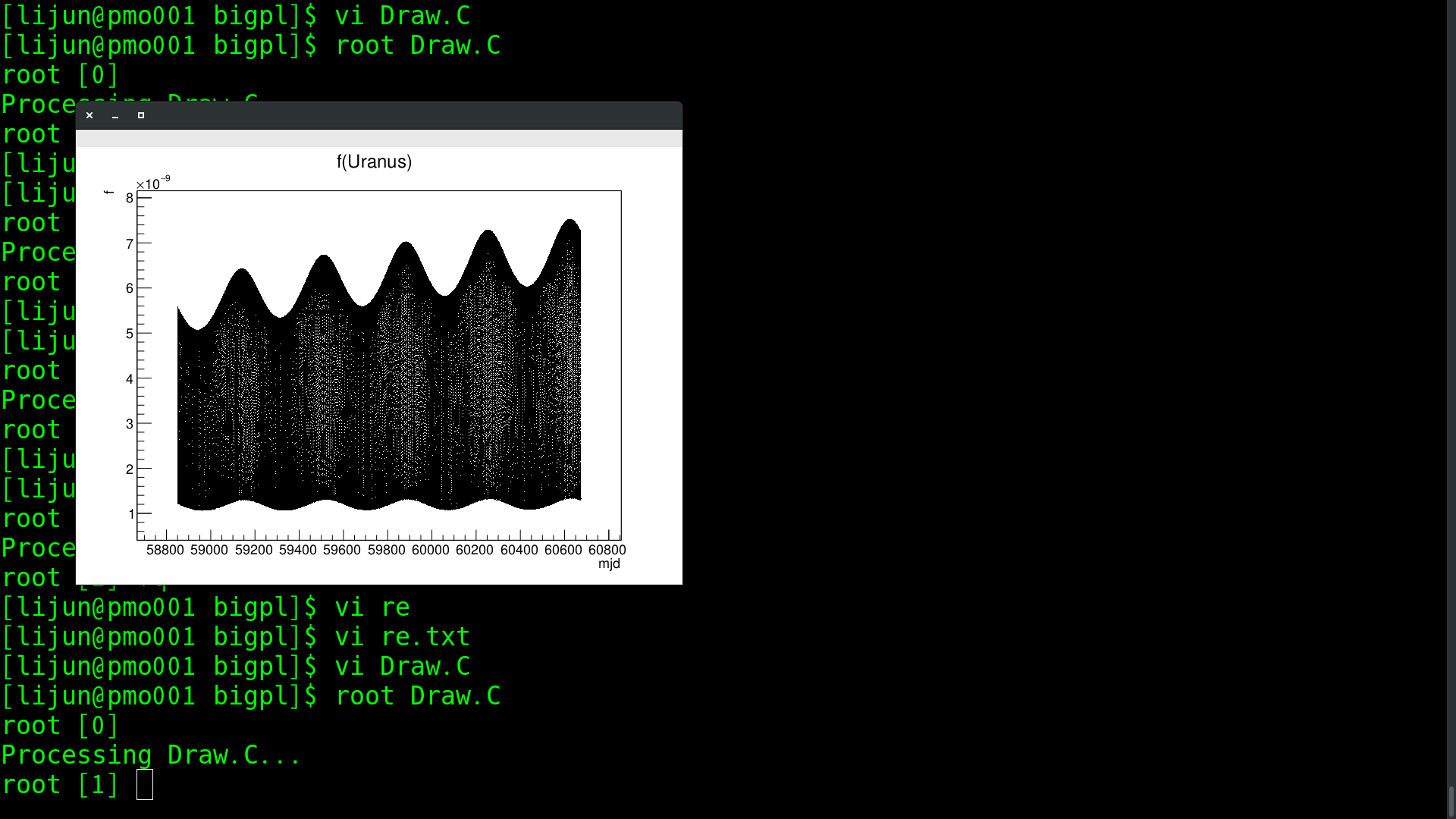
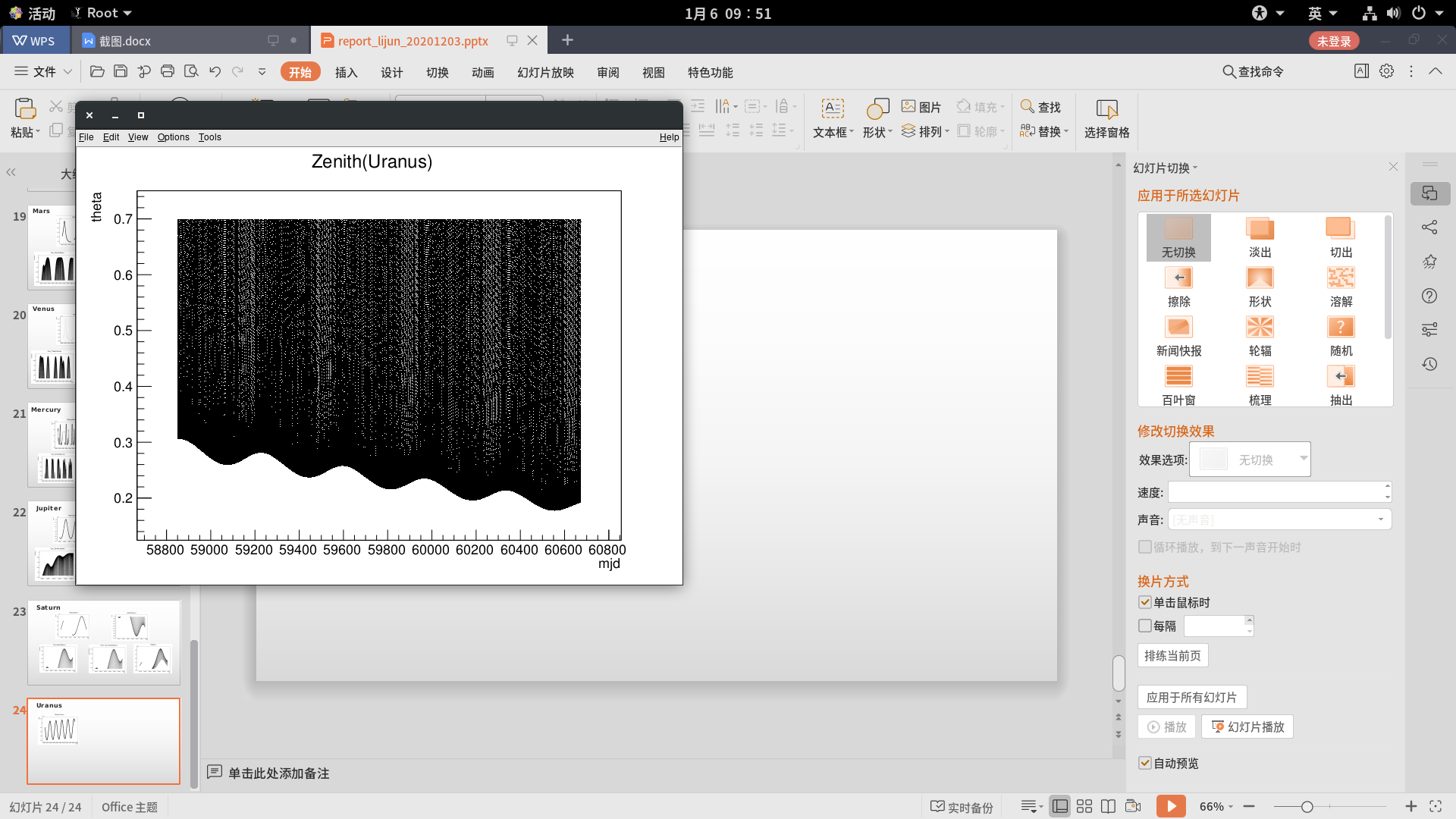
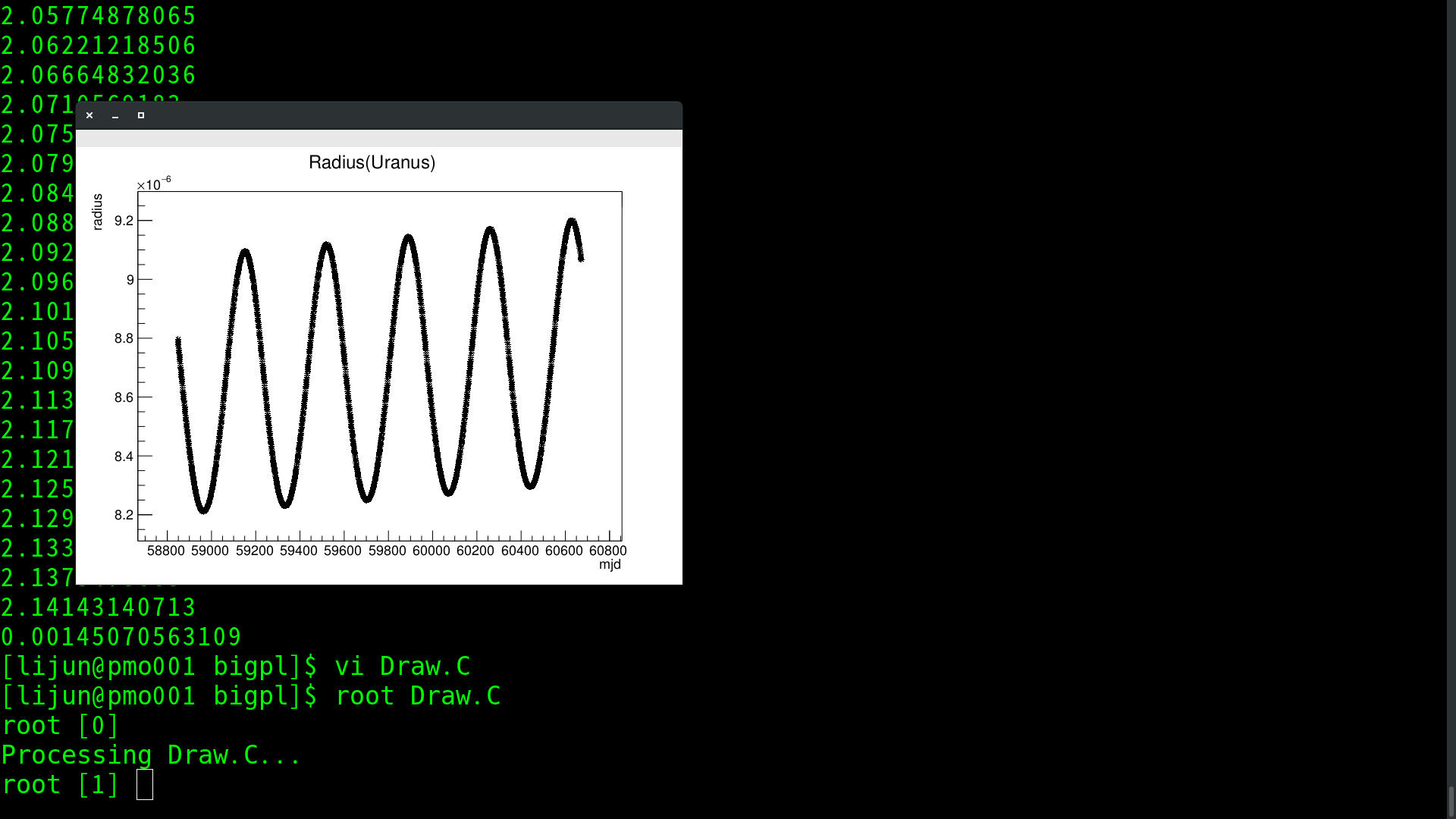
2.34 木星参数分析



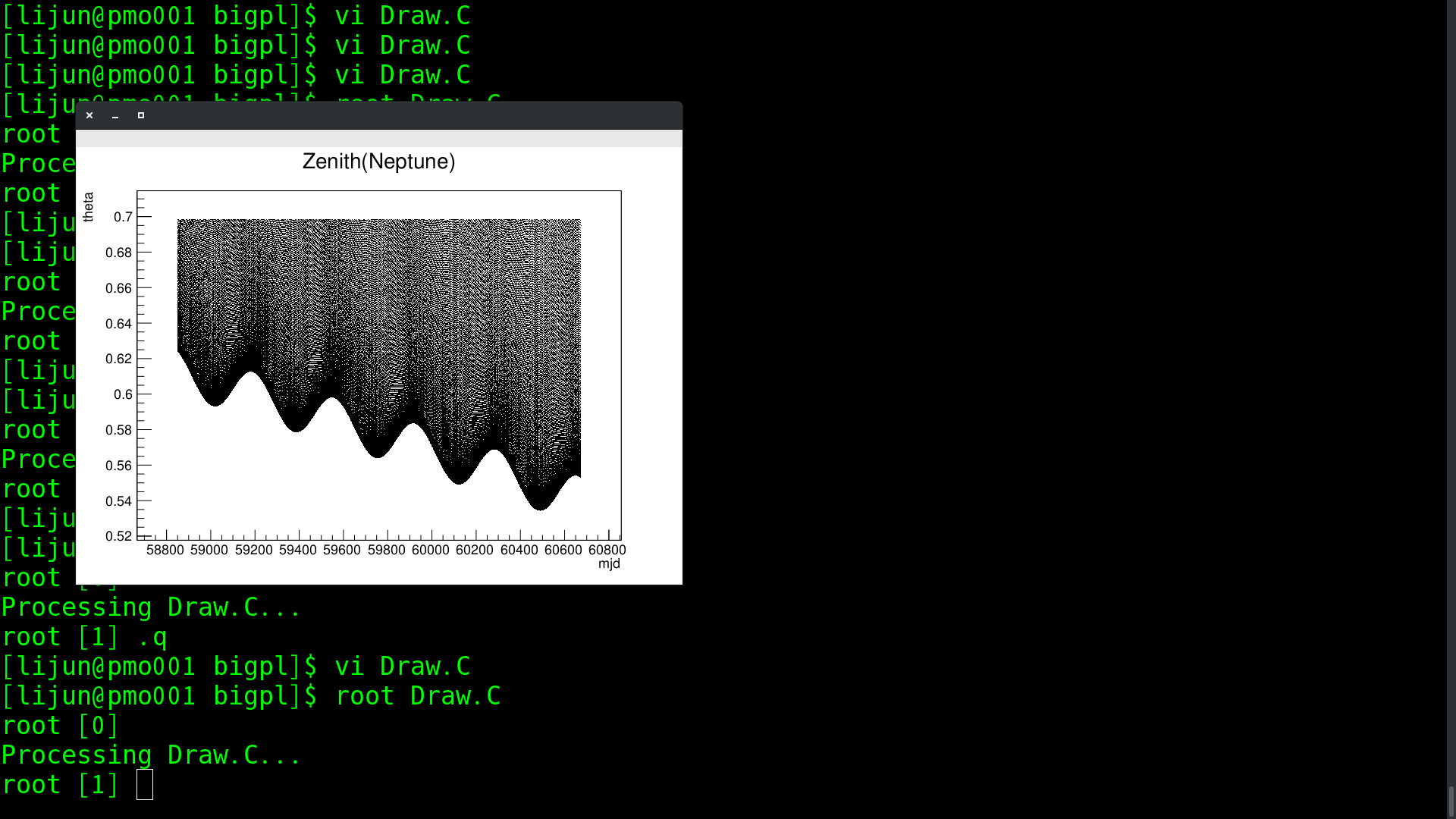
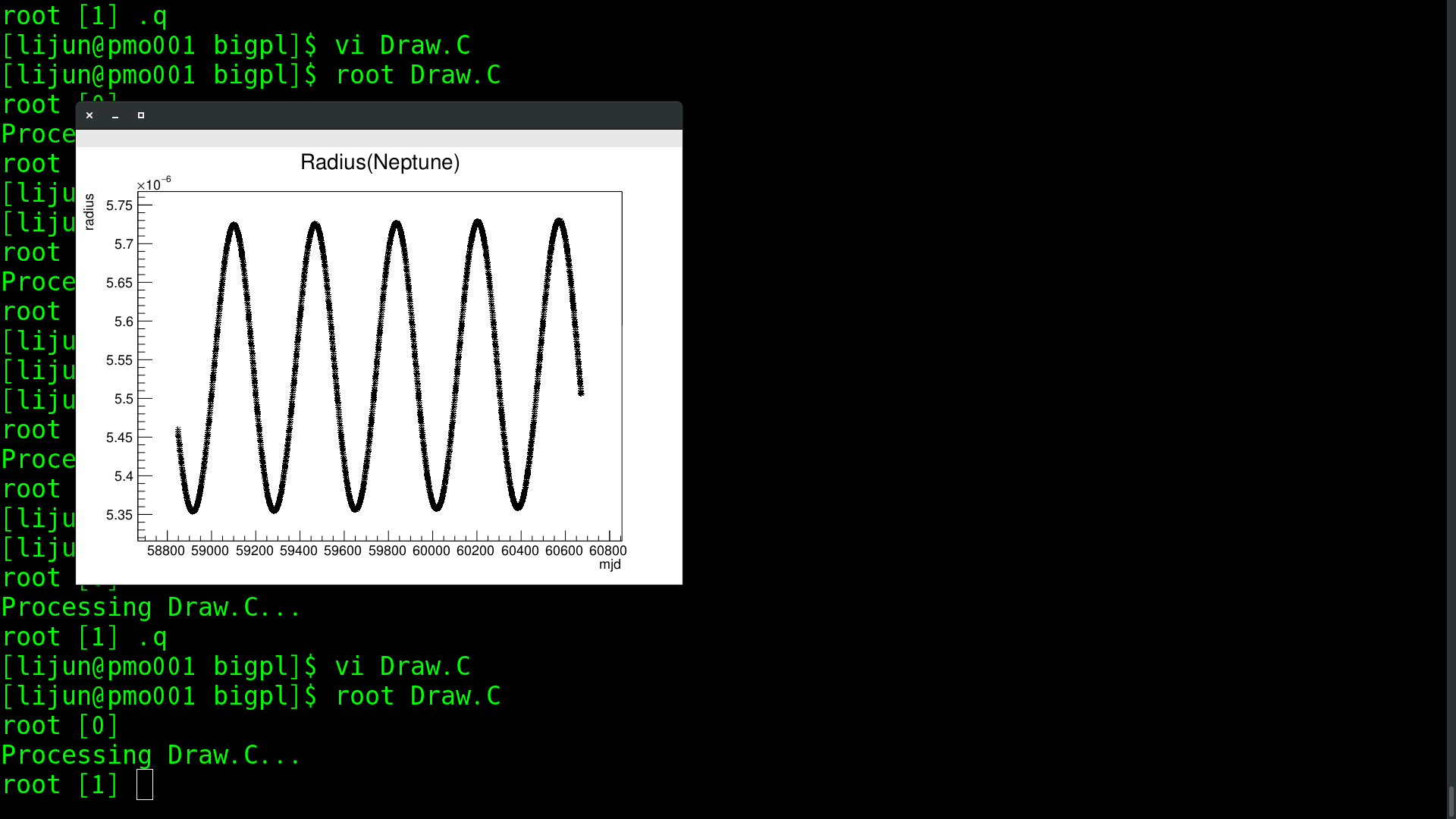
2.35 土星参数分析



# 2.36 天王星参数分析

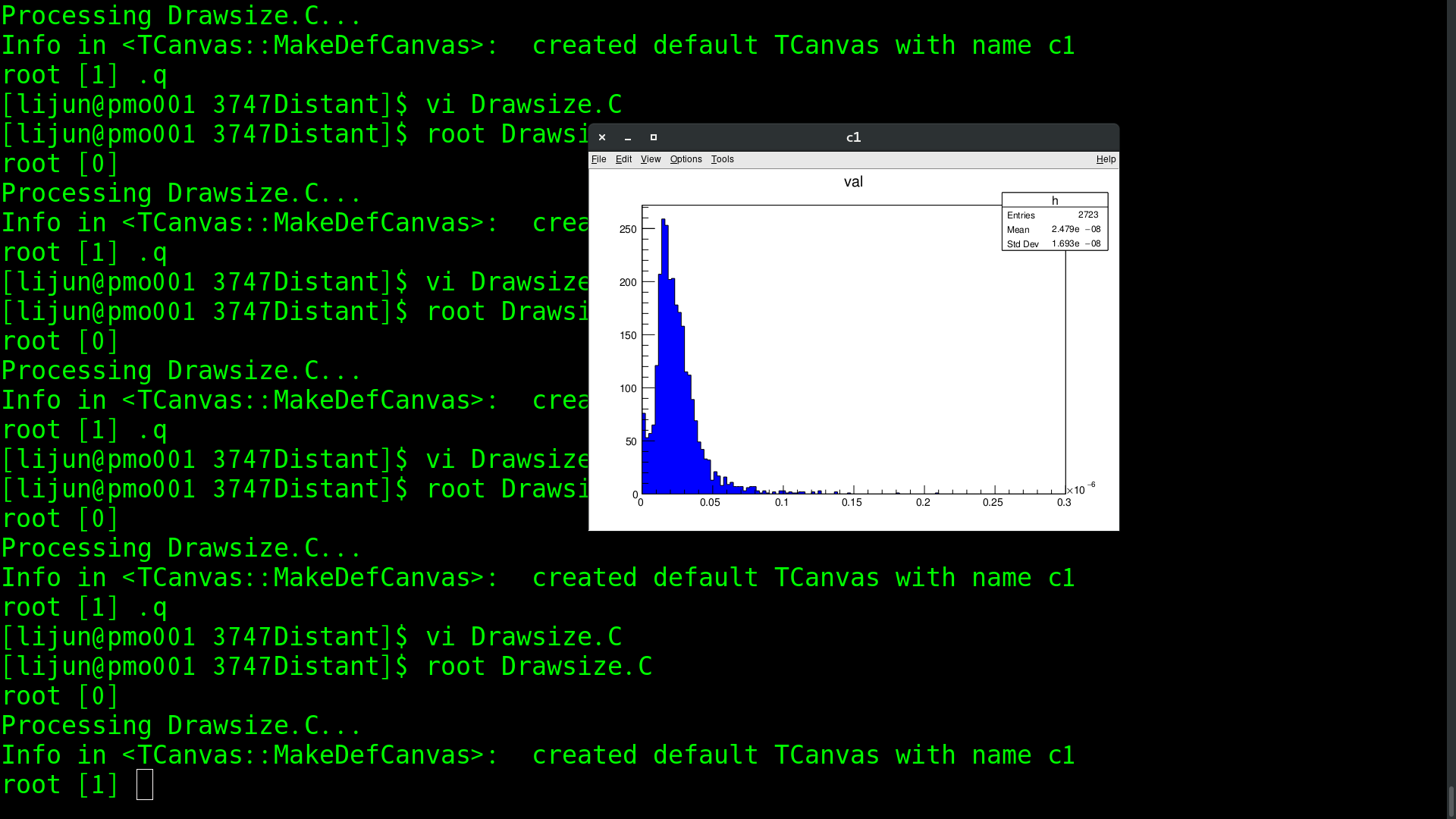
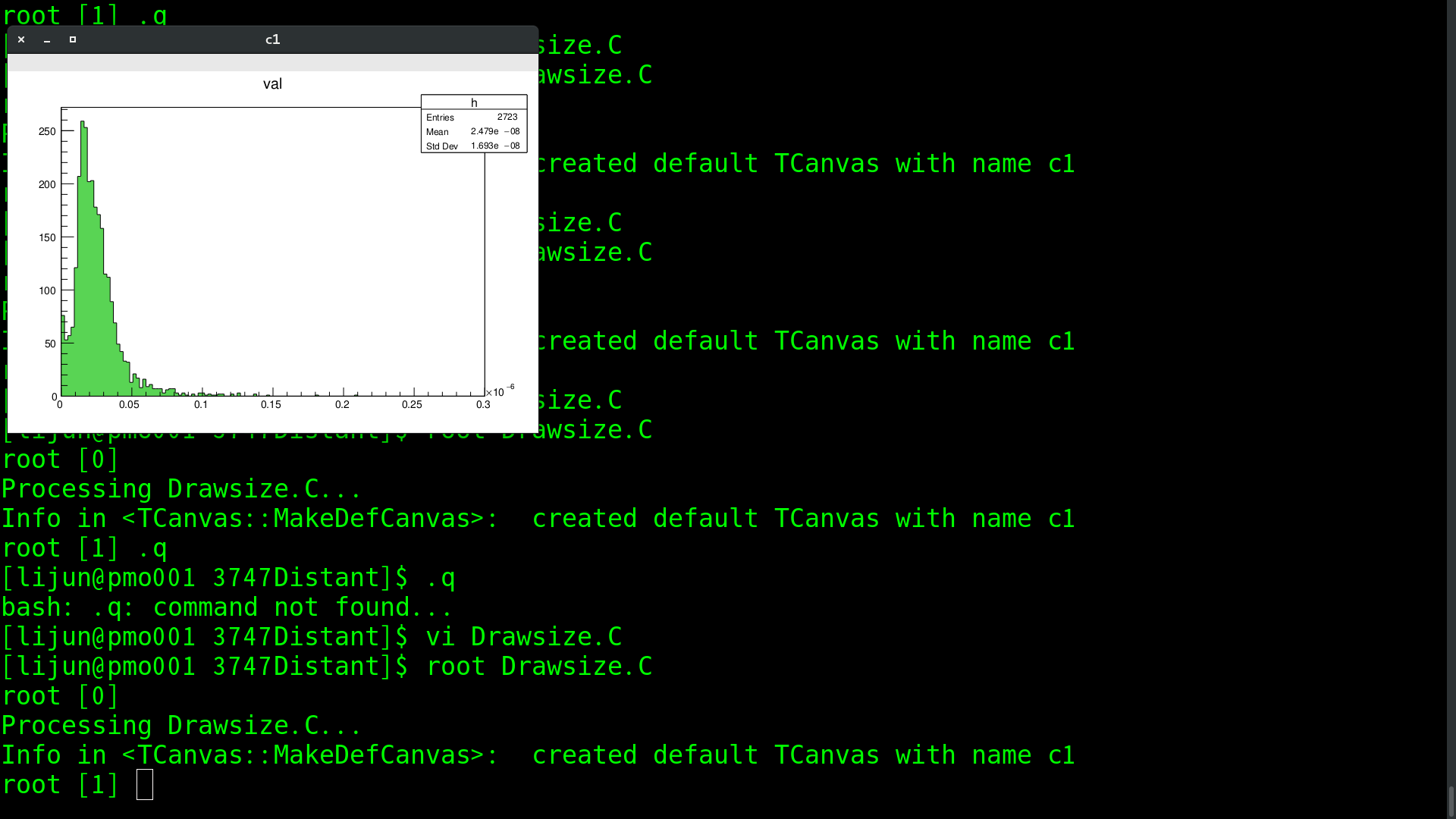


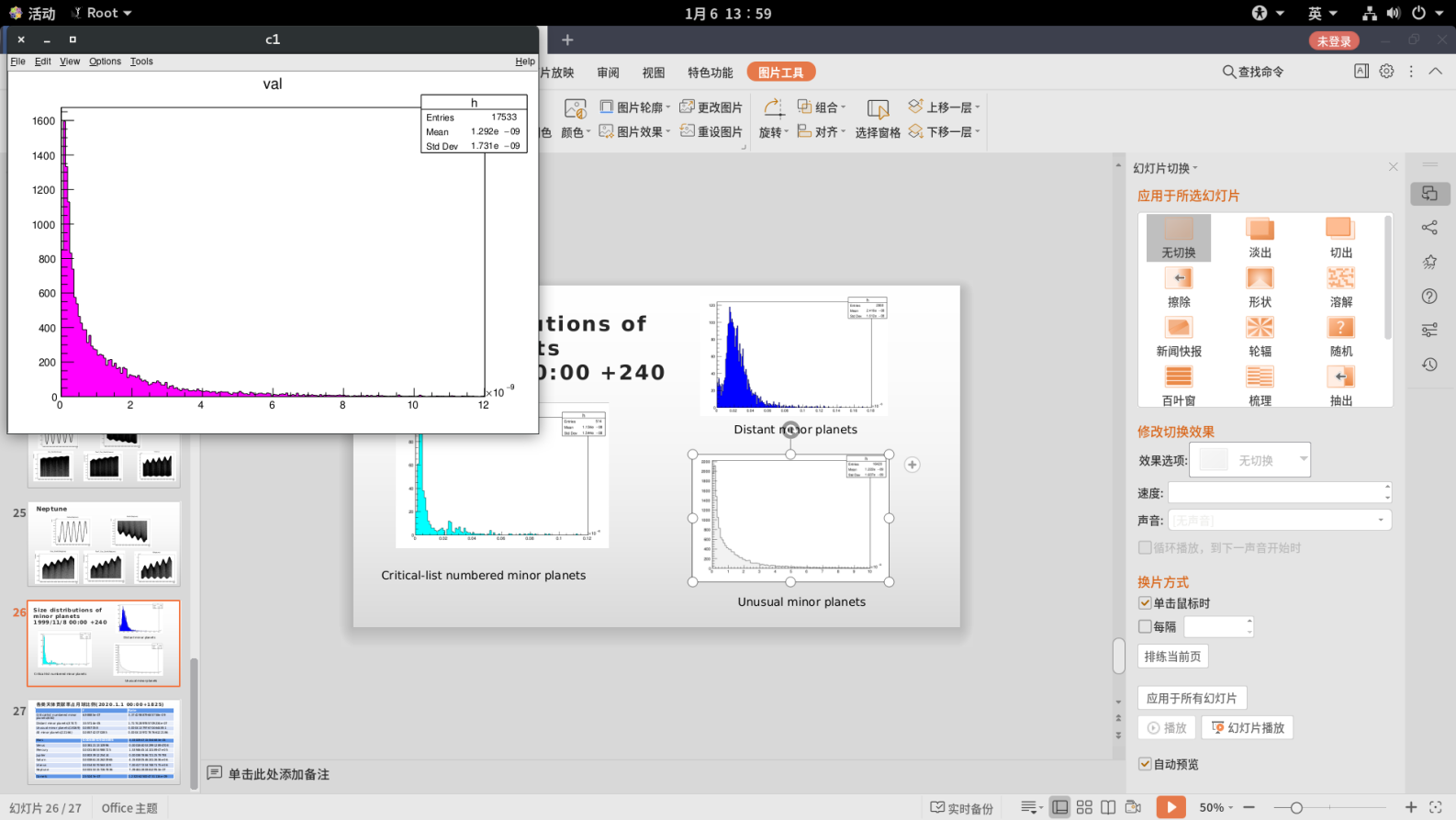
2.37 海王星参数分析



# 第三章 小行星及彗星对宇宙线的遮挡作用分析

## 3.1 小行星的分类和活动规律





## 3.2 小行星的参数分析

3.3彗星的基础知识

3.4彗星的遮挡率分析

### 3.5.2 检测书籍编号是否已存在

因为不可能存在重复的书籍编号。为了防止操作员的操作错误，所以需要提供一个书籍编号监测系统。如果该书籍号已存在，则不能进行添加，否则可以进行添加。

要检测书籍编号是否存在，需要在Dao类中添加一个方法，该方法用来接收输入的书籍编号，然后查询该书籍编号对应的书籍记录，其主要方法如下：

public static List selectBookInfo(String ISBN) {//根据图书编号查询图书

List list=new ArrayList();//保存所有查询到的书籍信息

String sql = "select \* from tb\_bookInfo where ISBN='"+ISBN+"'";

ResultSet rs = Dao.executeQuery(sql);//执行查询

……//插入数据，关闭连接

}

对书籍编号进行检测，一般是在操作员输入完书籍编号，并准备输入其他信息时来完成。这是需要新建一个ISBNFFocusListener监听类，该类继承了FocusAdoper类，并实现其中的focusLost，其实现代码如下：

class ISBNFocusListener extends FocusAdapter {

public void focusLost(FocusEvent e){

if(!Dao.selectBookInfo(ISBN.getText().trim()).isEmpty()){

JOptionPane.showMessageDialog(null, "添加书号重复！");

return;

}

}

}

### 3.5.3 图书信息添加

要添加书籍信息，首先要在Dao中添加一个Insertbook方法，通过该方法来接受用户输入的所有书籍信息，然后执行插入操作。其主要方法如下：

public static int Insertbook(String ISBN,String typeId,String bookname,

String writer,String translator,String publisher,Date date,Double price){

int i=0;

try{

String sql="insert into tb\_bookInfo(ISBN,typeId,bookname,writer,translator,"+ "publisher,date,price) values('"+ISBN+"','"+typeId+"','"+bookname+"',"+ "'"+writer+"','"+translator+"','"+publisher+"','"+date+"',"+price+")";

i=Dao.executeUpdate(sql);//执行插入操作

}

……

}

### 3.5.4 图书修改信息

修改图书信息同保存图书信息非常类似，首先在Dao中添加一个方法Updatebook。通过该方法来接收图书修改表单中提交的书籍信息。其代码如下：

public static int Updatebook(String ISBN,String typeId,String bookname,

String writer,String translator,String publisher,Date date,Double price){

int i=0;//更新记录数

try{ String sql="update tb\_bookInfo set ISBN='"+ISBN+"' , typeId='"+typeId+"',bookname='"+bookname+"',

writer='"+writer+"',

translator='"+translator+"',

publisher='"+publisher+"',

date='"+date+"',

price="+price+" where ISBN='"+ISBN+"'";

i=Dao.executeUpdate(sql);//执行更新

}

……

}

修改图书信息，是通过单击修改按钮执行的，上面我们已经为修改按钮注册了一个监听器，其对应的监听器类为UpdateBookActionLinter。下面来实现该监听器类，其实现代码如下：

class UpdateBookActionListener implements ActionListener {

public void actionPerformed(final ActionEvent e) {

int i=Dao.Updatebook(ISBNs, bookTypes, bookNames, writer， translators,publishers,Date.valueOf(pubDates),Double.parseDouble(prices));

if(i==1){//如果返回更新记录数为1，表示修改成功

JOptionPane.showMessageDialog(null, "修改成功"); Object[][]results=getFileStates(Dao.selectBookInfo());//重新获得书籍信息

DefaultTableModel model=new DefaultTableModel();//获得表格模型

table.setModel(model);//设置表格模型

model.setDataVector(results, columnNames);//设置模型数据和列名

}

}

}

## 3.6 读者信息管理模块

读者信息管理模块包含两个子模块，分别是读者信息添加和读者信息修改与删除。其中读者信息添加模块用来录入新的读者信息，读者信息修改与删除模块用来修改和删除已有的读者信息。其系统运行界面如图3.5所示。



图3.5 读者信息管理模块运行界面

### 3.6.1 读者信息添加

要添加读者信息，首先要在Dao中添加一个InsertReader方法，通过该方法来接受用户输入的所有读者信息，然后执行插入操作。其主要方法如下：

public static int InsertReader(String name,String sex,String age,String identityCard,Date date,String maxNum,String tel,Double keepMoney,String zj,String zy,Date bztime,String ISBN){

int i=0;

try{String sql="insert into tb\_reader(name,sex,age,identityCard,date,maxNum,

tel,keepMoney,zj,zy,bztime,ISBN)values('"+name+"','"+sex+"','"+age+"','"+identityCard+"','"+date+"','"+maxNum+"','"+tel+"',"+keepMoney+",'"+zj+"','"+zy+"',

'"+bztime+"','"+ISBN+"')";

System.out.println(sql);

i=Dao.executeUpdate(sql);//执行插入操作

}

……

}

### 3.6.2 读者信息修改与删除

修改与删除读者信息同保存读者信息非常类似，首先在Dao中添加两个方法UpdateReader及DelReader。通过这两个方法来接收图读者修改表单中提交的读者信息。其主要代码如下：

public static int UpdateReader(String id,String name,String sex,String age,String identityCard,Date date,String maxNum,String tel,

Double keepMoney,String zj,String zy,Date bztime,String ISBN){

int i=0;

try{String sql="update tb\_reader set

name='"+name+"',sex='"+sex+"',

age='"+age+"',identityCard='"+identityCard+"',

date='"+date+"',maxNum='"+maxNum+"',

tel='"+tel+"',keepMoney="+keepMoney+",

zj='"+zj+"',zy='"+zy+"',

bztime='"+bztime+"'where ISBN='"+ISBN+"'";

i=Dao.executeUpdate(sql);// 执行更新

}catch(Exception e){

e.printStackTrace();

}

Dao.close();//关闭连接

……

}

public static int DelReader(String ISBN){

int i=0;

try{ String sql="delete from tb\_reader where ISBN='"+ISBN+"'";

//System.out.println(sql);

i=Dao.executeUpdate(sql);

}catch(Exception e){

e.printStackTrace();

}

Dao.close();//关闭连接

……

}

## 3.7 图书订购管理模块

图书订购管理模块包含两个子模块，分别是新书订购和验收新书。其中新书订购模块用来录入新的图书信息，验收新书模块用来验收新书信息。其系统运行界面如图3.6和图3.7所示。



图3.6 新书订购管理界面



图3.7 图书验收界面

### 3.7.1 新书订购管理

要添加新书订购信息，首先要在Dao中添加一个InsertBookOrder方法，通过该方法来接受用户输入的所有新书订购信息，然后执行插入操作。其主要方法如下：

public static int InsertBookOrder(String ISBN,Date date,String number,String operator,String checkAndAccept,Double zk){

int i=0;

try{String sql="insert into tb\_order(ISBN,date,number,operator,checkAndAccept,zk) values('"+ISBN+"','"+date+"','"+number+"','"+operator+"',"+checkAndAccept+",'"+zk+"')";

i=Dao.executeUpdate(sql);//执行插入

}

……

}

### 3.7.2 图书验收

添加完图书订购信息后，我们还要对图书进行验收，首先要在Dao中添加一个SelectBookOrder方法，通过该方法来查找出所有录入的新书订购信息，再添加一个UpdateCheckBookOrder方法，通过该方法来对图书进行验收。其主要方法如下：

public static List selectBookOrder(boolean b) {

List list=new ArrayList();

String sql = "SELECT \* FROM tb\_order where ISBN='"+ISBN+"'";

ResultSet rs = Dao.executeQuery(sql);//执行查询

try {

while (rs.next()) {

Order order=new Order();

order.setISBN(rs.getString("ISBN"));

order.setDate(rs.getDate("date"));

order.setNumber(rs.getString("number"));

order.setOperator(rs.getString("operator"));

order.setZk("zk");

order.setCheckAndAccept("checkAndAccept");

list.add(order);

}

} catch (Exception e) {

e.printStackTrace();

}

Dao.close();//关闭连接

return list;

}

public static int UpdateCheckBookOrder(String ISBN){

int i=0;

try{

String sql="update tb\_order set

checkAndAccept=0 where ISBN='"+ISBN+"'";

i=Dao.executeUpdate(sql);//执行更新

}catch(Exception e){

e.printStackTrace();

}

Dao.close();//关闭连接

return i;

}

## 3.8 图书借阅管理模块

图书借阅管理模块包含三个子模块，分别是新书借阅、图书归还和图书搜索。其中图书借阅模块用来录入读者信息及所借阅的图书的信息，图书归还模块用来显示归还读者信息及所借阅的图书信息，图书搜索模块用来为读者提供图书搜索功能。其系统运行界面如图3.8、3.9和3.10所示。



图3.8 图书借阅管理界面



图3.9 图书会还管理界面

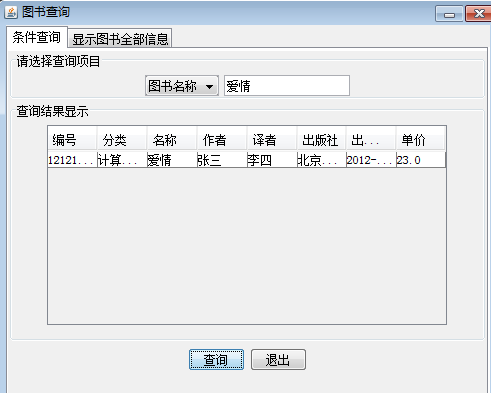


图3.10 图书查询界面

### 3.8.1 图书借阅

当读者想要对某图书进行借阅操作时，图书馆管理员必须将读者信息及所借阅的图书信息输入到数据库，此时，需要在Dao中添加一个InsertBookBorrow方法，通过该方法录入读者信息及所借阅的图书信息。其主要方法如下：

public static int InsertBookBorrow(String bookISBN,String readerISBN,String operatorId,Timestamp borrowDate,Timestamp backDate){

int i=0;

try{String sql="insert into tb\_borrow(bookISBN,readerISBN,operatorId,

borrowDate,backDate)values

('"+bookISBN+"','"+readerISBN+"','

"+operatorId+"','"+borrowDate+"','

"+backDate+"')";

i=Dao.executeUpdate(sql);//执行插入

}

……

}

### 3.8.2 图书归还

当读者到图书馆还书时，图书馆管理员需要通过输入读者信息查找出读者所借阅的图书信息，此时，需要在Dao中添加一个selectBookBack方法，通过该方法查询出读者所借图书的信息。其主要方法如下：

public static List selectBookBack(String readerISBN) {

List list=new ArrayList();

String sql = "SELECT a.ISBN AS bookISBN,

a.bookname, a.typeId ,b.id,b.operatorId,

b.borrowDate, b.backDate,

c.name AS readerName,

c.ISBN AS readerISBN FROM tb\_bookInfo

a INNER JOIN tb\_borrow b ON

a.ISBN = b.bookISBN INNER JOIN

tb\_reader c ON b.readerISBN = c.ISBN WHERE (c.ISBN = '"+readerISBN+"' and isback=1)";

System.out.println(sql);

ResultSet rs = Dao.executeQuery(sql);//执行查询

……

}

### 3.8.3 图书搜索

管理员可以通过图书搜索模块对图书馆所有的图书信息进行搜索。为了查询方便，可以通过图书名称进行查询，也可以通过图书作者进行查询。此时，需要在Dao中添加selectbookmohu和selectbookmohuwriter这两个方法，通过这两个方法查询出图书信息。其主要方法如下：

public static List selectbookmohu(String bookname){

List list=new ArrayList();

String sql="select \* from tb\_bookInfo

where bookname like '%"+bookname+"%'";

System.out.print(sql);

ResultSet s=Dao.executeQuery(sql);//通过图书名称执行查询

……

public static List selectbookmohuwriter(String writer){

List list=new ArrayList();

String sql="select \* from tb\_bookInfo

where writer like '%"+writer+"%'";

System.out.print(sql);

ResultSet s=Dao.executeQuery(sql);//通过图书作者执行查询

……

}

}

## 3.9 系统维护模块

系统维护模块包含两个子模块，分别是更改口令和用户管理。其中更改口令模块用来更改管理员登陆系统口令，用户管理模块用来添加新的用户及对用户进行修改和删除。其系统运行界面如图3.11、3.12和3.13所示

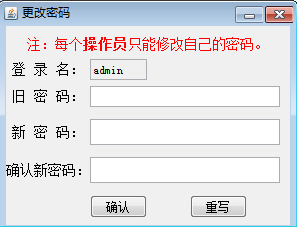


图3.11 更改密码界面

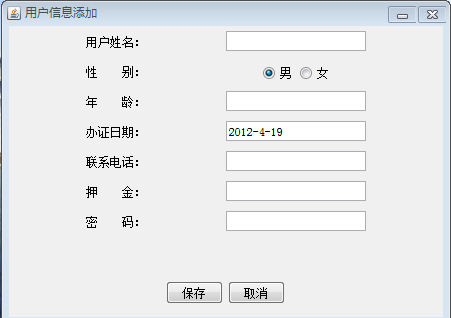


图3.12 用户信息添加界面

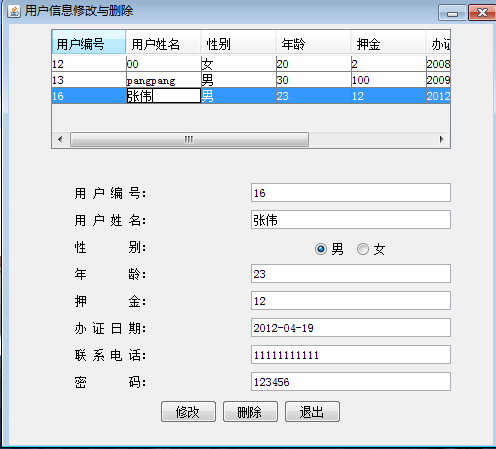


图3.13 用户信息修改与删除界面

### 3.9.1 更改口令

本图书馆管理系统是需要密码才能进入的，而且每个操作员都可以根据自己的喜好更改自己的密码，此时，需要在Dao中添加Updatepass方法，通过该方法来尽心口令的更改。其主要方法如下：

public static int Updatepass(String password,String name){

int i=0;

try{String sql="update tb\_operator set password='

"+password+"' where name='"+name+"'";

i=Dao.executeUpdate(sql);//执行更新

}

……

}

### 3.9.2 用户信息添加

要想添加新的用户时，需要将新的用户信息录入到用户表中。此时，需要在Dao中添加Insertoperator方法，通过该方法录入新的用户信息。其主要方法如下：

public static int Insertoperator(String name,String sex,int age,String identityCard,Date workdate,String tel,String password){

int i=0;

try{String sql="insert into tb\_operator

(name,sex,age,identityCard,workdate,tel,password) values('"+name+"','"+sex+"',"+age+",'

"+identityCard+"','"+workdate+"',

'"+tel+"','"+password+"')";

System.out.println(sql);

i=Dao.executeUpdate(sql);//执行插入

}

……

}

### 3.9.3 用户信息修改与删除

要想实现用户信息修改与删除，需要在Dao中添加Updateuser和Deluser方法，通过这两个方法实现用户信息的修改与删除。其方法如下：

public static int Deluser(int id){

int i=0;

try{String sql="delete from tb\_operator where id='"+id+"'"; //System.out.println(sql);

i=Dao.executeUpdate(sql);//执行用户信息删除操作

}catch(Exception e){

e.printStackTrace();

}

Dao.close();//关闭连接

return i;

}

public static int Updateuser(int id,String name,String sex,

int age,String identityCard,

Date workdate,String tel,String password){

int i=0;

try{String sql="update tb\_operator set name='"+name+"',

sex='"+sex+"',age="+age+",identityCard='

"+identityCard+"',workdate='"+workdate+"',

tel='"+tel+"',password='"+password+"' where id='"+id+"'";

i=Dao.executeUpdate(sql);//执行用户信息修改操作

}catch(Exception e){

e.printStackTrace();

}

Dao.close();//关闭连接

return i;

}

# 第四章 系统测试

## 4.1 测试项目

页面测试：逐个测试每个页面是否可以正确显示，显示页面是否美观，易懂，测试各菜单能否正常运行。

经测试，所有界面均可以正确显示，显示页面比较美观，易懂，各菜单均能正常运行。

功能测试：对每个菜单下的功能逐一测试，尽最大可能发现潜在的错误。

经测试，每个菜单下的功能均能实现，暂未发现其潜在的错误。

需求测试：根据需求分析的内容，测试系统是否和当初的设计一样。

经测试，所设计的图书馆管理系统基本满足当初的设计需求。

## 4.2 测试用例

根据本系统的测试项目需求，设计了部分测试用例，见表4.1。

表4.1 测试用例表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 用例  编号 | 测试  模块 | 输入 | 输出 | 预期结果 | 与预期结果 |
| 1 | 登陆  模块 | 用户名:admin  密 码:122410 | 进入主系统界面 | 进入主系统界面 | 相同 |
| 2 | 图书  信息  添加 | 输入图书编号：1111111111111  类别:程序设计  书名：设计  作者:周五  出版社:天津出版社  译者：赵六  出版日期: 2002-4-19  单价：24 | “添加成功！” | “添加成功！” | 相同 |
| 3 | 读者  信息  添加 | 姓名：张伟  性别：男  年龄：23  职业：学生  有效证件：工作证  证件号码：1111111111111  最大借书量：3  会员证有效日期：2013-4-19  电话：22222222222  押金：50  办证日期：2011-5-19  读者编号：1111111111111 | “添加成功！” | “添加成功！” | 相同 |
| 4 | 读者  信息  修改 | 姓名：张伟  性别：男  年龄：24  职业：学生  有效证件：工作证  证件号码：1111111111112  最大借书量：3  会员证有效日期：2013-4-19  电话：22222222222  押金：50  办证日期：2011-5-19  读者编号：1111211111111 | “修改成功！” | “修改成功！” | 相同 |
| 5 | 图书  搜索 | 图书名称：设计 | 编号：1111111111111  类别:程序设计  书名：设计  作者:周五  出版社:天津出版社  译者：赵六  出版日期: 2002-4-19  单价：24 | 编号：1111111111111  类别:程序设计  书名：设计  作者:周五  出版社:天津出版社  译者：赵六  出版日期: 2002-4-19  单价：24 | 相同 |
| 6 | 更改  口令 | 登陆名：admin  旧密码：admin  新密码：122410  确认新密码：122410 | “密码修改成功” | “密码修改成功” | 相同 |
| 7 | 用户  信息  添加 | 用户姓名：cp  性别：男  年龄：22  办证日期：2012-4-20  联系电话：11111111111  押金：22  密码：122410 | “添加成功” | “添加成功” | 相同 |

# 结束语

这次图书馆管理系统的设计真是让我绞尽脑汁，经过了好几个月才做好。其实制作管理系统我们现在看来已经不是难事，难的是如何去构思，如何去想。

做系统要用心去做，我从中也学到很多知识，知道如何去综合多门学科中的知识，编程能力也有了很大提高，另外也有很多心得体会。

本系统是一个最基本的图书馆管理系统，可扩展性很大，系统的执行效率也比较高。

这些天的设计，我深刻体会到软件开发是相当辛苦的，但成功以后的喜悦也是非常美妙的，投入的越多，获得的快乐与充实感越多。

本课题在选题及研究过程中得到周老师的亲切关怀和悉心指导下完成的。她严肃的科学态度，严谨的治学精神，精益求精的工作作风，深深地感染和激励着我。从课题的选择到项目的最终完成，周老师都始终给予我细心的指导和不懈的支持。周老师不仅在学业上给我以精心指导，同时还在思想、生活上给我以无微不至的关怀，在此谨向周老师致以诚挚的谢意和崇高的敬意。

# 参考文献

1. 日影模拟程序的开发及相关数据分析算法的研究 南云程
2. ARGO-YBJ实验的宇宙线月影及“膝”区能谱唯象研究 王博

[3]《Cosmic Rays and Particle Physics》 托马斯 K. 盖瑟 剑桥大学出版社 1990

[4]《ASP．NET网络系统程序开发》 陈宗兴等 中国水利水电出版社 2004

[5]《信息系统开发方法》 姜旭平等 清华大学出版社 1996

[6]《Java面向对象程序设计》 张桂珠等 北京邮电大学出版社 2003

[7]《Java程序设计实例教程》 毕光吉 冶金工业出版社 2007

[8]《软件制作工程》 李超 电子科技大学出版社 2005

[9]《软件工程导论第四版》 张海瀋 清华大学出版社 2003

[10]《软件测试方法和技术》 朱少民 清华大学出版社 2005

[11]《软件测试技术》 佟伟光等 人民邮电出版社 2005