密级:_____

中国科学院大学
University of Chinese Academy of Sciences

# 博士学位论文

## 天文资源无缝融合关键技术研究


作者姓名：樊东卫

指导教师：赵永恒 研究员 国家天文台

崔辰州 研究员 国家天文台

**Prof. Alexander Szalay Johns Hopkins University**

**Dr. Tamás Budavári Johns Hopkins University**


学位类别：理学博士

学科专业：天文技术与方法

培养单位：中国科学院国家天文台

**2013 年 5 月**

# Key Technique Research
# in Astronomical Resource Seamless Integration

**By**

**Dongwei Fan**

**A Dissertation Submitted to**

**University of Chinese Academy of Sciences**

**In partial fulfillment of the requirement**

**For the degree of**

**Doctor of Astronomical Technology and Method**

**National Astronomical Observatories,**

**Chinese Academy of Sciences**

**May, 2013**

# 摘　要

天文资源多种多样，如天文设备、天文胶片、分析软件、数字图片、星表、光谱、数值模拟数据等等，其中各式的观测、计算数据更是天文研究的根基。随着大量的天文仪器、设备、方法投入使用，每天每夜都在产出海量数据，天文开启全波段观测的同时也已经进入到了数据大爆炸时代。为了全面认识一个天体的物理特性，就需要了解它的全貌。而丰富的多波段数据将天体信息以前所未有的层次向天文学家展现出来。但是，如何在多个星表及其它形式数据中找到同一个天体的数据，在大数据时代将越来越困难。随着天文数据的增多，相应的天文工具、技术、服务也越来越多，并结合了众多新兴的信息技术。天文学家需要花费大量时间来了解如何使用这些服务与技术。如何减少天文学家在一些技术使用上的障碍，将众多天文资源整合、连接起来，使之可以无缝访问，愈发变得重要起来。本论文对天文资源整合的几个关键技术进行了研究，并得到了一些崭新的成果。

数据融合方面，比较实用的一种技术是交叉证认。本论文在快速交叉证认方法条带算法（Zones Algorithm）基础之上，提出了一种新颖的方法，将星表天区覆盖信息直接带入到交叉证认过程中，并使用与星表一样的数据索引方式。这在两个星表重叠区域较小或仅在指定区域内进行交叉证认时非常高效。同时由于星表覆盖图信息的加入，可以通过很简单的差集运算，快速了解两个星表中未能相互匹配的天体是由于曝光不足等原因未被另一个星表观测到，异或是因为该天体完全没在另一个星表的观测区域内。此新方法将应用到美国虚拟天文台交叉证认服务（Open SkyQuery）的新版交叉证认引擎中。

数据融合本身迫切需要对不同波段的数据也应用交叉证认方法。但是在射电星表中，有些天体，如类星体除了中心部分之外，还有两侧喷流结构。证认一个天体的时候需要将这两侧的结构也考虑进来，以方便找到此类结构天体的完整信息。本论文创造性地使用直线非对称模型来构造射电观测中出现的喷流，应用贝叶斯推断方法来计算对比射电源在模型中的各种组合方式的概率，以取得一个在模型中最可能的证认结果。也可以用其它模型来取代线性非对称模型，以使用此方法来寻找特定结构的天体信息。经试验，程序结果与澳大利亚射电天文学家手工证认的结果匹配程度较好。此方法可应用到SKA 等射电

望远镜观测结果与光学观测结果的交叉证认中。

本文还在资源的整合及简化使用方式上进行了创新。应用虚拟天文台技术，在维护本地数据文件——如FITS文件——的基础上，联结本地分析工具与网络数据、网络服务。尝试将本地计算机及网络上的数据、工具、服务无缝连接起来。更重要的是将整个过程透明化，使天文学家无须过多了解技术细节，而专注于数据的使用及科学研究。在具体实现上，使用了Eclipse RCP 框架来实现跨平台与插件化，以鼓励天文社区内更多的人通过插件的方式贡献自己的力量，并使得软件成为一个统一的资源管理平台。目前多个天文服务都已经以插件的形式整合到此平台中，并简化了使用流程。

本论文最后还讨论了关于天文学术报告视频资源的管理与发布问题。天文学术报告也是非常重要的天文学习、研究资源。但以往不受重视或缺乏技术能力来解决这些问题。本论文创新地提出了一个通过家用DV即可实施的低成本在线视频直播点播系统，并提供视频存档、往期视频点播等功能。未来希望在积累更多讲座、视频资源的基础之上，与科学数据的使用结合起来，更便于天文工作者使用。

**关键词：** 虚拟天文台,资源访问,数据融合,交叉证认

# Abstract

Various astronomical resources exist in the astronomy community, e.g. telescopes, films, analysis software, digital images, astronomy catalogs, spectrums and simulation data. Especially, the observation and calculation data are the basic resources for astronomy research. Huge data come with the operations of huge amount of astronomy instruments and techniques; astronomy is entering the full-band observation and data explosion era. We have to collect different kinds of data to study an object's real physical specification. The plenty multi-band data will lead astronomers to a new level to understand the universe. But how to find the same object's information in different catalogs or other data sets becomes a tough problem with huge data amount. More and more tools and techniques which utilize new technologies also come with the big data. Astronomers spend lots of time to learn the related knowledge to use these tools and services. How to reduce the barrier among the tools and astronomers, to join the astronomical resources and make them seamless accessible becomes an important issue. This dissertation studies several key techniques to integrate astronomical resources and get some new benefits.

Crossmatching is a useful technique to combine different data sets. Based on the very fast Zones Algorithm Crossmatching method, this dissertation presents a novel method to directly involve catalogs' sky coverage information during the crossmatching processes, which use uniform indices. The new method especially works well when two catalogs only have small common sky coverage area. With the benefit of the sky coverage information, we can do a simple `DIFFERENCE` operation between two data sets to detect the reason why some objects do not have counterparts in other catalog: they are not in the other catalog's sky coverage, or they are too faint to be detected by the telescope. This new method will be applied to the new crossmatch engine of Virtual Astronomical Observatory of United states.

Data integration urgently needs the technique to crossmatch different wave-

bands of data. But in radio catalogs, some objects e.g. quasar has a radio core in center and two radio lobes far away from the core. When crossmatching these objects, we also have to consider these two lobes to find their full information. This dissertation uses a geometry straight line model with different combination of the radio detections to simulate the core and lobes, and applies the Bayesian inferences method to find the best radio counterparts for an optical detection. Other geometry or other kinds of model could be utilized in this method to find the objects which fit the specified structure. We have done some experiments to check the effectiveness of this method; the result is closed to the crossmatching result by human eyes from Australian Radio Astronomers. It could be a choice to integrate huge optical catalogs and radio catalogs which might come from the SKA telescope.

This dissertation also tries to find new ways to integrate resources and makes the usage mode simpler. With techniques from Virtual Observatory and based on the maintenance of local FITS files, we have tried to join the local data and software with the data and services on the internet to make them seamless accessible. The more important aspect is to help astronomers to easily utilize the tools, and focus on the scientific research but not the technical details of the services. On the implementation, we use the Eclipse RCP framework to construct the software which is cross-platform and pluginable. The plug-in structure would encourage the astronomy community to contribute their implementation of data usage or other functions, and eventually make this tool to be a platform which integrates astronomy resources besides software. Some astronomy services are already usable in this platform and easy to use.

At the end of this dissertation, we discuss issues about the management of the videos of astronomy academic reports. Academic reports are also valuable resources for astronomy learning and research. But they usually go unnoticed. Sometimes, the reason is that people do not have the ability to solve the related problems. This dissertation proposes a low-cost scheme, which only needs a simple DV for family, to provide video web broadcasting and Video-On-Demand service. With the accumulation of reports, videos and related documents, we will be able to join these information to the scientific data sets and benefit the

astronomers.

**Keywords:** Virtual Observatory, Resource Access, Data Integration, Cross-match

# 目　　录

# 表　　格

# 插　　图

# 第一章　引言

天文学是一门观测的科学，数千年前古人便已开始对天空进行经年累月的观察、记录。天文学依赖于观测，通过肉眼、胶片或现代数字化的设备来记录数据。科学数据包含了观测数据及其它各式各样的数据，如数值模拟结果也是天文学研究的基础性资源。过去数十年来，随着探测器及信息技术的发展，天文学家对数据的获取能力得到了快速提升。天文领域的数据量也越来越大，天文学研究已从TB 时代进入到了PB 时代[1]，其数据质量及复杂性也随之快速提高[2]。在传统的光学观测数据之外，射电、伽玛射线、X射线等波段数据也越来越多，如图1.1[1] 向我们展示了银河系在不同波段观测的结果。另外还有各类宇宙学、星系演化等数值模拟的数据量也蔚为大观。32亿像素的大口径全天巡视望远镜（Large Synoptic Survey Telescope，LSST）[2][3]，每月将产生近500TB的天文图像[4]。泛星计划（Panoramic Survey Telescope and Rapid Response System，Pan-STARRS）[3][5]，由4部直径1.8米的望远镜构成，每一部望远镜的CCD由64x64的CCD 阵列构成，每块CCD为600x600像素，即每个相机约为14亿像素。约每30秒曝光一次，一张照片尺寸将有2GB，每晚产生的数据即达数TB。数据量如此之大，以至于Pan-STARRS的数据处理系统难于将每一幅图片都保存下来[6]。

海量数据的存储、处理及发布一直以来就是天文学所要面对的巨大挑战。技术的发展似乎总是无法满足天文学家的需求。当通常数据量仍在以GB来计算的时候，天文学家已经开始讨论TB级别的数据；而当我们终于进入了TB 时代，天文学家早以开始探讨PB量级的数据。面对现有软件、算法总是“不够好”的困境，天文学家与技术专家们需要绞尽脑汁去研究更为高效的数据存取设施、更有效的算法、更快速的处理设备以应对与日俱增的数据对传统数据处理、数据分析方法的挑战。

不同波段的数据反应了天体不同方面的性质，全波段观测有助于更全面地认识天体。但是海量的异种数据更加剧了数据处理的难度，如何将同一天体在

[1]银河系多波段观测结果图片来自http://www.astro.ljmu.ac.uk/gal_desc
[2]LSST http://www.lsst.org/lsst/
[3]Pan-STARRS http://pan-starrs.ifa.hawaii.edu/public/home.html

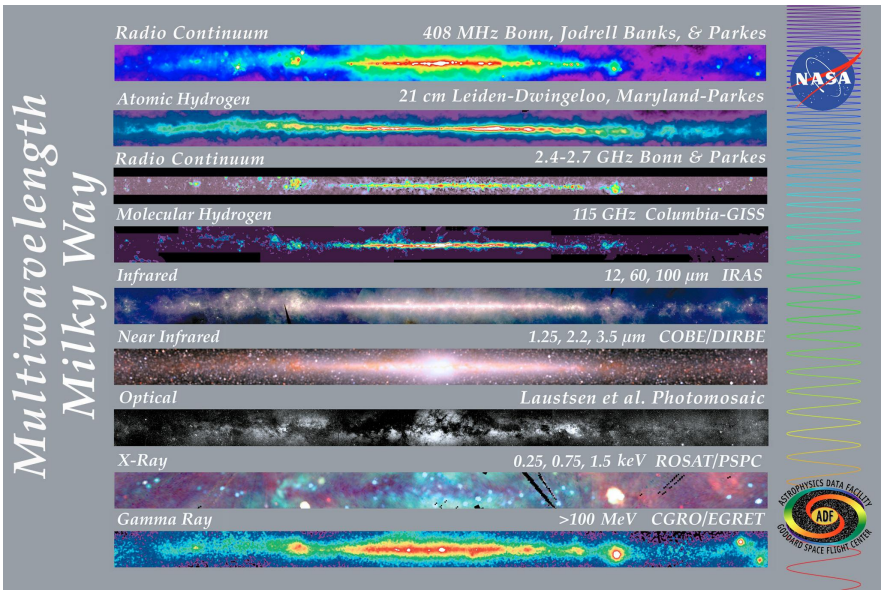


图 1.1: 银河系多波段观测结果展示。

不同波段的数据对应起来，变得异常困难。只靠人工处理数据显然不现实，天文学需要自动化的天文数据处理技术，来将多个波段的数据融合起来。

另一方面，通常各个望远镜或设备所生成的各式各样的数据都是独立保存并发布的。天文学家要向不同的机构查询、获取数据，甚至需要自己带上硬盘跑到各机构所在地拷贝数据。这个数据索取的过程可能要一遍一遍进行多次，非常费时、低效。

## 1.1 虚拟天文台

作为应对上述问题的一个尝试，1990年代，美国的天文学家及计算机科学家共同提出了虚拟天文台（Virtual Observatory，VO）的概念。利用计算机信息技术，虚拟天文台将全世界的天文研究资源无缝、透明地整合起来，为天文学研究及科学教育构建一个数据密集型的网络环境。它将帮助天文工作者冲破凌驾在数据之上的时空限制，并使得基于VO带来的数据共享及科学合作而产生突破性研究成为可能。

虚拟天文台这一概念吸引了来自天文界及计算机界的共同关注，已有19个国家和地区启动了自己的VO项目。2002年6 月，美国虚拟天文台（National

Virtual Observatory，NVO）及欧洲天体物理虚拟天文台（European Astrophysical Virtual Observatory，EURO-VO）与英国天文网格虚拟天文台（UK AstroGrid Virtual Observatory，AstroGrid）共同组建了国际虚拟天文台联盟（International Virtual Observatory Alliance，IVOA）。2002年10月，中国虚拟天文台（China-VO）项目也加入了IVOA。

国际虚拟天文台联盟的任务是促进工具、系统的开发与部署及各组织间的国际协调与合作，将天文数据的国际化应用整合成为一个综合的互操作的虚拟天文台[4]。IVOA 专注于标准规范的发展，并鼓励各方遵守这些规范以使得国际天文界能从中受益。目前IVOA有8个工作组，制定了一系列的数据服务规范[5][7]，并得到多个天文数据中心的大力支持。工作组包括

- 应用程序（Applciations）工作组。专注于为天文学家提供通过VO数据及服务进行天文学研究的软件工具。制定有应用程序简单通信协议（Simple Application Messaging Protocol，SAMP）[8]，规范适用VO 的软件间的互操作工作方式。

- 数据访问层（Data Access Layer）工作组。其任务是制定与规范VO远程数据获取的规范。天文数据服务须按这些标准来规范数据并发布服务，客户端则可以按标准访问相应服务，减少了数据使用者的使用难度。目前已经制定的标准包括简单锥形检索服务标准（Simple Cone Search，ConeSearch）[9][10]、简单图像访问协议（Simple Image Access Protocol，SIA）[11]、表格数据访问协议（Table Access Protocol，TAP）[12]、简单光谱访问协议（Simple Spectral Access Protocol，SSAP）[13]、简单光谱谱线访问协议（Simple Line Access Protocol，SLA）[14]。甚至为了方便进行数据检索，专门制定了一个天文数据检索语言（Astronomical Data Query Language，ADQL）[15]，可以对指定天球区域进行数据检索。

- 数据建模（Data Modeling）工作组。专注于观测数据与模拟数据的元数据间的逻辑关系，调查天文学家真正想要的数据查找、处理、解析方式，并提供一个框架来处理这些问题。目前已经建立测光数据模型（Photometry Data Model）[16]、光谱数据模型（Spectral Data Model）

[4]IVOA介绍http://www.ivoa.net/about/what-is-ivoa.html
[5]IVOA 规范http://www.ivoa.net/documents/

[17]、简单谱线数据模型（Simple Spectral Lines Data Model）[18]、模拟数据模型（Simulation Data Model）[19]等等。

- 网格与网络服务（Grid & Web Services）工作组。其目标是调查与定义VO环境下对网格技术与网络服务技术的应用标准。制订了单点登陆的验证机制（Single-Sign-On Profile: Authentication Mechanisms）[20]、程序参数定义语言（Parameter Description Language）[21]、基于REST技术的网络文件存储服务VOSpace（VOSpace specification）[22]等等协议。

- 语义（Semantics）工作组。关注天文领域的词、短语、句子及文本的含义与解释。制订对天体物理对象、数据类型、概念、事件或其他天文现象的标准描述方式，包括自然语言、查询、翻译、接口国际化等等。定义了VO中所使用的单位（Units in the VO）[23]、词典格式（Vocabularies in the Virtual Observatory）及简单知识系统（Simple Knowledge Organization System）[24]、以及资源统一描述方法[25][26][27] 等等。

- VO事件（VOEvent）工作组。目标是为天空中即时事件的描述、传递、归档与消息发布制标准，以统一的方式描述其内容与含义。为这一目标制订了天空事件报告元数据规范（Sky Event Reporting Metadata）[28]。

- 资源注册（Resource Registry）工作组。IVOA资源注册（IVOA Registry）为天文学家提供各VO 资源的位置、详细描述、使用方式等信息。此工作组主要定义IVOA Registry相关的各种标准，如IVOA统一资源标识符（IVOA Identifiers）[29]、规范IVOA Registry 接口（IVOA Registry Interfaces）[30]、资源元数据定义（Resource Metadata for the Virtual Observatory）[31] 等等。

- VO表格格式（VOTable）工作组。制定XML格式的VO 表格数据交换格式：VOTable[32]。

这些工作与标准共同组成了IVOA的架构，如图1.2[6] ，但这些过于细节的信息并不需要天文学家去了解，而只须由专业天文数据服务开发人员掌握并应用到

[6]IVOA架构图来自http://www.ivoa.net/documents/Notes/IVOAArchitecture/20101123/IVOAArchitecture-1.0-20101123.pdf

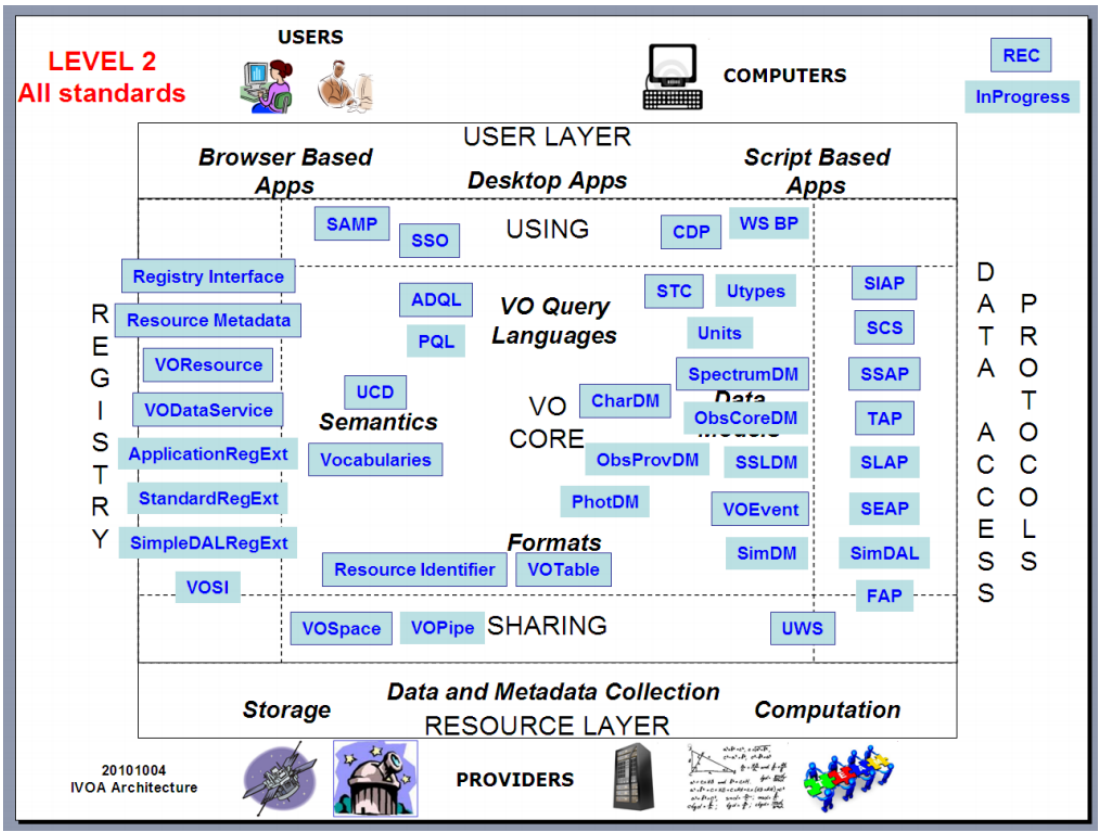


图 1.2: 国际虚拟天文台联盟IVOA的架构。

数据工作上。如果天文学家能掌握像ADQL这样的查询语言，将有助于其更快找到所需的数据。但是这也并不是必须学习的，可以由数据服务人员提供一些简单的图形界面进行辅助。

在虚拟天文台及天文信息学[34]等各领域的专业人士的共同努力下，多年来在互联网上发布了众多天文数据资源及服务。其中法国斯特拉斯堡天文数据中心（Strasbourg Astronomical Data Center，法语简写CDS）[7]及美国虚拟天文台（US Virtual Astronomical Observatory, VAO[8]——前称National Virtual Observatory, NVO[9]）尤为瞩目，拥有大量的数据资源与研发力量，各自提出了一系列的天文服务与标准，并广为全球天文界所使用。

[7]CDS http://cds.u-strasbg.fr/
[8]VAO http://www.usvao.org/
[9]NVO http://www.us-vo.org/old_index.html

法国斯特拉斯堡天文数据中心最为著名的服务是被广泛使用的天体名称解析及文献索引服务（Set of Identifications, Measurements and Bibliography for Astronomical Data，Simbad）[10]，可以通过天体坐标或名称查询一个天体的其他名称、坐标、文献信息等等。其星表服务（VizieR）[11] 目前提供了11052 个星表的查询服务，影响力巨大。此外，CDS还有交互式软件Aladin[12]、在线交叉证认服务（X-Match）[13]、文献服务（CDS Bibliographic Service）[14] 等一系列在线、桌面服务。

美国虚拟天文台立足于美国大量先进仪器带来的海量数据、高校科研力量及软件工业的成熟技术，维护着国际范围的天文资源注册与发布中心（NVO Directory）[15]、数据资源整合搜索工具（DataScope）[16]、在线交叉证认工具（Open SkyQuery）[17]、天区覆盖图服务（Footprint Service）[18][35][37]、光谱检索及分析服务（Spectrum Service）[19] 等等重要服务与工具。文档方面，在天文界影响力巨大的天体物理数据系统（SAO/NASA Astrophysics Data System，ADS）也在美国虚拟天文台的参与者之列。

中国虚拟天文台[36]虽实力较弱，但依托于为郭守敬天文望远镜——又称大天区面积多目标光纤光谱天文望远镜（Large Sky Area Multi-Object Fibre Spectroscopy Telescope，LAMOST）——及南极天文望远镜阵（Chinese Small Telescope ARray，CSTAR）等设备提供数据发布服务，同时独立自主进行相关研究，也走出了一套自己的道路。成为发展中国家虚拟天文台发展的参照项目之一。近年来，China-VO 独自研发了天体信息屏幕取词及检索工具SkyMouse[20][38]、虚拟天文台数据获取工具（Virtual Observatory Data Access Service，VO-DAS）[21][39][40][41][42]、FITS 头入库系统（FITS Header Archiving System，FitHAS）、及本论文工作之一的资源统一管理平台：天文资源管理

[10]Simbad http://simbad.u-strasbg.fr/simbad/
[11]VizieR http://vizier.u-strasbg.fr/viz-bin/VizieR
[12]Aladin http://aladin.u-strasbg.fr/aladin.gml
[13]X-Match http://cdsxmatch.u-strasbg.fr/xmatch
[14]CDS Bibliographyic Service http://cdsbib.u-strasbg.fr/cgi-bin/cdsbib
[15]NVO Directory http://nvo.stsci.edu/vor10/index.aspx
[16]DataScope http://heasarc.gsfc.nasa.gov/cgi-bin/vo/datascope/init.pl
[17] Open SkyQueryhttp://openskyquery.net/Sky/skysite/
[18]Footprint Service http://voservices.net/footprint
[19]Spectrum Service http://voservices.net/spectrum
[20]SkyMouse http://skymouse.china-vo.org/
[21]VO-DAS http://www.china-vo.org/vodas.html


图 1.3: 中国天文数据中心网站首页。

器（Astronomical Resource Manager，ARM）等工具。此外，China-VO还构建和维护了中国天文数据中心[22]，如图1.3，为郭守敬天文望远镜科学试观测数据、BATC 大视场多色巡天观测数据、CSTAR 中国之星测光数据、国家天文台2.16米望远镜观测数据提供管理、发布服务，并为多个世界知名的星表、数据库提供镜像。未来China-VO 还将在国内天文数据发布、资源整合等工作上继续努力，为国内外天文学家提供更好的服务。

## 1.2　交叉证认

天文星表处理是虚拟天文台的核心任务之一。星表是一系列天体信息的

[22]中国天文数据中心http://casdc.china-vo.org/

集合，它们因为有共同的特性、形态、起源或发现方式而被收集起来。更常见的它们是天文巡天观测的结果。星表包含了大量的天体信息及数据，如2微米全天巡天星表（Two Micron All Sky Survey，2MASS）包含了470,992,970个天体；美国海军天文台B1.0星表（United States Naval Observatory B1.0 Catalog，USNO B1.0）包含1,042,618,261个天体；2012 年8月发布的斯隆数字化巡天第9次数据（Sloan Digital Sky Survey Data Release 9，SDSS DR9）其星表包含了1,231,051,050个天体[43][44]。随着设备精度的进一步提高，未来的巡天项目还将提供更多更大的星表。多波段及时序研究均依赖于这些观测结果。这些研究最紧迫的需求之一，便是可以自动化地将多个独立的数据集有意义地联结起来的有效工具[45]。这类工具通常使用交叉证认（crossmatch）技术来将同源的多个观测结果对应起来。美国虚拟天文台2010 年执行计划[46] 中将交叉证认描述为：“交叉证认是对多次观测数据是否同源的一个可信地、物理有效地鉴定，通常这些观测发生在不同时间不同波段之上。”交叉证认技术中有一种主流方法，依靠观测结果在天球上的位置接近程度来判断两个观测结果是否源自同一天体。虽然它仍有一些缺点，比如没有给出匹配概率以及匹配边界是人为自行设定的，但它仍不失为一种实用的方法并已经被广泛使用。星表交叉证认主要集中在几个方面的发展，其中一些是重新对匹配的准确程度进行统计学研究[47][48]；其他的主要针对解决计算方面的问题[49][50][51]。这些努力与尝试从根本上改变了我们对观测的处理方式，并为下一代的分析工具及服务铺平了道路。

国内在交叉证认技术方面也有一些研究。如高丹[52][53][54] 研究了基于分层三角网格（Hierarchical Triangular Mesh，HTM）索引分区与kd-tree 找最近邻算法的交叉证认方法；而赵青[55][56][57] 在高丹工作基础上提出了基于MapReduce模型的分布式交叉证认技术，取得了不错的成果。

但是不管是国内还是国外的研究，均没有很好地考虑将星表天区覆盖图带入星表交叉证认技术中。它们或者完全没有包含这些信息，或者事后才将其包含在内。没有这些天区覆盖信息，将无法说明某些未能匹配的源是因为其亮度未达到观测极限，亦或是它没在另一星表的观测范围内。一些空间查询方法如前述的HTM或HEALPix[49] 可以在指定区域内检索源，但是它们都使用了与交叉证认本身不同的索引策略以提高查询速度，因而需要额外的步骤去重新组织经过交叉证认的数据。

本论文第四章讨论了一种新的方法，将空间限制直接带入到交叉证认中，在交叉证认开始之前排除掉无关区域的数据，并使用与交叉证认一样的数据索引方式。因而能够比之前的方法更快，并且还可以为观测源的光谱能量分布提供缺失源信息限制。

在将星表的天区覆盖信息应用到交叉证认技术中之前，首先需要解决的是如何描述这些信息。虽然可以知道一个星表覆盖在天空中的哪些区域，但是仍需解决如何用一种统一的方式将它们描述出来、并可以直接应用到交叉证认中去。本论文的解决方法是使用球面图形运算工具包（Spherical Toolkit），目前该工具包拥有C#、Microsoft SQL Server、Java 及本人完成的$C++$版本。Spherical Toolkit 主要包含了球面图形运算函数库（Spherical Library）及分层三角网格。在本论文中，将使用此Spherical Library 来对天球上的常见形状进行描述，并可对图形进行交集、并集计算，取得图形外框等等。

大型星表通常存储在数据库中，并使用适当的索引对数据进行排序，以方便对数据的检索。交叉证认算法若能直接在数据库中运行，将免于自行对数据进行提取、排序，而可直接操作数据。Jim Gray的条带算法（Zones Algorithm）[50]便是数据库交叉证认算法中的佼佼者。它通过对天球进行条带划分，给予每个条带一个编号（ZoneID），使用该编号ZoneID及赤经（right ascension，R.A.）对数据进行聚集索引，可以将坐标邻近的数据也在硬盘上物理邻近放置，提高了数据存取的效率。此外，它还通过邻近条带、赤经与赤纬范围限制快速过滤无关数据，使得最后两个天体的实际距离计算被限制在一个很小的球面正方形内。整个交叉证认过程，主要是在进行数据的大小逻辑对比而非数值计算，因而非常高效。美国虚拟天文台交叉证认服务（Open SkyQuery）因此将其选为了核心交叉证认技术[58]。本论文也通过对条带算法进行改进，将天区覆盖信息带入交叉证认过程，而进一步提高效率。由于本论文工作与条带算法关联甚大，将首先在第三章对条带算法进行详细阐述，以便在第四章中使用相同的标识、名称对算法进行描述。

## 1.3　光学星表与射电星表的交叉证认

基于条带算法的算法改进取得了成效，但是这样的算法仍主要只适用光学星表的交叉证认。天文研究需要多波段数据的整合才能更容易接近物理真实，通过结合一个天体的多方面的数据才能真正了解它的实质。为此，进一步的研

究放在了光学星表与射电星表的交叉证认技术方面。

一个天体的射电观测结果可能包含多个部分，如类星体的中心及两侧喷流，在星表中将有三条数据分别对应这三个组成部分。而光学星表对一个天体只有一个观测结果，即该天体的中心部分。在这里就需要将三个射电观测结果与光学观测结果证认为同一天体。

这一部分的工作需要为射电源设计一个模型，以模拟其几何形态或其他特征。再对某个光学源潜在的可证认的射电源做各种可能的假设组合以设置该模型中的各个成分。基于这样的一个模型将会有多个候选的证认结果，需要从中做出选择。这里的方法主要是基于Tamás Budavári与Alex Szalay关于交叉证认的概率论方面的研究。他们的研究主要基于贝叶斯假设推断[47]，详细方法将在第五章中进行介绍。他们的方法甚至可进一步应用于宇宙学事件的交叉证认[59]，如超新星爆发等等。

## 1.4 资源统一管理平台的研究

交叉证认技术注重的是数据的无缝融合。而当前天文的资源和服务的爆炸性增长，不但使得数据量越来越大，工具也随之越来越多了。以往数据分析工具被IRAF、MIDAS、IDL 主导的局面也随着各式工具的发展而被打破，天文学家可以有更多的选择[60]。但选择多了，困惑也多了。这些新工具使用的技术越来越复杂，大量使用了天文学家所不熟悉的计算机语言、规范，使得工具的使用门槛大大提高。IVOA中有相当数量的专业IT人员，虽然使得天文服务开发、工具、存储格式越来越规范，但是这些技术并不是天文学家所感兴趣的。天文学家的感兴趣的工作是科学研究，而不是工具。如果天文学家需要在工具的学习上花费大量时间，他们宁可使用已有的较繁琐的手段。这造成了一些很有用的工具及服务，却不被为天文学家所乐于使用。新的天文服务需要更易于天文学家使用，隐藏技术细节，提供简单、友好的访问环境。

在第六章中将描述我们在这一方面的一个尝试，探讨将本地资源与网络资源整合的技术方法，并构建一个简单易用的平台来实现应用技术。意图在一个平台内将本地数据与网络服务、本地应用程序甚至时下流行的云计算、云存储等简单联结起来，而无须天文学家去了解背后的机制。所有的这些努力将汇集成一个跨平台、插件化的富客户软件：天文资源管理器（Astronomical Resource Manager，ARM）。

## 1.5　视频资源的归档与发布

天文界虽然有完全结构化的数据如星表，易于存储和检索。但是很多非结构化资源如天文讲座视频讲稿等等，也是非常重要的资源，可用于科普、教育、传播等用途。然而，以往的各种培训、讲座经常出现由于通知不到位，或有人因在外出差而无法参加。又或者是非常重要的讲座，因为没有摄影留档而难以向更多人传播。许多重要信息就此绝版，引以为憾事。

这一方面是技术问题，以往摄像设备较复杂，需要专业人员进行操作。然而随着家庭摄像设备，如DV的普及，如何拍摄早已不是问题。而无法到现场参会的问题，可以通过在线直播来解决。国际上已有比较成功的案例，如太空望远镜科学研究所（Space Telescope Science Institute，STScI）的直播系统（STScI Webcasting）[23]。他们有足够的资金来购买专业拍摄设备、切换台及各种专业辅助工具来对讲座进行高质量地存档与发布。但是，对国内的现状而言，需要更为经济、易于部署的方案。本论文第七章提出一个经济、实用的方案，可使用家用DV及国内低成本的设备进行讲座拍摄，通过网络直播讲座、点播过往视频，发布与讲座有关的信息等等。为天文工作者提供更多学习、研究方面的便利。

## 1.6　论文章节安排

第一章，说明本论文工作的背景。描述天文界惊人的数据量及所面临的技术挑战，介绍虚拟天文台这一天文与信息技术相结合的产物，以及中国虚拟天文台所取得的成果，对天文数据、服务整理及融合技术的需求等等。

第二章，介绍球面图形运算工具包的两个组成部分：球面图形运算函数库及分层三角网格。详细描述球面图形运算函数库对球面图形的表示方法，以及分层三角网格对天球的划分方式。球面图形运算函数库将在后面章节中多次被应用到，也是第四章工作的基础。

第三章，详细介绍Jim Gray基于天球条带划分方式的快速交叉证认算法——条带算法——的思想、算法及其在数据库中的实现。条带算法是第四章的基本方法。

第四章，本人完成的一种新颖的交叉证认技术。该技术将星表的天区覆盖信息直接带入交叉证认算法中，可以只对两个星表的重叠区域进行检测，在条

[23]STScI Webcasting https://webcast.stsci.edu/webcast/

带算法的基础上进一步提高了交叉证认速度。此外，由于天区信息的加入，此技术还可以对星表缺失源进行快速检测。

第五章，提出一种基于贝叶斯因子比较的光学星表与射电星表的交叉证认方法。使用直线非对称模型，对邻近射电源各种组合方式的概率进行计算，分析并辨别最可能的组合方式。算法使用C#进行实现并与澳大利亚射电天文学家人工交叉证认的结果进行对比。

第六章，设计并实现了一个天文资源无缝整合的桌面应用软件——天文资源管理器。该软件支持多个操作系统平台，并采用插件化方式进行实现。通过各个插件集成了数个天文分析软件及天文网络服务。它所提供的多个扩展接口也利于天文社区开发力量的加入，以对其功能不断进行增强。

第七章，介绍了对天文视频、讲座资源的一种整合方式：设计与实现国家天文台在线直播系统。分析了系统的硬件与软件系统的架构及配置要求，说明了网络直播、视频点播的流程，并结合Widows Media Service 等平台进行了具体实施。

# 第二章　球面图形运算工具包

天空是天文学家工作的舞台，望远镜则是天文学家探索的利器。自1609年伽利略开创性地将望远镜用于天文观测以来，无数的望远镜一遍又一遍地在星空中搜寻，留下了一份又一份的观测纪录，汇集成各式各样的星表。天文数据浩如烟海，在集合存档之余，有个问题开始凸显出来：哪架望远镜观测了哪些天区？也即，一个星表所覆盖的是哪些区域？这个信息称为望远镜或星表的天区覆盖图（Sky Coverage），或称观测足迹（Footprint）。天区覆盖图可以方便地告诉天文学家，哪些区域已经被此望远镜观测过了。这在运营巡天望远镜时尤其有用，有利于制定望远镜的观测计划。天区覆盖图在研究中也非常重要，在几乎所有的统计学研究中，如光度函数、空间成团等，天区覆盖信息有着无法估量的价值。

对于单次观测，FITS[61]文件中所包含的信息是足够的，通过世界坐标系（World Coordinate System，WCS）可以知道它所覆盖的天区。但是对于使用了不同坐标系统的多次观测或多个天区的观测，尤其是有重叠区域的观测，问题就开始复杂起来。迫切需要有一个工具来精确、方便地对它们进行合并、计算面积等操作。

球面图形运算工具包（Spherical Toolkit）[1] 正是顺应了这一需求而诞生的。它是一个轻量级的软件包，可对天球上的几何形状进行精确计算。Spherical Tookit是由美国约翰霍普金斯大学（Johns Hopkins University）的Alex Szalay 教授及Tamás Budavári，György Fekete等人设计并实现的[62]。目前已经拥有C#、Microsoft SQL Server、Java、C++等多个平台的多个版本。其中C#和SQL[63]版本由Tamás Budavári 等人实现，Java版本由Deoyani Nandrekar-Heinis实现，C++版本由本人实现。SQL版本在美国斯隆数字巡天（Sloan Digital Sky Survey，SDSS）[2]的数据发布系统（SkyServer）[3][64] 上得到了持续地应用。在线交叉证认工具（Open SkyQuery）[4]、天区覆盖图

[1]Spherical Toolkit http://voservices.net/spherical/Default.aspx
[2]SDSS http://www.sdss.org/
[3]SkyServer http://skyserver.org/
[4]Open SkyQuery http://www.openskyquery.net/Sky/skysite/

服务（Footprint Service）[5] 等虚拟天文台服务也使用到了此图形操作库。技术上，C++版本也将可通过用户自定义函数的方式扩展MySQL函数库，使得MySQL也可以使用Spherical Toolkit对图形进行操作。这样做的好处是天文学家可以直接在数据库中高效地完成所需的计算和操作，而无须先将数据从数据库下载下来再单独写程序进行运算。基于SQL的用户自定义函数则可以避免了在数据存取上来回折腾的麻烦，提高了工作效率。

## 2.1 球面图形运算函数库

球面图形运算函数库（Spherical Library）是Spherical Toolkit的底层部分，负责球面图形的构造与运算。任意形状或大小的图形均可像搭积木一样动态生成，并计算出它们的面积。复杂如望远镜或天文星表的天区覆盖图也可以使用此库来生成。

为了便于理解Spherical Library的各种应用方式，首先需要了解其中所涉及的一些基本概念。主要是它所使用的坐标系统、计算方式、基本图形及其组合方法。

### 2.1.1 坐标系

Spherical Library基于两个坐标系，一是天文常用的赤经赤纬(*Right Ascension, Declination*)——或表示为$(\alpha, \delta)$——二维系统，其中赤经的取值范围为$[0°, 360°]$；赤纬的取值范围为$[-90°, 90°]$。赤经赤纬系统也可以很容易地应用于地球的经纬度系统。二是以天球球心为原点（也即地球球心为原点）的$(x, y, z)$三维空间迪卡尔坐标系（Cartesian Coordinate System），并令天球半径为1，单位为1，这样从球心指向球面上任一点所成向量均为单位向量$x^2 + y^2 + z^2 = 1$。其中J2000春分点的坐标为$(1, 0, 0)$，赤道坐标$(90°, 0°)$的点的坐标为$(0, 1, 0)$，天球北极点坐标为$(0, 0, 1)$，天球南极点坐标为$(0, 0, -1)$。$(\alpha, \delta)$ 与$(x, y, z)$两套坐标

[5]Footprint Service http://voservices.net/footprint/

系可以通过公式2.1~2.5相互转换。

$$x = \cos\delta\cos\alpha \tag{2.1}$$
$$y = \cos\delta\sin\alpha \tag{2.2}$$
$$z = \sin\delta \tag{2.3}$$
$$\alpha = \arctan\frac{y}{x} \tag{2.4}$$
$$\delta = \arcsin z \tag{2.5}$$

### 2.1.2　半空间

半空间（Halfspace）是球面上的一部分。通过一个平面切割球面取得，所截取的是该平面法向正向所指向的那一部分球面。如图2.1 [6]所示，左图中一平面将球面切割，如果该平面的法向朝上，则右图中白色部分即为所截得的半空间。该平面可以用一个向量及在该向量方向上的偏移来定义。如，指向北天极的向量(0,0,1)，在该向量方向上的偏移量设为0.85；即以(0,0,1) 为法向量的一个平面与球心的距离为0.85。球面剩下的另一部分，即下方大的球面被另一个平面所截得，该平面的法向指向南天极为(0,0,-1)，且该平面沿法向方向偏离原点-0.85单位。负值表示沿与法向相反的方向移动。如果不指明平面法向，它们看起来是似乎是一致的。偏移量D的取值范围为$[-1,1]$，极端情况如$D=-1$则平面与球面相切并截取到整个球面；若$D=1$则平面也与球面相切，但截取到的球面为空。

### 2.1.3　凸面

凸面（Convex）是任意有限个半空间的交集。几乎所有的简单形状都是凸面，如球面多边形，矩形等等。凸面由一系列半空间定义：$C=\{H_1\cap H_2\cap\cdots\cap H_n\}$。

比如想要得到天球上如图2.2四个端点为a、b、c、d的一个菱形区域，需要使用四个半空间来分别“画”出它的四个边框。它的四条边ab、bc、cd、da中分别属于1、2、3、4四个圆的一段圆弧，每个圆内的部分是天球上被截掉的部分，不属于半空间。因而，1、2、3、4 四个半空间用它们的球面的一部分拼出了菱形区域，但是这四个半空间的交集并不只是这个菱形，四个圆框外部的灰

[6]半空间切割示意图来自http://www.skyserver.org/htm/HtmPrimer/tut_primer.html

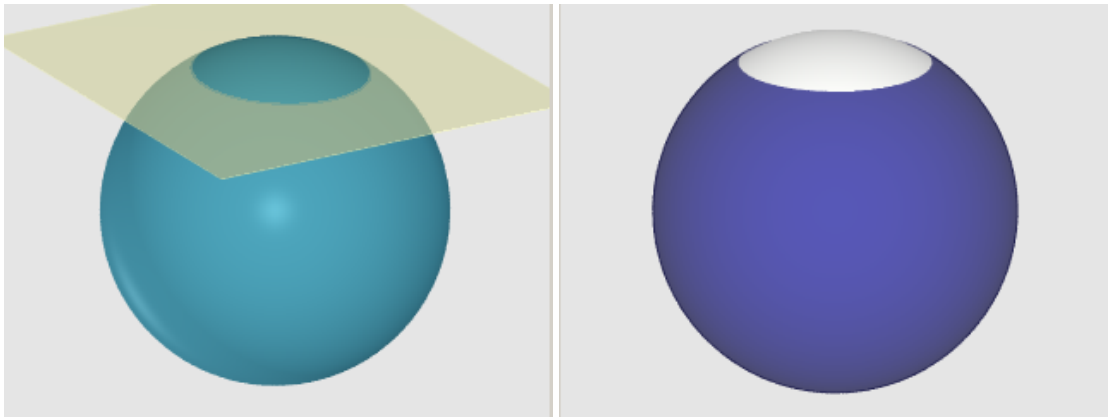

图 2.1: 半空间Halfspace由平面切割球面得到。左图中一平面将球面切割，如果该平面的法向朝上，则右图中白色部分即为所截得的半空间。

色部分也属于四个半空间的交集，图中被填充的部分即是四个半空间所构造的凸面。这时候，需要再增加一个可以覆盖菱形区域且不与外侧灰色部分重叠的半空间来与原来的半空间做交集，这样才能将菱形区域单独取出来。新增加的半空间即5号圆，与其它四个圆不同的是，5号圆内部的部分即是该半空间所覆盖的天球区域。这是在使用半空间构造凸面时需要注意的一个空间问题。

### 2.1.4 区域

区域（Region）是球面上的任意形状。它可以是由零个、一个或多个相连接的球面几何形状组成。为了保持数学上的一致，一个区域或者有一个有限的面积，或者为空。因而，一个点将被视为空区域（点没有面积）。一个区域可以是球面上的任意形状。它最简单的形式如一个圆所在球面上的圆内侧部分。更复杂的例子如球面多边形、矩形，或环绕球面的一个环。

简单地说，一个区域由0个或多个凸面的并集所构成：$R = \{C_1 \cup C_2 \cup \cdots \cup C_n\}$。

由之前的定义可知，一个半空间的交集就是它本身，即半空间就是它自身的凸面。一个凸面的并集也即它本身，即一个凸面也即一个区域。因而，一个半空间本身也是它自己的区域，也是最简单定义的区域。在Spherical Library中的函数所操作的对象都是区域。具体的半空间和凸面通常只在构造一些复杂区域时需要考虑。

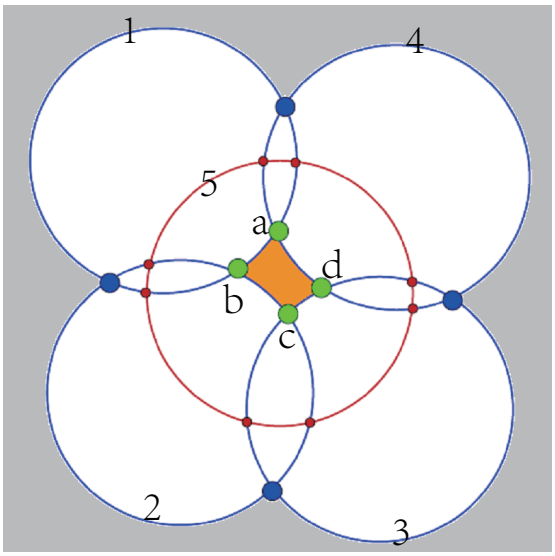


图 2.2: 外围四个圆（1、2、3、4）的边框所标示的半空间的交集是图中央及圆框外侧的被填充区域。为了将中央的菱形区域（其四个端点分别为a、b、c、d）单独取出来，还需要再增加一个覆盖了菱形区域而不与圆框外侧区域重叠的半空间（圆5）与原来的四个半空间做交集。

### 2.1.5　区域定义语言

Spherical Library 中的Halfspace、Convex、Region等概念可以描述球面上大多数的图形。但是，这些概念与常用数学上三角形、圆形、矩形、多边形等概念仍存在较大的差异。为了能够方便地使用日常数学语言来描述形状，Spherical Library设计了一个“区域定义语言”，通过文本的形式来描述一个或多个简单形状。Spherical Library的解释器可以对符合格式要求的形状描述信息进行解析，然后生成对应的Region 数据类型，这样就大大减少了函数库的使用难度。另外，形状交集、并集的计算结果也可以通过符合此语言的文本的形式输出。因而使用者也无须过多了解函数库内部的几何图形运算方法，降低了使用门槛。

区域定义语言的语法如表格2.1所示：

如一个中心位于$(44.2°, 33.1°)$、半径为5角分的球面圆定义为：

REGION CIRCLE J2000 44.2 33.1 5

示例代码如图2.3中的C++程序所示。

表 2.1: Spherical Library 区域定义语言

| | |
|---|---|
| {···}* | 包含0个或多个括号内成员 |
| {···}2 | 包含2个括号内成员 |
| {···}3+ | 包含3个或更多括号内成员 |
| null | 空字符串 |
| R | 角度，单位为角分 |
| D | 介于-1与1之间的实数 |
| | 用于表示从球心沿(ra,dec)或(x,y,z)所指方向的偏移量 |
| ra | 赤经，单位为度 |
| dec | 赤纬，单位为度 |
| x y z | 单位球面上的单位向量 |

| | | |
|---|---|---|
| regionSpec | := | `REGION{areaSpec}` * \| areaSpec |
| areaSpec | := | `circleSpec`\|`rectSpec`\|`polySpec`\|`hullSpec`\| convexSpec |
| convexSpec | := | `CONVEX J2000{ra dec D}` * |
| | \| | `CONVEX CARTESIAN{x y z D}` * |
| | \| | `CONVEX{x y z D}` * |
| | \| | null |
| rectSpec | := | `RECT J2000{ra dec}` 2 |
| | \| | `RECT CARTESIAN{x y z}` 2 |
| circleSpec | := | CIRCLE J2000 ra dec R |
| | \| | CIRCLE CARTESIAN x y z R |
| polySpec | := | `POLY J2000{ra dec}` 3+ |
| | \| | `POLY CARTESIAN{x y z}` 3+ |
| hullSpec | := | `CHULL J2000{ra dec}` 3+ |
| | \| | `CHULL CARTESIAN{x y z}` 3+ |

```
#include "./SphericalLib/global.h"
#include "./SphericalLib/Region.h"
#include "./SphericalHTM/Parser.h"

int main(int argn, char * argv[])
{
    std::string spec = "REGION CIRCLE J2000 44.2 33.1 5";
    boost::shared_ptr<Region> reg = Parser::compile(spec);
    reg->Simplify();
    std::cout<<"Area of this region"<<reg->getArea()<<std::endl;
    std::cout<<reg->ToString();
    return 0;
}
```

图 2.3: 通过“区域定义语言”来构造一个中心在$(44.2°, 33.1°)$、半径为5角分的球面圆。

## 2.2　分层三角网格

分层三角网格全称为Hierarchical Triangular Mesh（HTM），即对天区的递归的多层次三角形划分。HTM最先由Kunszt等人提出[65] [66]，现在已经发展到HTM2[67]。HTM直接建构于Shperical Library 之上，是对Spherical Library的扩展。如图2.4[7]所示。它从一个八面体开始，这也是它的第0 层。当把八面体投影到单位球面上，将产生8个球面三角，4个在北半球，4个在南半球。极点是各半球的4个球面三角所共有的顶点。极点的所有对边构成了赤道。可以想象把一个八面体放置到球体中，其两顶点即为球面的两极点，而其他四边则均匀分布在赤道上。而球面多边形的边则是八面体各条边在球面上的投影。

8个球面三角，按南北方向分别命名为N0至N3及S0至S3。我们将它们称为0级三角元素（Trixel）。每一个三角元素可被分解成四个更小的三角元素。如图2.5所示。其分解方法为，在球面三角各边取中点，并通过球形大圆的圆弧线段将三点连接起来。所有的新旧边及顶点构成了新的四个小三角的顶点与边。这种方法可以通过无限递归的方式将一个三角元素切分成越来越小的三角元素。

这种划分方式很自然地提供了一种对三角元素进行命名的方法。每一个三角元素有三个顶点分别被命名为0、1或2。其各自对应的边上的中点命名为0’、

[7]本小节关于HTM的图解来自http://www.skyserver.org/htm/index.html

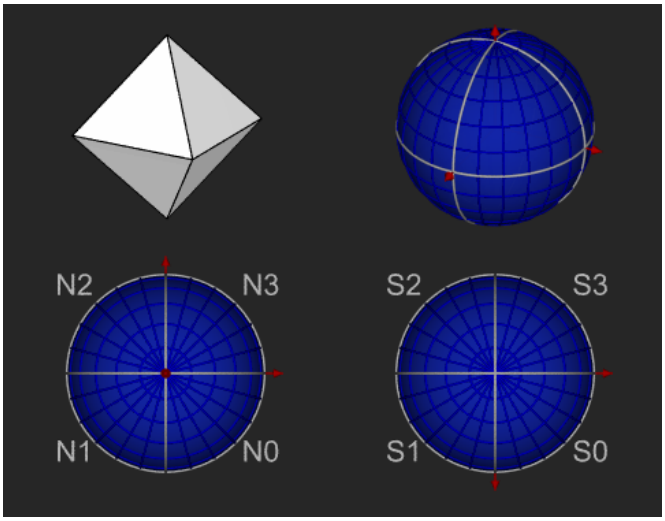


图 2.4: 八面体及其球面投影的三个视图。

1’与2’。新的被切分出来的小三角元素们通过在父三角的编号后方添加0、1、2、3进行命名。与父三角共享同一个顶点的，添加的是该顶点在父三角中的编号（如子三角与父三角共享的顶点为0，则其编号将被命名为父三角的编号+0），剩下的中央的三角则添加3。这样一来，小三角的编号将变得越来越长，而其长度也标示了其所处的划分层级。也即在这样的划分方式中的顶点们通过一个先导的1 bit、0 级的[0…7]及其子划分的[0…3]进行编号。这也给三角元素及其中点赋予了一个64bit的唯一编号，我们将此编号称为HtmID。

HtmID最小的有效编号是8。虽然此划分方法可无限地进行下去，但是64bit的长度将会在第31层划分时被分配完毕。而25层划分实际上已经足够精确，它所指向的是地球上大概0.6米的长度或单位球面上约0.02角秒的宽度。注意，此编码制并不是对正整数的完全覆盖，也并不意味着所有的数字都对应着一个有效的HtmID。

一个区域可由多个三角元素覆盖，并且它们可能并不是都处于同一划分层级。如果某个形状的内部区域已经完全覆盖了某一个三角元素，则该三角元素无须再进行划分。而边缘的很多碎片将划分出许多小三角元素。为了便于处理多个不同层级的HtmID，我们可以统一使用一个特定层级的编号（范围）来描述所有的三角而将无须所有的三角元素都划分到这一层级。从实际应用角度出发，我们将其定为20级。因而，所有低于20级的三角元素，都可

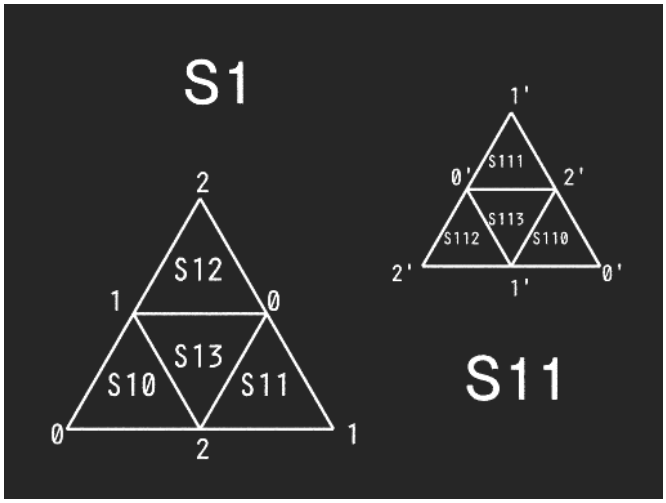


图 2.5: HTM的编号方式图解。

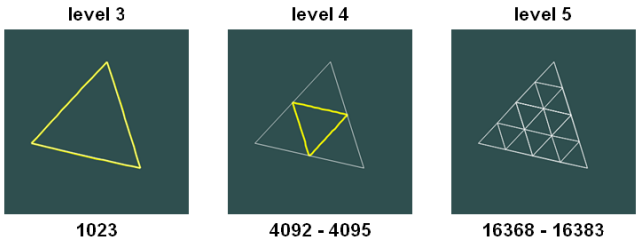


图 2.6: 同一个区域的三个不同层次划分。

由一系列的20级三角元素的HtmID来描述。这些HtmID是一串连续的整数。如图2.6所示，一个3级HtmID所指明的区域被编号为2012。到了第5级，该区域已被划分了两次，其所有5级子三角的HtmID范围从16368至16383。而到了20级，其HtmID范围是17575006175232 - 17592186044415。

而如何将位置与HtmID联系起来呢？如图2.7所示代码，指定一个划分级别12，能得到一个覆盖了指定点的三角元素的HtmID。另一方面，此HtmID指向的三角元素可被转化成三个点分别对应其三个顶点。

通过三角划分出来的位置信息，HTM还可以解决诸如：某个区域覆盖了哪些小三角（如图2.8），并进一步找出哪些天体在此区域中，只需要对比两者的HtmID范围即可；判断某个天体是否在某个区域中；某个天体附近有哪些天体，等等与区域有关的问题。这些查找能力在面对大星表时尤为重要，在统计

```
#include "./SphericalLib/global.h"
#include "./SphericalLib/cartesian.h"
#include "./SphericalHTM/global.h"
#include "./SphericalHTM/Trixel.h"
using namespace Spherical::Htm;
using namespace Spherical;

int main(int argn, char * argv[])
{
    long long htmid=Trixel::CartesianToHid (1, 0, 0, 12);
    std::cout<<htmid<<std::endl;//htmid=260046848
    Cartesian_PTR a,b,c;
    Trixel::ToTriangle(htmid, a, b, c);
    std::cout<<a->ToString()<<std::endl;
    std::cout<<b->ToString()<<std::endl;
    std::cout<<c->ToString()<<std::endl;
    return 0;
}
```

图 2.7: 取得覆盖J2000春分点的12级小三角元素的HtmID及该三角元素的三个顶点的坐标。

分析、数据挖掘等研究中占有相当重要的位置。近年来如SDSS等包含数亿条数据的大星表都放置在数据库中，与数据直接结合的SQL版本的HTM更展现出了它的强大威力。但是目前的SQL版本只能应用于Microsoft SQL Server，这是一个商用的收费数据库。天文界广泛地使用着免费的MySQL及Postgre SQL数据库，它们目前仍然无法直接受益于HTM强大的区域检索能力。C++版本的Spherical Toolkit有望将HTM带到免费数据库领域。

```
#include "./SphericalLib/global.h"
#include "./SphericalLib/Region.h"
#include "./SphericalHTM/global.h"
#include "./SphericalHTM/Parser.h"
#include "./SphericalHTM/Cover.h"
#include "./SphericalHTM/HidRanges.h"
using namespace Spherical::Htm;
using namespace Spherical;

int main(int argn, char * argv[])
{
    Region_PTR reg = Parser::compile("REGION CIRCLE J2000 44.2 33.1 5");
    reg->Simplify();

    std::vector<Int64Pair> table = Cover::HidRange(reg);
    std::vector<Int64Pair>::iterator it = table.begin();
    for(;it!=table.end();it++)
    {
        std::cout<<(*it).lo<<", "<<(*it).hi<<std::endl;
    }
    return 0;
}
```

图 2.8: 取得以$(44.2°, 33.1°)$为中心、半径为5角分的圆所覆盖区域内的小三角的编号范围。

## 2.3　小结

面对天文界对球面图形的计算需求，球面图形操作工具包（Spherical Toolkit）应运而生。其基础部分的球面图形运算函数库（Spherical Library）是一个小巧、强大的函数库，可在天球上“画”出所需要的各种图形，并对图形进行交集、并集、面积计算等操作。比较复杂的应用还可将其用于望远镜或星表的观测区域的描绘。这在天文数据统计分析、数据挖掘等应用中也是非常重要的。Spherical Toolkit的另一个重要成员：三角分层网格（HTM）直接建构于Spherical Library之上，已经成功应用于SDSS巡天项目的数据发布服务上。实现了星表索引、天区覆盖检索、邻近天体查找等非常基础也非常重要的功能。目前Spherical Toolkit只支持Microsoft SQL Sever数据库。但是本人所实现的C++ 版本可以用于扩展MySQL 或Postgre SQL的函数库，使得在天文领域使用得更广泛的这两类数据库也能受益于Spherical Toolkit的强大能力。

# 第三章　高效交叉证认算法——条带算法及其数据库实现

条带算法（Zones Algorithm）[68][69]由微软研究院的Jim Gray[1]提出的一个快速的交叉证认算法，在数据库中表现尤为优异。与分层三角划分方法不同，Zones Algorithm把天球按一定赤纬间隔划分成一个个环形带（或称条带），如图3.1。每一条带的最基本信息即为其赤纬范围$[\texttt{DecMin}, \texttt{DecMax})$。由此可赋予每一条带一个编号（ZoneID），该编号可由公式3.1求得，其中赤纬$\delta$可选取条带中央一点的赤纬，即令$\delta = \frac{(\texttt{DecMin}+\texttt{DecMax})}{2}$。若令每个条带的高度为$h$，则整个天球将被划分为$\lceil \frac{180}{h} \rceil$ 个条带，编号范围$\left[0, \lfloor \frac{180-h/2}{h} \rfloor\right]$。一个天体所处的条带同样可以通过公式3.1计算得出，由此也可通过对比两个天体的ZoneID 来判断是否处于邻近条带中。

$$\texttt{ZoneID} = \left\lfloor \frac{\delta + 90^\circ}{h} \right\rfloor \tag{3.1}$$

[1]Jim Gray http://research.microsoft.com/en-us/um/people/gray/

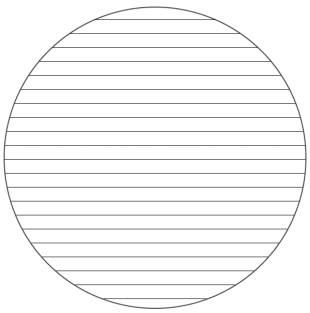

图 3.1: Zones Algorithm 将天球划分成了一条条赤纬度间隔相等且相互平行的环形带（条带）。

## 3.1 条带算法对赤纬的过滤

Zones Algorithm的基本思路是尽量少做计算、多做比较，从而提高计算效率。它只在特定的一个很小的以天体坐标$(\alpha, \delta)$为中心，交叉证认临界值$\theta$的2倍为边长的正方形区域内进行精确检查。而凡是在以天体为中心，半径为$\theta$的圆内的天体，则认为它们是同一天体。

赤纬在球面上是均匀分布的，因而很容易可获得正方形区域在赤纬方向上的覆盖范围，即$(\delta-\theta, \delta+\theta)$。但是，每次比较都要做一次浮点数的减法和加法仍然过于浪费计算资源了。Zones Algorithm用了更为巧妙的办法，先对每个天体的ZoneID进行检查。如果两个天体的ZoneID相差一定数值，则它们绝对不可能匹配，这样就只需要做一次整数运算。进一步的，ZoneID是一个有限范围的值，可以将所有临近的ZoneID做成一个对比表ZoneZone缓存在内存中。只要一对ZoneID没有出现在对比表中，则两个天体不会匹配。

以$\theta = 7.0'' \approx 0.00194°$，条带的高度$h = 7.1'' \approx 0.00197°$为例。若一天体$X$的赤纬为$\delta = 44.1°$，则其$\texttt{ZoneID} = \lfloor \frac{44.1}{0.00197} \rfloor = 22385$。它在赤纬方向上的搜索范围为$(44.1-0.00194, 44.1+0.00194) = (44.09806, 44.10194)$。下边界44.09806所在条带$\texttt{ZoneID} = \lfloor \frac{44.09806}{0.00197} \rfloor = 22384$，上边界44.10194所在的$\texttt{ZoneID} = \lfloor \frac{44.10194}{0.00197} \rfloor = 22386$，也即与天体$X$匹配的其他天体只能出现在$X$的相邻$\lceil \frac{\theta}{h} \rceil = \lceil \frac{7.0}{7.1} \rceil = 1$个条带内，也就是$[22385-1, 22385+1] = [22384, 22386]$。在对比表ZoneZone中，只需要保存如表格3.1所示三个记录。在条带$\texttt{ZoneID} = 22385$内的其他天体也均有此性质，因而只要任何天体与条带$\texttt{ZoneID} = 22385$内的天体进行匹配时，若其本身ZoneID不在此表中所列出的$\{22384, 22385, 22386\}$三个数值内，则无须对它们进行进一步的比较。需要注意的是，当$\theta > h$时，相邻条带不只是相邻的3个，要用公式$\lceil \frac{\theta}{h} \rceil$来确定条带搜索范围。

表 3.1: 对于$\theta = 7.0''$、$h = 7.1''$的情况，在邻近条带对比表中保存的与$\texttt{ZoneID} = 22385$相关的三行数据

| ZoneID1 | ZoneID2 |
|---|---|
| 22385 | 22384 |
| 22385 | 22385 |
| 22385 | 22386 |

由于每个条带内均有数个天体，因而，在ZoneZone表内对ZoneID进行对照要比直接对天体的赤纬搜索范围进行对照要迅速得多。但是ZoneID的过滤只是很粗糙的一步过滤，用于快速去掉多数的无效候选天体。然后仍然需要判断在邻近条带中的天体是否是在$(\delta-\theta,\delta+\theta)$的范围内。能进行这一步对比的天体只可能是在$X$天体的邻近条带中，这时候做两次算术运算的时间代价已经很低了，也是值得的。

## 3.2　条带算法对赤经的过滤

实现赤纬方向上的过滤之后，还需要对赤经方向进行比对，以过滤掉在相邻条带内与天体在赤经方向上距离超过$\theta$的天体。但是与赤纬不同，赤经的分布是不均匀的。在天球不同的纬度上，同样角度的赤经所覆盖的距离是不同的。而交叉证认对天体进行匹配使用的是距离判断。因而，若想要通过对比赤经来过滤掉距离较远的天体，需要先计算出不同纬度上一段距离所能覆盖的赤经范围Alpha。Alpha可通过公式3.2进行计算，它与$\theta$及$\delta$相关的，在不同的赤纬上均不相同。简单起见，对一个条带使用同一个Alpha值，即最靠近极点的$\delta$边界值。另外，在靠近北（南）极点时，如$|\delta|>89$，需要搜寻的区域可能包含极点在内，角度的膨胀会变得很严重，需要把整个360度都考虑进来。这时候可以直接令$\mathtt{Alpha}=180$。之所以是180而不是360，是因为赤经过滤的方法是检测其它天体与是否在$X$的赤经$\alpha$附近的$(\alpha-\mathtt{Alpha},\alpha+\mathtt{Alpha})$范围内，令$\mathtt{Alpha}=180$即包含了$X$所在条带上的所有天体。

$$\mathtt{Alpha}=\left|\arctan\left(\frac{\sin\theta}{\sqrt{\cos(\delta-\theta)\cos(\delta+\theta)}}\right)\right| \tag{3.2}$$

通过ZoneID的粗过滤及对赤纬及赤经过滤后，留下了以$\theta$的两倍为边长的正方形区域。整个的过滤过程可以用图3.2来表示。三个条带表示ZoneID粗过滤的过滤范围，内侧虚线正方形框表示赤纬过滤及赤经过滤范围。最中央的实线圆表示在此圆内的天体都将被视为与天体匹配。正方形区域内的天体需要精确地计算它们与$X$的实际距离。由于直接使用赤经赤纬$(\alpha,\delta)$进行距离计算比较复杂。Zones Algorithm使用了2.1.1中所提到的三维空间笛卡尔坐标系。将$(\alpha,\delta)$先转换为$(x,y,z)$三维坐标，再计算两个天体的三维空间距离。通常匹配边界值$\theta$都是用角度来表示的，如上文中的$\theta=7.0''$。需要把它所对应的球面

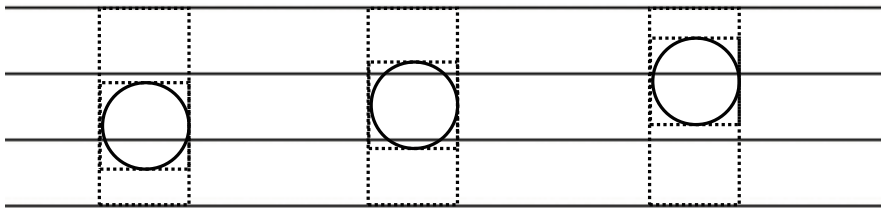

图 3.2: $\theta < h$时，同一个条带上的三个不同位置天体的搜索范围。三个条带表示ZoneID粗过滤的过滤范围，内侧虚线正方形框表示赤纬过滤及赤经过滤范围。最中央的实线圆表示在此圆内的天体都将被视为与天体匹配。

圆弧长度转换为三维坐标直线距离dist。可用公式3.3来完成转换。另外，因为星表中的天体数量巨大，$(\alpha, \delta)$到$(x, y, z)$的转换应该在交叉证认的准备阶段完成，以避免巨量重复计算。

$$\texttt{dist} = \sqrt{4\sin^2\left(\frac{\theta}{2}\right)} \tag{3.3}$$

## 3.3 环绕处理

赤经$\alpha$的取值范围为$[0°, 360°]$，由于条带是一个环，它的起点0°亦是终点360°。但是，在数值上0与360却是相差最大的数值。原本在$\alpha = 0°$附近的天体，只要它们的距离足够近就需要进行匹配计算。但现在如果有一对很接近的天体$a$、$b$，如图3.3，a在$\alpha = 0°$左边一点，如$\alpha_a = 359.99999°$；而b在$\alpha = 0°$ 右边一点点，如$\alpha_b = 0.00001°$。它们很可能能够被证认为是同一天体，这时候却因为$b$不在$(0.00001° - \texttt{Alpha}, 0.00001° + \texttt{Alpha})$的范围内而被跳过了。

针对$\alpha = 0°$附近出现的这种情况，有两种解决方案。一是扩展赤经的过滤条件：除了$\alpha_a \in (\alpha_b - \texttt{Alpha}, \alpha_b + \texttt{Alpha})$之外，如果满足$\alpha_a - 360 \in (\alpha_b - \texttt{Alpha}, \alpha_b + \texttt{Alpha})$或$\alpha_a + 360 \in (\alpha_b - \texttt{Alpha}, \alpha_b + \texttt{Alpha})$，则$a$不能被$b$的赤经范围过滤掉；二是当$\alpha_a + \texttt{Alpha} > 360$时，在星表中添加一条与$a$相同的数据，但是赤经修改为$\alpha_a - 360$。第一种方法的劣势是增加了计算量，但是对星表没有影响。第二种方法通过增加星表数据量来降低计算量，以空间换时间。另外，因为Alpha值是与匹配边界值$\theta$相关的，如果$\theta$很大，最坏的情况

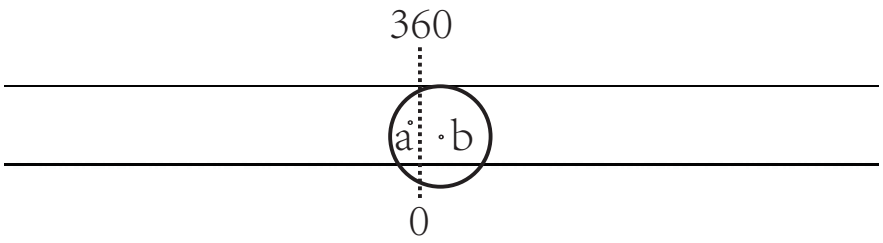


图 3.3: $\alpha = 0°$处出现了环绕问题，左侧的点a将有可能被漏掉，虽然它在右侧点b的有效搜索匹配范围内。

下，可能需要把半张星表做一次$\alpha - 360$之后添加到原来的星表上，即星表增大了50%。不过，如果保持使用同一个很少$\theta$，如$\theta = 7.0''$，则星表可能只需要扩张0.2%即可。

增大星表的方法还有一个问题，就是位于$\alpha = 0°$左侧的两个天体（称之为$c$、$d$）有可能会被匹配两次。因为$c$原始的赤经$\alpha_c$ 能直接通过$(\alpha_d - \texttt{Alpha}, \alpha_d + \texttt{Alpha})$的过滤，修改后的$\alpha_c - 360$也能通过$d$ 在星表中新增的$\alpha_d - 360$对应的$(\alpha_d - 360 - \texttt{Alpha}, \alpha_d - 360 + \texttt{Alpha})$的过滤。为了避免这种情况，可以添加一个判断：如果$\alpha_c$或$\alpha_d$同时为负，则不能穿过过滤器。因为只是简单地符号判断，第二种方法仍然要比第一种方法要高效。

## 3.4　条带算法的数据库实现

由于整个过程是纯粹的数值计算，不需要调用第三方函数库，也无须特别的平台支持，Zones Algorithm很适合直接在数据库中运行，并且与数据库系统没有依赖。只要是关系型数据库就可以运行使用标准SQL语法实现的Zones Algorithm交叉证认算法，真正做到了多平台可插拔运行。Zones Algorithm尤其适合关系型数据库还因为它可以充分利用数据库的索引机制。在Zones Algorithm中几乎所有的数据表都使用了聚集索引$(\texttt{ZoneID}, \alpha)$，即以ZoneID优先的顺序对数据进行排序，相同ZoneID的数据则按赤经$\alpha$ 排序。由于聚集索引直接决定了数据的物理存储，这意味着，所有的数据在硬盘上的存储也是按ZoneID优先的顺序排列，相同条带内的数据再按赤经$\alpha$ 的顺序排列。这样邻近的天体也就将保存在邻近的物理存储区域上。即使是不同条带上的数据，也依照$\alpha$ 进行排列的。而交叉证认本身查找的就是这些ZoneID 很近以及$\alpha$很近的

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.ZoneDef') AND type in (N'U'))
    DROP TABLE dbo.ZoneDef
GO

CREATE TABLE dbo.ZoneDef (
    ZoneID INT NOT NULL,
    DecMin FLOAT NOT NULL,
    DecMax FLOAT NOT NULL,
    Alpha FLOAT NOT NULL,
    PRIMARKEY(ZoneID)
)
GO
```

图 3.4: 条带定义表ZoneDef主要保存其编号、赤纬上下边界与Alpha值。

数据。这样，Zones Algorithm 极大提高了数据读取的IO性能，一定程序上减少了计算机硬件中最为缓慢的文件读取速度的影响。

SQL版本的Zones Algorithm需要将一些中间结果或辅助信息保存到数据表中。它们主要有条带定义表（ZoneDef），邻近条带对比表（ZoneZone），两个或多个星表索引（XXXIndex）等。

### 3.4.1 条带定义表

条带定义表（ZoneDef）的定义如图3.4所示。ZoneDef主要用途是保存整个天球被划分的情况，即被划分为多少个条带，每个条带的上、下边界值是多少，还有条带中最大的Alpha值。填充ZoneDef表也极简单，如图3.5所示。只需要从南天极点的赤纬值$-90$开始，每次步增一个指定的条带高度$h$，这里是$h = 7.1'' = 7.1/3600.0°$，直到到达或刚刚越过北天极点。Alpha值的处理较麻烦一些，因为计算步骤较复杂，需要建立一个专门的函数来进行处理。如图3.6建立了一个函数ZonefAlpha，并使用UPDATE 语句更新整个ZoneDef表中的Alpha值。

### 3.4.2 星表索引表

星表索引表至少应有两个，即参与交叉证认的两个星表$X$、$Y$，它们的索引表可分别表示为为诸如XIndex和YIndex。由于原始星表已经有一个聚集索引，不能在其中再添加一个形如(ZoneID, $\alpha$) 的聚集索引。如图3.7通常可以另

```
DECLARE @zoneHeight FLOAT, @maxZone BIGINT, @minZone BIGINT, @zoneDec FLOAT
SET @zoneHeight = 7.1/3600.0
SET @minZone = 0
SET @maxZone =  FLOOR(180.0/@zoneHeight)

WHILE @minZone <= @maxZone
BEGIN
    SET @zoneDec = @minZone * @zoneHeight - 90;
    INSERT dbo.ZoneDef VALUES (@minZone, @zoneDec, @zoneDec+@zoneHeight, -1)
    SET @minZone = @minZone + 1
END

UPDATE dbo.ZoneDef SET DecMax=90.0 WHERE DecMax>90
```

图 3.5: 为ZoneDef表写入整个天球的划分信息。

外再专门建一个索引表，保存天体的赤道坐标$\alpha, \delta$，天体在星表中的唯一编号ObjID以及单位球面上的三维空间笛卡尔坐标$(x, y, z)$，尤其是该天体所在条带的编号ZoneID。

填充索引表时，最重要的是从原始星表拿到赤经赤纬坐标及唯一编号。如图3.8所示，在插入赤道坐标信息的同时可将三维坐标$(x, y, z)$按公式2.1∼2.3计算出来，同时还可以根据公式3.1计算出它的ZoneID。之所以在代码中在表一级上加锁WITH (TABLOCK)，是为了防止数据库操作日志过大，也是为了防止其他程序也对这个表做写入操作，保证数据完整的同时也可以稍微提高一点速度。

将索引表建立并填充起来之后，还需要考虑解决$\alpha = 0°$附近的环绕问题。如果采用第3.3节中的方案一，则无须在此做更多处理。但通常选择的是更快速的方案二，因而，有必要在索引表建立之后加入所需的冗余信息。如图3.9所示，若一个天体的赤经$\alpha$与它所处条带上的最大的Alpha值相加大于360°即意味着它有可能会被0°右边的天体所匹配。因而需要将一行新的数据插入到XIndex表中，该新记录与该天体原来的数据相同，只是需要将赤经更新为$\alpha -$ Alpha。由于匹配边界值很小，实际增加的数据并不会很多。比如SDSS DR6主星表有超过2.3 亿条数据，而添加的冗余数据只有321条。

### 3.4.3　邻近条带对比表

邻近条带对比表（ZoneZone）用于提示数据库某一个条带上的数据需要与

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.ZonefAlpha')
    AND type in (N'FN', N'IF', N'TF', N'FS', N'FT'))
    DROP FUNCTION dbo.ZonefAlpha
GO

CREATE FUNCTION dbo.ZonefAlpha(@theta float, @dec float)
RETURNS FLOAT AS
BEGIN
    IF ABS(@dec)+@theta > 89.9 RETURN 180
        RETURN(DEGREES(ABS(ATAN(
            SIN(RADIANS(@theta))
            / SQRT(ABS(
                COS(RADIANS(@dec-@theta)) * COS(RADIANS(@dec+@theta))
            ))
        ))))
END


DECLARE @zoneHeight FLOAT, @theta FLOAT,
SET @theta = 7.0/3600.0
SET @zoneHeight = 7.1/3600.0

UPDATE dbo.ZoneDef
    SET alpha = CASE WHEN ABS(DecMax) < ABS(DecMin)
        THEN dbo.ZonefAlpha(@theta, DecMin - @zoneHeight / 100)
        ELSE dbo.ZonefAlpha(@theta, DecMax + @zoneHeight / 100)
    END
```

图 3.6: Alpha计算函数的构造，及ZoneDef表中Alpha值的更新。

哪几个条带上的数据进行对比。如果只使用赤纬搜索范围$(\delta-\theta, \delta+\theta)$，虽然也能找到所有需要的天体，但是数据库引擎无法迅速过滤掉不相干的别的条带上的数据，即数据引擎将搜索整个星表！因而有必要使用此ZoneZone表来提示数据库，再利用星表索引上已经建好的以ZoneID为主的聚集索引，所有的数据都可以快速找到，并且集中在相邻的物理地址上。如图3.10所示，ZoneZone表简单地就是存储两个星表的有效条带的对应编号。为了避免在交叉证认过程中回ZoneDef表去找Alpha值，填充ZoneZone表的过程中也将Alpha也保存了一份，对应于第二个星表中的条带。

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.XIndex') AND type in (N'U'))
    DROP TABLE dbo.XIndex
GO

CREATE TABLE dbo.XIndex (
    ZoneID INT NOT NULL,
    ObjID BIGINT NOT NULL,
    RA FLOAT NOT NULL,
    Dec FLOAT NOT NULL,
    Cx FLOAT NOT NULL,
    Cy FLOAT NOT NULL,
    Cz FLOAT NOT NULL,
    PRIMARY KEY(ZoneID, RA, ObjID)
)
GO
```

图 3.7: 星表索引表存储了天体在星表中的唯一编号ObjID、赤道坐标(RA, Dec)、单位球面三维坐标$(x, y, z)$ 以及天体所在条带编号ZoenID。

```
DECLARE @zoneHeight FLOAT
SET @zoneHeight=7.1/3600.0

INSERT dbo.XIndex WITH (TABLOCK)
SELECT CONVERT(INT,FLOOR(([DEC]+90.0)/@zoneHeight)) ZoneID,
ObjID, RA, DEC,
    COS(RADIANS(DEC))*COS(RADIANS(RA)) Cx,
    COS(RADIANS(DEC))*SIN(RADIANS(RA)) Cy,
    SIN(RADIANS(DEC)) Cz
FROM CatalogX
ORDER BY ZoneID, RA, ObjID
```

图 3.8: 从原始表获取的最重要信息是天体的编号及赤道坐标，剩下的信息均可通过赤道坐标计算出来。

```
INSERT dbo.XIndex WITH (TABLOCK)
SELECT t.ZoneID, ObjID, RA-360, Dec, Cx,Cy,Cz
FROM dbo.XIndex t
    JOIN dbo.ZoneDef d on d.ZoneID = t.ZoneID
WHERE RA + d.Alpha > 360
```

图 3.9: $\alpha = 0°$附近的环绕问题的第二个解决方案，是在索引表中加入冗余数据，以空间换时间。

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.ZoneZone') AND type in (N'U'))
    DROP TABLE dbo.ZoneZone
GO

DECLARE @n INT
SET @n=CEILING(7.0/7.1)

CREATE TABLE dbo.ZoneZone (
    ZoneID1 INT NOT NULL,
    ZoneID2 INT NOT NULL,
    Alpha2 FLOAT NOT NULL,
    PRIMARY KEY(ZoneID1, ZoneID2)
)
GO

INSERT dbo.ZoneZone WITH (TABLOCK)
SELECT Z1.zoneid, Z2.zoneid, d2.alpha
FROM (SELECT DISTINCT ZoneID FROM dbo.ZoneTable1) z1
    JOIN (SELECT DISTINCT ZoneID FROM dbo.ZoneTable2) z2
        ON Z2.zoneid between Z1.zoneid - @n and Z1.zoneid + @n
    JOIN dbo.ZoneDef d2 ON d2.ZoneID = Z2.ZoneID
ORDER BY 1, 2
```

图 3.10: ZoneZone表保存的是两个星表的有效ZoneID及它们的对应关系，并保存一份Alpha值以方便查找。

### 3.4.4 交叉证认过程

在准备好了星表索引XIndex、YIndex，邻近条带对照表ZoneZone，计算出所需条带的Alpha 值之后。最后的交叉证认过程可以开始进行了。如图3.11所示，整个的交叉证认过程从第一个星表的索引出发，先到ZoneZone表查看需要到第二个星表的哪些条带上查找数据，这是第一步的ZoneID粗过滤。第二步需要确定通过了第一步过滤的数据是否在正方形过滤区域的有效赤经范围内。第三步则是检查通过前两步过滤的数据是否在实际有效的赤纬范围内，而不仅仅是有效邻近条带内。这里还采用了处理环绕问题的第二种方法，需要判断两个数据是否都是冗余数据——交叉证认不在冗余数据内进行，以避免重复结果。这样经过四步过滤得到了以匹配边界值两倍长度为边长的正方形区域内的数据，最后一步只需要确定这些数据是否处在以第一个天体为圆心，匹配边界值

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.ZoneMatch') AND type in (N'U'))
    DROP TABLE dbo.ZoneMatch
GO

DECLARE @dist2 FLOAT = 4 * POWER(SIN(RADIANS(7.0/60.0/60.0/2.0)), 2);

SELECT t1.objid as ObjID1,
    t2.objid as ObjID2
INTO dbo.ZoneMatch
FROM dbo.XIndex t1
    INNER LOOP JOIN dbo.ZoneZone zz on zz.zoneid1 = t1.zoneid
    INNER LOOP JOIN dbo.YIndex t2 on zz.zoneid2 = t2.zoneid
        and t2.ra between t1.ra - zz.Alpha2 and t1.ra + zz.Alpha2
        and t2.dec between t1.dec - @theta and t1.dec + @theta
        and ( t1.RA >= 0 or t2.RA >= 0 )
WHERE (t1.cx-t2.cx) * (t1.cx-t2.cx)
    + (t1.cy-t2.cy) * (t1.cy-t2.cy)
    + (t1.cz-t2.cz) * (t1.cz-t2.cz) < @dist2
```

图 3.11: 交叉证认过程利用预先准备好的星表索引、邻近条带对比表及赤经与赤纬过滤值迅速获取最小范围内的候选数据，再精确计算距离以确定其有效性。

为半径的圆内即可。这一步是通过空间距离计算进行的，@dist2即是将通常以球面圆弧表示的匹配边界$\theta$转换为空间距离后的数值的平方。空间距离的计算也非常简单，计算两个空间向量的差的模即可。

代码中的LOOP JOIN联接关键字非常重要，不同的关键字的时间效率大为不同。关系数据库支持三种联接方式，LOOP JOIN、MERGE JOIN及HASH JOIN。如表3.2所示，可以看到三种联接方式下的交叉证认的效率相差极远。而只使用JOIN，由数据库自己选择联接方式的话，它所选择的方式更接近于HASH JOIN。之所以不适合HASH JOIN是因为星表索引已经按照应用需求排好序，数据库再做一次哈希排序没有意义。MERGE JOIN适合于一对一联接的场景，而交叉证认需要多对多进行对比、过滤，也不适合。而看起来最为原始低效的多对多两重循环的LOOP JOIN反而与交叉证认相得益彰。更简单地说，LOOP JOIN更适合天文星表交叉证认的应用场景，而数据库引擎并不能发现这一点。因而，需要在代码予以明确指出这一点，强制要求数据库引擎使

表 3.2: SQL Server数据库中四种联接方式在Zones Algorithm交叉证认程序中的耗时对比

| KEYWORD | PERFORMANCE |
|---|---|
| LOOP JOIN | 449s |
| MERGE JOIN | 310650s |
| HASH JOIN | 25599s |
| JOIN | 26111s |

用LOOP JOIN算法进行数据联接操作。这也告诉了我们，算法并没有那么明显的优劣，关键是要在合适的场景使用适当的算法才能提高效率。数据库引擎虽然已经很强大很智能，但是它也未必能找到最快速的方式，必要的时候还是需要给它提示。

## 3.5 小结

Zones Algorithm是一个非常高效的天文星表交叉证认算法。它能充分利用关系数据库的聚集索引机制，高效地读取数据。尤其是它的三层过滤方式都只是简单地做数值比较，或者是简单的算术运算，极其快速地过滤掉了多数的无效数据。最后只在一个很小的正方形区域内对数据的实际有效性进行检查。最重要的是它可以完全由SQL实现的，可被几乎所有的关系数据库支持，真正实现了多平台可插拔运行。

# 第四章　结合覆盖图的星表交叉证认算法与缺失源检测及数据库实现

条带算法（Zones Algorithm）是一种非常快速的交叉证认算法，也是美国虚拟天文台交叉证认服务Open SkyQuery的核心算法。但是与其它交叉证认算法一样，它们都没有将星表的天区覆盖信息考虑在内。典型的如图4.1[1]所示斯隆数字化巡天第5次数据（Sloan Digital Sky Survey's 5th Data Release，SDSS DR5）和GALEX星系演化探测器第2次数据（Galaxy Evolution Explorer's 2nd Public Release，GALEX GR2）的天区覆盖图，其中主要由蜂窝状小格所组成的区域为GALEX GR2的天区覆盖图。天区覆盖信息也是星表信息的重要组成部分，善加利用也能进一步优化交叉证认算法。假设这样一个场景，星表A 与星表B 要做交叉证认，但是实际上它们的观测区域只有很小一部分重叠，如图4.2 示例。比较理想的情况下，只需要对重叠区域的数据进行交叉证认。但是，几乎所有的交叉证认算法都不理会这些信息，它们仍将按部就班地准备数据，遍历整个星表。本章讨论的就是如何将天区覆盖信息融入到Zones Algorithm 交叉证认算法中。需要提到的是，应用HTM可以快速地在指定区域找到该区域内的天体，Open SkyQuery 便是利用了此项技术来在指定区域内进行交叉证认。但是它利用的索引是`HtmID`，与Zones Algorithm 大为不同，由于一个数据表只能有一个聚集索引，此项方法的效率大打折扣。这里所讨论的方法将能够使用`ZoneID`为索引，与Zones Algorithm整合得更自然。相关论文已经发表在太平洋天文学会技术刊物（Publications of Astronomical Society of the Pacific, PASP）上[70]。

[1]SDSS DR5 和GALEX GR2天区覆盖图图片来自http://voservices.net/footprint/

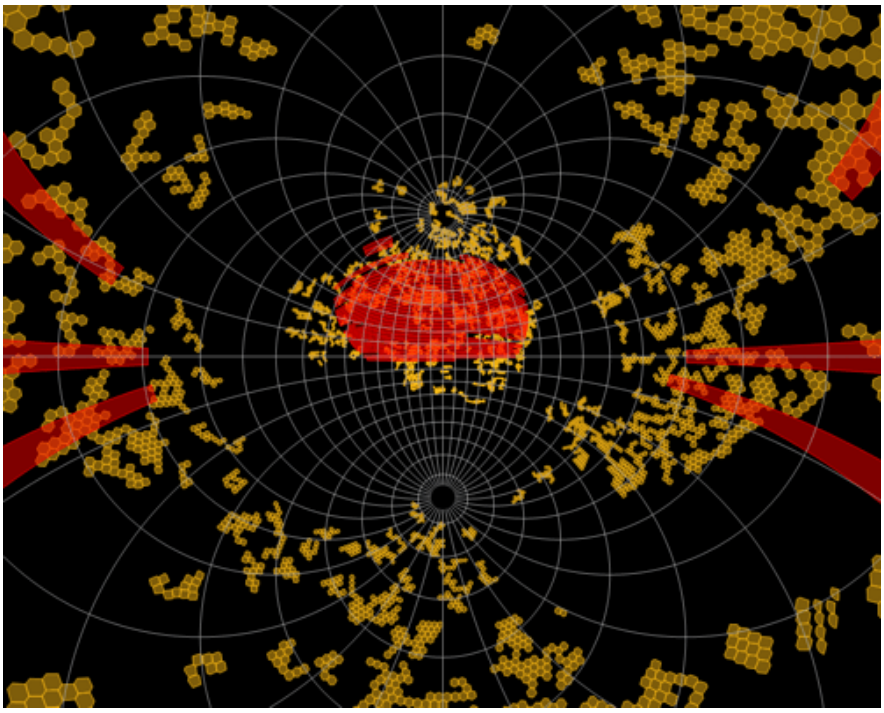

图 4.1: SDSS DR5的天区覆盖图及主要由蜂窝状小格所组成的GALEX GR2的天区覆盖图。

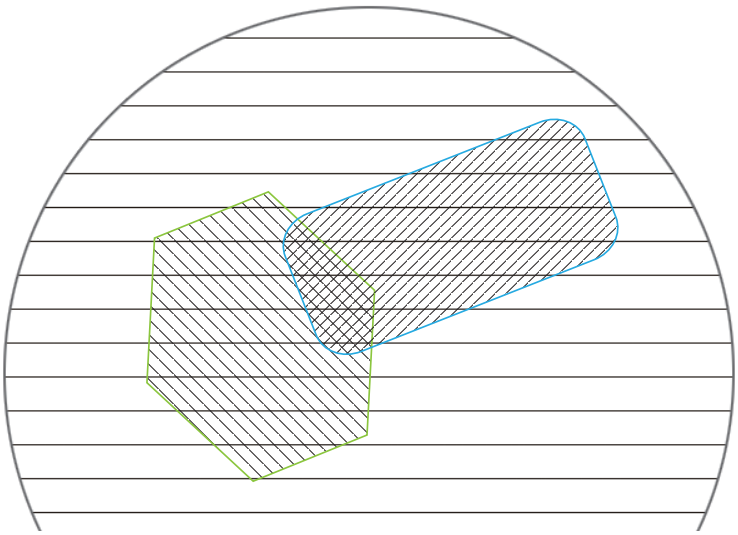

图 4.2: 两个球面区域只有很小部分重叠的情形示例。

## 4.1　使用条带片段模拟覆盖图

在Zones Algorithm中使用天区覆盖图信息，最重要的一条就是如何将条带与覆盖图联系起来。这里所采用的方法就是使用条带片段来模拟覆盖图。因为整个天球都已经被条带划分过了，而所有的覆盖图显然都在天球上，所有的覆盖图也就都被条带所切割。只需要找到覆盖图与条带有交集的部分，记录下这一部分条带的信息即可。如图4.3，使用环带片段来模拟一个圆，阴影区域表示的条带片段完整地将整个圆覆盖住了。可以看到全部的条带片段的面积要比原来的圆大了一些，但是因为条带的高度通常很窄，比如7.1″，条带片段的模拟效果与圆非常接近。这实际上与计算机屏幕上使用像素点阵来显示圆的效果类似，只要分辨率够高，显示的效果与实际情况相差很小。这样一来，一个圆就被分解成了数个条带片段，而这些片段的信息非常简单，只有三个数据：条带编号ZoneID、左边界的赤经值RAMin及右边界的赤经值RAMax。有一个明显地好处，就是可以使用ZoneID和RAMin 对数据进行排序存放，与Zones Algorithm一致，保证了相关数据也存储在相邻区域。

使用条带片段来描述天球上的一个形状，最主要的优点就是简化了计算。因为条带片段只有三个数据，不管是做交集还是并集运算，都只需先检查片段们是不是在同一个条带上，即ZoneID是否相等；然后再对比左右边界值来判断条带片段是否有交集，这甚至都不需要算术运算而只需对数据大小进行逻辑对比。对于天区覆盖图而言这尤其重要，因为天区覆盖图通常都是很复杂的几何图形，如果直接计算它们的交集，将消耗大量的时间，并且计算结果只能使用在这两个星表的相关操作中，非常低效。这可能也是之前的交叉证认算法都不考虑天区信息的原因。使用条带片段来描述天区覆盖图也可以方便地添加新数据。在需要添加新天区的信息时，可以使用条带片段模拟这一区域后，快速合并到原来的数据中去。而两个星表的覆盖图交集计算也可以等量替换成它们各自的条带片段模拟表的交集计算，两个表的交集计算时间将大大低于两个复杂几何图形的交集运算。这时候，两个星表所重叠的面积就可以在需要的时候快速生成，而不用担心太多时间消耗。最关键的是，每个星表的条带片段覆盖图都是各自独立的，只在需要的时候将相关的两个表进行对比即可，每个表可以被重复使用。这样即便生成条带片段覆盖图需要消耗较长时间，也只是一次性的工作，效率比直接对两个覆盖图的几何图形运算要高多了。

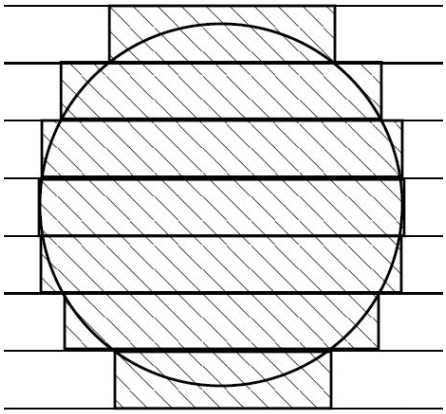

图 4.3: 使用阴影区域的条带片段来模拟天球上的一个圆。

## 4.2 条带片段并集、交集算法

合并条带片段即是将首尾能够相连的片段连接到一起，它们有两个条件：一是要在同一个条带上，即要有同一个ZoneID；二是一个片段的左边界（或右边界）在另外一个片段的赤经覆盖范围内。如图4.4中左侧部分的条带片段们可被合并成一段，而右侧部分继续保持其原样。如若直接对每一个片段进行对比，则整个合并过程需要多重循环。这里设计了一新的方法，每次寻找一个可以合并的片段的最小（左）边界，然后寻找最大（右）边界，最后将左右边界成对合并即完成了整个合并过程。

寻找最大或最小边界，第一步需要将所有片段都放到同一个集合（同一个表）INTERVALS中，并按ZoneID、RAMin、RAMax的顺序排序。排序后，所有可以连接在一起的片段都将按RAMin从小到大的顺序排列。这时候一个可合并片段的最小（左）边界的特征是，这个左边边界不处于任何它的左边片段之内；同样的，最大（右）边界的特征是，这个右边边界不处在任何它的右边片段之内。算法如图4.5，因为可以存在多个同值的左边、右边边界，需要把重复值去掉。对于每一个左边界值，在同一条条带上寻找在它右边离它最近的一个右边界即可，它们便是可合并区域的左右边界。

与合并算法相比，交集算法要简单得多。只需要将两个集合中在同一个条带上的片段一一进行对比，如果一个片段的左边界或右边界落入另一个片段的

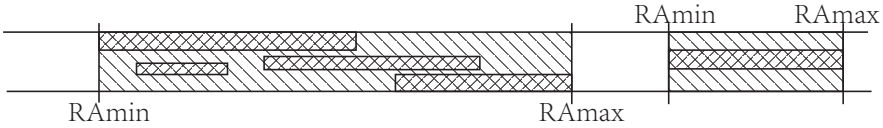


图 4.4: 左侧部分的条带片段们可被合并，而右边的条带仍保持原样。

```
merge all segments to one set #INTERVALS

sort the #INTERVALS by ZoneID, RAMin, RAMax

SELECT ZoneID, RAMin
INTO #LEFTBOUNDARY
FROM #INTERVALS o1
WHERE NOT EXISTS(
    SELECT RAMin    FROM #INTERVALS O2
    WHERE o1.ID<>o2.ID AND o1.ZoneID=o2.ZoneID AND
        o1.RAMin > o2.RAMin AND o1.RAMin<=o2.RAMax )

remove duplicate rows from #LEFTBOUNDARY

SELECT ZoneID, RAMax
INTO #RIGHTBOUNDARY
FROM #INTERVALS o1
WHERE NOT EXISTS(
    SELECT RAMin FROM #INTERVALS o2
    WHERE o1.ID<>o2.ID AND
        o1.ZoneID=o2.ZoneID AND
        o1.RAMax >= o2.RAMin AND o1.RAMax<o2.RAMax )

remove duplicate rows from #RIGHTBOUNDARY

INSERT MERGERESULT(ZoneID, RAMin, RAMax)
SELECT o1.ZoneID, o1.RAMin, Min(o2.RAMax) RAMax
FROM #LEFTBOUNDARY o1
    JOIN #RIGHTBOUDNARY o2 ON o1.ZoneID=o2.ZoneID AND o2.RAMax>RAMin
GROUP BY o1.ZoneID, o1.RAMin
```

图 4.5: 条带片段的合并算法。其核心是先找到可合并区域的最左边界，再找到最右边界，将两个边界记录到一个新的数据集中即可。

```
INSERT INTERSECTION
SELECT o1.ZoneID ZoneID,
    CASE WHEN o1.RAMin>o2.RaMin THEN o1.RAMin
        ELSE o2.RAMin END RaMin,
    CASE WHEN o1.RAMax>o2.RaMax THEN o2.RAMax
        ELSE o1.RAMax END RaMax
    FROM INTERVALS1 o1
        JOIN INTERVALS2 o2 ON o1.ZoneID=o2.ZoneID
        AND (o1.RAMin BETWEEN o2.RAMin AND o2.RAMax
            OR o2.RAMin BETWEEN o1.RAMin AND o1.RAMax)
```

图 4.6: 条带片段的交集算法。只需一一对比两个片段，若有交集则记录最大左边界及最小右边界。

赤经范围内，则记录下有交集的两个片段最大的左边界RAMin值，及最小的右边界RAMax；若无重叠则直接略过。得到的新边界即是两个片段的交集。算法如图4.6所示。

## 4.3 覆盖图信息与交叉证认的结合及缺失源检测

由于实际进行的星表覆盖图的交集计算是在它们的条带片段中进行的，得到的交集也是一些片段的集合（FOOTPRINTINTERSECT），其效果如图4.7如所示。FOOTPRINTINTERSECT可用于限制Zones Algorithm对第一个星表的搜索范围，只查找那些位于两个星表重叠区域内的天体。然后在这些天体的附近寻找它们在第二个星表中的候选匹配天体。FOOTPRINTINTERSECT也可以用于邻近条带对比表（ZoneZone）的生成，条带片段也包含了条带的编号ZoneID 信息。并且，这样生成的ZoneZone对比表要比原来的方法小一些，因为它只包括了重叠区域的信息。而原来的方法需要将第一个星表的所有条带列出来。另外需要注意的是，将FOOTPRINTINTERSECT用于Zones Algorithm交叉证认时，也需要处理赤经$\alpha = 0°$处的环绕问题，否则在Zones Algorithm中增加的$\alpha < 0°$数据将被忽略而影响结果。

天区覆盖信息的加入，还带来了一个新的应用：可以应用于星表缺失源的检测。由于望远镜设备的性能不同，或曝光不足等原因，两个星表虽然观测了同一个天区，但是观测到的天体却不尽相同。如星表A中的一天体c没有被星表B找到，同样的星表B中的一天体d没有被星表A 找到。这时候，就称B相对于A缺失了c，称A相对于B缺失了d。在以往想要知道A表中天体为什么没

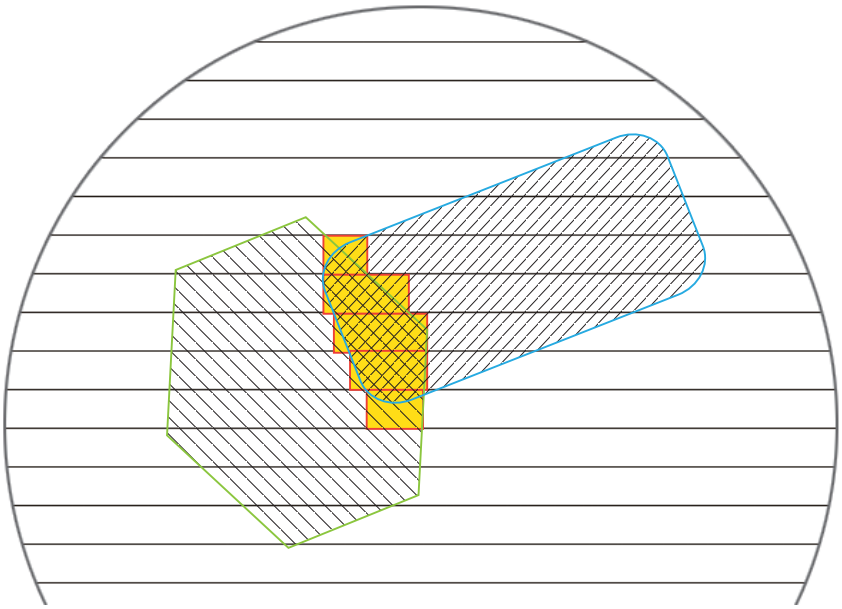

图 4.7: 使用条带片段来模拟两个覆盖图的重叠区域。

```
SELECT c.ObjID
FROM FOOTPRINTINTERSECT AS o
    JOIN CatalogB AS c ON o.ZoneID = c.ZoneID
        AND c.RA BETWEEN o.RAMin AND o.RAMax
EXCEPT
SELECT ObjID2 FROM MatchedObjects
```

图 4.8: 查找处于A表观测范围内却未被A表匹配的天体。

在B表中出现——是因为没在B的观测范围内呢，还是因为B观测了但是没找到——是一件很困难的事。而现在则因为观测天区信息的加入而变得轻而易举。其检测过程简单而直接，如图4.8所示，要找到A表所缺失的B表中的天体，只需要先找到B表在两个星表重叠区域中的所有天体，然后扣除已被A表匹配的天体得到一个天体集合。此集合中的天体即处于A表的观测范围内，而未被A找到，它们可能预示了A中的一些潜在的问题。整个检测过程快速而有效。

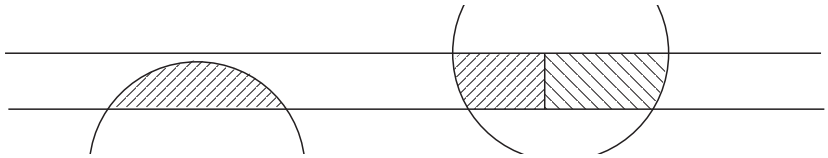

图 4.9: 一个条带与覆盖图相交的典型情形，覆盖图的某个组成部分的末端在条带内，或是中间部分在条带内，且它们有可能被赤经$\alpha = 0^\circ$ 线所分割。

## 4.4 数据库实现

由于现成的星表、Zones Algoirthm代码、天区覆盖信息均在数据库中，这里也直接采用了数据库对新算法进行实现。天区覆盖图是通过Spherical Library进行描述的，如图4.1中的GALEX GR2的图由多个六边形构成，每一个六边形对应的是望远镜一次观测的区域，在数据库里就是一个数据表中的一行Spherical Library区域描述语言文本。两个星表的天区覆盖图本是可以直接进行交集计算的，但是覆盖图的各个组成部分相当庞杂，且几何运算复杂耗时，计算结果只能在两个星表中应用也不经济。因而新算法采用的办法是先将覆盖图映射到Zones Algorithm中的条带上。实际上所要进行的计算就先变成了天区覆盖图与条带的交集计算。

### 4.4.1 覆盖图的条带片段集

这里的覆盖图在天球上是用Spherical Library描述的，因而所有图形的边都是天球上一个圆的一部分。对于一个条带而言，某个覆盖图与其相交的情形典型地如图4.9所示。覆盖图的某个组成部分可能是其末端在条带内，也可能是中间部分在条带内，且它们有可能被$\alpha = 0^\circ$分割而出现Zones Algorithm中的环绕问题。可以看到，覆盖图与条带的交叉部分的左、右边界可以从两个图案的交点处获得，即交点的最小值可取为左边界，交点的最大值可取为右边界。Spherical Library中的`fGetOutlineArcs`函数可用于取得这些交点。但是这同时也带来了两个问题。

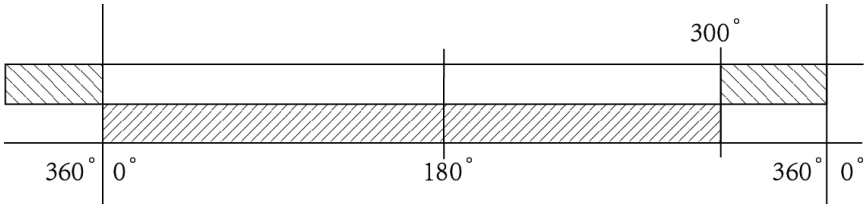


图 4.10: 取最小赤经值为左边界，最大赤经值为右边界的作法无法正确取得横跨$\alpha = 0°$的片段的范围。

问题一，一个片段可能横跨了赤经线$\alpha = 0°$。这时候，之前的取交叉点赤经最小值为左边界，取交叉点赤经最大值为右边界就不再成立。如图4.10所示，$(330°, 360°) \bigcup (0°, 30°)$及$(30°, 330°)$片段的赤经最小值均为30°，最大值均为330°。此时算法将无法取得片段正确的赤经的范围。此时需要考虑将条带切分为两部分，一部分为$[0°, 180°]$，另一部分为$[180°, 360°]$。这样，一个片段的右侧将不会越过360°，从而解决了问题。这样操作带来的直接问题就是交叉点的计算量增倍，但因为这一操作只需要进行一次，所得到的条带片段将可任意使用，因而仍是可接受的。

问题二，如图4.11所示，有时一个圆是在条带中凸出的，这时候只考虑交点的范围将使凸出部分被遗漏掉。这种情况发生的条件是，圆弧的圆心在此条带上，这时候需要做一个专门处理，当凸出情况出现在左侧时，需要将圆心坐标减去圆半径所对应的赤经范围，即`Alpha`值，将此结果与左边的两交点做比较，取最小值。当凸出情况出现在右侧时，需要将圆心坐标加上圆半径所对应的`Alpha`值，将此结果与右边的两交点做比较，取最大值。具体判断凸出情况的算法伪代码如图4.12所示，其中`RA`是圆弧圆心，`RA1`与`RA2` 是一个圆弧上的两个端点的赤经值，两个端点按顺时针方向排列。

#### 4.4.1.1　定义条带的两个部分

在处理覆盖图与条带的交集运算之前，首先需要解决的一个问题就是如何使用Spherical Library描述条带，以供条带与覆盖图进行计算。如图4.13定义了一个SQL函数`fRegionZone`专门用于生成条带区域。其构造方法是先构造一个范围从南天极点到条带上边界`dmax`的半空间，然后构造一个从北天极点到条带下边界`dmin`的半空间，两者的交集即是一个条带。但是为了解决一

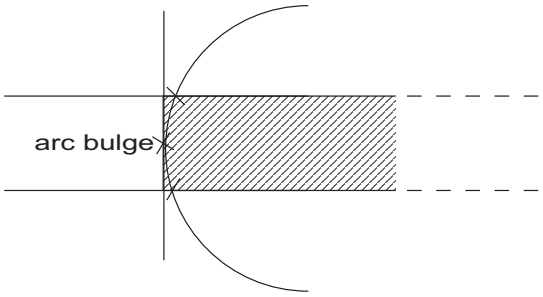


图 4.11: 当圆弧的圆心处于本条带中时，会出现圆弧的外沿越过两边交点的情形，需要特别处理。

```
If RA1>RA and RA1-RA<180,
    RA1 is at the right side of RA.
    Choose the max(RA1, RA2, RA+Alpha);
else if RA1<RA and RA-RA1>180),
    RA1 is at the right side of RA,
    and 0<=RA1<180, 180<RA<=360.
    Choose the max(RA1, RA2, RA1+Alpha-360);
else if RA1>RA and RA1-RA>180,
    RA1 is at the left side of RA,
    and 180<RA1<=360, 0<=RA<180.
    Choose the min(RA1, RA2, RA-Alpha+360);
else RA1 is the left side of RA.
    Choose the min(RA1, RA2, RA-Alpha).
```

图 4.12: 判断凸出情况的算法伪代码。一个条带与覆盖图的边界可通过两者的交点来取得，但是需要处理覆盖图凸出的问题。

个片段横跨$\alpha = 0°$的问题，条带还需要与东半球或西半球做交集，从而分别取得$[0°, 180°]$及$[180°, 360°]$的区域。条带两区域的描述信息都将保存入Zones Algorithm 中的条带定义表`ZoneDef`中，不再赘述这个过程。

#### 4.4.1.2 取得覆盖图在条带上的边界

一个典型的SDSS DR6的覆盖区域如图4.14所示，可以看到它是一个弯曲的带状区域。因为不知道它的最顶点在哪，或者在别的情形不知道最低点在哪，这时候，哪些条带需要与这个带状区域做交集以取得条带模拟片段就变成了一个很棘手的问题。最稳妥的办法就是所有的条带都与它作几何交集运

```
CREATE FUNCTION fRegionZone(@dmin FLOAT,
    @dmax FLOAT, @b0to180 INT)
RETURNS VARBINARY(MAX)
AS
BEGIN
    DECLARE @bins VARBINARY(MAX), @binn VARBINARY(MAX)
    , @bin2 VARBINARY(MAX),@bin3 VARBINARY(MAX)
    --北边的半空间
    SET @bin1 = dbo.fConvexAddHalfspace(null,
        0, 0,0,+1,  SIN(RADIANS(@dmin)));
    --南边的半空间
    SET @bin1 = dbo.fConvexAddHalfspace(@bin1,
            0, 0,0,-1, -SIN(RADIANS(@dmax)));
    SET @bin1 = dbo.fSimplifyBinary(@bin1);

    IF(@b0to180=1)
        --东半球
        SET @bin2 = dbo.fConvexAddHalfspace(null,
            0, 0, +1,0, 0)
    ELSE
        --西半球
        SET @bin2 = dbo.fConvexAddHalfspace(null,
            0, 0, -1,0, 0)
    SET @bin2 = dbo.fSimplifyBinary(@bin2);
    SET @bin3 = dbo.fIntersect(@bin1, @bin2);
    RETURN @bin3;
END
```

图 4.13: 产生条带的东西两部分的函数，每一部分都是三个“半空间”的交集。

算，但是时间上显然划不来。因为条带高度只有$h = 7.1''$，整个球面被分割成了$\lceil \frac{180\times60\times60}{7.1} \rceil = 91268$个条带。而这个带状区域显然并没有覆盖那么多的条带。Spherical Library 里面有一个`fGetPatches`函数可以在一定程度上帮助解决这个问题，它可求得这个带状区域的外切圆。这样，可以将计算限制在这个外切圆的赤纬范围内。这显然也并不是一个很令人满意的方法，仍然在很大部分计算是无用的。但是，目前这个已经是相对较好的解决方法了。图4.15示范了使用`fGetPatches`取得相关条带上下界信息，并实施图形交集计算的过程。由于条带被分成了东西半球两部分，一个条带需要与同一个图形做两次计算。这也增加了计算成本，需要在进一步的研究中给予优化。

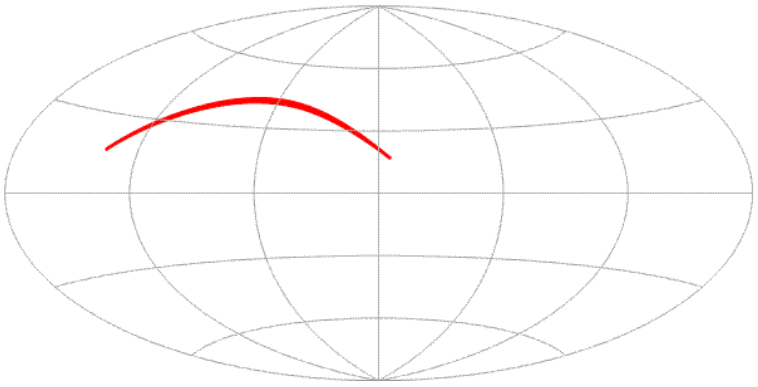

图 4.14: 一个典型的SDSS DR6的区域覆盖片段。

做完交集计算之后，需要确定各个条带上的片段范围，也即要找每个片段的左、右边界。这在上文中已经讨论过了，主要是需要注意圆弧凸出的问题。前述的解决凸出问题的算法已经被写入了一个叫作`fGetOutlineExt`的SQL函数中，边界计算过程如图4.16。同样由于条带被分割了两部分，需要分两部分分别计算。最后需要注意的是，各个覆盖图的组成部分可能本身也是有重叠的。这意味着一些片段是可以合并的，这就需要应用前文所讨论的条带片段合并算法来简化整个条带片段模拟结果。最后的结果保存在一个`FA_SDSSDR6Primary`表中，此表须按条带编号、左边界、右边界$(\texttt{ZoneID}, \texttt{RAMin}, \texttt{RAMax})$ 设置聚集索引顺序，以将相关的数据存储在相邻物理区块内。

SDSS DR6条带片段模拟图的生成方法与过程也适用于其他的星表的覆盖图的处理。最核心的过程是确定需要做交集计算的条带范围，计算交集。然后进一步地计算出各个有效条带片段的左右边界，并注意合并连接在一起的条带。减少重复数据的同时也能减少后续计算的次数。

### 4.4.2 覆盖图的交集及生成ZoneZone的新方法

得到了所需星表覆盖图的条带片段模拟表之后，就无须再考虑原始的天区覆盖图了，之后的相关操作都可以建立在这些模拟表上。两个星表的覆盖图的交集也就可以在它们各自的模拟表中进行了。如前所述两个模拟表的交集运算，即是将两个表中同一条带上的片段边界一一进行对比，取最大的左边界和最小的右边界。因为各个模拟表都已经对可连接的条带片段进行了合

```
--保存星表覆盖图每一部分的描述信息及需要做交集计算
的条带上下边界范围
INSERT F_SDSSDR6 WITH(TABLOCKX) (RegionBinary, DecMin, DecMax)
SELECT t.RegionBinary RegionBinary,
    f.dec-f.radius/60.0 DecMin, f.dec+f.radius/60.0 DecMax
FROM #F_SDSSDR6_Temp t
    CROSS APPLY sph.fGetPatches(t.RegionBinary) f


IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'[dbo].[FZ_SDSSDR6Primary]') AND type in (N'U'))
    DROP TABLE [dbo].[FZ_SDSSDR6Primary]
GO

CREATE TABLE FZ_SDSSDR6Primary(
    id BIGINT IDENTITY (1,1)NOT   NULL,
    ZoneId INT,
    Intersect1 VARBINARY(MAX),
    Intersect2 VARBINARY(MAX),
    PRIMARY KEY(id, zoneid)
)
GO
--对覆盖图的每一部分与其相关的条带做交集计算
INSERT FZ_SDSSDR6Primary WITH(TABLOCKX) (ZoneId, Intersect1, Intersect2)
SELECT z.ZoneId ZoneId,
    sph.fIntersect(g.RegionBinary, z.RegionBinary1) Intersect1,
    sph.fIntersect(g.RegionBinary, z.RegionBinary2) Intersect2
FROM dbo.F_SDSSDR6 g
    JOIN dbo.Z_ZoneDef z ON
        z.DecMin BETWEEN g.DecMin AND g.DecMax
        OR z.DecMax BETWEEN g.DecMin AND g.DecMax
GO
--删除无用数据
DELETE FROM FZ_SDSSDR6Primary where Intersect1 IS NULL AND Intersect2 IS NULL
```

图 4.15: 使用fGetPatches取得星表覆盖图上下界信息，并实施星表覆盖图与相关条带交集计算的过程

并，因而两个表的交集部分不可能再产生可连接的片段，也就无须再对交集表进行整理。整个计算过程非常快速，可以在几秒内完成。与直接进行几何计算的动辄几个小时相比，可用性大大增强，可以在需要的时候再即时计算。需要注意的是，为了与Zones Algorithm中处理环绕问题的解决方案保持一致，这里也需要添加一些冗余数据。当一个片段的右边界加上该条带的赤经扩展范围Alpha之后，若值大于360，则需要添加一个新的片段。如图4.17，新的片段保持ZoneID值一致，但RAMin、RAMax要分别减去360。

模拟表的交集也仍是一个模拟表，其中同样记录了(ZoneID, RAMin, RAMax)等信息。尤其是ZoneID信息，可以用于Zones Algorithm算法中邻近条带对比表（ZoneZone）的生成。如图4.18，相较于原来的方法，仅是将星表换成了覆盖图交集的模拟表。最大的优势就是模拟表远远小于数亿条记录的星表，速度大大加快。且在原来的方法中，需要记录所有同时出现在两个星表中的ZoneID，而实际在很多条带上两个星表并没有交集。因而原来的ZoneZone表中有很多无效的邻近条带信息，增加了在星表中检索数据的范围。新的方法则尽量减少了这一情况，检索范围缩减到了只在两星表重叠区域所涉及的条带内。

### 4.4.3 新的交叉证认方法及缺失源检测

新的交叉证认方法主要是加入了星表的重叠区域模拟表，如图4.19所示，主要的改动是将两星表的重叠区域模拟表FI_SDSSDR6Primary_GALEXGR3AIS-Primary 直接列在FROM关键字之后，并限制第一个星表的搜索范围，其他部分维持不变。之所以将该表直接放在FROM关键后面，是因为它通常是最小的一个表，甚至可能为零。数据库引擎通过这个小表的信息可以快速制定执行策略，略过不必要的搜索范围。代码中第一个JOIN没有像其他子句那样有一个LOOP的限制，也是为了让数据库能自己选择较合适的执行策略，实际运行表明，标明LOOP要比不标明要慢几秒。

在得出两个星表的交叉证认结果之后，即可轻而易取地进行缺失源检测。如图4.20，先找到星表CX_SDSSDR6Primary在两个星表重叠区域中的天体，然后排除掉已经被匹配的天体，得到的即是GALEXGR3AISPrimary覆盖了却未能观测到的天体。所需的仅仅是两个简单的SELECT选择语句和EXCEPT求差操作，简洁快速。

```
IF  EXISTS (SELECT * FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.FN_SDSSDR6Primary') AND type in (N'U'))
    DROP TABLE dbo.FN_SDSSDR6Primary;
GO

CREATE TABLE FN_SDSSDR6Primary(
    id INT IDENTITY(1,1),
    ZoneId INT NOT NULL,
    RaMin FLOAT NOT NULL,
    RaMax FLOAT NOT NULL,
    PRIMARY KEY (id, ZoneId, RaMin, RaMax)
)
GO

INSERT FN_SDSSDR6Primary WITH(TABLOCKX) (ZoneId, RaMin, RaMax)
SELECT r.ZoneId ZoneId, MIN(f.ra1) RaMin, MAX(f.ra1) RaMax
FROM FZ_SDSSDR6Primary r
    JOIN dbo.Z_ZoneDef z ON r.ZoneId=z.ZoneId
    CROSS APPLY dbo.fGetOutlineExt(r.Intersect1, z.DecMin, z.DecMax, 0) f
WHERE r.intersect1 IS NOT NULL
GROUP BY r.id, r.zoneid

INSERT FN_SDSSDR6Primary WITH(TABLOCKX) (ZoneId, RaMin, RaMax)
SELECT r.ZoneId ZoneId,MIN(f.ra1) RaMin, MAX(f.ra1) RaMax
FROM FZ_SDSSDR6Primary r
    JOIN dbo.Z_ZoneDef z ON r.ZoneId=z.ZoneId
    CROSS APPLY dbo.fGetOutlineExt(r.Intersect2, z.DecMin, z.DecMax, 1) f
WHERE r.intersect2 IS NOT NULL
GROUP BY r.id, r.zoneid;
```

图 4.16: 计算条带上与覆盖图重叠片段的有效边界。

```
INSERT FI_GALEXGR3AISPrimary_SDSSDR6Primary WITH(TABLOCKX)
SELECT a.ZoneId, a.RAMin-360, a.RAMax-360
FROM FI_GALEXGR3AISPrimary_SDSSDR6Primary a
    JOIN Z_ZoneDef z ON a.ZoneId=z.ZoneId AND a.RAMax+z.Alpha>360
```

图 4.17: 星表覆盖图交集模拟表也需要处理环绕问题，需要添加部分冗余数据。

```
IF  EXISTS (
    SELECT *
    FROM sys.objects
    WHERE object_id = OBJECT_ID(N'dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary')
        AND type in (N'U')
    )
    DROP TABLE dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary
GO

CREATE TABLE dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary(
    ZoneID1 INT NOT NULL,
    ZoneID2 INT NOT NULL,
    Alpha2 FLOAT NOT NULL
)
GO

ALTER TABLE dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary
ADD CONSTRAINT PK_ZZ_GALEXGR3AISPrimary_SDSSDR6Primary
PRIMARY KEY ( ZoneID1, ZoneID2 )
GO

INSERT dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary WITH (TABLOCKX)
SELECT Z1.zoneid, Z2.zoneid, d2.alpha
FROM (
    SELECT DISTINCT ZoneID
    FROM dbo.FI_GALEXGR3AISPrimary_SDSSDR6Primary
    ) z1
    JOIN (
    SELECT DISTINCT ZoneID
    FROM dbo.FI_GALEXGR3AISPrimary_SDSSDR6Primary
    ) z2
    ON Z2.zoneid BETWEEN Z1.zoneid - 1 AND Z1.zoneid + 1
JOIN dbo.Z_ZoneDef d2 ON d2.ZoneID = Z2.ZoneID
ORDER BY 1, 2
```

图 4.18: 使用星表覆盖图交集的条带片段模拟表来生成邻近条带对比表ZoneZone，速度更快，内容更有效。

```
IF  EXISTS (SELECT * FROM sys.objects
WHERE object_id = OBJECT_ID(N'dbo.M_GALEXGR3AISPrimary_SDSSDR6Primary')
AND type in (N'U'))
DROP TABLE dbo.M_GALEXGR3AISPrimary_SDSSDR6Primary
GO

DECLARE @theta float, @dist2 float;
SELECT @theta = 7.0/3600.0;
set @dist2 = 4 * power(sin(radians(@theta/2)), 2);

SELECT t1.objid as objid1,
    t2.objid as objid2,
    60*120*degrees(asin(sqrt(
        (t1.cx-t2.cx) * (t1.cx-t2.cx)
            + (t1.cy-t2.cy) * (t1.cy-t2.cy)
            + (t1.cz-t2.cz) * (t1.cz-t2.cz)
        )/2)) Sep
INTO dbo.M_GALEXGR3AISPrimary_SDSSDR6Primary
FROM dbo.FI_GALEXGR3AISPrimary_SDSSDR6Primary as ol1
    JOIN dbo.CX_GR3AISPrimary as t1
        on (t1.zoneid=ol1.ZoneID and t1.ra between ol1.ramin and ol1.ramax)
    INNER LOOP JOIN dbo.ZZ_GALEXGR3AISPrimary_SDSSDR6Primary zz
        on zz.zoneid1 = t1.zoneid
    INNER LOOP JOIN dbo.CX_SDSSDR6Primary t2 on zz.zoneid2 = t2.zoneid
        and t2.ra between t1.ra - zz.Alpha2 and t1.ra + zz.Alpha2
        and t2.dec between t1.dec - @theta and t1.dec + @theta
        and ( t1.RA >= 0 or t2.RA >= 0 )
WHERE (t1.cx-t2.cx) * (t1.cx-t2.cx)
    + (t1.cy-t2.cy) * (t1.cy-t2.cy)
    + (t1.cz-t2.cz) * (t1.cz-t2.cz) < @dist2
```

图 4.19: 新的交叉证认方法，只是多了一个包含两星表重叠区域信息的小表，用以限制第一个星表的搜索范围。

```
SELECT objid
INTO GR3AISDropoutFromDR6
FROM CX_SDSSDR6Primary s
JOIN FI_GALEXGR3AISPrimary_SDSSDR6Primary o
ON s.zoneid=o.zoneid AND s.ra BETWEEN o.ramin AND o.ramax
EXCEPT
SELECT DISTINCT objid2
FROM M_GALEXGR3AISPrimary_SDSSDR6Primary m
```

图 4.20: 缺失源检测仅需要两个简单的SELECT语句和一个EXCEPT操作。

## 4.5 效率对比

为了与改造前的Zones Algorithm作对比，我们使用GALEX[71][72] GR3全天主星表与SDSS DR6[73]主星表进行交叉证认。它们的天区覆盖图均可从美国虚拟天文台的Footprint Service[2][3]获得。GALEX GR3有约5千5百万条数据，SDSS DR6 主星表约有2.3 亿条数据，它们的重叠区域面积经约为3689平方度。将两个天体的匹配边界设置为7″。所有的计算在一台配置为Intel Xeon E5430 CPU （2.66GHz、双CPU共8核）、24GB内存的服务器上进行。服务器所使用的操作系统为64位Windows Server 2008，数据库为Microsoft SQL Sever 2005-9.00.5057.00 开发版。

加入天区覆盖图前后的Zones Algorithm算法各部分的时间消耗可以从表4.1看到。其中使用条带片段模拟天区覆盖图的部分消耗了大量时间，其主要原因是形状的几何计算速度较慢（毫秒级），而形状的数量太多。GALEX GR3的天区覆盖图由15721 个形状组成，主体是小六边形；SDSS DR6 的天区覆盖图由43 个形状组成，数据虽然少，但是跨越的天区很大，与大量条带相交。而整个天球被切成了91270 个条带，每个条带再被分成两部分则一共是182540 个条带部分。但需要强调的是这一部分的计算只需要进行一次，当完成条带片段对天区覆盖图的模拟之后，这些条带片段即可取代天区覆盖图用于其他与天区有关的计算。实际上是使用了新的方式来描述天区覆盖图，也可在其他场合使用此种方式描述的覆盖图，这个时间消耗还是可以接受的。

[2]GALEX GR3 AIS覆盖图http://voservices.net/footprint/details_group.aspx?id=169
[3]SDSS DR6 覆盖图http://voservices.net/footprint/details_group.aspx?id=197

表 4.1: Zones Algorithm加入天区覆盖图前后效率对比

| 步骤 | 原Zones Algorithm | 加入天区覆盖图后 |
|---|---|---|
| ZoneDef | 00:02:08[1] | common proc. |
| 模拟天区覆盖图 | N/A | 05:48:27 |
| 星表索引 | 00:30:44 | common proc. |
| 天区覆盖图交集 | N/A | 00:00:06 |
| ZoneZone | 00:01:01 | 00:00:03 |
| 交叉证认 | 00:06:48 | 00:05:25 |

[1] 时间格式为“时:分:秒”

条带定义表ZoneDef与星表索引生成部分是新旧两方法都要用到的，标记为“common proc.”。最终的交叉证认部分，Zones Algorithm 花费了6 分48 秒，而加入天区信息之后减少为5 分25 秒，节省了约20% 的时间。考虑到数据在硬盘上存储也需要相当的时间，这个结果是相当令人满意的。交叉证认部分之外的两个星表的覆盖图交集计算时间为6 秒。新的ZoneZone表生成方法仅耗时3秒，而Zones Algorithm的方法需要1 分01 秒。在这一部分，星表交集的计算时间也完全被抵偿了，并且还节约了50多秒。

为了查看时间消耗与星表重叠区域面积的相关性，我们人为地将重叠区域限制到一定的赤经范围（cutting on R.A.）或赤纬范围（cutting on Dec.），以减小面积。再在新的限制区域内进行交叉证认，得到了图4.21。可见，当面积减小的时候，交叉证认的时间消耗也随之减少，整个图表上基本呈线性减少趋势。图中面积约为500平方度的部分，所截取的是SDSS DR6数据比较密集的区域。这里虽然面积减小了，但是因为包含的条带并未减少。数据库需要扫描的天体并未明显下降，时间减少得不甚明显。

交叉证认结束之后，仅需要很短的时间即可完成缺失源检测。经过29秒的检测，我们得到了一个数据：GALEX GR3中约有420万个天体处于SDSS DR6的观测区域中却未被SDSS DR6观测到。与交叉证认的时间效率分析方法相同，两个星表的重叠区域也按赤经或赤纬切割后用于缺失源检测，得到了图4.22。缺失源检测的时间消耗也随着两个星表的重叠面积的减少而减少，以至到了重叠面积为0时，消耗时间为0。

以上的检测在进行前都使用了DBCC DROP CLEANBUFFERS语句，以清除掉数据库缓存。但是操作系统的文件系统仍有可能保存了一部分数据在内存中，以致图表中出现了部分数据点明显处于趋势线以下的情形。

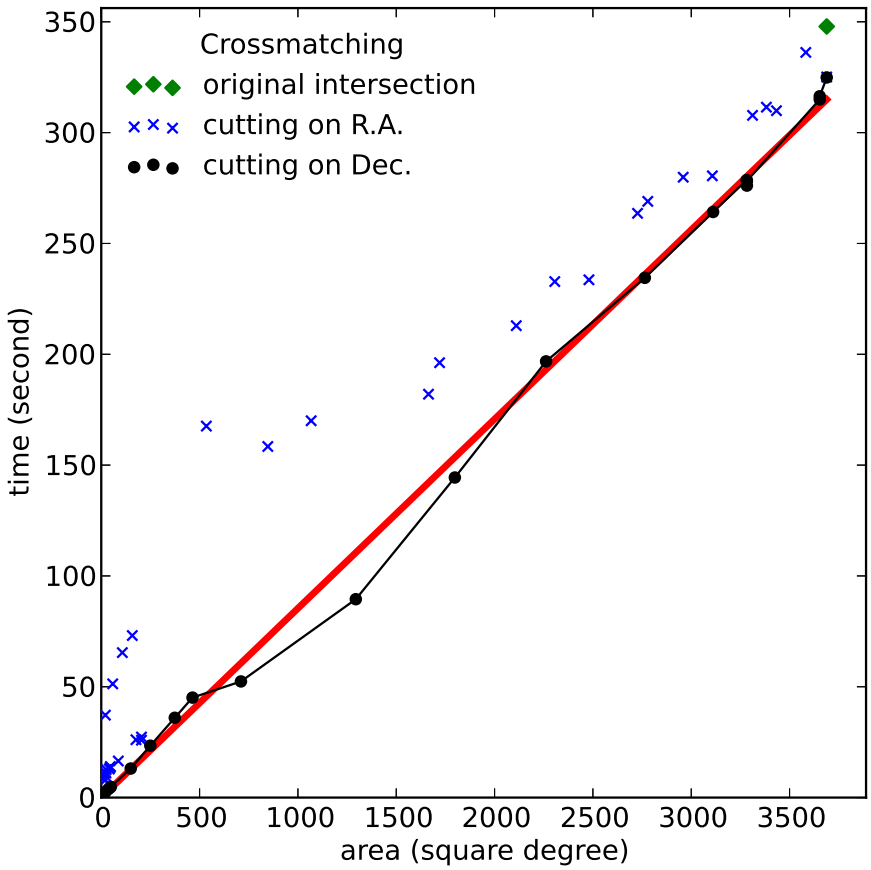


图 4.21: 随着重叠区域的减少，交叉证认的时间消耗也随之下降，基本呈线性减少趋势。

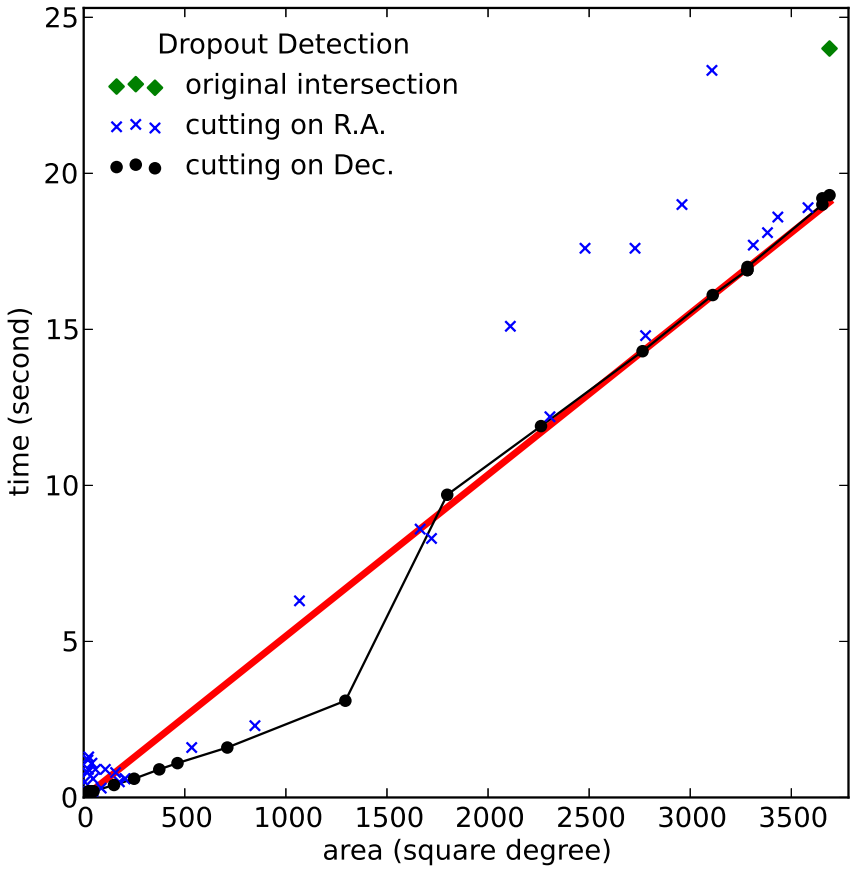


图 4.22: 缺失源检测的时间消耗基本也随重叠区域面积的减小而减少。

## 4.6 小结

可以看到随着星表天区覆盖信息的加入，条带算法得到了进一步的加速。且拥有了新的特性，可以对缺失源进行检测，为星表数据分析对比提供了新的信息。虽然条带片段对覆盖图的模拟过程非常耗时，但是幸好这一步只需要进行一次，相较它所带来的好处而言，还是可接受的。后续研究可以考虑提高这一部分的计算速度。比如使用更快速的图形操作库，减少无效计算等等。

本章中的条带模拟片段集合的生成方法也适用于天球上的任意形状，可以扩展成在任意指定区域内进行交叉证认的技术。即视任意区域为两星表的重叠区域，将其模拟片段集合应用到本章的交叉证认技术中，整个过程将非常迅速。可以做成一个Web Service服务供客户端远程调用，或单独做成一个网站系统来使用，天文学家可以只在自己感兴趣的区域中做多星表的交叉证认。此项技术将应用到美国虚拟天文台维护的星表交叉证认服务Open SkyQuery[4] 的新版交叉证认引擎中，对于中国虚拟天文台未来推出类似的服务也是一个重要的技术积累。

此外，由于在此算法中使用了Spherical Library，因而目前只能在Windows平台的Microsoft SQL Server中使用此程序。但是因为有了C++版本的Spherical Toolkit，未来将有望能在MySQL或Postgre SQL中也使用类似程序。

[4]Open SkyQuery http://www.openskyquery.net/Sky/skysite/

# 第五章　基于直线非对称几何模型的射电星表交叉证认方法

前面章节所在讨论的交叉证认技术，主要都是用于在一个星表或一次观测中一个天体只记录有一个坐标的情形，比如光学星表间的交叉证认。但是，有些天体在某些波段的观测结果中可能就包含有多个坐标。一些天体的射电观测结果除了自身之外，还可能记录到了它的喷流，如星系双极喷流。这样，一个天体在射电星表中可能就有不止两个观测目标。也即一个光学目标对应的射电目标可能是单个射电目标，也可能是两点甚至三点。有时候还可能与光学源对应的中间目标不明显甚至没有记录，而两端射电目标显著。从观测经验上看，两侧的喷流一般地都与中心连成一线，或形成一个较小的张角。本章将尝试使用直线非对称几何模型对射电星表（其中有些天体带有喷流）与光学星表进行交叉证认，并与澳大利亚射电天文学家手工交叉证认的结果进行对比。

未来将有平方公里级望远镜阵列（Square Kilometre Array，SKA）[1]等大规模射电望远镜（阵列）投入使用，产生出海量的射电观测数据。纯手工的交叉证认方式，显然太过于低效。本章讨论的方法将在一定程度上满足对于大规模射电波段星表与光学星表自动交叉证认的需求。

此处的交叉证认方法依然基于观测目标间的距离计算，主要引入贝叶斯方法结合射电源的几何分布进行分析，对各种假设进行评估并取得概率最大的匹配结果。除了通过特定几何模型信息来寻找候选匹配信息，此方法的优越之处在于它还可以很容易地加入观测目标的其他证据，如物理指标、光谱能量分布等。随着这些信息地加入，交叉证认的结果将会越来越准确。

## 5.1　贝叶斯因子

面对一系列的观测结果，我们会想知道它们是不是出自同一天体。当这些观测结果的坐标分散到整个天球上时，它显然不太可能源自同一天体。但若它们的坐标只有毫厘差距，我们可以说它们非常可能是同源的。但是，这“可能”的程度是多大呢？

首先，我们需要考查“观测精度”这一概念。当对观测结果进行位置校准

[1]SKA http://www.skatelescope.org/

时，精度可以通过对比天体测量标准或系统误差获得。但是，观测到的坐标仍有一定的概率偏离真实位置，这一概率通常符合正态分布。星表一般使用一个$\sigma$ 值来表示其观测精度，如$\sigma = 0.1''$。我们可以使用概率密度函数来描述对天体的测定，这一概率函数在天球不同位置可能是不同的。我们定义了一个模型$M$，通过一个三维标准向量$\boldsymbol{m}$表示天球上的一个观测目标，$p(\boldsymbol{x}|\mathbf{m}, M)$ 表示一个准确位置为$\boldsymbol{m}$的天体在位置$\boldsymbol{x}$被观测到的概率。这是一个标准化的概率密度函数，即其积分为1。

$$\int d^3x \cdot p(\boldsymbol{x}|\mathbf{m}, M) = 1 \tag{5.1}$$

对于单个观测目标$\boldsymbol{x}_1$，应用贝叶斯定理取得其验后概率密度，即其准确位置$\boldsymbol{m}$为$\boldsymbol{x}_1$的概率为

$$p(\boldsymbol{m}|\boldsymbol{x}_1, M) = \frac{p(\boldsymbol{x}_1|\boldsymbol{m}, M)\, p(\boldsymbol{m}|M)}{p(\boldsymbol{x}_1|M)} \tag{5.2}$$

其中，$\boldsymbol{m}$在天球上的验前概率$p(\boldsymbol{m}|M)$用狄拉克符号$\delta$表示为，

$$p(\boldsymbol{m}|M) = \frac{1}{4\pi}\delta(|\boldsymbol{m}| - 1) \tag{5.3}$$

由全概率公式可得，$\boldsymbol{x}_1$在天球上的验前概率

$$p(\boldsymbol{x}_1|M) = \int d^3m \cdot p(\boldsymbol{m}|M)\, p(\boldsymbol{x}_1|\boldsymbol{m}, M) \tag{5.4}$$

不同望远镜的多次观测带来了不同精度的数据，我们引入贝叶斯推断以研究它们源自同一天体的可信程度。给定一个假设$H$，所有的数据都来自同一天体$\boldsymbol{m}$；再给定一个对立假设$K$，所有的观测数据都来自不同天体，且它们都不源自$\boldsymbol{m}$。有坐标集合$D = \{\mathbf{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_g\}$ 对应$n$ 次不同观测。通过计算$H$和$K$的验前概率与验后概率，得到贝叶斯因子

$$B(H, K|D) = \frac{P(H|D)\,/P(H)}{P(K|D)\,/P(K)} \tag{5.5}$$

应用贝叶斯定理，得

$$B(H, K|D) = \frac{p(D|H)}{p(D|K)} \tag{5.6}$$

贝叶斯因子即两个假设的似然函数的比值。实际的计算发在生$H$和$K$的参数化模型中，并在整个参数空间中分别对似然函数的概率密度函数进行积分。$H$假

设所有观测对象都是同一目标，因而可以用一个共同位置$\boldsymbol{m}$来对它进行参数化。对于$D$中的各次独立观测结果，它们的联合概率密度函数即是它们的观测精度的乘积$p_1 \cdot p_2 \cdot \cdots \cdot p_n$，积分过程可简化为

$$p\left(D|H\right) = \int d^3m \cdot p\left(\boldsymbol{m}|H\right) \prod_{i=1}^{n} p_i\left(\boldsymbol{x}_i|\boldsymbol{m}, H\right) \tag{5.7}$$

另一方面，对立假设$K$将被不同的位置$\{\mathbf{m}_i\}$ 参数化，其概率为各次观测的概率密度函数的积分的乘积

$$p\left(D|K\right) = \prod_{i=1}^{n} \left[\int d^3m_i \cdot p\left(\boldsymbol{m}_i|K\right) p_i\left(\boldsymbol{x}_i|\boldsymbol{m}_i, K\right)\right] \tag{5.8}$$

当贝叶斯因子$B\left(H, K|D\right)$数值远大于1时，可认为$H$成立，即所有的观测数据来自同一天体；当它小于1时，则认为$K$ 成立，即观测数据对应不同天体；如与1保持一个数量级，则可认为两种假设都存疑，需要进一步检验。

贝叶斯因子法的优势在于，它可以随着新证据的加入而不断迭代进行下去。一旦一个证据的贝叶斯因子如$\frac{p(D|H)}{p(D|K)}$ 被计算出来，它即能成为后面新证据的验前概率的一部分。根据贝叶斯因子的定义方式，新旧两个贝叶斯因子可直接相乘而取得新的贝叶斯因子，并进一步对假设$H$ 和$K$进行检验。这不但可以充分利用已有的计算，还可以不断加入新的数据、证据对假设进行检验。

## 5.2　从赤道坐标到天球切平面坐标

由于涉及到几何计算及多重积分，为简化计算，我们在光学源$x_0$处做天球切平面。令$x_0$在切平面上的坐标为$(0, 0)$，并将所有在其附近的射电源坐标也投影到这个切平面上。对射电源坐标进行投影的同时将赤道坐标转为切平面的上笛卡尔坐标。这样，空间三维立体几何计算变成了相对简单的二维平面几何计算，一定程度上也符合了观测时的实际情形，即望远镜对目标天区的观测结果就是一个平面图。这一步操作将空间三维坐标变成了平面二维坐标，也将后续的空间三重积分变成了平面二重积分，减少了计算复杂度。

与前几章一致，天球仍然被视为一个以球心为原点，半径为1的球面。下面需要求出定义切平面的两个单位向量，西向量$\boldsymbol{w}$ 及北向量$\boldsymbol{n}$。$x_0$的赤道坐标$(\alpha, \delta)$在XoY平面的投影向量为

$$\boldsymbol{x}_{\mathbf{0}}' = \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix} \tag{5.9}$$

因$\cos\alpha^2 + \sin\alpha^2 = 1$，此向量也是单位向量。西向量$\boldsymbol{w}$在切平面上，因而须与$\boldsymbol{x_0'}$ 相垂直并指向西，可表示为

$$\boldsymbol{w} = \begin{pmatrix} \sin\alpha \\ -\cos\alpha \\ 0 \end{pmatrix} \tag{5.10}$$

$x_0$在XoZ平面上的投影向量为

$$\boldsymbol{x_0''} = \begin{pmatrix} \cos\delta \\ \sin\delta \end{pmatrix} \tag{5.11}$$

则其北向量也须与此向量相垂直，并指向北。可表示为

$$\boldsymbol{n'} = \begin{pmatrix} -\sin\delta \\ 0 \\ \cos\delta \end{pmatrix} \tag{5.12}$$

但是这个$\boldsymbol{n'}$指向的是赤经0°处的北天极方向，还须将$\boldsymbol{n}$旋转至$\alpha$处。因而完整的$\boldsymbol{n}$应为

$$\boldsymbol{n} = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} -\sin\delta \\ 0 \\ \cos\delta \end{pmatrix} = \begin{pmatrix} -\sin\delta\cos\alpha \\ -\sin\delta\sin\alpha \\ \cos\delta \end{pmatrix} \tag{5.13}$$

得到$\boldsymbol{w}$与$\boldsymbol{n}$之后，就可以将其它天球坐标投影到$x_0$处的切平面上了。如点$r'$的坐标为$(\alpha', \delta')$，它在$x_0$的切平面上的坐标$(p, q)$ 的计算方法为

$$p(r') = (r' - r_c) \cdot \boldsymbol{n} = \begin{pmatrix} x' - x \\ y' - y \\ z' - z \end{pmatrix} \cdot \boldsymbol{n} \tag{5.14}$$

$$p(\alpha', \delta') = \begin{pmatrix} \cos\delta'\cos\alpha' - \cos\delta\cos\alpha \\ \cos\delta'\sin\alpha' - \cos\delta\sin\alpha \\ \sin\delta' - \sin\delta \end{pmatrix} \cdot \begin{pmatrix} -\sin\delta\cos\alpha \\ -\sin\delta\sin\alpha \\ \cos\delta \end{pmatrix} \tag{5.15}$$

$$q(r') = (r' - r_c) \cdot \boldsymbol{w} = \begin{pmatrix} x' - x \\ y' - y \\ z' - z \end{pmatrix} \cdot \boldsymbol{w} \tag{5.16}$$

$$q(\alpha', \delta') = \begin{pmatrix} \cos\delta' \cos\alpha' - \cos\delta\cos\alpha \\ \cos\delta' \sin\alpha' - \cos\delta\sin\alpha \\ \sin\delta' - \sin\delta \end{pmatrix} \cdot \begin{pmatrix} \sin\alpha \\ -\cos\alpha \\ 0 \end{pmatrix} \tag{5.17}$$

## 5.3　直线对称模型

为了与射电星表进行交叉证认，需要给光学观测坐标和射电观测坐标的几何关系设立一个模型。由于典型的双侧喷流表现出的几何形态都是基本与中心连成一线。首先建立的模型是直线对称模型，即射电源的双极喷流与射电源中心成一条直线，且喷流的两瓣与中心的距离是相等的。若用向量$\boldsymbol{m}$ 表示射电源中心，$\boldsymbol{m}'$表示一侧的一瓣，则另一瓣$\boldsymbol{m}''$ 可表示为$\boldsymbol{m}'' = 2\boldsymbol{m} - \boldsymbol{m}'$。

如图5.1，对于一个光学源，其附近可能有0个或数个射电源。任一射电源可能是与该光学源一样的中心，称为CORE；或是喷流瓣，称为LOBE 或与光学源无关，即NONE。以图5.1椭圆框中的两个射电源为例，他们可以构成的组合有：

$$\begin{gathered} (NONE, NONE) \\ (NONE, CORE) \\ (NONE, LOBE) \\ (CORE, NONE) \\ (CORE, LOBE) \\ (LOBE, NONE) \\ (LOBE, LOBE) \end{gathered} \tag{5.18}$$

三个射电源则更为复杂，如图5.1的矩形框，不仅包含了上述的所有组合，还可以构成直线对称模型所需的三个部分：(CORE,LOBE,LOBE)。任一种组合都不能有超过一个CORE，也不能超过两个LOBE，而NONE 则可以有数个。如果有超过三个以上射电源，也同样受此限制。如图5.1中央的光学源，实际上需要考虑虚线圆内的所有射电源的组合情况。分别计算这些假设，并对比它们的值，以取得对光学源而言在直线对称模型中最可能的一种假设，也即最符合：光学源附近有一个射电源，两侧有距离相等的喷流瓣与射电源连成一线。

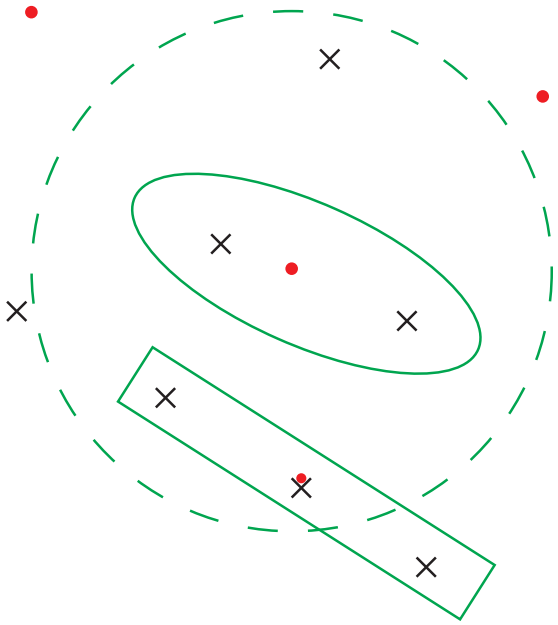

图 5.1: 光学源与射电源在天球上的分布示例，圆点表示光学源，交叉线表示射电源。

假设有四个射电源$D = \{y_0, y_1, y_2, y_3\}$，它们其中一个(CORE,LOBE,LOBE,NONE)假设的似然函数计算公式为

$$
\begin{aligned}
& p(\texttt{CORE}, \texttt{LOBE}, \texttt{LOBE}, \texttt{NONE}) \qquad (5.19)\\
= & \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) L_{x0}(\boldsymbol{m}_0) L_{y0}(\boldsymbol{m}_0) \right.\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1 | \boldsymbol{m}_0) L_{y1}\left(\boldsymbol{m}_1\right) L_{y2}\left(2\boldsymbol{m}_0 - \boldsymbol{m}_1\right) \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) L_{y3}(\boldsymbol{m}_2) \right\}
\end{aligned}
$$

可以看到$y_0$作为CORE是与$x_0$共同对应的是同一个中心$m_0$，它们的概率密度函数直接相乘来做积分。$y_1$与$y_2$作为LOBE对应的是$m_1$且与$m_0$相关，$y_1$、$y_2$的概率密度函数直接相乘作积分之后再与$x_0$、$y_0$作积分。而NONE，与$m_0$、$m_1$均无关，它的概率密度函数的积分直接与前面的积分相乘。把一个LOBE 去掉，它们其中

一个(CORE,LOBE,NONE,NONE)假设的似然函数计算公式变为

$$\begin{aligned} & p(\texttt{CORE},\texttt{LOBE},\texttt{NONE},\texttt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)L_{x0}(\boldsymbol{m}_0)L_{y0}(\boldsymbol{m}_0)\right. \\ & \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)L_{y1}\left(\boldsymbol{m}_1\right)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)L_{y2}(\boldsymbol{m}_2)\right\} \\ & \left\{\int d^2m_3\cdot p(\boldsymbol{m}_3)L_{y3}(\boldsymbol{m}_3)\right\} \end{aligned} \tag{5.20}$$

若将CORE去掉，则它们其中一个(NONE,LOBE,LOBE,NONE) 假设的似然函数计算公式为

$$\begin{aligned} & p(\texttt{NONE},\texttt{LOBE},\texttt{LOBE},\texttt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)L_{x0}(\boldsymbol{m}_0)\right. \\ & \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)L_{y1}\left(\boldsymbol{m}_1\right)L_{y2}\left(2\boldsymbol{m}_0-\boldsymbol{m}_1\right)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)L_{y0}(\boldsymbol{m}_2)\right\} \\ & \left\{\int d^2m_3\cdot p(\boldsymbol{m}_3)L_{y3}(\boldsymbol{m}_3)\right\} \end{aligned} \tag{5.21}$$

对于只有一个LOBE的情形，则它们其中一个(NONE,LOBE,LOBE,NONE) 假设的似然函数计算公式为

$$\begin{aligned} & p(\texttt{NONE},\texttt{LOBE},\texttt{NONE},\texttt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)L_{x0}(\boldsymbol{m}_0)\right. \\ & \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)L_{y1}\left(\boldsymbol{m}_1\right)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)L_{y0}(\boldsymbol{m}_2)\right\} \\ & \left\{\int d^2m_3\cdot p(\boldsymbol{m}_3)L_{y2}(\boldsymbol{m}_3)\right\} \\ & \left\{\int d^2m_4\cdot p(\boldsymbol{m}_4)L_{y3}(\boldsymbol{m}_4)\right\} \end{aligned} \tag{5.22}$$

把所有LOBE去掉，只考虑(CORE)的情况，则变回到了通常的交叉证认情形。似然函数计算公式变为

$$\begin{aligned} & p(\mathtt{CORE},\mathtt{NONE},\mathtt{NONE},\mathtt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)L_{x0}(\boldsymbol{m}_0)L_{y0}(\boldsymbol{m}_0)\right\} \\ & \left\{\int d^2m_1\cdot p(\boldsymbol{m}_1)L_{y1}(\boldsymbol{m}_1)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)L_{y2}(\boldsymbol{m}_2)\right\} \\ & \left\{\int d^2m_3\cdot p(\boldsymbol{m}_3)L_{y3}(\boldsymbol{m}_3)\right\} \end{aligned} \tag{5.23}$$

若所有的射电成员均独立，且与$m_0$无关，则它们的(NONE,NONE,NONE,NONE)假设的似然函数计算公式为

$$\begin{aligned} & p(\mathtt{NONE},\mathtt{NONE},\mathtt{NONE},\mathtt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)L_{x0}(\boldsymbol{m}_0)\right\} \\ & \left\{\int d^2m_1\cdot p(\boldsymbol{m}_1)L_{y0}(\boldsymbol{m}_1)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)L_{y1}(\boldsymbol{m}_2)\right\} \\ & \left\{\int d^2m_3\cdot p(\boldsymbol{m}_3)L_{y2}(\boldsymbol{m}_3)\right\} \\ & \left\{\int d^2m_4\cdot p(\boldsymbol{m}_4)L_{y3}(\boldsymbol{m}_4)\right\} \end{aligned} \tag{5.24}$$

以上所有的$D=\{y_0,y_1,y_2,y_3\}$的位置是可以互换的，也即需要计算它们所有可能的组合。然后对比结果来取得一个最好的假设。

下面需要考量各个概率密度函数的选择。正态分布（高斯分布）是颇为常见的一种分布，许多的效应都表现出了概率显著朝某一点聚集的趋势，球面上也同样表现出了这一效应[74][75]。因而我们选择了使用了正态分布概率密度函数来评估天体之间的距离，即$L_{x0}$、$L_{y0}$、$L_{y1}$、$L_{y2}$、$L_{y3}$ 等概率密度函数均选用了正态分布。如$L_{x0}$表示为

$$L_{x0}(\boldsymbol{m}_0)=g(\boldsymbol{x}_0|\boldsymbol{m}_0,\Sigma_{x0}) \tag{5.25}$$

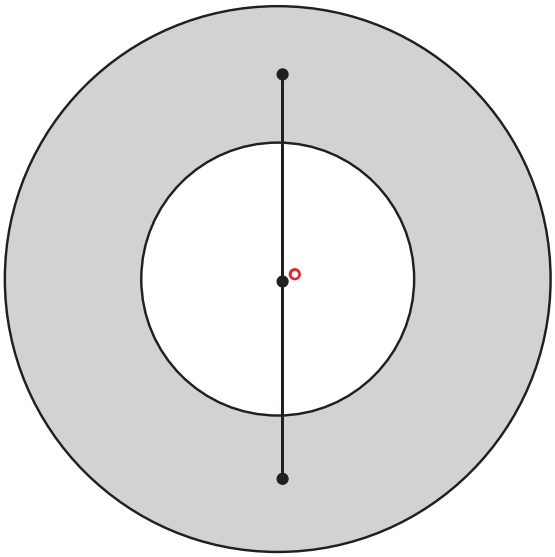

图 5.2: 当LOBE位于图中灰色区域时，它有一个相同的大于0的概率值，其它区域为零。图中三个黑色代表三个射电源，而空心点表示光学源的位置。

$g$即高斯分布（Gaussian Distribution）的英文首字母，因为是概率密度函数，用小写。而对于天体在球面分布的验前概率$p(\boldsymbol{m}_i)$，则依天球表面积

$$a = 4\pi r^2 = 4\pi \left(r/pi \times 180 \times 60 \times 60\right)^2 = 534638377792.47 arcsec^2 \tag{5.26}$$

设定为$p(\boldsymbol{m}_i) = \frac{1}{a} = 1.8704231524E-12$，即假设每平方角秒上有一个天体。这对于精度差于$1''$的星表是足够的。对于LOBE所对应的坐标$m_1$，与中心源$m_0$有关，需要加入它与$m_0$的关系。$p(\boldsymbol{m}_1|\boldsymbol{m}_0)$ 可采用的较直接的一个概率分布如图5.2所示，当LOBE位于图中灰色区域时，它有一个有效的概率值，其它区域为零。画出其概率分布曲线则形如5.3，实际上就是均匀分布概率密度函数。用$R$表示大圆的半径，$r$表示小圆半径，则此均匀分布概率密度函数的计算公式为

$$p\left(m_1|m_0\right) = \begin{cases} \left[\left(R^2 - r^2\right)\pi\right]^{-1} & 若 r < |m_1 - m_0| < R \\ 0 \end{cases} \tag{5.27}$$

对于$m_1$的概率密度函数$p(m_1|m_0)$，除了均匀分布概率密度函数，还可以考虑瑞利分布（Rayleigh Distribution）、对数正态分布（Log-normal Distribution）。使用它们的理由是：离$m_0$越远的$m_1$越不可能是LOBE。瑞利分布的概率

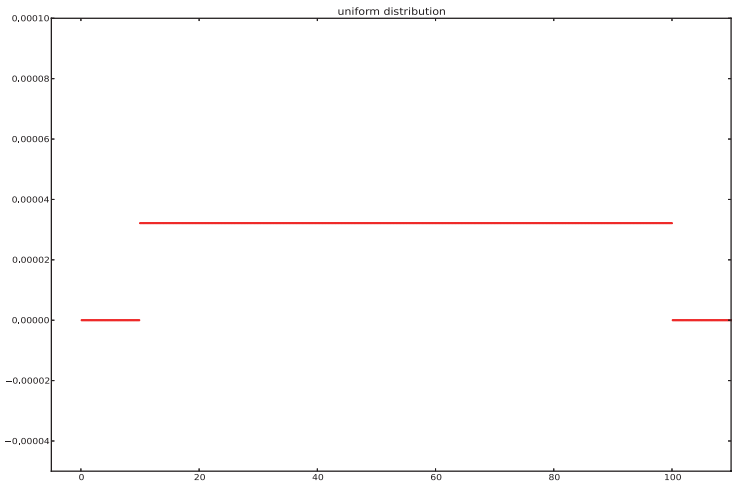


图 5.3: 均匀概率分布，当位于有效范围内时，概率密度函数值为一个常数；当位于有效范围外时，概率密度函数值为0。

密度函数为

$$p(k|\sigma) = \frac{k}{\sigma^2} e^{-k^2/2\sigma^2} \tag{5.28}$$

如图5.4，它有一个$\sigma$值来改变曲线的形态。瑞利函数的均值（$mean$）、方差（$var$）与$\sigma$的关系为

$$mean = \sigma\frac{\pi}{2}, var = \frac{4-\pi}{2}\sigma^2 \tag{5.29}$$

对数正态分布的概率密度函数为

$$\ln N(\mu, \sigma^2) = \frac{1}{x\sqrt{2\pi\sigma^2}} e^{-(\ln x-\mu)^2/2\sigma^2} \tag{5.30}$$

如图5.5，通过参数$\mu$、$\sigma$改变其概率密度函数曲线形态。它的均值、方差与参数$\mu$、$\sigma$的关系为

$$mean = e^{\mu+\sigma^2} \tag{5.31}$$

$$var = \left(e^{\sigma^2} - 1\right) e^{2\mu+\sigma^2} \tag{5.32}$$

$$\mu = \ln mean - \frac{\sigma^2}{2} \tag{5.33}$$

$$\sigma = \sqrt{\ln\left(\frac{var}{mean^2} + 1\right)} \tag{5.34}$$

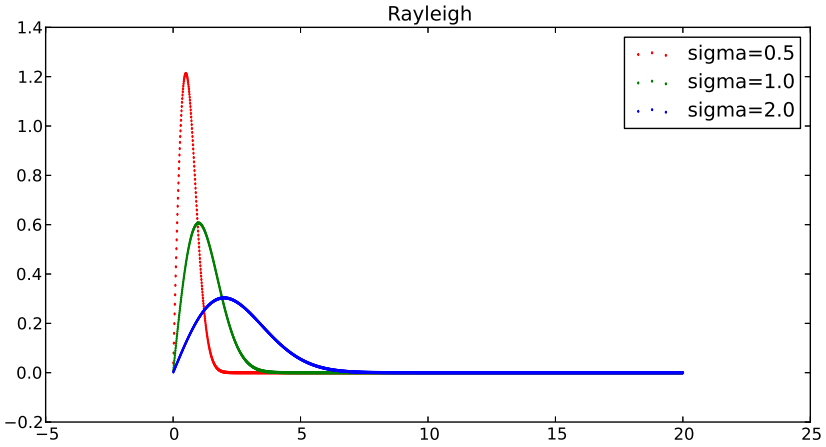


图 5.4: 瑞利分布概率密度函数在不同$\sigma$值下的曲线形态。

在确定三类概率分布之后，最终的假设计算公式也确定下来。以前述最复杂的(CORE,LOBE,LOBE,NONE)为例，其计算公式是

$$
\begin{aligned}
& p(\texttt{CORE}, \texttt{LOBE}, \texttt{LOBE}, \texttt{NONE}) \qquad (5.35)\\
= \; & \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) g\left(\boldsymbol{x}_0 | \boldsymbol{m}_0, \Sigma_{x0}\right) g\left(\boldsymbol{y}_0 | \boldsymbol{m}_0, \Sigma_{y0}\right) \right.\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1 | \boldsymbol{m}_0) g\left(\boldsymbol{y}_1 | \boldsymbol{m}_1, \Sigma_{y1}\right) g\left(\boldsymbol{y}_2 | 2\boldsymbol{m}_0 - \boldsymbol{m}_1, \Sigma_{y2}\right) \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) g\left(\boldsymbol{y}_3 | \boldsymbol{m}_2, \Sigma_{y3}\right) \right\}
\end{aligned}
$$

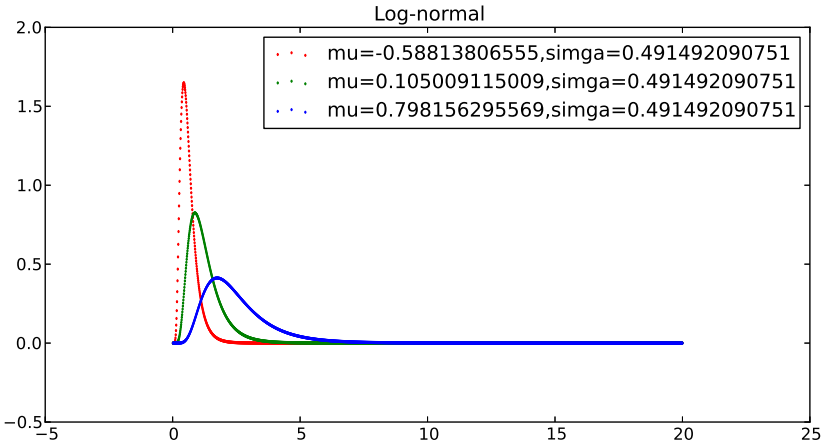


图 5.5: 对数正态分布概率密度函数在不同$\mu$、$\sigma$值下的曲线形态。

## 5.4 直线非对称模型

考虑到由于望远镜观测视角等原因，所观测到的双极喷流在坐标上不一定是对称的，我们给第二个瓣增加了在直线上的位移比率$k$。第二个瓣的位置范围变为

$$2\boldsymbol{m}_0 - \boldsymbol{m}_1 + k(\boldsymbol{m}_0 - \boldsymbol{m}_1) = (2+k)\boldsymbol{m}_0 - (1+k)\boldsymbol{m}_1 \tag{5.36}$$

由于增加了一个变量$k$，在涉及到两个LOBE的计算中将需要对$k$ 也做一次积分。与前述积分均为二重积分不同，只需对$k$ 做一重积分即可。而随着积分层次的增加，实际计算量也将随之增加数倍。前述的射电源的各种组合的似然函数计算也需要相应做一些改变，但实际只改变带有两个LOBE的形式，即(CORE,LOBE,LOBE)或(LOBE,LOBE)。

增加$k$后的候选假设(CORE,LOBE,LOBE,NONE)的计算公式变为

$$
\begin{aligned}
& p(\texttt{CORE},\texttt{LOBE},\texttt{LOBE},\texttt{NONE}) \qquad (5.37)\\
= \;& \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) L_{x0}(\boldsymbol{m}_0) L_{y0}(\boldsymbol{m}_0) \right.\\
& \int dk \cdot p(k)\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) L_{y1}\left(\boldsymbol{m}_1\right) L_{y2}\left[(2+k)\boldsymbol{m}_0 - (1+k)\boldsymbol{m}_1\right] \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) L_{y3}(\boldsymbol{m}_2) \right\}\\
= \;& \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) g\left(\boldsymbol{x}_0|\boldsymbol{m}_0, \Sigma_{x0}\right) g\left(\boldsymbol{y}_0|\boldsymbol{m}_0, \Sigma_{y0}\right) \right.\\
& \int dk \cdot p(k)\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) g\left(\boldsymbol{y}_1|\boldsymbol{m}_1, \Sigma_{y1}\right) g\left(\boldsymbol{y}_2|(2+k)\boldsymbol{m}_0 - (1+k)\boldsymbol{m}_1, \Sigma_{y2}\right) \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) g\left(\boldsymbol{y}_3|\boldsymbol{m}_2, \Sigma_{y3}\right) \right\}
\end{aligned}
$$

而增加$k$后的候选假设(NONE,LOBE,LOBE,NONE)的计算公式变为

$$
\begin{aligned}
& p(\texttt{NONE}, \texttt{LOBE}, \texttt{LOBE}, \texttt{NONE}) \qquad (5.38)\\
= & \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) L_{x0}(\boldsymbol{m}_0) \right.\\
& \int dk \cdot p(k)\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) L_{y1}\left(\boldsymbol{m}_1\right) L_{y2}\left[(2+k)\boldsymbol{m}_0 - (1+k)\boldsymbol{m}_1\right] \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) L_{y0}(\boldsymbol{m}_2) \right\}\\
& \left\{ \int d^2 m_3 \cdot p(\boldsymbol{m}_3) L_{y3}(\boldsymbol{m}_3) \right\}\\
= & \left\{ \int d^2 m_0 \cdot p(\boldsymbol{m}_0) g\left(\boldsymbol{x}_0|\boldsymbol{m}_0, \Sigma_{x0}\right) \right.\\
& \int dk \cdot p(k)\\
& \left. \int d^2 m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) g\left(\boldsymbol{y}_1|\boldsymbol{m}_1, \Sigma_{y1}\right) g\left(\boldsymbol{y}_2|(2+k)\boldsymbol{m}_0 - (1+k)\boldsymbol{m}_1, \Sigma_{y2}\right) \right\}\\
& \left\{ \int d^2 m_2 \cdot p(\boldsymbol{m}_2) g\left(\boldsymbol{y}_0|\boldsymbol{m}_2, \Sigma_{y0}\right) \right\}\\
& \left\{ \int d^2 m_3 \cdot p(\boldsymbol{m}_3) g\left(\boldsymbol{y}_3|\boldsymbol{m}_3, \Sigma_{y3}\right) \right\}
\end{aligned}
$$

可以看到在$k$的积分计算部分，$k$也有一个概率分布，与对天体所使用的概率分布一样，对它所采用的也是正态分布，只是限于一维。即第二个瓣即便与第一瓣不甚对称，也仍非常可能是在$2\boldsymbol{m}_0 - \boldsymbol{m}_1$ 附近。

## 5.5 重点抽样积分方法

对于前述的积分计算，并不是所有公式都有一个解析解，因而对大部分的公式只能采取计算机数值计算的方式。在积分的数值计算方面，最有影响力的便是蒙特卡罗（Monte Carlo）方法，又称计算机随机模拟方法。它通过对随机抽样进行统计取得近似值。因而涉及如何产生随机数，数据统计及误差估计。尽管它的计算结果的精度不很高，但是它能迅速地给出一个近似值，在应用上也极具价值。一般有随机投点法、平均值法、分层抽样、重要抽样法、相关抽样法等计算方法。

重要抽样法（Importance Sampling）是蒙特卡罗方法中的一种降方差方法，由Marshall[76][77] 提出。其原理起源于数学上的变量代换方法的思想，即

$$\int_0^1 f(x)dx = \int_0^1 \frac{f(x)}{g(x)}g(x)dx = \int_0^1 \frac{f(x)}{g(x)}dG(x) \tag{5.39}$$

此后随机点的选择将按$G(x)$函数分布，而不是简单的均匀变化的$dx$。被积函数由$f(x)$变为$\frac{f(x)}{g(x)}$。计算方法也变为产生$n$个$G(x)$分布的随机数作为$x$，然后代入$\frac{f(x)}{g(x)}$进行计算并统计它们相加的和$sum$，最后$\frac{sum}{n}$ 即为所求的积分的近似值。与均匀取随机数方法相比，重要抽样法的随机数是在我们所关注的区域产生的，如正态分布的的随机数大量出现在平均值附近，离平均值越远随机数越少。这样就大量减少了无效计算，减小了误差。

重要抽样法的方差为

$$var = \frac{[\sum f(x_i)]^2 - \sum [f(x_i)^2]}{n} \tag{5.40}$$

二维重点抽样法与前述方法几近相同，只是增加了一个维度，积分区域从线变成面。如计算二维区域$D$上的一个定积分$\iint_D f(x,y)dxdy$。则取概率分布函数$G(x,y)$的概率密度函数为$g(x,y)$，定积分变为

$$\iint_D f(x,y)dxdy = \iint_D \frac{f(x,y)}{g(x,y)}g(x,y)dxdy = \int_D \frac{f(x,y)}{g(x,y)}dG(x,y) \tag{5.41}$$

需要产生出n个服从$G(x,y)$的二维随机变量$(x_i,y_i)$，并计算$\frac{f(x_i,y_i)}{g(x_i,y_i)}$，最后$\frac{1}{n}\sum_n \frac{f(x_i,y_i)}{g(x_i,y_i)}$ 即为二重积分的近似值。很显然的一个好处，便是原本二重积分需要二重循环，现在只需要一重循环即可，二维的维度被放置到随机变量$(x,y)$上了。

### 5.5.1　多维正态分布概率密度函数相乘

因为(CORE,LOBE,LOBE)等假设的似然函数的计算中大量使用了概率密度函数，我们很自然地采用了重要抽样法来计算积分，一来可以降低计算复杂度，二来也可以减少方差。但这面临两个难题，一是如何产生二维正态分布随机数；二是虽然我们试图产生二维正态随机数的方式来减少对正态分布概率密度函数的直接计算，但若一个积分中存在两个正态分布概率密度函数时，如带有CORE 的假设或带有(LOBE,LOBE)的假设。只取其中一个正态分布的随机数，

则另外一个正态分布概率密度函数仍需要进行计算，这样计算量并未减少。当算式中存在二个二维正态分布概率密度函数相乘的情况，最好是将它们合二为一，以减少一半的计算量。

多维正态概率密度函数可表示为

$$\begin{aligned} g(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) &= (2\pi)^{-d/2}\left|\boldsymbol{\Sigma}\right|^{-1/2}\exp\left[-\frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)^{T}\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{x}-\boldsymbol{\mu}\right)\right] \\ &= (2\pi)^{-d/2}\left|\boldsymbol{\Sigma}\right|^{-1/2}\exp\left\{-\frac{1}{2}\left[-(\boldsymbol{x}-\boldsymbol{\mu})\right]^{T}\boldsymbol{\Sigma}^{-1}\left[-(\boldsymbol{x}-\boldsymbol{\mu})\right]\right\} \\ &= g(\boldsymbol{\mu}|\boldsymbol{x},\boldsymbol{\Sigma}) \end{aligned} \tag{5.42}$$

而两个多维正态分布概率密度函数的乘积[78][79] 可表示为

$$g_p(\boldsymbol{\mu},\boldsymbol{\Sigma}) = g(\boldsymbol{\mu}_1,\boldsymbol{\Sigma}_1)\cdot g(\boldsymbol{\mu}_2,\boldsymbol{\Sigma}_2) \tag{5.43}$$

$$\text{其中}\boldsymbol{\Sigma} = \left(\boldsymbol{\Sigma}_1^{-1}+\boldsymbol{\Sigma}_2^{-1}\right)^{-1} \tag{5.44}$$

$$\boldsymbol{\mu} = \left(\boldsymbol{\Sigma}\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1+\boldsymbol{\Sigma}\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\mu}_2\right) \tag{5.45}$$

两个$n$维正态分布概率密度函数的乘积虽然仍然是$n$维正态概率密度函数，但它不再标准化，它的积分是一个常数$C$。需要对它的积分除以该常数$C$以使其重新规范化。

$$C = (2\pi)^{-\frac{d}{2}}\left|\boldsymbol{\Sigma}\right|^{\frac{1}{2}}\left|\boldsymbol{\Sigma}_1\right|^{-\frac{1}{2}}\left|\boldsymbol{\Sigma}_2\right|^{-\frac{1}{2}}exp\left[\frac{1}{2}\left(\boldsymbol{\mu}^{T}\boldsymbol{\Sigma}\boldsymbol{\mu}-\boldsymbol{\mu}_1^{T}\boldsymbol{\Sigma}_1\boldsymbol{\mu}_1-\boldsymbol{\mu}_2^{T}\boldsymbol{\Sigma}_2\boldsymbol{\mu}_2\right)\right] \tag{5.46}$$

由公式5.42可知，正态分布概率密度函数中的两个变量$\boldsymbol{x}$与$\boldsymbol{\mu}$可互换位置，即$\boldsymbol{x}$与$\boldsymbol{\mu}$的距离，亦可描述为$\boldsymbol{\mu}$与$\boldsymbol{x}$ 的距离。这也是后续简化编程实现的一个重要转换手段。仍以假设(CORE,LOBE,LOBE,NONE)为例，将原始的积分方程变换为

$$\begin{aligned} & p(\texttt{CORE},\texttt{LOBE},\texttt{LOBE},\texttt{NONE}) \\ = & \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)g\left(\boldsymbol{m}_0|\boldsymbol{x}_0,\Sigma_{x0}\right)g\left(\boldsymbol{m}_0|\boldsymbol{y}_0,\Sigma_{y0}\right)\right. \\ & \int dk\cdot p(k) \\ & \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)g\left(\boldsymbol{m}_1|\boldsymbol{y}_1,\Sigma_{y1}\right)g\left(\boldsymbol{y}_2|(2+k)\boldsymbol{m}_0-(1+k)\boldsymbol{m}_1,\Sigma_{y2}\right)\right\} \\ & \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)g\left(\boldsymbol{m}_2|\boldsymbol{y}_3,\Sigma_{y3}\right)\right\} \end{aligned} \tag{5.47}$$

由于

$$
\begin{aligned}
& g\left(\boldsymbol{y}_2|(2+k)\boldsymbol{m}_0-(1+k)\boldsymbol{m}_1,\Sigma_{y2}\right) && (5.48)\\
= \;& (2\pi)^{-d/2}\left|\Sigma_{y2}\right|^{-1/2} && (5.49)\\
& \exp\left\{-\frac{1}{2}\left[(k+2)\boldsymbol{m}_0-(k+1)\boldsymbol{m}_1-\boldsymbol{y}_2\right]^T\Sigma_{y2}^{-1}\left[(k+2)\boldsymbol{m}_0-(k+1)\boldsymbol{m}_1-\boldsymbol{y}_2\right]\right\}\\
= \;& \frac{(2\pi)^{-d/2}}{(k+1)^d}\left|\frac{\Sigma_{y2}}{(k+1)^2}\right|^{-1/2} && (5.50)\\
& \exp\left\{-\frac{1}{2}\left[\frac{k+2}{k+1}\boldsymbol{m}_0-\boldsymbol{m}_1-\frac{\boldsymbol{y}_2}{k+1}\right]^T\left[\frac{\Sigma_{y2}}{(k+1)^2}\right]^{-1}\left[\frac{k+2}{k+1}\boldsymbol{m}_0-\boldsymbol{m}_1-\frac{\boldsymbol{y}_2}{k+1}\right]\right\}\\
= \;& \frac{1}{(k+1)^d}g\left[\boldsymbol{m}_1|\frac{(k+2)}{(k+1)}\boldsymbol{m}_0-\frac{\boldsymbol{y}_2}{(k+1)},\frac{\Sigma_{y2}}{(k+1)^2}\right] && (5.51)
\end{aligned}
$$

最终，

$$
\begin{aligned}
& p(\texttt{CORE},\texttt{LOBE},\texttt{LOBE},\texttt{NONE}) && (5.52)\\
= \;& \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)g\left(\boldsymbol{m}_0|\boldsymbol{x}_0,\Sigma_{x0}\right)g\left(\boldsymbol{m}_0|\boldsymbol{y}_0,\Sigma_{y0}\right)\right.\\
& \int dk\cdot p(k)\\
& \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)g\left(\boldsymbol{m}_1|\boldsymbol{y}_1,\Sigma_{y1}\right)\frac{g\left(\boldsymbol{m}-1|\frac{(2+k)}{1+k}\boldsymbol{m}_0-\frac{\boldsymbol{y}_2}{(1+k)},\frac{\Sigma_{y2}}{(k+1)^2}\right)}{(1+k)^2}\right\}\\
& \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)g\left(\boldsymbol{m}_2|\boldsymbol{y}_3,\Sigma_{y3}\right)\right\}\\
= \;& \left\{\int d^2m_0\cdot p(\boldsymbol{m}_0)g_p\left(\boldsymbol{m}_0|\boldsymbol{x}_0,\boldsymbol{y}_0,\Sigma_{x0},\Sigma_{y0}\right)\right.\\
& \int dk\cdot p(k)\\
& \left.\int d^2m_1\cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0)\frac{1}{(k+1)^2}g\left(\boldsymbol{m}_1|\boldsymbol{y}_1,\frac{k+2}{k+1}\boldsymbol{m}_0-\frac{\boldsymbol{y}_2}{k+1},\Sigma_{y1},\Sigma_{y2}\right)\right\}\\
& \left\{\int d^2m_2\cdot p(\boldsymbol{m}_2)g\left(\boldsymbol{m}_2|\boldsymbol{y}_3,\Sigma_{y3}\right)\right\}
\end{aligned}
$$

上述公式中的矩阵$\Sigma_{x0}$或$\Sigma_{yi}$可以是一个源在赤经和赤纬上的误差，也可以指定为常量，按数据精度或需求而定。如，对于光学源我们所使用的$\Sigma_{x0}$为

$$\Sigma_{x0}=\begin{pmatrix}0.2^2 & 0\\ 0 & 0.2^2\end{pmatrix} \tag{5.53}$$

### 5.5.2 针对假设比较的简化

从前文中对各类假设的公式描述可以看到，其中$p(\boldsymbol{m}_0)$是一个常数$\frac{1}{a}$（其中$a$为整个天球的表面积，单位为$arcsec^2$），可以将此常数提取出积分符号。此外，像形如

$$p(\boldsymbol{m}_2)\int d^2m_2 \cdot g\left(\boldsymbol{m}_2|\boldsymbol{y}_3,\Sigma_{y3}\right) \tag{5.54}$$

的公式中只有一个正态分布概率密度函数，这意味着，它的积分为1。因而，此公式的值为$\frac{1}{a}$。根据这两个信息，所有的公式都可以进行简化。为方便起见，使用一对大括号$\{\cdot\}$来表示一个积分计算。则前述的各类假设的似然函数可以重新描述为

$$\begin{aligned} &p(CORE,LOBE,LOBE,NONE) \\ &=\frac{1}{a}\{x_0y_0y_1y_2\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^2}\{x_0y_0y_1y_2\} \end{aligned} \tag{5.55}$$

$$\begin{aligned} &p(CORE,LOBE,NONE,NONE) \\ &=\frac{1}{a}\{x_0y_0y_1\}\cdot\frac{1}{a}\{y_2\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^3}\{x_0y_0y_1\} \end{aligned} \tag{5.56}$$

$$\begin{aligned} &p(NONE,LOBE,LOBE,NONE) \\ &=\frac{1}{a}\{x_0y_1y_2\}\cdot\frac{1}{a}\{y_0\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^3}\{x_0y_1y_2\} \end{aligned} \tag{5.57}$$

$$\begin{aligned} &p(NONE,LOBE,NONE,NONE) \\ &=\frac{1}{a}\{x_0y_1\}\cdot\frac{1}{a}\{y_0\}\cdot\frac{1}{a}\{y_2\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^4}\{x_0y_1\} \end{aligned} \tag{5.58}$$

$$\begin{aligned} &p(CORE,NONE,NONE,NONE) \\ &=\frac{1}{a}\{x_0y_0\}\cdot\frac{1}{a}\{y_1\}\cdot\frac{1}{a}\{y_2\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^4}\{x_0y_0\}=\frac{1}{a^4}\cdot C_{x0y0} \end{aligned} \tag{5.59}$$

$$\begin{aligned} &p(NONE,NONE,NONE,NONE) \\ &=\frac{1}{a}\{x_0\}\cdot\frac{1}{a}\{y_0\}\cdot\frac{1}{a}\{y_1\}\cdot\frac{1}{a}\{y_2\}\cdot\frac{1}{a}\{y_3\}=\frac{1}{a^5} \end{aligned} \tag{5.60}$$

其中p(NONE, NONE, NONE, NONE)=$1/a^5$恒定是一个常数，而p(CORE, NONE, NONE, NONE)的积分有解析解，剩下的带有LOBE的假设均需进行积分计算。

由于p(NONE, NONE, NONE, NONE)是一个常数，可以让所有其它的假设都除以这个常数，得到该假设与假设(NONE, NONE, NONE, NONE)的似然函数的比值，即两个假设的贝叶斯因子。

$$B(CORE, LOBE, LOBE) \tag{5.61}$$
$$= \frac{p(CORE, LOBE, LOBE, NONE)}{p(NONE, NONE, NONE, NONE)} = \frac{\frac{1}{a^2}\{x_0 y_0 y_1 y_2\}}{\frac{1}{a^5}} = a^3 \cdot \{x_0 y_0 y_1 y_2\}$$

$$B(CORE, LOBE) \tag{5.62}$$
$$= \frac{p(CORE, LOBE, NONE, NONE}{p(NONE, NONE, NONE, NONE)} = \frac{\frac{1}{a^3}\{x_0 y_0 y_1\}}{\frac{1}{a^5}} = a^2 \cdot \{x_0 y_0 y_1\}$$

$$B(LOBE, LOBE) \tag{5.63}$$
$$= \frac{p(NONE, LOBE, LOBE, NONE}{p(NONE, NONE, NONE, NONE)} = \frac{\frac{1}{a^3}\{x_0 y_1 y_2\}}{\frac{1}{a^5}} = a^2 \cdot \{x_0 y_1 y_2\}$$

$$B(LOBE) \tag{5.64}$$
$$= \frac{p(NONE, LOBE, NONE, NONE}{p(NONE, NONE, NONE, NONE)} = \frac{\frac{1}{a^4}\{x_0 y_1\}}{\frac{1}{a^5}} = a \cdot \{x_0 y_1\}$$

$$B(CORE) \tag{5.65}$$
$$= \frac{p(CORE, NONE, NONE, NONE}{p(NONE, NONE, NONE, NONE)} = \frac{\frac{1}{a^4} \cdot \{x_0 y_0\}}{\frac{1}{a^5}} = a \cdot C_{x0y0}$$

可以看到，各类贝叶斯因子的计算不再与大于3个射电星源之外的其它射电源相关，且常数$a$的幂次与射电源的数目相等。其它的射电源的似然函数的积分都是常数，且都在分子、分母中出现而相互抵消了。更重要的是，计算出来的各个贝叶斯因子仍可以直接用于各个假设间的对比，如

$$\begin{aligned} &\frac{B(CORE, LOBE, LOBE)}{B(LOBE, LOBE)} \\ &= \frac{a^3 \cdot \{x_0 y_0 y_1 y_2\}}{a^2 \cdot \{x_0 y_1 y_2\}} = \frac{\frac{1}{a^2} \cdot \{x_0 y_0 y_1 y_2\}}{\frac{1}{a^3} \cdot \{x_0 y_1 y_2\}} \\ &= \frac{p(CORE, LOBE, LOBE, NONE)}{p(NONE, LOBE, LOBE, NONE)} \end{aligned} \tag{5.66}$$

这些贝叶斯因子都相当于对应的各个假设的似然函数除以一个常数，而两个假

设的似然函数的对比也是一个相除的过程，常数被抵消掉了。

这样，经过重重简化后，实际需要计算的内容变成B(CORE, LOBE, LOBE)、B(LOBE, LOBE)、B(CORE)、B(LOBE)、B(NONE)等5 类，而其中显然B(NONE) = p(NONE, NONE, NONE, NONE) / p(NONE, NONE, NONE, NONE) = 1。则对剩下的四类贝叶斯因子的计算公式归纳为

$$
\begin{aligned}
& B(CORE, LOBE, LOBE) \qquad (5.67)\\
= \; & a^3 \int d^2m \cdot p(\boldsymbol{m}_0) \cdot g_p\left(\boldsymbol{m}_0|\boldsymbol{x}_0, \boldsymbol{y}_0, \Sigma_{x0}, \Sigma_{y0}\right)\\
& \int dk \cdot p(k)\\
& \frac{1}{(k+1)^2} \int d^2m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) \cdot g_p\left(\boldsymbol{m}_1|\boldsymbol{y}_1, \frac{k+2}{k+1}\boldsymbol{m}_0 - \frac{y_2}{k+1}, \Sigma_{y1}, \frac{\Sigma_{y2}}{(k+1)^2}\right)
\end{aligned}
$$

$$
\begin{aligned}
& B(LOBE, LOBE) \qquad (5.68)\\
= \; & a^2 \int d^2m \cdot p(\boldsymbol{m}_0) \cdot g\left(\boldsymbol{m}_0|\boldsymbol{x}_0, \Sigma_{x0}\right)\\
& \int dk \cdot p(k)\\
& \frac{1}{(k+1)^2} \int d^2m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) \cdot g_p\left(\boldsymbol{m}_1|\boldsymbol{y}_1, \frac{k+2}{k+1}\boldsymbol{m}_0 - \frac{y_2}{k+1}, \Sigma_{y1}, \frac{\Sigma_{y2}}{(k+1)^2}\right)
\end{aligned}
$$

$$
\begin{aligned}
& B(CORE, LOBE) \qquad (5.69)\\
= \; & a^2 \int d^2m \cdot p(\boldsymbol{m}_0) \cdot g_p\left(\boldsymbol{m}_0|\boldsymbol{x}_0, \boldsymbol{y}_0, \Sigma_{x0}, \Sigma_{y0}\right)\\
& \int dk \cdot p(k)\\
& \int d^2m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) \cdot g\left(\boldsymbol{m}_1|\boldsymbol{y}_1, \Sigma_{y1}\right)
\end{aligned}
$$

$$
\begin{aligned}
& B(LOBE) \qquad (5.70)\\
= \; & a^3 \int d^2m \cdot p(\boldsymbol{m}_0) \cdot g\left(\boldsymbol{m}_0|\boldsymbol{x}_0, \Sigma_{x0}\right)\\
& \int dk \cdot p(k)\\
& \int d^2m_1 \cdot p(\boldsymbol{m}_1|\boldsymbol{m}_0) \cdot g\left(\boldsymbol{m}_1|\boldsymbol{y}_1, \Sigma_{y1}\right)
\end{aligned}
$$

$$
B(CORE) = a \cdot C_{x0y0} \qquad (5.71)
$$

## 5.6　程序实现

经过了理论准备及相关公式推导，至此已可以对此算法进行具体实现。实际需要分三步进行。第一步使用以前的方法（如前述的Zones Algorithm）对光学星表和射电星表进行初步的交叉证认，需要设置一个较大的匹配边界，找出在一个光学源周围一定范围内的多个射电源。此做法的目的是期望能在初步的结果集中包含所有与该光学源有关的射电源，尤其是喷流瓣。第二步是对与一个光学源有关的射电源进行组合，并使用本章中的方法求出各个组合的对全NONE假设的贝叶斯因子。第三步便是综合所有的似然函数数值、射电源的组合情形等信息，求出对一个光学源或射电源的最可能的假设。其中第一步的方法在前面几章已有说明，不再赘述。本节重点描述使用计算程序求解各个似然函数的值的一些重要事项。

依据本章前面几节的内容，具体实现方面遇到的问题主要有三个。一是如何对各个射电源进行组合；二是如何取得一、二维正态分布随机数；最后一个是积分求贝叶斯因子值。

### 5.6.1　组合的算法

经过第一遍的星表交叉证认之后，可得到如下结构的信息。其中，一个光

```
Hyp{
    Integer Core;
    List<Integer> Lobes;
    int GetCount() { return Lobes.Count+Core==-1?0:1;} }
}
```

图 5.6: Hyp数据结构用于保存一个假设里面的CORE、LOBE成员在射电源数组Radios 中的位置。

学源$oid$ 可对应1个至数个（$>3$）射电源$rid$。

$$\left\{\begin{array}{l} \text{光学源信息}\left\{\begin{array}{l} oid, \text{光学源编号} \\ ora, \text{光学源赤经} \\ odec, \text{光学源赤纬} \end{array}\right. \\ \text{射电源信息}\left\{\begin{array}{l} rid, \text{射电源编号} \\ rra, \text{射电源赤经} \\ rdec, \text{射电源赤纬} \end{array}\right. \end{array}\right.$$

需要按所有可能的方式将这些射电源设为CORE、LOBE 或NONE成分，分别对不同组合计算其似然函数。主要的限制为，每种组合不能有超过一个的CORE，不能有超过两个LOBE。遵守这一限制的情况下，所有的源都可以是CORE或LOBE或NONE。

首先，所有的与一个光学源相关的射电被保存到一个数组Radios 中，然后设计一个用于保存一个组合的数据结构Hyp。如图5.6，它包含了一个$Core$指向CORE成员在Radios中的位置、一个$Lobes$ 列表保存各个LOBE 成员在Radios中的位置。如前所述，通过对Hyp中CORE、LOBE的非空成员计数，可以得到$a$的幂次。 准备好Radios射电源数组及Hyp数据结构之后，可以通过如图5.7算法取得所有可能的假设。算法思想是，先确定CORE，再从剩下的射电源中选择不重复的(LOBE, LOBE)组合。最后可以考虑加入单个LOBE及全NONE的情形。CORE和LOBE均可以为空，用-1表示。

### 5.6.2 随机数生成

一维正态分布随机数的生成算法直接使用了《Numerical Recipes》[2][80]一书中的Box-Mulles算法[81]，其C# 实现方式如图5.8所示。 为了获得较好的正

[2]Numerical Recipes http://www.nr.com/

态分布随机数，在积分计算的全程只使用一个标准一维正态分布随机数生成器ran。而二维正态分布随机数直接来源于一维的正态分布随机数，设平均值向量$\boldsymbol{\mu}=\begin{pmatrix} x \\ y \end{pmatrix}$、协方差矩阵$\Sigma=\begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}$、已经生成的两个一维标准正态分布随机数$\begin{pmatrix} x' \\ y' \end{pmatrix}$，其算法[82]为

$$x_{new} = x+\sigma_x x' \tag{5.72}$$

$$y_{new} = y+\rho\sigma_y x'+\sigma_y\sqrt{1-\rho^2}y' \tag{5.73}$$

$$= y+(cov_{xy}x'+\sqrt{\sigma_x^2\sigma_y^2-cov_{xy}^2})/\sigma_x \tag{5.74}$$

如图5.9所示为C#版本的二维高斯随机数生成程序。通过指定一维随机数生成器及正态分布定义中的平均值向量M、协方差矩阵Σ，可产生出一个类型为Vector2ext的数据，其中包含了两个随机数。其所生成的(CORE, LOBE, LOBE)型的随机坐标点分布可从图5.10中看到，生成效果与预期符合。

### 5.6.3　积分的计算

解决了穷举组合各类假设问题，并得到了一、二维正态分布随机数生成器之后。重点所要实现的便是各个假设的似然函数的计算，也即各种积分的计算。首先需要将积分计算中的所有射电源的赤道坐标投影到光学源处的切面上。投影计算的程序如图5.11，其中cra及cdec表示光学源的赤经赤纬，单位为弧度，因为C#库中的sin等三角函数使用的单位是弧度。tra及tdec 则是一个射电源的赤经赤纬，单位也是弧度。相关射电源的坐标都需要做转换，转换结束之后。得到的坐标集合为以光学源为原点，即$(0,0)$，射电源坐标为围绕$(0,0)$的一系列二维实数坐标。

构成每个二维正态分布的各向量及矩阵运算较为复杂，可以使用一个现成的矩阵操作类库Mapack[3]，并使用一个Det或Gaussian 类来保存一个多维正态分布的所有成员。主要是记录平均值向量M、矩阵Σ的逆阵F等等。如图5.12所示积分计算程序即是本章所要实现的最核心的计算。它融合了前述的贝叶斯因子的最终算法，通过对各种假设组合的类型判断是否可以直接计算解析解，如(CORE)类型的假设；是否需要将两个二维正态分布函数合并，即带

[3]Mapack https://github.com/lutzroeder/Mapack

有CORE或LOBE, LOBE 的假设。同时通过公式5.40 计算积分过程的近似方差，并保存到err中，而积分结果保存到sum0 中。hyp.Prior中保存的即是前文中a（以平方角秒为单位的天球面积）的幂值，其数据最大可以达到$10^{31}$。方便起见，所有的结果都做了对数操作，即求$\log_{10}(sum0)$。因而，后续对各种假设的似然函数的对比就不再是相除，而是相减。积分计算的三重循环的循环次数可按需要进行设置，如第一层$m_0$ 有关的循环次数为40，第二层k有关的循环次数取30，第三层$m_1$有关的循环次数取80。则在最复杂的情况下，程序需要计算$40 \times 30 \times 80 = 96000$次才能完成积分并得到一个贝叶斯因子。

```
List<Hyp> hyps
List<int> cores
n=len(Radios)

for i=-1;i<n;i++
    cores.Add(i)

List<int[]> lobes

lobes.add(new int[]{-1,-1})
for(i=-1;i<n;i++)
    for(j=i+1;j<n;j++)
        lobes.Add(new int[]{j,i})

m=len(lobes)
for(i=0;i<=n;i++)
    for(j=0;j<m;j++)
        if cores[i]!=lobes[j][0] && cores[i]!=lobes[j][1]{
           Hyp hyp;
           hyp.Core=cores[i];
           if(lobes[j][0]!=-1) hyp.Lobes.Add(lobes[j][0]);
           if(lobes[j][1]!=-1) hyp.Lobes.Add(lobes[j][1]);
           hyps.add(hyp);
        }

for(i=0;i<=n;i++){
    Hyp hyp;
    hyp.Core=-1;
    if(core[i]!=-1) hyp.Lobes.Add(cores[i]);
    hyps.Add(hyp);
}
```

图 5.7: 穷尽Radios射电源数组中各种可能的假设，保存为一个Hyp序列。

```
public class Ran1
{
private long IA,IM,IQ,IR,iy,idum;
private long[] iv = new long[32];
    private double AM = 1.0/2147483647.0;
    private int NTAB = 32;
    private double NDIV = 1.0+(2147483647.0-1.0)/32.0;
    private double RNMX = 1.0-1.2e-7;
    private int iset = 0;private double gset;
    public Ran1(){
        IA=16807; IM=2147483647;IQ=127773;
        IR=2836; iy=0; idum=1;
    }
    public double NextDouble(){
        int j;  long k;   double temp;
        if (idum <= 0 || iy == 0) {
                if (-idum < 1) idum=1;
                else idum = -idum;
                for (j=NTAB+7;j>=0;j--){
                    k=idum/IQ;
                    idum=IA*(idum-k*IQ)-IR*k;
                    if (idum < 0) idum += IM;
                    if (j < NTAB) iv[j] = idum;
                }
                iy=iv[0];
        }
        k=idum/IQ;  idum=IA*(idum-k*IQ)-IR*k;
        if (idum < 0) idum += IM;
        j=Convert.ToInt32(iy/NDIV)%NTAB;
        iy=iv[j]; iv[j] = idum;
        if ((temp=AM*iy) > RNMX) return RNMX;
        else return temp;
    }
    public double NextGasdev(){
        double fac, rsq, v1, v2;
        if (iset == 0){
            do{
                v1 = 2.0 * this.NextDouble() - 1.0;
                v2 = 2.0 * this.NextDouble() - 1.0;
                rsq = v1 * v1 + v2 * v2;
            } while (rsq >= 1.0 || rsq == 0.0);
            fac = Math.Sqrt(-2.0 * Math.Log(rsq) / rsq);
            gset = v1 * fac;  iset = 1;
            return v2 * fac;
        }else{iset = 0;   return gset;}
    }
}
```

图 5.8: 使用C#实现《Numerical Recipes》中Box-Mulles一维正态分布随机数生成算法。

```
public class GaussianRandom2D{
    public Matrix M { get; private set; }
    public Matrix Sigma { get; private set; }
    public Ran1 RAN { get; private set; }
    public long Seed { get { return RAN.Idum; } }
    public GaussianRandom2D(Matrix mean, Matrix sigma, Ran1 ran) {
        M = mean;
        Sigma = sigma;
        RAN = ran;
    }
    public Vector2ext NextSample(){
        double rannor1 = RAN.NextGasdev();
        double rannor2 = RAN.NextGasdev();
        double sigma1 = Math.Sqrt(Sigma[0, 0]);
        double y1 = M[0, 0] + sigma1 * rannor1;
        double y2 = M[1, 0] + (Sigma[1, 0] * rannor1 +
            Math.Sqrt(Sigma[0, 0] * Sigma[1, 1]
            - Sigma[1, 0] * Sigma[1, 0]) * rannor2) / sigma1;
        return new Vector2ext(y1, y2);
    }
}
```

图 5.9: 基于一维正态分布随机数生成的二维正态分布随机数。为了保证有较好的正态分布随机数，一次积分计算中只使用一个一维正态分布随机数生成器。

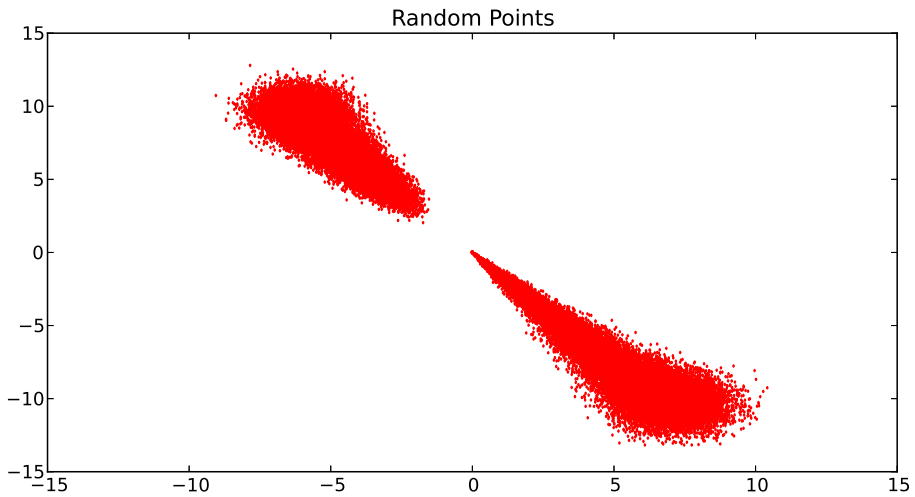


图 5.10: 通过程序所生成的(CORE, LOBE, LOBE)型的随机坐标分布示例。

```
double cra, cdec;//optical object's ra and dec, in radians
double sinRA = Math.Sin(cra);
double cosRA = Math.Cos(cra);
double sinDec = Math.Sin(cdec);
double cosDec = Math.Cos(cdec);

double tra, tdec;//radio object's ra and dec, in radians
double sinRAp = Math.Sin(tra);
double cosRAp = Math.Cos(tra);
double sinDecp = Math.Sin(tdec);
double cosDecp = Math.Cos(tdec);

double north = Constant.Radian2Arcsec
    * (
        -sinDec * cosRA * (cosDecp * cosRAp - cosDec * cosRA)
        - sinDec * sinRA * (cosDecp * sinRAp - cosDec * sinRA)
        + cosDec * (sinDecp - sinDec)
        );
double west = Constant.Radian2Arcsec
    * (
        sinRA * (cosDecp * cosRAp - cosDec * cosRA)
        - cosRA * (cosDecp * sinRAp - cosDec * sinRA)
        );
```

图 5.11: 将射电源坐标投影到以光学源坐标为原点的球面切平面上，(cra,cdec)为光学源赤道坐标，(tra,tdec) 为射电源赤道坐标。

```
double sum0 = 0; double sum0s = 0;
if(0 == hyp.Lobes.Count){
double sum0 = hyp.Core.HasValue ? Optical.Multiply(Radios[hyp.Core]).S * hyp.Prior : 1;
return new Vector2ext(Math.Log10(sum0), 0);   }
SymMatrix2 Sigma2 = hyp.Lobes.Count == 2 ? new SymMatrix2(0.2, 0.2, 0) : null;
Gaussian first = null;
if (hyp.Core!=-1){first = Optical.Multiply(Radio[hyp.Core]);
}else first = Optical;
Matrix Sigma0 = first.F.Inverse; Gaussian second = null; Matrix Sigma1 = null;
if (1 == hyp.Lobes.Count){
    second = Assoc.Radio[hyp.Lobes[0]];   Sigma1 = second.F.Inverse;
}Ran1 random = new Ran1();
GaussianRandom2D rand0 = new GaussianRandom2D(first.M, Sigma0, random);
for (int i = 0; i < SampleCounts[0]; i++){
    Vector2ext m0 = rand0.NextSample();  double sum1 = 0;   double int1 = 1;
    if (hyp.Lobes.Count == 1){
        GaussianRandom2D rand1 = new GaussianRandom2D(second.M, Sigma1, random);
        for (int j = 0; j < SampleCounts[1]; j++){
            Vector2ext m1 = rand1.NextSample();
            double r2 = m0.Substract(m1).ModSquare();
            sum1 += prior(r2);
        }   int1 = sum1 / SampleCounts[1];
    }else if (hyp.Lobes.Count == 2){
        for (int c = 0; c < SampleCounts[1]; c++){
            double k = random.NextGasdev(0, KSigma);
            if (k < -1) continue;     double sum2 = 0;    double int2 = 0;
            Vector2ext m0_2k = m0.Multiply(2 + k);
            double k1 = k + 1;    Det l2 = Assoc.Radio[hyp.Lobes[1]];
            second = new Det((m0_2k.X - l2.X) / k1, (m0_2k.Y - l2.Y) / k1, l2.Sigma / k1);
            second = second.Multiply(Assoc.Radio[hyp.Lobes[0]], false);
            Sigma1 = second.F.Inverse;
            GaussianRandom2D rand1 = new GaussianRandom2D(second.M, Sigma1, random);
            for (int j = 0; j < SampleCounts[1]; j++)   {
                Vector2ext m1 = rand1.NextSample();
                double r2 = m0.Substract(m1).ModSquare();
                sum2 += prior(r2);//Math.Log(first2.Evaluate(m1));
            }   int2 = sum2 * second.S / k1 / k1 / SampleCounts[1];  sum1 += int2;
        }
        int1 = sum1 / SampleCounts[1];
    }   sum0 += int1; sum0s += int1 * int1;
}//end x0,y0
sum0 = sum0 * first.S * hyp.Prior / SampleCounts[0];
sum0s = hyp.Prior * hyp.Prior * first.S * first.S * sum0s / SampleCounts[0];
sum0s = Math.Sqrt((Math.Abs(sum0s - sum0 * sum0)) / SampleCounts[0]);
double err = sum0s / sum0 / Math.Log(10);
Vector2ext result = new Vector2ext(Math.Log10(sum0), err);
```

图 5.12: 此程序融合了前述各种假设与全NONE的贝叶斯因子的计算方法，积分保存在sum0 中，误差保存在err中。由于数值较大，所有结果都做了求对数操作，误差值也针对对数的情形做了调整。

## 5.7 试验及数据分析

为了对方法的有效性进行检验，本文使用了SWIRE 3[4]（Spitzer Wide-area InfraRed Extragalactic）[83]在CDF-S[5]（Chandra Deep Field South）区公开的星表与ATLAS[6]（Australia Telescope Large Area Survey）[84]射电星表进行交叉证认（两个星表在这一区域几乎完全重叠），并与ATLAS射电天文学家通过肉眼及M-test方法[85]（M即Magliocchetti）进行交叉证认的结果进行对比。其中ATLAS在该区域的星表约有700个射电源坐标，而SWIRE3 在此区域的星表中含有约11 万个源。取三层循环的次数分别为40、30、80，通常在CPU频率约为2.9GHz的计算机上单线程跑完此程序大约需要56分钟，可得到32万6 千余个贝叶斯因子，其中有效数据（即值不为$-Infinity$或$NaN$）约有5万2千条。有效数据个数随$m_1$ 的概率密度函数$p(\boldsymbol{m}_1|\boldsymbol{m}_0)$ 不同而有所不同。

取$p(\boldsymbol{m}_1|\boldsymbol{m}_0)$ 为均匀分布概率函数，且均值为22 时，我们得到了结构与数据如表5.1 类似的SWIRE3源周围射电源的各种组合方式对全NONE假设的贝叶斯因子的值。如前所述，贝叶斯因子`factor` 及误差估计`err`的数值较大，为方便起见，均对它们取对数。表中`swire`列表示的是SWIRE3源的名称，`Radios`列显示的是射电源的名称组合，其属性成分一一对应于`combi`列的信息。另外，表格的数据量较大，为方便分析，所有的数据都被导入到了Microsoft SQL Sever数据库中，并使用SQL、Python 等语言及工具进行辅助分析。

由于每个SWIRE3源可有数个相关射电源，而一个射电源可以是数个SWIRE3 源的相关射电源，这导致数据的分析比较复杂。从ATLAS射电天文学家的手工交叉证认结果中可以得到以下信息，如表5.2：(CORE, LOBE, LOBE)是极少出现的情形，(LOBE, LOBE) 型相对稍多，(CORE, LOBE) 甚至比(CORE, LOBE, LOBE) 更少，而(CORE)最多，这也与平日的经验一致。需要说明的是，ATLAS 射电天文学家给我们提供的星表与他们论文[84]中的星表不同，他们使用了后期稍大一点的星表——没有提供给我们。因而需要把那部分的源去掉，导致了统计结果与他们论文稍有不同。此外，(LOBE)型不在ATLAS射电天文学家的统计范围内，并且容易干扰(CORE)型的分析，在数据分析初始阶段便被排除了。而表中的`double`是(CORE, LOBE)加(LOBE, LOBE)，在ATLAS 的论文中，称这两种类型为`double`，也即一个SWIRE3 源

[4]SWIRE http://www.spitzer.caltech.edu/
[5]SWIRE天区覆盖信息http://irsa.ipac.caltech.edu/data/SPITZER/SWIRE/
[6]ATLAS http://www.atnf.csiro.au/research/deep/index.html

表 5.1: 贝叶斯因子计算结果示例

| swire | radios | combi | factor | err |
|---|---|---|---|---|
| SWIRE3_J033323.75-281328.1 | C573 | Core | 3.31 | 0 |
| SWIRE3_J033323.75-281328.1 | C573,C570 | Core,Lobe | 11.25 | 1.58E-09 |
| SWIRE3_J033323.75-281328.1 | C574,C570 | Lobe,Lobe | 8.5 | 0.33 |
| SWIRE3_J033323.75-281328.1 | C573,C574 | Core,Lobe | 11.25 | 1.58E-09 |
| SWIRE3_J033323.75-281328.1 | C573,C574,C570 | Core,Lobe,Lobe | 12.81 | 0.3 |
| SWIRE3_J033323.78-272407.0 | C572 | Core | 11.09 | 0 |
| SWIRE3_J033324.04-281323.2 | C570,C574 | Lobe,Lobe | 16.39 | 0.2 |
| SWIRE3_J033324.04-281323.2 | C573 | Core | 7.8 | 0 |
| SWIRE3_J033324.04-281323.2 | C573,C574 | Core,Lobe | 15.75 | 1.65E-09 |
| SWIRE3_J033324.04-281323.2 | C573,C570 | Core,Lobe | 15.75 | 1.6E-09 |
| SWIRE3_J033324.04-281323.2 | C573,C570,C574 | Core,Lobe,Lobe | 24.21 | 0.18 |

对应有两个射电源的情形。在一些描述中，他们还使用了`Core-jet`的描述，可与(CORE, LOBE)相对应，即一个中心源加一个喷流。而(CORE, LOBE, LOBE)在ATLAS射电天文学家的描述中称为`triple`。

假设计算保存在一个`#table`的表中，我们采用了如下步骤来对其进行分析、处理

- 人为地给较好的(CORE)设定一个标准，比如设为10，这意味着这个射电源有$\frac{10}{10+1} \approx 90.9\%$ 的可能性是一个`CORE`。作为`CORE`的射电源不可以再作为任何源的`LOBE`，即从`#table`中删除所有使用这样的射电源为`LOBE`的假设。整个过程如图5.13 所示。这一点也可以在积分计算阶段加以考虑，先只做(CORE)型的计算，对于超过10的射电源不再让它作为`LOBE`。这样将可以减少相当一部分计算量，尤其是带有`LOBE` 的积分计算本来就是整个计算中消耗最大的部分。

- 由于(CORE, LOBE, LOBE)有效数据结果较少，从它开始比较快一些。需查找所有SWIRE3源最大的(CORE, LOBE, LOBE)型假设的贝叶斯因子，如果这假设里面涉及的射电源的最大的(CORE, LOBE,

表 5.2: ATLAS射电天文学家手工交叉证认结果的统计信息

| 类型 | 数量 | 百分比 |
|---|---|---|
| CORE,LOBE,LOBE | 10 | 1.647% |
| LOBE,LOBE | 22 | 3.624% |
| CORE,LOBE | 5 | 0.8237% |
| double | 27 | 4.448% |
| CORE | 559 | 92.092% |
| complex | 11 | 1.812% |

LOBE)型贝叶斯因子也是在此假设中，则它们被选为triple型结果。然后从#table删除所有带有triple结果中所涉及的SWIRE3源与射电源。整个过程如图5.14所示。

- 查找所有SWIRE3源各自最大的(CORE, LOBE)或(LOBE, LOBE) 型假设的贝叶斯因子，如果这些假设里面的射电源的最大的(CORE, LOBE)或(LOBE, LOBE) 贝叶斯因子也是在此假设中，则它们被选为double型结果。为了不影响后续的分析，需要从#table中删除所有涉及double型结果中的SWIRE3、射电源的结果。整个过程如图5.15所示。

- 在#table结果集中剩下的结果里面寻找每个SWIRE3源最大的(CORE)型假设，如果它对应的射电源的最大的(CORE) 型贝叶斯因子也在此假设中，则它们被选为core型结果。整个过程如图5.16 所示。

经过这样的计算，对于本节开始时设定的程序参数，即选用的三重循环计算次数分别为40、30和80，$\boldsymbol{m}_1$ 的概率密度函数使用平均值为22的均匀分布概率密度函数。对照ATLAS射电天文学家的交叉证认结果，得到表5.3 中的对照结果。可以看到，几乎找到了所有的triple 型结果，double 型的错误率较高，而core 型准确率也非常不错。

通过对比数据分析错误原因，triple 遗漏掉的一个是因为两个LOBE所形成的角度偏大，导致在直线模型中的计算结果较差，甚至比(CORE, LOBE)型要低，因而被程序选为double。如图5.17的上方两图为被程序遗漏掉的triple组合的实际观测图像，下方图为三个射电源成员（C030、C034、

```
WITH core(radios, maxcore)
    AS (SELECT radios, MAX(factor)
        FROM #table
        WHERE combi='core' AND factor>10 GROUP BY radios)
DELETE FROM #table
WHERE EXISTS (
    SELECT TOP 1 *
    FROM dbo.gSplitRadioComponent(radios, combi) g
        JOIN core c ON g.cid=c.radios AND g.component='lobe'
    )
```

图 5.13: 如果有射电源的(CORE)型贝叶斯因子大于10，则它不能作为LOBE存在，需从结果集中把那些将它作为LOBE的结果删除。图中gSplitRadioComponent函数用于取得射电源所对应的成分，以判断它是作为CORE还是LOBE存在。

表 5.3: 计算结果与ATLAS射电天文学家手工交叉证认结果对比

| 类型 | 计算结果 | 匹配数 | 遗漏数 |
|---|---|---|---|
| triple | 14 | 9 | 1 |
| double | 53 | 17 | 10 |
| core | 550 | 520 | 39 |

C038）在光学源的切平面上的投影。而错误判断的几个结果，则主要是因为其中的射电源实际对应有较好的SWIRE3 源，但是该源未在我们试验用的SWIRE3 星表中出现，而在ATLAS 射电天文学家的结果中他们看到了这些源。

对于double型结果，遗漏掉的一些结果，主要原因是程序发现了两个射电源对应了更好的一个SWIRE3源，而这一SWIRE3源与ATLAS 不同。但是实际上，程序选出的这个SWIRE3源较弱，或SWIRE3源在图像上的光斑很大，给出的坐标并不是真正的中心导致了计算偏差。在后续的应用中，可以加上对亮度等信息的贝叶斯因子以加强其结果，使得正确结果可以显现出来。对于剩下的不正确的计算出来的double结果，有可能是潜在的double型而未被发现。也可能是因为模型本身的原因，凡是在搜索范围的直线上的两射电源都会被认为是double 型的，这同样需要添加更多信息加以矫正。

对于core型结果，实际上可以再将一些结果去除，因为有些贝叶斯因子仅有2.34989839753225甚至更低，这不能说明它们是很好地匹配了SWIRE3源。但是在结果也能看到有11.0953478594329如此高贝叶斯因子的配对未被ATLAS的射电天文学家发现，而低如3.11781848072623的结果却能在他们的结果中出现。有可能是他们遗漏了，也可能说明了程序潜在的问题。需要进一步和他们进行交流。

实际上，我们不止使用了平均值为22的均匀分布概率密度函数，我们还使用了瑞利分布、对数正态分布概率密度函数，并使用不同的平均值。并对比他们的结果，得到了一些有意思的结果。如图5.18~5.20，每个图中的三条曲线分别表示只被程序发现的结果（program菱形）、只被ATLAS 人工发现的结果（eye圆形）、程序与人工方式均发现的结果（intersection星形）。

可以看到，不管是哪种分布都存在一个临界的平均值，即各图下方两条曲线（program及eye）的交叉点，当在那个平均值附近取值进行计算时，可以得到非常好的匹配结果。而平均值远离该临界值时，匹配结果开始逐渐变差。三种分布函数的情况大致相同，这也说明了$\boldsymbol{m}_1$的概率密度函数类型的取舍对计算结果影响并不太大。但是平均值会对计算结果有较大影响，如果能计算出该临界值在何处，将可大大提高计算准确率。

对应于三种概率密度函数的曲线，对数正态分布概率密度函数的曲线下降最快，其临界值最小；而瑞利分布概率密度函数曲线相对较缓，临界值比对数正态分布要大一些；均匀分布的概率密度函数曲线是分段曲线，其临界值最大。但前二者的计算范围包含了所有相关的射电源，而均匀分布不计算均值×2范围之外的射电源。因而，临界平均值所体现的信息，可能隐含说明了准确匹配的射电源组合们大多处在均匀分布概率密度函数的临界平均值内。ATLAS星表中的射电源与SWIRE3源的距离大约都在该临界值附近。但是，这只是ATLAS独有的情况呢，还是一种较为普遍的现象？需要更进一步的研究、分析。

```
WITH tabletriple(id,swire,radios, combi, factor, err)
    AS (SELECT * FROM #table WHERE CHARINDEX(',', combi, 7)>0),
    swiremax(swire, maxfactor)
    AS(SELECT swire, MAX(factor) FROM tabletriple GROUP BY swire),
    radiomax(cid, maxfactor)
    AS(SELECT g.cid, MAX(t.factor) FROM tabletriple t
    CROSS APPLY dbo.gSplitRadioComponent(t.radios, t.combi) g GROUP BY g.cid)
SELECT DISTINCT tmp.*
    , gs.radio1, gs.component1, rm1.maxfactor corefactor
    , gs.radio2, gs.component2, rm2.maxfactor lobefactor1
    , gs.radio3, gs.component3, rm3.maxfactor lobefactor2
INTO #triple
FROM tabletriple tmp
    CROSS APPLY dbo.gSplitRadioComponentTriple(tmp.radios, tmp.combi) gs
    JOIN swiremax sm ON sm.swire=tmp.swire AND tmp.factor=sm.maxfactor
    JOIN radiomax rm1 ON gs.radio1=rm1.cid AND tmp.factor=rm1.maxfactor
    JOIN radiomax rm2 ON gs.radio2=rm2.cid AND tmp.factor=rm2.maxfactor
    JOIN radiomax rm3 ON gs.radio3=rm3.cid AND tmp.factor=rm3.maxfactor
ORDER BY tmp.swire;

-- remove the compoenents of #triple from #table
WITH radiosintriple(id) AS (SELECT r.id
    FROM radio r JOIN #triple tr ON CHARINDEX(r.id, tr.radios)>0)
DELETE FROM #table
WHERE swire IN (SELECT swire FROM #triple)
   OR EXISTS (SELECT TOP 1 *
       FROM dbo.gSplitRadioComponent(radios, combi) g
           JOIN radiosintriple r ON r.ID=g.cid);
```

图 5.14: 查找每个SWIRE最好的(CORE, LOBE, LOBE)型假设的贝叶斯因子，如果所包含的所有射电源的最好的(CORE, LOBE, LOBE) 贝叶斯因子也在此假设中，则此假设被选为`triple`型结果。为了不影响后面的结果，需要把`triple`型结果所涉及的SWIRE3、射电源从结果集`#table`中去除。

```
WITH tabledouble(id,swire,radios, combi, factor, err)
    AS (select * from #table
    WHERE CHARINDEX(',', combi, 7)=0 AND CHARINDEX(',', combi)>0),
    swiremax(swire, maxfactor)
    AS(select swire, max(factor) FROM tabledouble GROUP BY swire),
    radiomax(cid, maxfactor)
    AS(SELECT g.cid, max(t.factor) FROM tabledouble t
    CROSS APPLY dbo.gSplitRadioComponent(t.radios, t.combi) g GROUP BY g.cid)
SELECT tmp.*, gs.*
INTO #double
FROM tabledouble tmp
    CROSS APPLY dbo.gSplitRadioComponentDouble(tmp.radios, tmp.combi) gs
    JOIN swiremax sm ON sm.swire=tmp.swire AND tmp.factor=sm.maxfactor
    JOIN radiomax rm1 ON rm1.cid=gs.radio1 AND tmp.factor=rm1.maxfactor
    JOIN radiomax rm2 ON rm2.cid=gs.radio2 AND tmp.factor=rm2.maxfactor;

WITH radiosindouble(id)
    AS (SELECT r.id    FROM radio r
    JOIN #double tr ON CHARINDEX(r.id, tr.radios)>0)
DELETE FROM #table
WHERE swire IN (SELECT swire FROM #double)
   OR EXISTS (SELECT TOP 1 *
   FROM dbo.gSplitRadioComponent(radios, combi) g
   JOIN radiosindouble r ON r.ID=g.cid);
```

图 5.15: 查找每个SWIRE最好的(CORE, LOBE)或(LOBE, LOBE)型假设的贝叶斯因子，如果所包含的所有射电源的最好的(CORE, LOBE) 或(LOBE, LOBE)贝叶斯因子也在此假设中，则此假设被选为double型结果。为了不影响后面的结果，需要把double 型结果所涉及的SWIRE3、射电源从结果集#table中去除。

```
WITH tablecore (id,swire,radios, combi, factor, err)
    AS (SELECT * FROM #table WHERE combi='core')
,     swiremax (swire, maxfactor)
    AS (SELECT swire, MAX(factor) FROM tablecore GROUP BY swire)
,     radiomax (cid, maxfactor)
    AS(SELECT g.cid, MAX(factor) FROM tablecore t
    CROSS APPLY dbo.gSplitRadioComponent(t.radios, t.combi) g GROUP BY g.cid)
SELECT t.* INTO #core FROM tablecore t
    JOIN swiremax sm ON t.factor=sm.maxfactor AND t.swire =sm.swire
    JOIN radiomax rm ON t.factor=rm.maxfactor AND t.radios=rm.cid
```

图 5.16: 从剩下的结果中寻找每个SWIRE3源最好的(CORE)型假设，如果对应的射电源的最好(CORE)型贝叶斯因子在此假设中，则它们被选为core型结果。

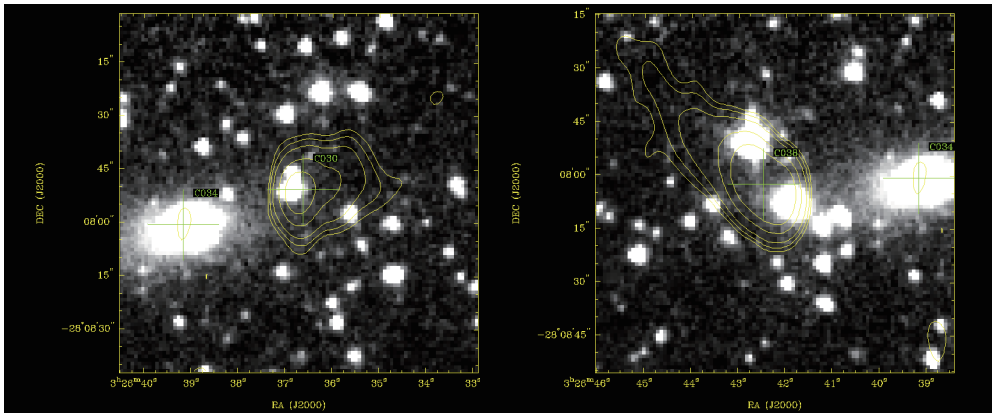



图 5.17: 被程序遗漏掉的一个triple 组合。中心射电源为C034，两侧瓣为C030与C038。上图为它们的实际观测图像。下图为三个射电源在C034附近一光学源的球面切平面上的投影情况，可以看到两个LOBE之间的夹角偏大，这导致了它们在直线模型中的概率值偏低。

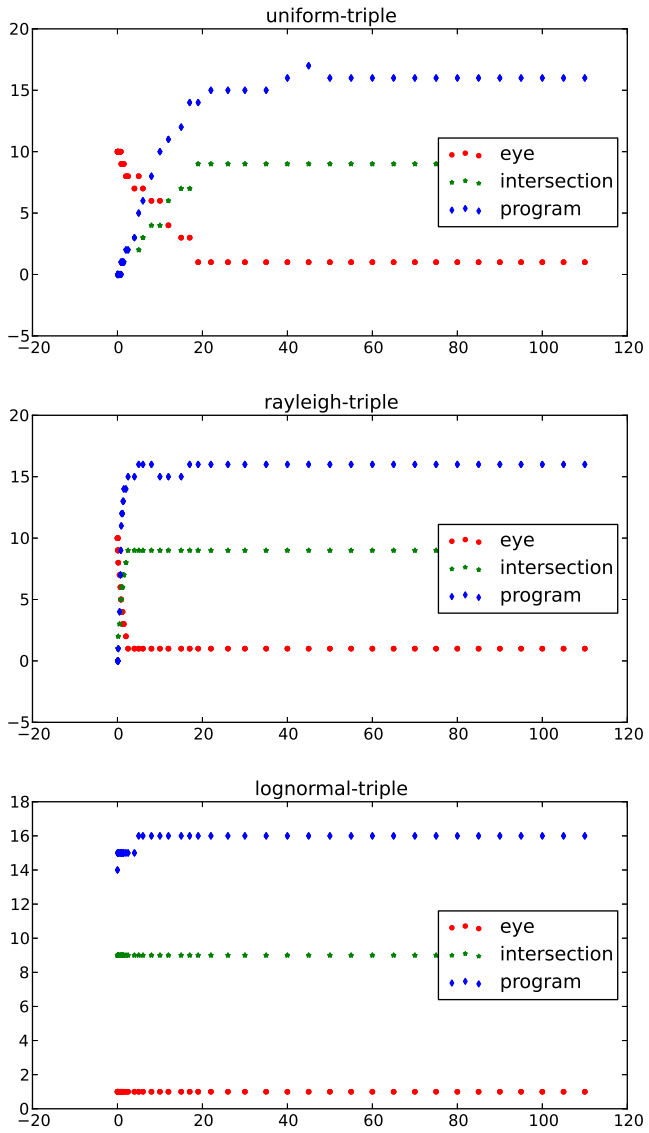


图 5.18: 三种概率密度函数在`triple`型结果中的表现，横轴表示不同的均值，纵轴表示结果数目。可以看到`triple`的匹配效果大至相同，都顺利得到了9个准确的匹配结果，且结果稳定。

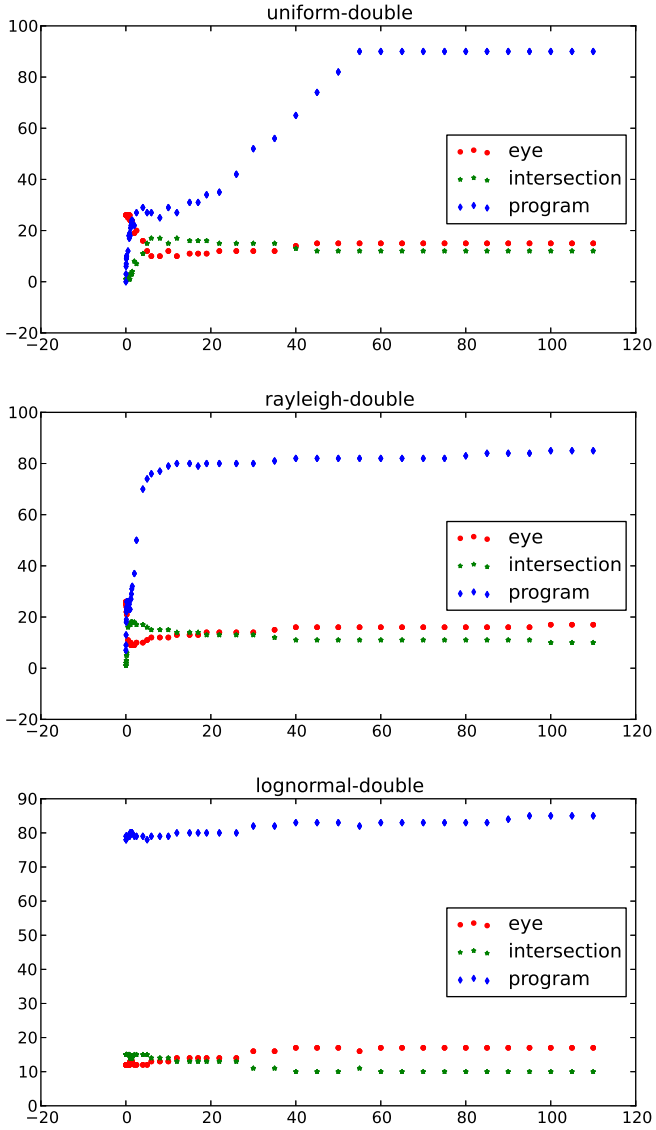


图 5.19: 三种概率密度函数在double型结果中的表现。总体表现都差强人意，对数正态分布相对表现稍好，表明此模型对double型适应性不佳，需要进一步改进。

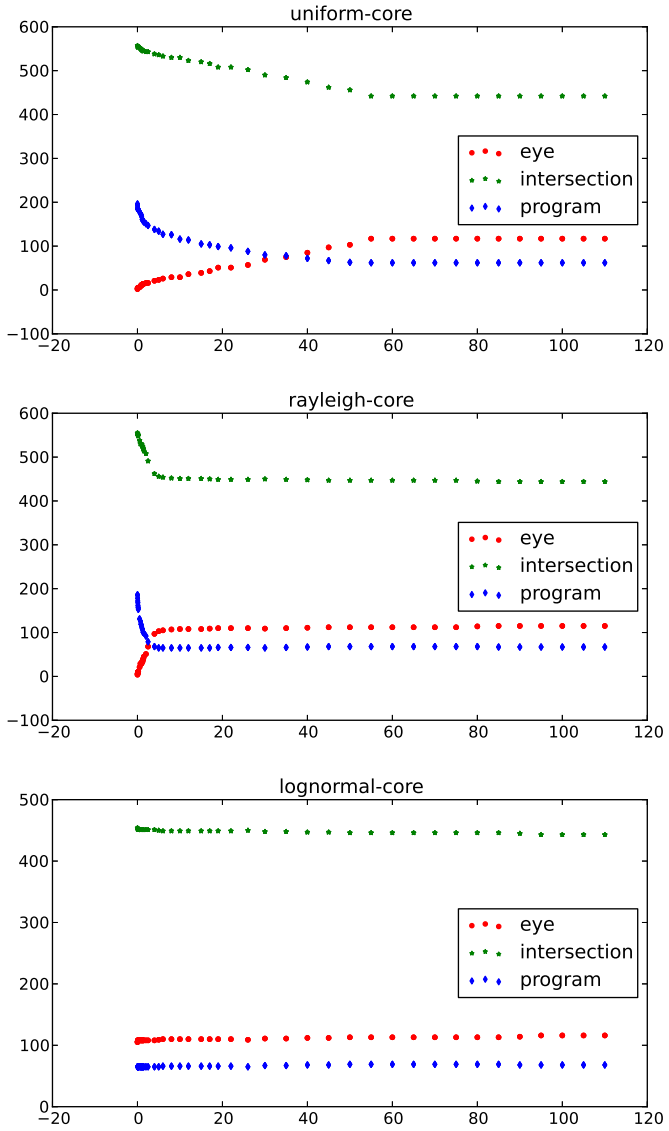


图 5.20: 三种概率密度函数在core型结果中的表现。三种概率密度函数表现相当，均匀分布的表现似乎相对好一些，平均值的临界值较大。

## 5.8　小结

本章研究了使用直线非对称模型对光学星表与可能带有喷流射电源的射电星表的进行交叉证认的一种基于贝叶斯假设推断方法。通过对射电源组合所构成的各种可能的假设进行计算，综合比较分析各个假设的贝叶斯因子数据，取得对于光学源或射电源在模型中概率较高的组合结果。此方法可应用到未来的大规模射电波段巡天（如SKA）观测结果与光学观测结果的数据融合中来。

由于此方法还存在一些问题，比如直线非对称模型不能适应喷流瓣夹角较大的情形。另外对于(CORE, LOBE)与(LOBE, LOBE)的表现也比较差强人意。后续需要继续改进模型，并针对更多的星表进行计算，以进一步验证此算法的有效性。程序方面，将会把相关的程序改写成SQL 函数或过程。利用数据库的多线程及数据调度能力，使得程序可以更有效率地运行。或者考虑使用分布式或并行大规模运算的策略，减少程序的计算时间。

# 第六章　资源统一管理平台

前面几章主要着力于研究如何联结多个星表数据的算法，即交叉证认技术。但任何算法都需要被具体应用才能体现出其价值，而并非所有的天文学家都擅长于某种编程语言。通常他们习惯于使用某几种语言或工具，他们的关注点在于研究本身而不是本末倒置于技术。这时候天文技术的研究者就要在这中间为天文学家搭起一座桥梁，使得天文学家可以更便捷地获得所需数据，而无须在如何获取数据上耗费大量宝贵时间。

这实际上也是虚拟天文台的使命之一。本章将描述的资源统一管理平台：天文资源管理器（Astronomical Resource Manager，ARM）——曾称FITS 文件管理器（FITS Manager，FM）[86][87]——作为中国虚拟台的项目之一，致力于帮助天文学家管理数据，获取数据、服务。此项目也是中国虚拟天文台计划（China-VO）[1]和印度虚拟天文台计划（VO-India[2]的合作研究项目，目前得到了自然科学基金天文联合基金（编号U1231108）的支持。

## 6.1　天文资源管理器的定位与技术选择

天文资源管理器的关注点在于如何方便天文学家获取数据、访问服务，而不是直接处理数据。直接处理数据的工作可以转交给天文学家最熟悉的工具，如IDL[3]、IRAF[4]、MIDAS[5]、Aladin[6][88]、fv[7]、SAOImage DS9[8] 等等。数据又分本地数据或远程、异地数据。天文界最常见的，也是ARM 最直接要面对的便是天文研究者计算机上众多的FITS 文件。

FITS是天文界的行业规范，应用领域窄。对通用软件如Picasa[9]、ACDSee[10]

[1]China-VO http://www.china-vo.org/
[2]VO-India）http://voi.iucaa.ernet.in/~voi/
[3]IDL http://www.exelisvis.com/
[4]IRAF http://iraf.noao.edu/
[5]MIDAS http://www.eso.org/sci/software/esomidas/
[6]Aladin http://aladin.u-strasbg.fr/
[7]fv http://heasarc.gsfc.nasa.gov/docs/software/ftools/fv/
[8]SAOImage DS9 http://hea-www.harvard.edu/RD/ds9/
[9]Picasa http://picasa.google.com/
[10]ACDSee http://www.acdsee.com/

等而言，对此专业格式进行支持的代价过高，又无商业利益。像许多很便利、实用的功能如FITS图像缩略图显示、FITS 头信息检索等，均无法从通用软件得到支持。而在专业软件中又只能手工一次一次操作，或者需要写脚本进行处理，过于繁琐。

综合这些问题，ARM首要解决的就是如何对本地FITS文件进行快速可视化及检索。一些较通用的文件管理功能，如收藏夹、文件注释、通过路径快速定位文件夹等，也必须包含其中，以提供与各操作系统桌面环境相接近的使用体验。还需要与本地应用软件相结合，形成互操作的环境。更高级的还可以通过提取FITS文件信息与远程网络服务联动，一站式获取更多数据。

天文界除了FITS外，还有众多其他数据格式。且即便只是FITS 格式也过于灵活，每个望远镜的FITS格式可能都不尽相同。因而需要给ARM足够的灵活性，以方便对不同数据格式进行定制。考虑到开发力量较小，而各功能模块需逐步按需求慢慢添加，为了适应这种开发方式，ARM计划采用插件化的开发方式。以插件的方式添加功能，也减少了各部分之间的相互依赖，降低维护成本。

多平台支持也是需要考虑的一个问题，天文圈内Windows、Linux 诸多发行版、Mac系统都有人在用。若使用C/C++之类的语言，则需要针对不同平台分别编程。C++或许可以使用Qt[11]进行跨平台开发，也可以支持插件化。但是，一些函数库缺少C/C++版本。而且，C/C++ 开发难度高，学习曲线偏长。Java在这一方面的优势就较为明显，语法简单，学习难度小。而且各种函数库比较全，天文界很多函数库都有相应Java版本。跨平台方面更是Java所长，Java迅速崛起为主流语言的原因之一便是它所号称的“一次编译，到处运行”。而且，Java 得到了IT业界老牌力量Oracle、IBM等的支持，尤其是IBM所推出的Eclipse 系列编译器，更成为了Java主流的跨多平台编译器。

Java的插件化程序开发方面，Eclipse本身便是基于插件平台Eclipse Rich Client Platform（Eclipse富客户端平台，Eclipse RCP）的杰作[89]，所有的功能都由插件组成。Eclipse RCP拥有自己独立的插件框架，而自Eclipse3 R3[90]版本开始，IBM 更将Eclpise插件框架建构在服务网关开放接口（Open Services Gateway initiative，OSGi）[12] 之上，使得整个架构更为灵活。IBM还为Eclipse开发了比Java标准图形库AWT/Swing 更为轻量级的SWT/JFace，在不同平台

[11]Qt http://qt.digia.com/

[12]OSGi http://www.osgi.org/Main/HomePage

上发布的程序界面与该平台上的风格相同，而不像AWT/Swing 一样千篇一律，且与各平台都格格不入。最重要的是SWT/JFace 的响应速度及可定制化程度较AWT/Swing更高。Java的一次编译到处的特性也得以保留。比较麻烦的一点就是基于Eclipse RCP的程序需要针对不同平台分别发布，因为SWT/JFace不是Java自带的标准图形库，在发布Eclipse RCP 时需要把SWT/JFace 相关的包带上，而各个包在各个平台是不同的。但这一点也被以插件化的方式给解决了，Eclipse社区提供了一个DeltaPack[13] 的插件，可以在发布时一次性生成所有需要支持的平台的发布包。这些工作都是一次性的，在功能升级时也只需要更新相关的插件即可，并不需要重新发布。

插件化带来的一个直接的好处，便是可以选择自己需要的功能插件来使用，去除无关插件，从而建立一个个性化的使用环境。Eclispe 本身就是通过不同的插件组合，形成了Eclipse IDE for Java EE Developers、Eclipse IDE for C/C++ Developers、Eclipse for RCP and RAP Developers、Eclipse for Testser等等适应不同人群需要的产品[14]。Eclipse RCP 可以通过在线或者link 方式来安装插件，自动发现新插件，并在启动时加载这些插件。相应的，Eclipse也提供了删除机制来去掉一些插件，如link 方式安装的插件直接删除dropins目录下的*.link 文件即可。甚至也可以手工删除掉一些文件以去除特定功能。需要注意的是在删除一个文件的时候，要小心是否有插件依赖于此文件，有时可能还需要清除一下缓存以避免启动错误。

Eclipse已经是一个非常成熟的产品，背后有IBM这样的大企业支持，并且有一个活跃的开发社区为它提供各式各样的功能插件，这些插件也都能支持基于Eclipse RCP构架的其他产品。Eclipse RCP提供了完备的帮助系统插件、多语言支持、鲁棒的插件更新及安装界面[91]，这些都大大减少了在开发工作上的人力消耗。基于以上便利性，我们选用了Java作为开发语言，并使用Eclipse RCP作为插件平台，对ARM进行开发。

## 6.2　天文资源管理器的结构

ARM鼓励天文社区中有编程经验者加入其中，以插件的方式增加各自需要的功能，利用社区的力量扩展软件的能力。使得ARM成为一个资源、服

[13]Eclipse 3.7.2版本的DeltaPack http://download.eclipse.org/eclipse/downloads/drops/R-3.7.2-201202080800/#DeltaPack

[14]Eclipse Downloads http://www.eclipse.org/downloads/

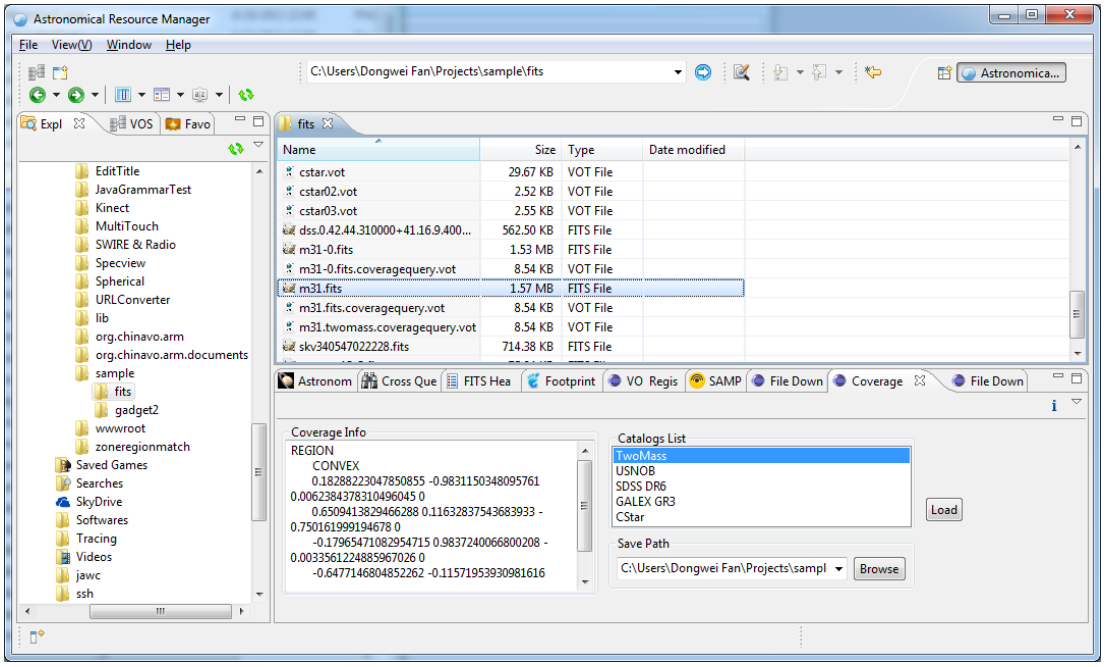


图 6.1: ARM当前的一个界面。

务平台，而不仅仅是一个软件。ARM当前版本的一个界面如图6.1所示。 整个ARM目前包含了20余个插件，主要由如下几大部分组成：

- 文件信息查看。如FITS头查看，文件列表、图标、缩略图等。
- 文件管理。如资源管理器，文件（夹）收藏夹，文件搜索，支持在FITS头内搜索等。
- 信息检索。这是最主要增强的一项内容，可在ARM内直接调用Sesame（包含Simbad、NED等服务）检索、ADS 检索、arXiv检索、SkyMouse检索、DataScope、天文名词检索等等。
- 内嵌专业工具。内嵌了ds9、fv、Aladin、Topcat、VOSpec 等专业天文软件。可随ARM 在多个平台发布，如Windows、Linux、Solaris等及其各自32位或64 位版本。
- 远程文件管理。主要是与虚拟天文台VOSpace 进行了联结，进一步的还将考虑与其他云存储的联结。

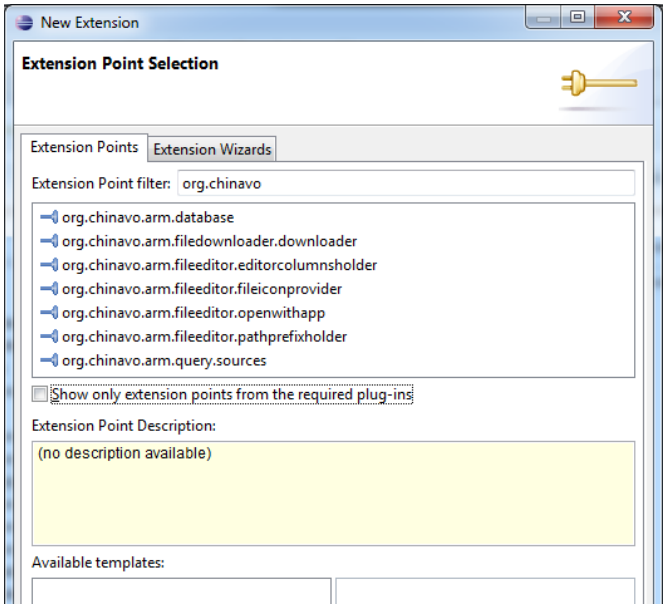


图 6.2: ARM当前所定义的扩展接口，以供更多开发者使用。

- 数据与服务整合。如通过分析FITS图像信息，将其天区覆盖信息上传到Footprint Service，或利用前述第四 章的技术在星表中查询并获取其覆盖区域内的天体。

- 辅助小工具。如天文文件批量下载工具、NASA “Astronomy Picture of the Day”展示及桌面壁纸设置等

在这些插件的实施过程中，ARM的一些插件定义了如图6.2中几个扩展接口，供其他功能插件使用，其他开发者也可以通过这些接口来实现自己的功能[92]。其中，ID 为“org.chinavo.arm.fileeditor”的文件列表插件（简称fileeditor）直接与文件相关，它提供了最多的接口，它也是整个ARM 的中心。如图6.3 所示，ARM 中所有的与文件有关的插件都将依赖于此插件。“editor”在其名称里指的是Eclipse RCP的一种控件模式，与视图“view”类似，但editor可以出现在RCP的“编辑区”内，并在同一个RCP 程序中有多个实例[93]，适用于在ARM中显示多个文件列表。在最新的Eclipse 4 Juno[15] 中，editor与view的界限已经开始变得非常模糊。

[15]Eclipse Juno http://www.eclipse.org/juno/

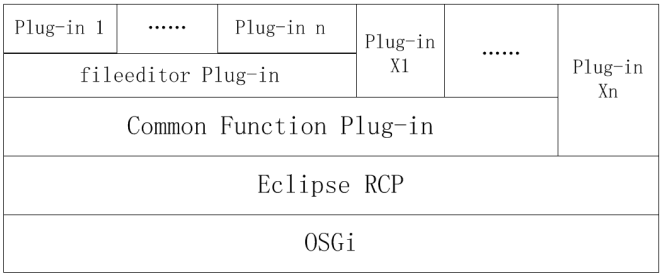


图 6.3: ARM系统结构及插件关系。

## 6.3 文件列表插件

天文资源管理器，顾名思义便是以天文资源为中心，围绕天文资源来综合各种服务。天文资源中最基本的即是各种天文文件。因而文件列表插件便作为了整个ARM 的中心。它负责使用各种方式来显示文件列表，并提供扩展接口为其他插件提供当前所选中的文件的信息。

文件列表插件最重要的一项任务，即是“告知”别的插件，当前有哪些被选中了、当前有哪些文件要被打开等等。这样，别的插件就可以针对选中的文件来完成自己的任务。ARM通过提供Eclipse RCP Selection Service[94] 来解决这一问题。这是Eclipse RCP 本身的一个机制，数据提供者可以注册服务，数据接收者需要监听系统Selection Service的变化。即别的插件可以通过监听系统Selection Service来获知文件列表中的变化。这样极大的降低了文件选择列表与其他插件的耦合度。文件列表只管说明选中了哪些文件，并给出了Java 本身定义的File数据对象；而别的插件也只需要接收这些通用的数据对象，而无须知道文件列表是如何实现的。只要遵守这一约定，其中任何一个插件本身就可以按自己的需要进行变化，而不会对别的插件造成影响。这也是软件工程中的“对接口编程”的思想。

文件列表插件本身可以提供多样化的内容显示，如表格状的文件详细信息显示方式，或者文件图标显示方式、图片缩略图显示方式。但是文件类型如此多样化，只在一个插件里面完成这些任务将遥遥无期。因而，文件列表插件定义了多个接口，由其它插件来为它提供文件属性、图标、缩略图等消息。这些接口的基本信息如下，

- 文件信息提供者接口，编号为“org.chinavo.arm.editorcolumnsholder”，对应这个接口的Java Interface是org.chinavo.arm. fileeditor. columns. ColumnInfoIntr。用于提供表格状文件列表中的一列信息，如文件名、文件最近修改日期、文件类型等等。文件列表还可以通过新添加的这一列信息来对文件进行排序。

- 文件图标提供者接口，编号为“org.chinavo.arm.fileeditor.fileiconprovider”，对应的Java Interface是org.chinavo.arm.fileeditor.fileicon.FileIconProvider。新插件可以注册自己所关心的文件格式如.fits、.txt等，文件列表在显示这些文件的时候，将会向扩展插件“索要”这些类型的文件的图标或缩略图。由于Java在处理图片方面并无优势，扩展插件可以把生成的图标缓存到内存或数据库中，以加快下一次的显示速度。ARM本身提供了一系列的公共函数库（即结构图中的Common Function Plug-in）来支持这些操作。

- 文件路径响应接口，编号为“org.chinavo.arm.fileeditor.pathprefixholder”，对应的Java Interface是org.chinavo.arm.path.PathPrefixListener。插件可以自定义自己的文件路径规则，如file://表示文件系统路径，fav://表示收藏夹路径。文件列表同时了一个文本框，可以在文本框输入文件夹的路径以快速进入所需文件夹。文件列表插件对输入的路径进行分解，把路径传递给不同协议的接收插件。通过这一机制，ARM不仅可以支持自定义路径协议，也可以支持ftp，ssh等已有协议。

- 文件打开接口，编号为“org.chinavo.arm.fileeditor.openwithapp”，对应的Java Interface为org.chinavo.arm.fileeditor.openwithapp.AppInterface。这是ARM 实现与本地程序整合的关键技术。插件可以注册特定的文件类型，如.fits，也可以指定为“*”以关联任意类型文件。当在文件列表中右击扩展名为.fits的文件时，将显示所有注册了.fits 类型的插件名称。选择其一，文件列表将把当前文件信息发送给该插件，插件接收到信息后就可以自己决定如何打开该.fits文件，如调用fv、ds9 等数据分析软件。fv、ds9 等软件实体可以被放到Eclipse RCP插件内，封装成jar包或仍以文件夹的形式存在。通过Eclipse RCP内部的定位机制，FileLocator 工具类[16]

[16]FileLocator工　具　类http://help.eclipse.org/indigo/index.jsp?topic=%2Forg.eclipse.platform.doc.isv%2Freference%2Fapi%2Forg%2Feclipse%2Fcore%2Fruntime%2FFileLocator.html

能够定位可执行文件的绝对路径，这样可以免于让使用者自己配置fv等软件。插件在调用这些程序的时候还可以带上特定的指令，来完成一些自动化的工作。另外一种与本地软件协同的机制即虚拟天文台的应用程序简单通信协议（SAMP），将在后面相关插件中进行说明，ARM 主要通过这两种方式来与本地软件协同工作。

当然了，文件列表插件还需要提供一个入口来让其它插件将文件装入列表中。这通过一个约定好了的数据对象org.chinavo.arm.fileeditor.input.InpuData来传递文件列表，同时还要求数据提供者实现一个接口org.chinavo.arm.fileeditor.input. InputDataProvider以响应文件列表对刷新文件列表的要求。维持系统稳定的首要任务是保持这些数据结构及接口不变，所有的通信通过接口进行。

## 6.4 覆盖区域查找插件

从一个FITS图像文件可以看到一个区域的图像，从一个星表也可以获取一个区域的天体列表。Aladin等软件可以将两类信息重叠起来对比这两种结果。但是，如何以可视化的方式快速获得一个图像上的天体列表呢？通过FITS中的世界坐标系（World Coordinate System，WCS）信息可以获取一个FITS图像四个顶点的坐标。通过顺时针或逆时针连接这几个坐标，可画出矩形的四条边，也就可以知道图像所覆盖的矩形区域的形状。进而可以在指定星表中查询在这个区域内的天体信息，再封装成Aladin可以识别的数据文件。Aladin将图像与星表进行重叠放置，即能对照图像上各部分所对应的天体。效果如图6.4所示，这便是覆盖区域查找插件（Coverage Query）插件的工作：在星表中查找一个FITS 图像上的天体。

Coverage Query插件完成这一功能的过程包含了三个工作：获得图像所覆盖区域的信息；在星表中查询这一区域中的天体，并保存为文件；在Aladin等可视化软件中显示图像及星表。第一步可以利用第二章中的Spherical Library完成。首先通过WCS 按顺时针顺序，即按[1,1]、[1,naxis1]、[naxis2,naxis1]、[naxis2,1] 的顺序，从FITS文件中获取到四个顶点的赤道坐标[E0, E1, E2, E3]，使用公式2.1~2.3 转换为单位球面上的三维笛卡尔单位向量[D0, D1, D2, D3]。两个交叉向量可以决定一个平面，也即Sperical Library中的半空间（Halfspace）的切面。相邻两个顶点的向量的叉积决定了切面的法向，共有四个法向[C0, C1, C2, C3]，须注意令该法向指向图像的中心。这样，四个半空间的交集所

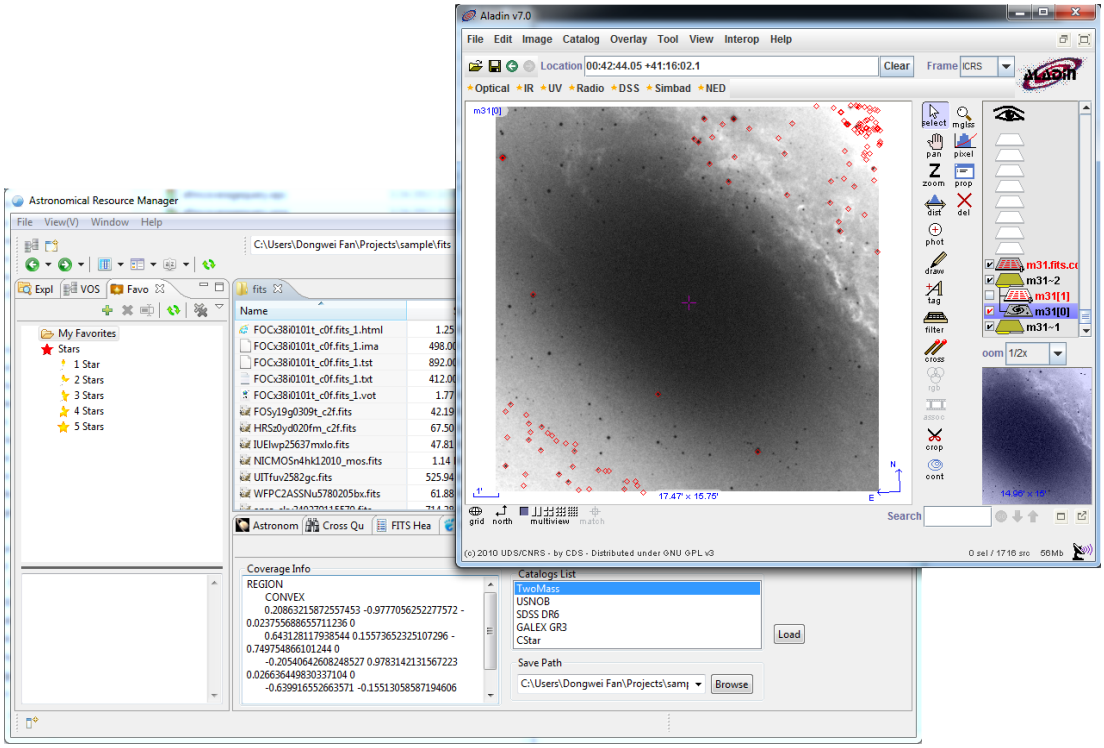


图 6.4: 覆盖区域查找插件通过分析FITS图像文件获得其覆盖区域，然后在TwoMASS星表中查找该区域内的天体列表并保存为VOTable文件。Aladin可以叠放这两个文件以对照图像和星表。

形成的凸面Convex就是图像的覆盖区域了，交集的外框就是图像的四条边。用Spherical Library的区域描述语言来描述这个凸面，如图6.5，就可将该矩形描绘出来。

这里存在一个问题，[D0, D1, D2, D3] 在天球上的实际排列可能是按顺时针方向，也可能是按逆时针方向。需要保证两个向量的叉积的方向指向图像中心，如果方向取反了，则后续取到的将是一个空集。如果是顺时针排列，则应按逆时针方向做叉积，即[C0, C1, C2, C3]=[D03, D32, D21, D10]；如逆时针排列，则应按顺时针方向做叉积，即[C0, C1, C2, C3]=[D01, D12, D23, D31]。

第二步，则需要将该区域描述语言文本提交到一个星表查询服务上，并将结果保存为VOTable。ARM实际上为了这一功能自行实现了一个简单的Web Service，利用第四 章中的技术，可以轻松在星表中查找到指定区域内的天

```
REGION
    CONVEX
    C0x C0y C0z 0
    C1x C1y C1z 0
    C2x C2y C2z 0
    C3x C3y C3z 0
```

图 6.5: 通过对FITS图像的四个顶点做叉积可以获得四个半空间的法向，四个半空间的交集即是FITS图像的覆盖区域。

体。该服务的核心是纯SQL语句，如图6.6，使用条带片段（Zones Intervals）来模拟该区域，然后查找处在这些片段中的天体。从前面几章可知，以条带编号（ZoneID）为聚集索引过滤数据更为迅速一些。这些SQL 语句可以直接在安装了Spherical Library 的数据库服务器上运行，而Web Service中的代码可以使用JDBC 或其他方式连接到数据库运行这段SQL语句即可获得结果。查询结果将被Web Service 保存为VOTable[17] [95]格式。VOTable标准[32]是国际虚拟天文台联盟IVOA定义的数据存放标准，已经成为联盟内数据传输的通用格式，Aladin、Topcat等软件均支持此格式。与FITS 相比，VOTable 的定义更为明确，其中所使用的单位、名称都由IVOA 语义工作组[18] 通过统一内容描述（Unified Content Descriptors，UCD）给予定义。一个VOTable的例子如图6.7 所示，这里的Web Service即以此文件为模板，以一个天体的数据为一行把数据封装到< TABLEDATA >标签中。也有专门为VOTable设计的工具库如STILTS[19][96]，TOPCAT等VOTable 处理软件就是通过此库完成的[97]，但在这里因为功能比较单一，直接使用文本模板比较快捷。

第三步，在Aladin中打开FITS图像文件及保存星表查询结果的VOTable文件。这两个文件可以通过ARM的SAMP插件来通知Aladin打开，也可以直接使用为Aladin 所扩展的“文件打开接口”插件来打开。这一步非常简单，后续的发展中可以考虑将这一步精简掉。

整个的三步流程看起来颇为复杂，但是对使用者而言，仅需要鼠标操作四次。第一步是点击一个FITS图像文件，Coverage Query插件即自动使用区域描述语言描述FITS图像所覆盖的区域；选择所需要的星表，如2MASS；点

[17]VOTable http://www.ivoa.net/Documents/VOTable/20040322/PR-VOTable-1.1-20040322.html
[18]IVOA Semantics Working Group http://wiki.ivoa.net/twiki/bin/view/IVOA/IvoaSemantics
[19]STILTS http://www.star.bristol.ac.uk/~mbt/stilts/

```
DECLARE @regionstring VARCHAR (1000),
    @region VARBINARY(max), @database VARCHAR(255), @execsql varchar(max)

SET @database = 'dbo.TwoMass'
SET @regionstring = 'REGION CONVEX
0.20863215872557453 -0.9777056252277572 -0.023755688655711236 0
0.643128117938544 0.15573652325107296 -0.749754866101244 0
-0.20540642608248527 0.9783142131567223 0.026636449830337104 0
-0.639916552663571 -0.15513058587194606 0.7526229513868136 0'
SET @region = sph.fSimplifyString(@regionstring)
SELECT dec-radius/60.0 decmin, dec+radius/60.0 decmax
INTO #patches
FROM sph.fGetPatches(@region);

SELECT z.ZoneId ZoneId, sph.fIntersect(@region, z.RegionBinary1) Intersect1
    , sph.fIntersect(@region, z.RegionBinary2) Intersect2
INTO #intersection
FROM dbo.ZoneDef z
    JOIN #patches p ON z.DecMin BETWEEN p.decmin AND p.decmax
    OR z.decmax BETWEEN p.decmin AND p.decmax;

CREATE TABLE #intervals(
    ZoneID INT not null,
    RaMin FLOAT not null,
    RaMax FLOAT not null,
    Alpha FLOAT,
    PRIMARY KEY(ZoneID, RaMin, RaMax)
);

INSERT #intervals(ZoneID, RaMin, RaMax)
SELECT r.ZoneId ZoneId,MIN(f.ra1) RaMin, MAX(f.ra1) RaMax
FROM #intersection r
    JOIN dbo.ZoneDef z ON r.ZoneId=z.ZoneId AND r.Intersect1 IS NOT NULL
    CROSS APPLY dbo.fGetOutlineExt(r.Intersect1, z.DecMin, z.DecMax, 1) f
GROUP BY r.ZoneId

INSERT #intervals(ZoneID, RaMin, RaMax)
SELECT r.ZoneId ZoneId,MIN(f.ra1) RaMin, MAX(f.ra1) RaMax
FROM #intersection r
    JOIN dbo.ZoneDef z ON r.ZoneId=z.ZoneId AND r.Intersect2 IS NOT NULL
    CROSS APPLY dbo.fGetOutlineExt(r.Intersect2, z.DecMin, z.DecMax, 1) f
GROUP BY r.ZoneId
set @execsql = 'SELECT c.* FROM #intervals i inner loop JOIN ' + @database
    + ' c ON i.zoneid=c.zoneid AND c.ra BETWEEN i.ramin AND i.ramax'

EXEC(@execsql)
```

图 6.6: 使用条带片段来模拟FITS图像的覆盖区域，然后在星表中寻找处于这些片段中的天体。

```
<?xml version="1.0"?>
<VOTABLE version="1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://vizier.u-strasbg.fr/xml/VOTable.xsd">
<DEFINITIONS> <COOSYS ID="myJ2000" equinox="2000." epoch="2000." system="eq_FK5" />
</DEFINITIONS>
<RESOURCE name="objects in specified region">
<TABLE name="results">
<DESCRIPTION>selected objects</DESCRIPTION>
<FIELD name="Name" ucd="ID_MAIN" datatype="char" arraysize="8*" />
<FIELD name="RA" ucd="POS_EQ_RA_MAIN" ref="J2000" datatype="float" width="9"
 precision="6" unit="deg"/>
<FIELD name="Dec" ucd="POS_EQ_DEC_MAIN" ref="J2000" datatype="float" width="6"
 precision="2" unit="deg"/>
<DATA><TABLEDATA>
 <TR><TD>5630702138442</TD><TD>10.728245</TD><TD>41.145565</TD></TR>
 <TR><TD>5630702138451</TD><TD>10.744733</TD><TD>41.148071</TD></TR>
 <TR><TD>5630702138467</TD><TD>10.775519</TD><TD>41.148735</TD></TR>
  </TABLEDATA></DATA>
 </TABLE>
</RESOURCE> </VOTABLE>
```

图 6.7: 一个简单的VOTable格式文件内容。

击按钮将区域描述文本发送到Web Service，插件自动下载Web Service查询结果。第四步，选择刚下载的VOTable文件及FITS图像文件，发送到Aladin上，Aladin即自动重叠展示这两个文件。整个过程简单省事，使用者无须了解任何中间过程所涉及到的技术。

在第四章中的交叉证认技术发展成在任意区域内的交叉证认服务Web Service之后，可以通过ARM 将FITS 图像区域直接发送给Web Service，这样天文学家就无须了解Spherical Library也可以迅速完成在自己感兴趣区域内的多星表交叉证认工作。

## 6.5 天区覆盖图插件

前面几章中均提到过天区覆盖图服务（Footprint Service），其中存放了数个天文望远镜或星表的天区覆盖图信息，如SDSS、GALEX 等等。从中我们可以直观地了解到，哪些望远镜观测过我们所感兴趣的区域，哪些星表带有这些区域的数据。

Footprint Server本身也是基于Spherical Library实现的，它使用了区域描述语言来描述各个天区覆盖图。Footprint Service 并不只限于提供一些大星表的天区覆盖信息以及相关检索功能。它还提供了网页界面[20] 来添加自定义的覆盖图，甚至还专门编写了一个Web Service[21]，以鼓励有相关需要的天文工作者将自己所感兴趣的区域描绘出来，并保存到Footprint Service中。

但是，不管是使用Footprint的网页界面还是Web Service，使用者均需要学习区域描述语言。这对于普通使用者而言，过于繁琐。而在第6.4节的Coverage Query插件相关描述中，可以看到，通过WCS取得FITS图像四个顶点的坐标后可以通过空间向量叉乘取得四个半空间的法向量，进而可以用区域描述语言将图像的覆盖图描述出来。之后只需要调用Footprint Service 的Web Service即可将这一图像所覆盖的区域上传到服务器中。如图6.8的上图所示，只需要设置一次用户的GUID，这个ID 在Footprint网站上注册用户之后即可获得；选择FITS图像文件，插件即自动生成图像所覆盖的区域描述信息，简单单击右下角的“Call Footprint Service”，这个图像的天区覆盖信息就被上传到Footprint Service数据库中了。在网站上登录该GUID所对应的用户，如图6.8的下图就可看到刚上传的覆盖图信息。通过这一插件，可以一次性地将多个FITS文件的覆盖图信息上传到Footprint 网站上，并获得这些图合并后的综合消息，这对于了解手上的FITS图像文件所涉及的天区是极有用的。

[20]Footprint Service 创建覆盖图的网页界面http://voservices.net/footprint/process_preview.aspx

[21]Footprint Service 的Web Service http://voservices.net/footprint/ws_v1_1/FootprintServices.asmx

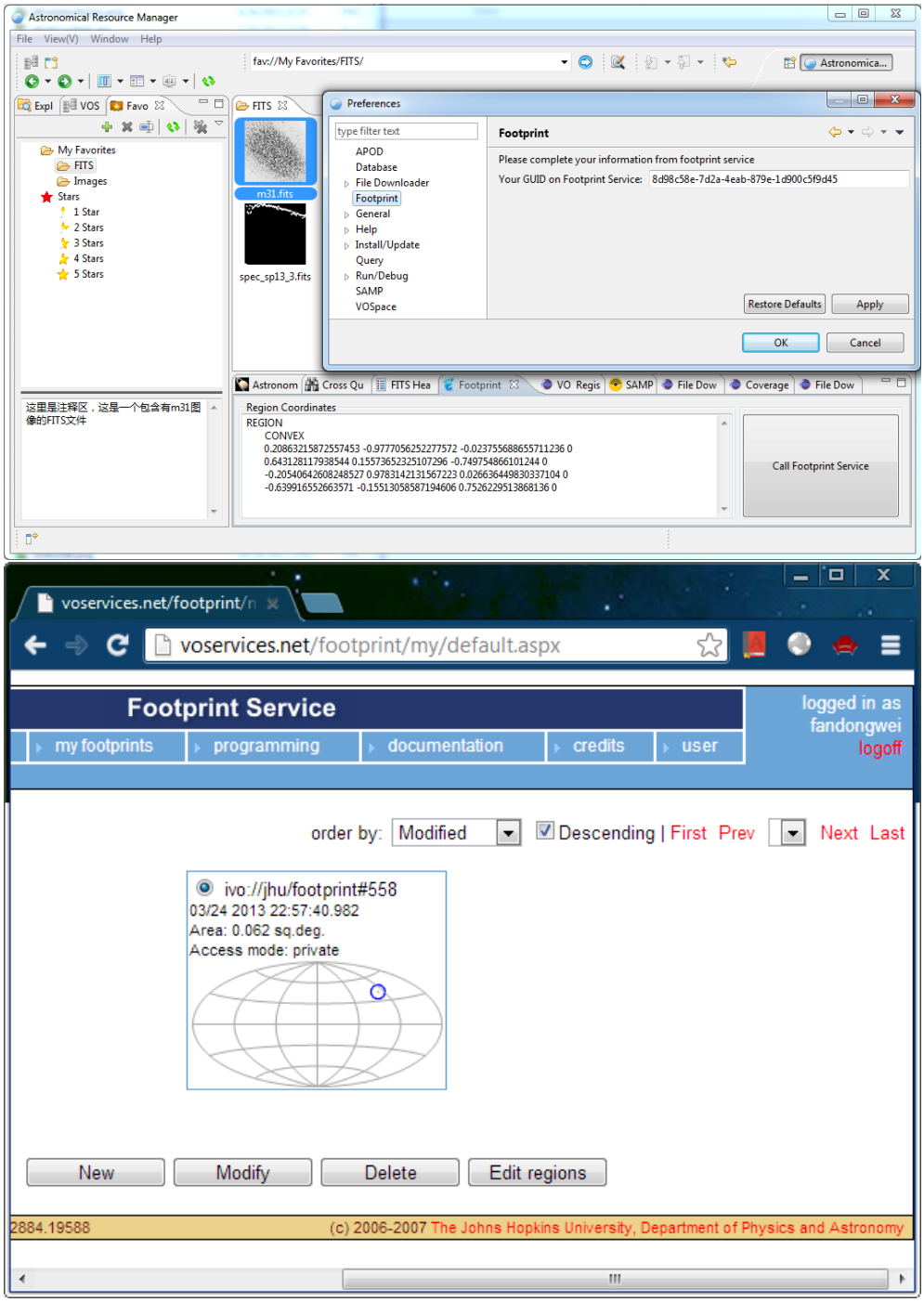


图 6.8: 上图所示的天区覆盖图插件中，仅需选择FITS图像文件，然后单击右下角的按钮，即可把该FITS图像所覆盖的区域信息上传到Footprint Serive中GUID所指定的用户目录下。下图中登录Footprint Service网站，就能看到刚上传的覆盖图信息。

## 6.6　应用程序通信插件

当今天文界所使用的软件越来越多，越来越专门化，不同的设备可能都独自开发了一套软件对自己的数据进行处理。而天文学家所使用的工具也多种多样，不再可能实现一个大而全的软件来满足人的需求。退而求其次，可使用多个软件来满足不同需求。但多个软件使用起来又过于复杂，诸如文件的打开、坐标定位等操作在无谓地重复进行着。这时候需要一个机制，让所有的软件一起工作，表面上它们仍然是独立的，但是操作起来却像是同一个软件。应用程序简单通信协议（Simple Application Messaging Protocol，SAMP）[22][98]，便是国际虚拟天文台用来实现这一目的的一个应用程序间的通信标准[8]。它通过称为MTypes[23] 的数据格式在各个程序间进行通信，互通有无。MTypes 中定义了数个消息类型，用以传递不同的数据，各个程序可按自己的需要选择是否支持某个MTypes 消息类型。SAMP当前版本为2012 年4 月11 日发布的1.3版本[24]，建构于XML-RPC[99] 通信协议之上。XML-RPC最初由Dave Winer 及微软发布[100]，并不断演进成了后来流行的Web Service的基础:简单对象访问协议SOAP[101]。

SAMP在本地计算机上设置了一个消息交换中心，称为Hub。所有的应用程序都要连接到Hub，通过declareMetadata()在Hub中注册自己的基本信息，再通过declareSubscriptions()告知Hub自己可以处理的消息类型，由Hub赋予每个程序一个唯一的ID，包括Hub 自身也有一个“hub”的ID。Hub掌控整个通信系统，应用程序可以向Hub查询其他程序的信息并通过Hub进行通信。所有的消息都要首先交到Hub，由Hub分析后，按指定类型来将该转发到指定ID。有三种通信方式，

- 通知（Notify）。一个程序A向Hub发送消息，说明A要向某个ID 的程序B发送一个指定类型消息，A 即结束当前通信；程序B 从Hub 收到消息，通信完成，B 自行决定要不要做处理

- 异步请求（Asynchronous Call）。程序A向Hub发送一个带有标记tag的消息，说明要向某个ID的程序B的索取一个消息，通信未结束，但是A可以先进行其他工作；程序B接收到Hub转发来的消息，处理后向Hub传

[22]SAMP http://wiki.ivoa.net/twiki/bin/view/IVOA/SampInfo
[23]MTypes http://wiki.ivoa.net/twiki/bin/view/IVOA/SampMTypes
[24]SAMP当前版本http://www.ivoa.net/Documents/SAMP/

送A所需消息，完成B 的通信；Hub将B回复的消息发送给A，A接收消息后，完成通信，A 可通过查看tag了解消息所对应的是自己的哪个请求。

- 同步请求（Synchronous Call）。与异步请求类似，但是A 在给Hub发送消息后程序即阻塞，直到Hub给它返回程序B 的回复后才完成通信，然后继续运行程序。

另外还有广播（Notify All）方式，实际上就是Notify通知的的变形，只是由一对一变成一对多。

目前已经有20多个软件支持SAMP[25]，并有五种不同语言或功能的SAMP工具包。1.3版本的SAMP甚至可以让网页给本地应用程序发送消息，使得远程服务也与本地应用有机结合起来了。一个简单的应用，如在不同程序间处理同一个文件。如图6.9所示，在ARM选中一个FITS文件，然后在当前连接到Hub的程序中选择一个Aladin，使用Notify 方式通过一个image.load.fits[26]类型的消息发到Aladin 上。image.load.fits类型的MType消息定义为：载入一个二维FITS图像文件。整个原始消息如图6.11所示，消息体中主要提供的信息是Aladin 的编号“C4”及FITS文件的路径“URL”。编号为C4的Aladin接收到此消息后即可从此URL中载入该FITS 文件，这样两个程序就可以共同对这个FITS文件进行处理了。实际编程中可以通过各种SAMP类库来简单地完成这一过程，如图6.10 使用JSamp[27] 库来迅速构建好消息并发送，无须自己完成这一长串XML文本。MTypes中还定义了其它一些很实用的消息，如coord.pointAt.sky 可以告诉其他程序，当前程序正在点击操作天球上的哪个坐标，这时，另外的程序就可以根据这个消息也把自己的界面也调整到该坐标上。这样很容易就可以在多个程序上查看到同一天区、同一位置的数据、图像。由于各个程序的专长各不相同，这实际上相当于把多个程序整合成了一个程序，同时在操作一样的数据。这也是SAMP 所要达到的目标：让多个程序协同工作形成合力，而不是费力去开发一个不可能实现的大而全的程序。

SAMP协议还有一些保留的消息类型。主要用来描述协同系统中发生的一些事件，如samp.hub.event. register 用于提示有新的程序进入系统，samp.hub.event. unregister表示有程序离开系统。

[25]支持SAMP的软件列表http://wiki.ivoa.net/twiki/bin/view/IVOA/SampSoftware

[26]image.load.fits 类型消息http://wiki.ivoa.net/twiki/bin/view/IVOA/SampMTypes#image_load_fits

[27]JSamp库http://software.astrogrid.org/doc/p/jsamp/1.3-3/index.html

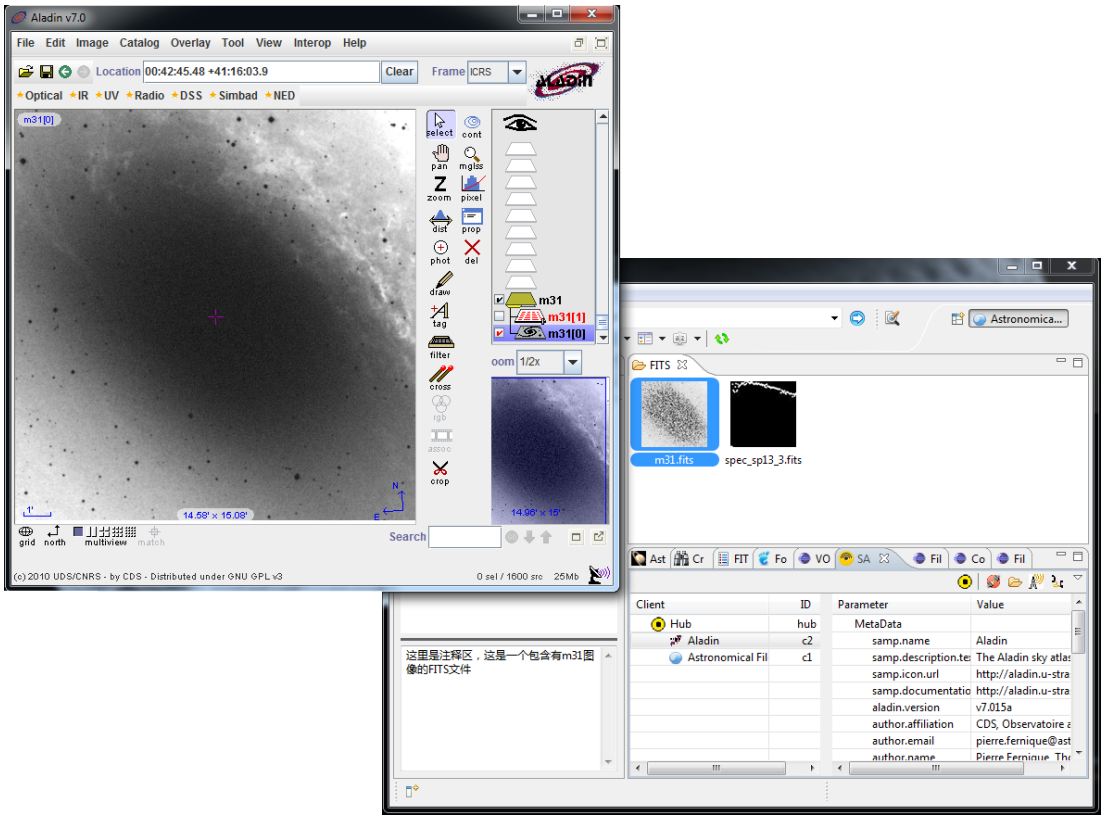


图 6.9: ARM通过SAMP通知Aladin打开指定的FITS文件。

```
Message  m = new Message("image.load.fits");
String path = "http://127.0.0.1:31719/export/3/m31.fits";
m.addParam("url", path);
m.addParam("table-id", path);
m.addParam("name", f.getName());
try {
    conn.notify("C4", m);
} catch (SampException e) {}
```

图 6.10: ARM使用JSamp来发送一个Notify消息，通知C4程序打开指定地址的一个m31.fits二维图像文件。

```
<?xml version='1.0' encoding='UTF-8'?>
<methodCall>
  <methodName>samp.hub.notify</methodName>
  <params>
    <param>
      <value>k:1_hubvkuifmynbapll</value>
    </param>
    <param>
      <value>c4</value>
    </param>
    <param>
      <value>
        <struct>
          <member>
            <name>samp.mtype</name>
            <value>image.load.fits</value>
          </member>
          <member>
            <name>samp.params</name>
            <value>
              <struct>
                <member>
                  <name>name</name>
                  <value>m31.fits</value>
                </member>
                <member>
                  <name>table-id</name>
                  <value>http://127.0.0.1:31719/export/3/m31.fits</value>
                </member>
                <member>
                  <name>url</name>
                  <value>http://127.0.0.1:31719/export/3/m31.fits</value>
                </member>
              </struct>
            </value>
          </member>
        </struct>
      </value>
    </param>
  </params>
</methodCall>
```

图 6.11: 向ID为C4的程序发送一个Notify通知的原始消息。主体是一个image.load.fits的MType消息，告知C4 打开指定地址的一个m31.fits 文件。

## 6.7　VOSpace插件

VOSpace是国际虚拟天文台联盟（IVOA）制定的一个基于Representational State Transfer（REST）技术的网络文件存储服务规范。VOSpace规范[22]定义了如何获取、存放文件，未定义物理实现的细节，因而各机构可按自己的情况将文件进行分布式存放或直接放到一个存储阵列，而VOSpace服务的使用者完全不需要了解这些信息。此外，VOSpace 服务之间也可以通过pullToVoSpace及pushFromVoSpace 等操作实施数据互传。实际上可将VOSpace 视为虚拟天文台体系内的“云存储”服务，为天文学家以统一的方式进行文件存储、共享提供便利。

目前有数家机构对VOSpace规范进行了具体实现，如法国CDS[28]，而加拿大CANFAR云计算平台将VOSpace作为其输出数据存放标准[29]。笔者借在美国约翰霍普金斯大学学习的机会，也与美国虚拟台的VOSpace实现者进行了深入交流，并通过ARM插件的方式对VOSpace进行支持，以作为ARM对云存储服务及单点登录SSO 机制进行支持的一次尝试。如图6.12，ARM需要通过美国虚拟天文台的单点登录服务SSO[30] 获得对VOSpace访问的访问令牌（Access Token），通过该令牌即可按照VOSpace 规范获取、下载一个文件的信息，或者创建、上传一个文件。进一步的还可以实现如SkyDrive、Dropbox等云存储类似的本地文件与远程文件的同步服务。

VOSpace的单点登录SSO实现对虚拟天文台的未来非常重要，由于虚拟天文台的服务越来越多。而很多服务都需要使用者通过一个用户名来管理个人数据，使用者每使用一个新服务都需要注册一次不仅麻烦，而且容易造成账户、密码管理混乱。也使得ARM 无法很好地整合网络服务。SSO 的出现极大改善了这一状况，它的理念便是通过一个用户名、密码访问所有服务。每个服务可以向使用者申请访问其个人信息，而使用者仅需要对该服务进行授权，而无需再创建单独的用户名，这实际上即是推特、新浪微博等大型平台所使用的开放授权OAuth[31] 形式。未来随着更多的服务的虚拟天文台服务加入对SSO与OAuth 的支持，ARM 将能真正地将众多服务无缝连接起来。

[28]CDS VOSpace http://cds.u-strasbg.fr/resources/doku.php?id=vospace

[29]CANFAR的VOSpace存　储https://wiki.cadc-ccda.hia-iha.nrc-cnrc.gc.ca/canfar/index.php/In_Depth_VOSpace

[30]SSO https://vaossotest.ncsa.illinois.edu/openid/

[31]OAuth http://oauth.net/

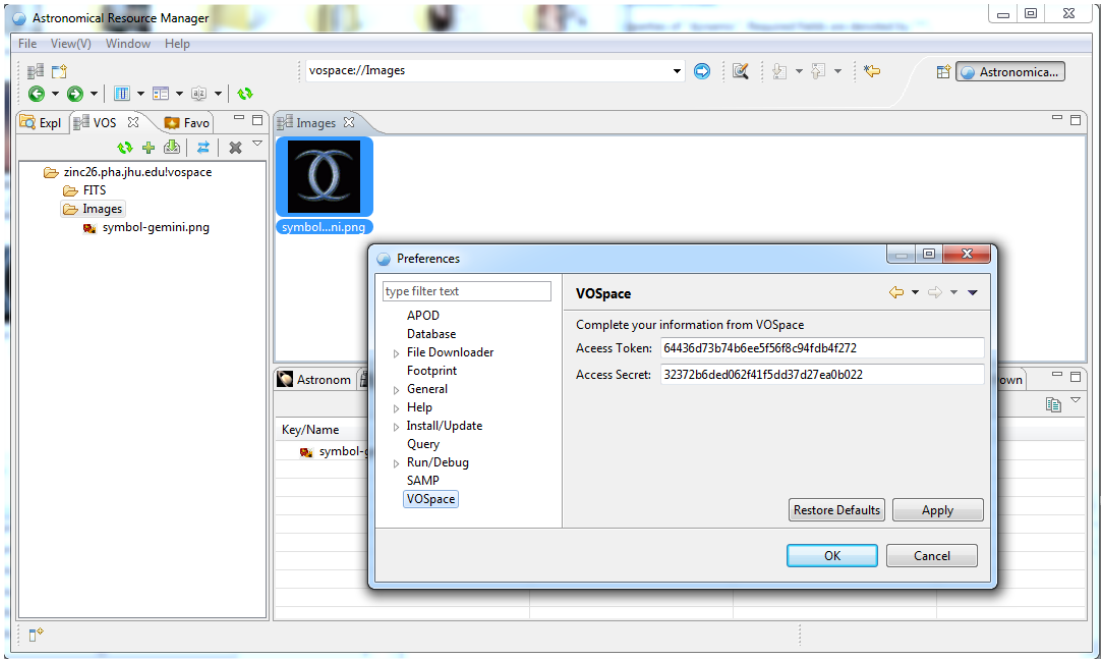


图 6.12: VOSpace插件可以直接访问到美国虚拟台的VOSpace服务。

## 6.8 服务检索插件

服务检索插件（Cross Query）主要用于直接对一些常用天文网络服务进行查询，在ARM输入关键字或者点击FITS文件后就可以直接从各个常用引擎获取信息。可以直接调用Web Service网络服务，也可以通过构造HTTP GET查询网址的方式获取某些查询的结果。目前可查询的服务、网站如图6.13中下拉列表所示，包括：

- 法国斯特拉斯堡天文中心（CDS）的天体名称解析服务Sesame Web Service[32]，它集成了CDS的Simbad[33] 与NASA/IPAC的NED[34] 名称解析服务及星表查询服务VizieR[35][102]。

- ADS，即The SAO/NASA Astrophysics Data System[36]。天文界最常用的

[32]Sesame Web Service http://cds.u-strasbg.fr/cgi-bin/Sesame
[33]Simbad http://simbad.u-strasbg.fr/simbad/
[34]NED http://ned.ipac.caltech.edu/
[35]VizieR http://vizier.u-strasbg.fr/vizier/
[36]ADS http://adswww.harvard.edu/

文献检索系统。

- arXiv[37]，非常受欢迎的非营利文献电子预印本发布网站。
- DataScope[38]，美国虚拟天文台的一个资源整合引擎，可从数百个由虚拟天文台技术驱动的服务中检索图像、星表等信息。
- CDS Bibliographic[39]，CDS的文献索引。
- DSS[40]，即STScI Digitized Sky Survey，直接获取哈勃望远镜在指定区域观测所获得的图像。
- AstroDict[41]，基于天文学名词审定委员会编定的天文学名词推荐译名制作的中英文双向查询系统[33]，可以查找到一些天文学名词对应的中英文名称，以便于规范使用天文学名词。

ADS、VizieR等服务在中国科学院国家天文台均有镜像，访问速度要比带宽较窄的越洋连接更快一些。Cross Query插件中可以很方便地做到直接从国家天文台的镜像网站查询，提高速度，而这一切对使用者都是透明的。

以上这些服务都是通过一个“查询源添加接口”的扩展点添加到Cross Query插件中的，即一个查询就是一个插件。这样做的目的是便于进一步添加更多服务，而前端保持一个简单的输入框及按钮，所有的复杂操作交由各扩展插件来具体完成。这个扩展点的编号为org.chinavo.arm.query.sources，对应的查询所要实现的Java Interface为org.chinavo.arm.query.sources.SourceIntr。从使用效果上看，查询接口要进一步完善，查询过程可能要进一步简化。需要继续征询意见以做改进。

[37]arXiv http://arxiv.org/
[38]DataScope http://heasarc.gsfc.nasa.gov/cgi-bin/vo/datascope/init.pl
[39]CDS Bibliographic http://cdsbib.u-strasbg.fr/cgi-bin/cdsbib
[40]DSS http://stdatu.stsci.edu/cgi-bin/dss_form
[41]天文学名词http://www.lamost.org/astrodict/index.php

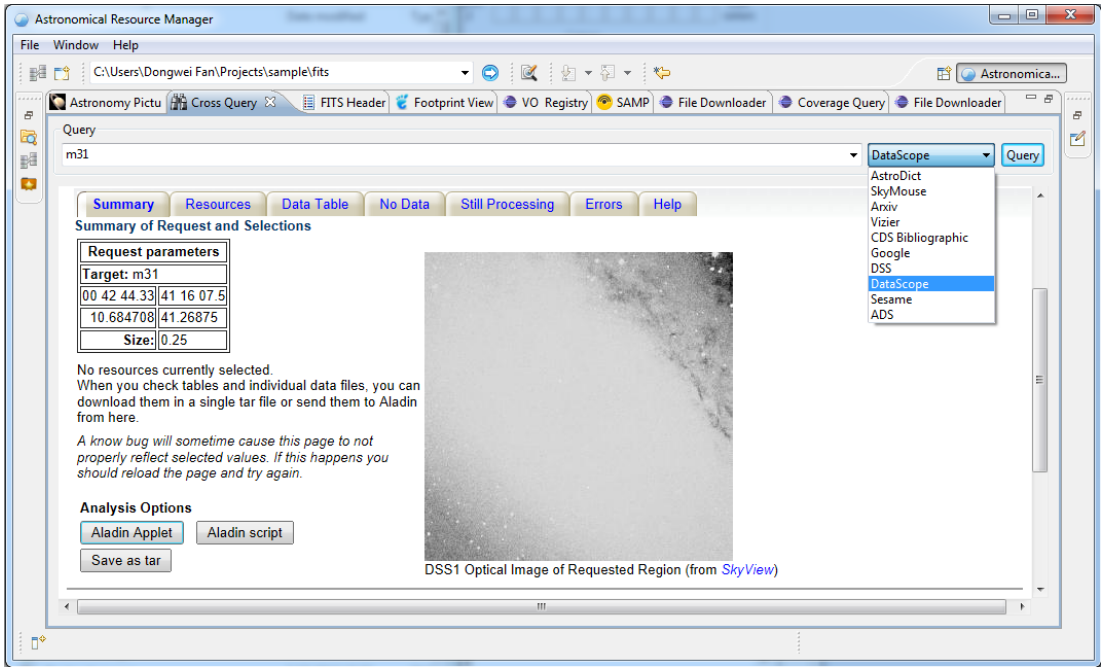


图 6.13: 服务检索插件在DataScope上查询m31的结果。

## 6.9 资源管理器及收藏夹插件

Explorer资源管理器和Favorites收藏夹插件是文件列表插件的主要数据来源。其中资源管理器提供本地系统上的文件夹层次目录，基本与操作系统自带的文件管理器类似，Windows下的界面如图6.14。图中可以看到，资源管理器还实现了文件删除、重命名、新建文件等操作。这需要在文件列表插件（fileeditor）的右键菜单中添加菜单项，这在Eclipse RCP体系中非常容易做到，仅需要扩展org.eclipse.ui.menus接口即可，如图6.15，最关键的是定位到fileeditor的右键菜单的位置，即popup:org.chinavo.arm.fileeditor. editor，这在Eclipse RCP中称为资源定位符，是Eclipse RCP本身自带的扩展框架，为程序开发带来了极大的便利。图中还可以看到资源管理器插件也实现了fileeditor的文件路径响应接口，对file:// 及诸如C:\ 这样的默认路径进行响应。当在地址栏中输入路径“C:\ Users\ Dongwei Fan\ Desktop”时，fileeditor插件发现资源管理器插件可以响应这一路径，并把这一路径送交给资源管理器插件；资源管理器把路径展开到Desktop文件夹，并载入此文件夹内的文件列表，封装成InputData 类型送回给fileeditor；fileeditor收到InpuData，展示成文

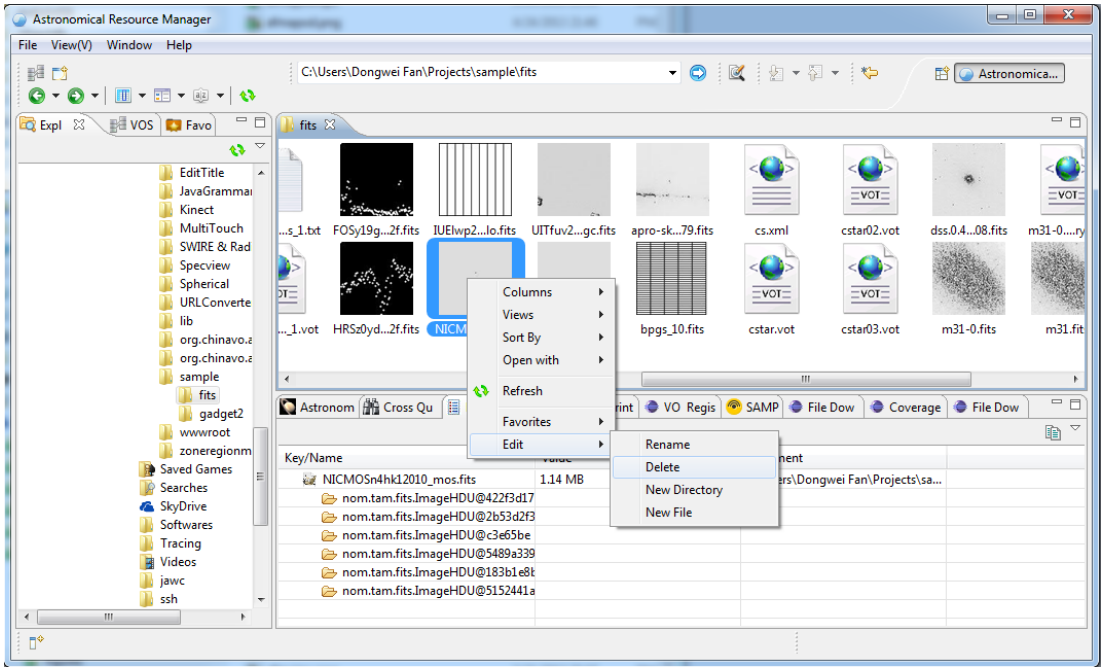


图 6.14: ARM资源管理器插件及右键菜单效果。

件列表或图标列表，必要的时候向图标提供者插件中索取文件图标。

收藏夹插件同样通过这样的方式在fileeditor的右键菜单中添加了菜单项，如图6.16所示，用来将选中文件添加到收藏夹中。而为了便于存取收藏的文件的信息，并给文件提供注释，使用了数据库。ARM默认使用了Apache Derby[42]数据库，这是一个纯Java实现的嵌入式关系型数据库，非常便于随ARM跨平台发布。ARM 也考虑到了使用其他数据库的情形，并提供了数据库扩展接口org.chinavo.arm.database，只需要制作一个插件来实现这一接口即可。但是从安装时的复杂程度考虑，嵌入式的Derby完全无须配置，对使用者要求低，最适合ARM的使用情境。收藏夹插件所使用的数据表结构如图6.17，其中`fileinfavorites` 及`comment`文件路径直接保存了文件的绝对路径，而不是专门使用一个File 表，目的是减少对其他表的依赖，也可以提高查询效率。

[42]Apache Derby http://db.apache.org/derby/

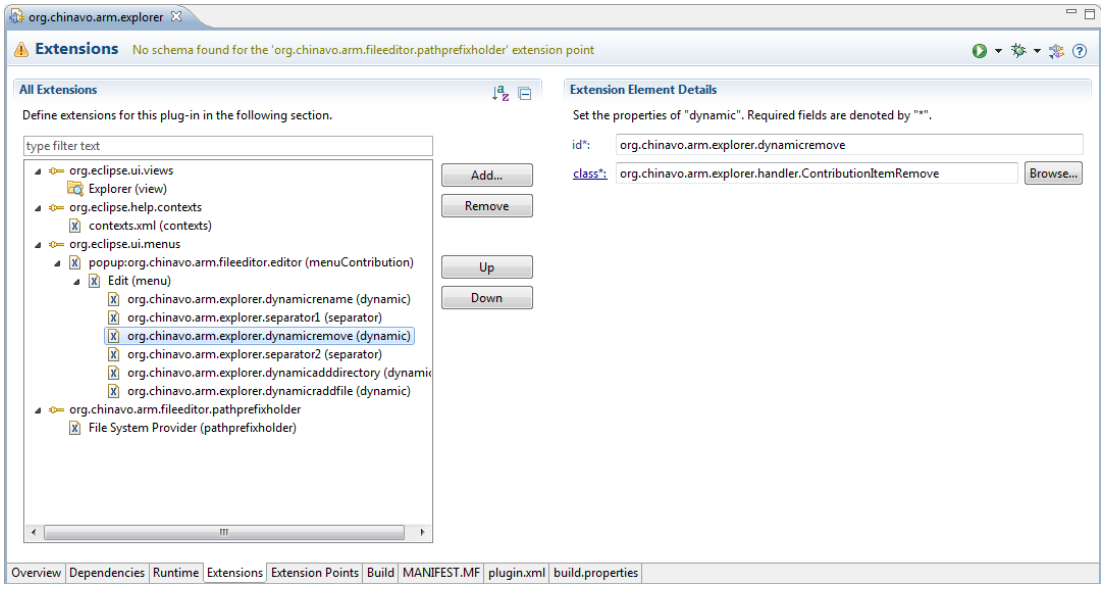


图 6.15: 资源管理器插件在文件列表插件中添加右键菜单项的方式。

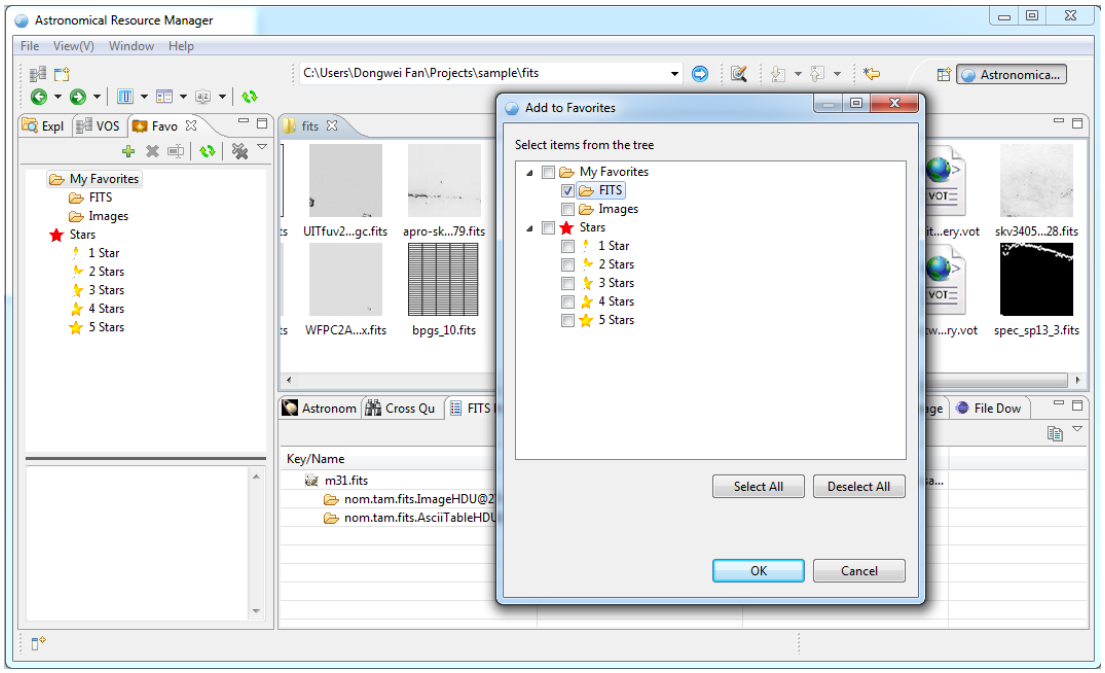


图 6.16: 收藏夹插件在文件列表插件中通过右键操作来把文件添加到指定收藏夹中。


图 6.17: 收藏夹插件所使用的数据表结构。

## 6.10　FITS头查看及文件检索插件

FITS头查看插件（FITS Header）及文件检索插件（File Search）均专门为FITS 进行了设计。如图6.18，FITS Header 列出每个FITS头中的信息，并按FITS格式的约定，每一行分为关键字KEY、数值VALUE、及注释COMMENT。文件检索除了对文件列表中的文件的文件名进行检查之外，当发现当前文件是FITS文件时，还进入文件中搜索FITS头信息，若遇到匹配字符串则将此文件列入结果集中。因为检索是在当前文件列表中进行，可以通过巧妙设置检索关键字来一步步过滤结果，取得最终想要的文件。由于涉及到FITS 文件的处理，这两个插件都使用了FITS文件的Java 类库nom.tam.fits[43]。因为有多个插件使用这个Java类库，按Eclipse RCP插件化的理念，此类库也被做成了一个插件并输出所有类成员以供其他插件在需要时引用。

[43]nom.tam.fits 库http://fits.gsfc.nasa.gov/fits_libraries.html#java_tam

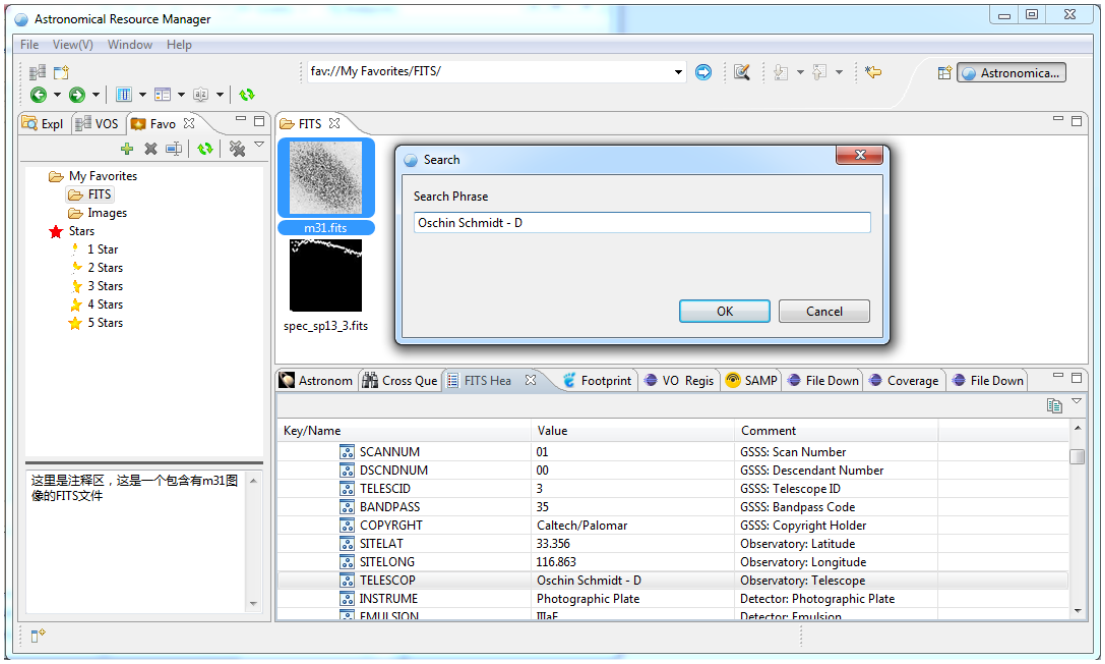


图 6.18: 图中底部为FITS头查看视图，中间的对话框为文件检索关键字输入框。

## 6.11 天文每日一图插件

天文每日一图（Astronomy Picture of the Day，APOD）是美国航空航天局的一个网站系统[44]，每天都更新一些与天文有关的图像或视频，并带有讲解文本。除了有科普的作用之外，它所提供的大图片还可以作为计算机桌面壁纸，在全世界广大天文爱好者及天文工作者中受到广泛欢迎。它在社交网站拥有众多的跟随者，并产生了各种语言约20 种镜像站点[103]；被应用到了课堂上[104]；还有人专门开发了程序用于每日自动从APOD 下载图片并设置为桌面壁纸，如APOD Wallpaper[45]。遗憾的是，它只支持windows系统，因而ARM制作了一个插件，如图6.19所示，用于查看每天的APOD信息。并提供了工具对往期图片、视频地查看与下载，且可以将指定日期的图片设置为Windows、Linux、Mac的壁纸。其中比较复杂的是壁纸设置部分，无法通过Java来直接实现，需要针对不同平台分别编写可执行程序来执行壁纸设置任务。

[44] APOD http://apod.nasa.gov/apod/

[45] APOD Wallpaper https://sites.google.com/site/apodwallpaper/

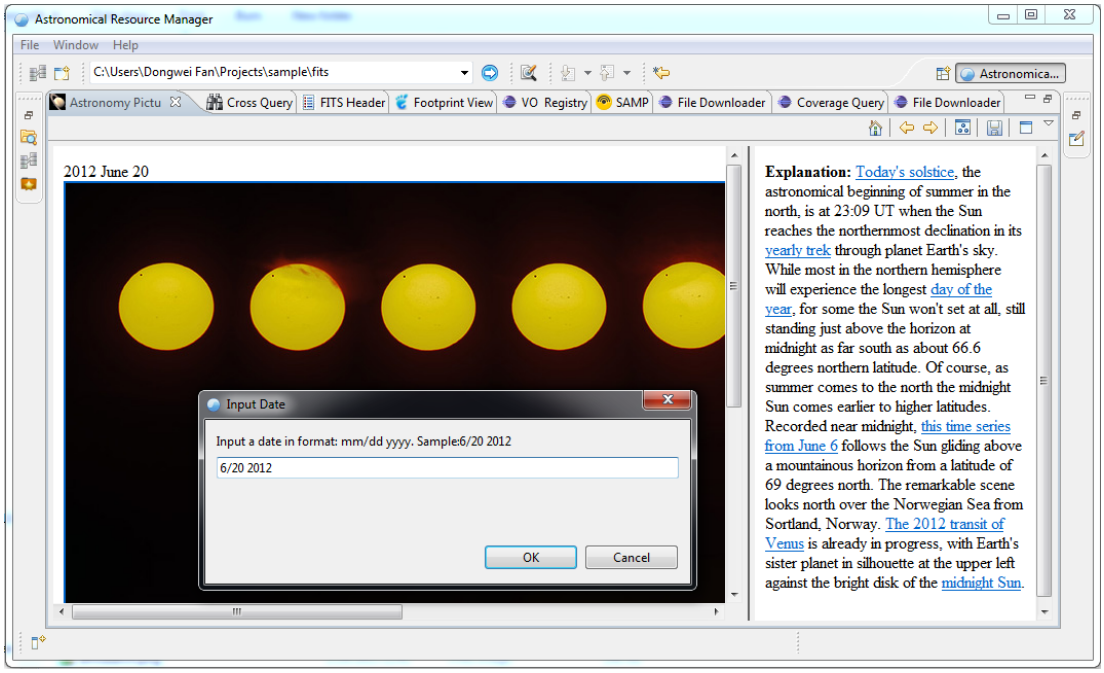


图 6.19: ARM的天文每日一图小插件，可以读取指定日期的图片、视频，并设置图片为桌面壁纸。

## 6.12　天文文件下载插件

天文文件下载插件File Downloader与APOD插件一样，是一个附加的小工具，用于帮助从一个天文数据中心下载一些有规律的小文件。它满足的是这样的需求：一些天文中心提供了某种服务，如NVSS Postage Stamp Server[46]，可以通过一个赤道坐标及查询范围或指定图像尺寸来生成数据文件并提供下载。如果手上只有几个小区域的数据要下载，手工操作是没有问题的。但是若是数百上千个，则显然是个重复无聊的苦力活。这个小插件就是为了帮助下载这种服务中的数据文件的，它通过使用者所给出的目标网址、天体坐标列表等信息自动构建文件下载地址，并主动下载文件到指定目录。如图6.20，首先需要指定待下载文件所在的网址，如http:// www.somedomain.org /query? ra={0} &dec={1} ；然后指定文件存放的位置及文件名的格式；{0} 及{1} 对应的是第三步中的第一列及第二列信息，每行及每列如何进行分隔可自行设定。点击“Start”后等待程序完成工作即可。

[46]NVSS Postage Stamp Server http://www.cv.nrao.edu/nvss/postage.shtml

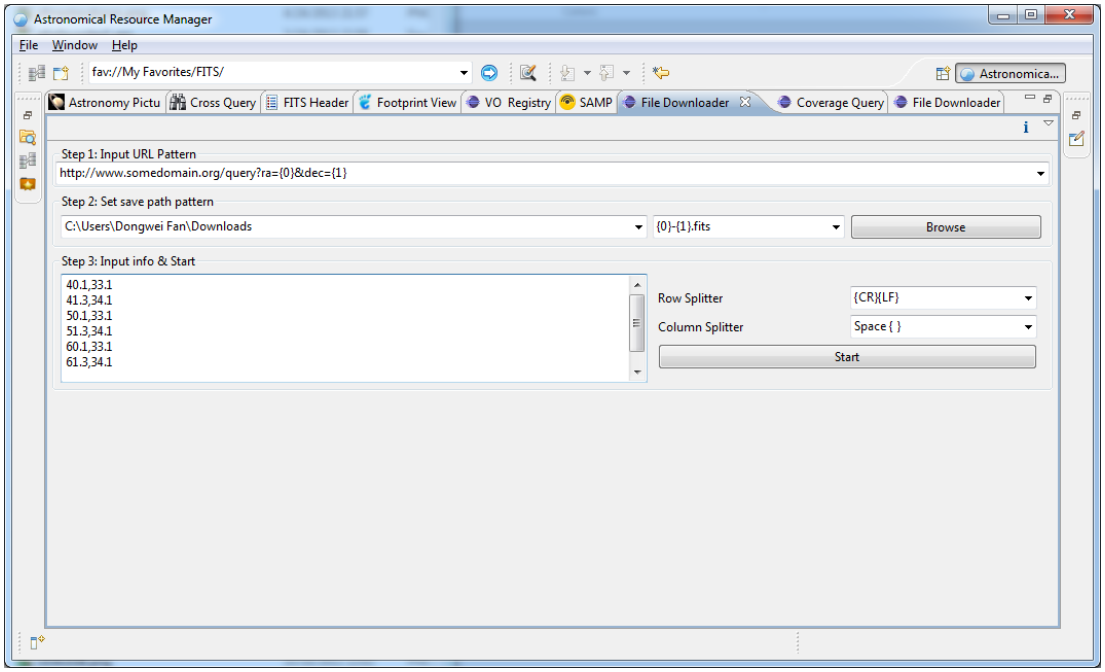


图 6.20: ARM天文文件下载小插件，可批量从指定网址下载所需数据文件。

## 6.13 小结

资源统一管理平台：天文资源管理器ARM，专注于对天文数据的管理而不是处理。ARM 直接管理本地数据文件，在此基础上通过扩展接口、SAMP、调用Webservice等方式，直接整合各个本地应用程序以及网络服务，使之形成合力，达成$1+1>2$的整体效果。ARM 所使用的跨平台插件化体系，可以非常方便地对各个部分进行增强，而无须改变原来的代码。还可以依照不同的需求，对插件进行搭配、组合以适应不同人群的需要。这非常有利于调动社区的力量，最终使得ARM 成为一个平台而不单纯是一个管理工具，这也是ARM所努力达成的目标。

ARM将继续改进程序的使用体验，简化操作，并与更多的天文服务整合。前面几章所讨论的天文星表交叉证认算法，就可以通过适当的方式引入到整个体系中来，作为中国虚拟天文台服务的一部分的同时也能在ARM 上做定制化应用。中国科学院最近加快信息化建设、推进云计算平台的发展，天文领域云是其中很重要的一个组成部分，由于ARM 灵活的插件化特性，亦有望作为云服务的客户端之一加入到平台中来，为天文学家提供更好的服务。

# 第七章　学术讲座视频资源的发布

天文资源的整合不仅包括数据资源的整合，各种讲座、教程也是很重要的学习、研究资料。国家天文台作为科研单位，每年都会举行许多次学术会议、研讨会、学术讲座、培训班等等。其中不乏诺贝尔奖获得者、邵逸夫天文学奖获得者等知名天文学家参与。这些活动增进了国内外学术圈的交流，促进了国内天文学研究的发展。

然而，并不是所有的培训班每年都会举行，年度培训的内容也不尽相同。有些著名天文学家更是难得来华做一次讲座，弥足珍贵。而目前并没有对讲座、会议进行录像、存档的习惯。这其中还包含了设备、技术方面的困难。此外还有一个问题，就是无法进行网络直播。许多无法在现场的人就没办法看到他所感兴趣的讲座及培训。

在国际上，有不少天文机构都有自己的一套学术讲座直播系统。美国航空航天局（National Aeronautics and Space Administration，NASA）下属的太空望远镜研究所（Space Telescope Science Institute，STScI）的系统叫作STScI Webcasting[1]，用于直播在研究所大演讲厅内的讲座。在其网站系统上可点播历次讲座的视频，下载各种格式的视频文件及课件、海报等等。该网站还提供了类似日历的功能，用于提醒最近要进行的讲座。按最近刚结束的讲座、本周内的讲座、将要进行的讲座等等方式进行分类，还可以在线进行搜索，非常便利。

以2012年10月11日的一次讲座为例。在9月17日，STScI的讲座邮件组就开始公告该讲座的摘要，时间、日期、地点、主讲人等等，甚至还有与讲座有关的配图。在网站首页的“Upcoming Webcasts”（即将开始的网络直播）可以查看到相应链接。10月8日在邮件组发出了第二次通知，并介绍了远程收看、收听讲座的方式及相关事项。开始可以在首页的“This Week’s Webcasts”（本周网络直播）中查看到该次讲座信息。10 月11日，讲座当天，再次通过邮件组通知讲座即将开始。可以直接到会场或通过网络观看讲座。由于网络直播有延迟，按邮件中的提示，还可以选择打电话收听。在讲座的提问时间可以通过电

[1]STScI Webcasting https://webcast.stsci.edu/webcast/

话向演讲人提问，与演讲人对话。讲座结束之后不久，即可下载到讲座的录像及演讲人提供的其他材料。进入下一周后，在“Recent Webcasts”（最近网络直播）中仍可以找到该日的讲座。之后，该讲座将被存档，可以在“Webcast Archives”（网络直播存档）中找到或搜索到。除了订阅STScI的讲座邮件组以外，该网站还提供了RSS信息源，可以通过RSS阅读器来获得最新的讲座信息。

NASA本身则走得更远，甚至已经做成了网络电视台的形式：NASA TV[2]。用于直播火箭发射、航天器着陆等等。NASA TV还提供了手机应用程序下载，观众在手机上也可以看到NASA TV的内容。部分望远镜（天文台）的网站也提供了视频直播的功能。如，欧洲南方天文台（European Southern Observatory，ESO）的直播系统（ESOcast）[3]；美国夏威夷凯克（W.M. KECK）天文台提供了往期讲座的在线观看网页[4]。

国内天文系统也有过网络直播的经验，如国家天文台联手多家单位共同发起的“2009国际天文年日全食多路联合直播”[5]就是一次成功的实践。直播信号被国内外多家媒体引用，产生了广泛的影响。也凸显了网络对于天文知识传播的作用。上海天文台承办的天之文网站利用网络论坛功能提供了一个“天文直播室”[6]，组织天文学者开展讲座，在线直播并提供对往期视频内容的在线观看。但这些都是针对天文科普的应用。目前亟需的是将学术讲座的内容也直播并录像并公开到网络上，既可保存下珍贵的讲座内容，也方便天文专业学生的学习。也是公众了解天文研究的一个途径。

## 7.1 架构分析

从讲座现场录像，到观众最终可从其个人计算机上收看的过程，需要经过一系列的数据转换、缓存和传输。视频信号要经历一连串的硬件和软件。如图7.1 所示（圆角外框表示一个硬件设备，框内矩形表示一个软件系统，箭头旁边的文字说明其传输的内容），硬件方面有摄像设备、视频采集设备、网络

[2]NASA TV http://www.nasa.gov/multimedia/nasatv/index.html

[3]ESOcast http://www.eso.org/public/videos/archive/category/esocast/

[4]凯克天文台Podcast http://keckobservatory.org/education/podcast

[5]日全食多路联合直播让“长江日全食，全球看得见”http://www.nao.cas.cn/xwzx/zhxw/200908/t20090827_2449121.html

[6]天之文网站的“天文直播室”http://www.astron.ac.cn/list-47-1.htm

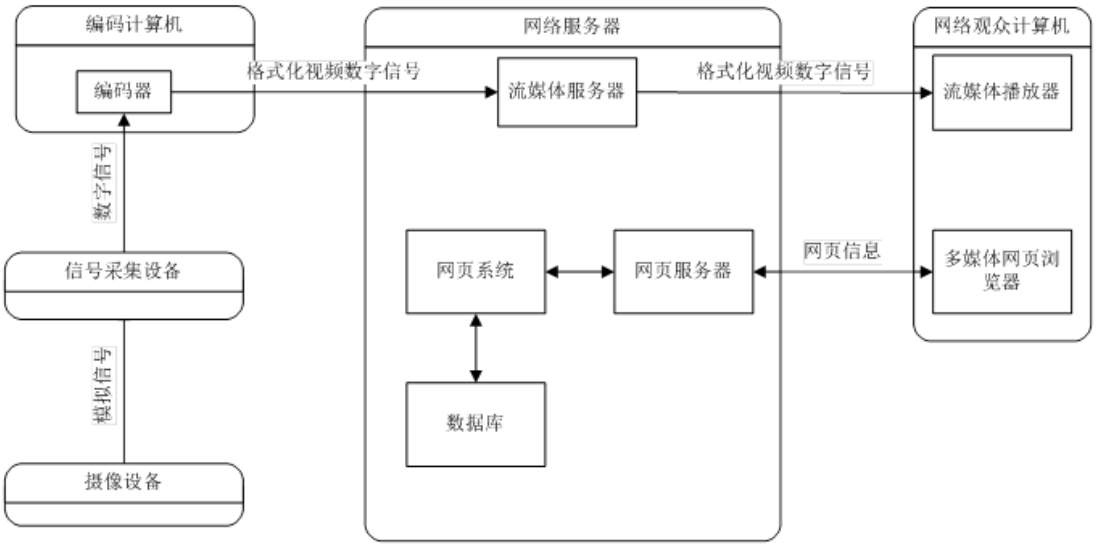


图 7.1: 视频直播系统的软硬件架构。

服务器；软件方面有编码软件、流媒体服务器、流媒体播放器，网页系统还需要数据库、网页服务器及浏览器。

### 7.1.1　现场信号转换

摄像设备负责将现场的画面、信号记录下来。几乎所有的摄像设备都带有实时输出视频、音频信号的端口。可通过这些端口获取到现场信号，并进行进一步的处理。摄像设备同时还带有录像功能，在往端口输出信号的同时，仍可将现场录像保存到自身的存贮介质上。这是第一手的资料存贮，因而在影像存档中非常重要。可用于较高清晰度影片的制作。但是，由于此类资料占用空间较大，在存贮容量有限的情况下，需酌情删除价值较小的原始资料。

摄像设备所输出的是模拟信号，计算机无法直接对模拟信号进行处理。因而，需要一个设备来对模拟信号进行采样并转换成计算机可处理的数字信号。这种设备即是采集设备，在台式机上，它通常以扩展卡的形式存在，又称为采集卡。不同的采集卡的采样能力不尽相同，需要按照实际需求进行选择。

采集卡所生成的数字信号，还需要经过编码器转换成适宜发布的格式，如WMV、AVI、FLV等等。这需要一台专门的计算机来完成。实际上需要的是安装在该计算机上的编码器。由于编码器需要消耗的资源较大，要求此计算机

有较强的计算能力，并能长时间稳定工作。实际上，由于计算机技术的飞速发展，笔记本电脑在一定程度上已可满足需求。但是容易使笔记本电脑工作温度过高，需要注意对笔记本进行通风、降温。编码器本身已经具备了对外发布现场信号的能力。但是由于编码计算机的计算能力不够强、带宽有限，需要将信号发布到专业的流媒体服务器上，由流媒体服务器来完成对数据的缓存及发布，并接受较大规模的网络访问。编码器在将数据发布到流媒体服务器上的同时，亦可同时将信号保存在计算机上。这是广播级别的文件，可将此文件保存，用于日后在网上再次发布。

从信号发布的角度看，以上步骤已经完成了现场对外广播的流程。但是，这时候远程网络上的观众在获知访问地址的情况下，只能看到视频，无法再得到其他信息。更严重的是，很多人无法得知流媒体服务器的数据发布地址。这时候，需要一个网页系统来完成相关信息的公布。

### 7.1.2 视频相关信息的发布

网页系统首要公布的是流媒体数据的发布地址。更方便的方法可在网页上嵌入一个流媒体播放器，通过网页脚本自动载入视频。这样用户只需要查看网页就可观看到视频。网页系统还需要公布的信息如：讲座的主题、演讲人的个人信息、演讲的课件。最好还能提供对已结束的讲座视频进行检索、在线观看、文件下载功能。这往往需要结合数据库服务器、网页系统、网页服务器来完成。通过网页将视频信息汇总，最终远程网络观众将可以同步地看到讲座的过程。利用流媒体服务的点播功能或服务器的文件下载功能，观众还可以在事后再重复观看、学习讲座内容。这无疑将大大方便由于种种原因无法出现在讲座现场的观众，而随时随地在有网络连接的情况下看到自己所需要的信息。

## 7.2 系统实现

本系统主要采用使用成较低的硬件及使用较广泛的Microsoft系列软件来实现。录像机采用普通家用DV，采集卡使用天敏科技（10moons 400uv）天敏UV200视频采集棒[7]，编码计算机使用便于移动的个人笔记本电脑（安装Windows 7 操作系统、Windows Media Encoder 9编码器），加上拥有独立IP及网络域名的网络服务器一台（安装Windows Server 2008操作系统、Windows

[7]UV200 http://www.10moons.com/productInfo.asp?PID=481

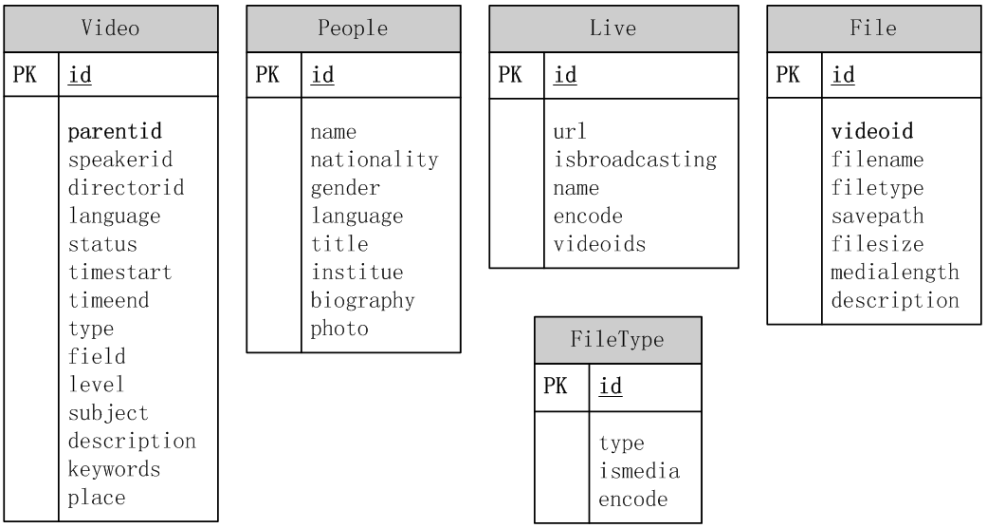


图 7.2: 与视频信息相关的数据库系统设计。

Media Service流媒体服务器、Windows SQL Server 2008数据库系统、Windows Internet Information System——IIS网页服务器）。由于Windows Media Service流媒体服务器对视频的编码格式为WMV，因而要求网络上远程访问的观众的计算机要有能支持MMS流媒体的播放器（如Windows Media Player）及多媒体网页浏览器（带Windows Media Player播放器插件）。

本系统的视频转换、传输部分均基于已有的成熟工业产品，具体实施时仅需要做一些连线、配置及临场数据传输即可。因而在具体实现上的主要着墨于网页系统，对信息管理和发布下功夫。

### 7.2.1　数据库结构

为了支持对数据的统一存储和检索，数据库是当前较好的选择。针对一个讲座的活动流程，本系统的数据库需要存储的信息如图7.2所示，包括：

与讲座视频本身直接相关的（Video表）：讲演人、主持人、讲座所使用的语言、讲座进行状态、开始及结束时间、类型、专业领域、难度、主题、简介、关键字、地点等等。有时候一个培训带有好几个讲座，可以通过设定这些讲座的parentid来将某个培训设为其父项，以便于查找同一培训内的所有的视频。

讲演人及主持人的信息归入People表，主要展示其简历。直播地址需要在流媒体系统中单独配置，由Live表管理，在停止直播之后需要从数据库中删


图 7.3: 对视频信息进行管理的后台页面。

除。当某视频在某直播点的videoids字段中时，表明此视频正在被直播中。前端网页据此进行相关提示，引导网络观众打开相应链接，观看直播。和一个视频相关的文件的情况较为复杂，可以是图片、讲稿、视频等文件。因而需要通过文件类型（FileType）来指定，不同文件的存放位置可不相同。如可用于点播的视频文件将被保存在一个指定的文件夹中。而讲座的海报亦可通过预定义一个文件类型来指明。相关网页根据这些信息进行适当地显示。

### 7.2.2 网页系统实现

单纯对于信息发布来说，实际上也可以使用现成的开源产品，如应用相当广泛的内容管理系统（Content Management System，CMS）Drupal[8]。但是，它不易对视频信息进行针对性管理，对数据库的定制也颇为复杂。因而，本系统还是采取了独立开发的方式。

本系统使用了IIS网页服务器上运行效率较高的ASP.net + C#进行具体实现。分前、后台两部分。后台页面负责信息的录入；前台页面负责综合信息进行展示，并对视频点击量、文件的下载量进行统计。

对视频信息进行管理的一个后面页面如图7.3所示。基本上采取的是以视频为中心的信息管理方式。演讲人信息、文件类型、直播点均为一次即可录入完毕，为视频信息的录入做准备。视频的录入除了基本信息录入之外，还应可多次添加新的相关文件，如图7.4为点击某一视频的“相关文件”链接之后所显示的与此视频有关的文件信息。

一个用于显示直播中的视频讲座前台页面如图7.5所示。尽量在一个页面内放入数据库中所保存的与此视频有关的所有信息。最重要的是提供了一个播

[8]Drupal http://drupal.org/

| id | video_id | file_name | filetype_id | host | savepath | file_size | media_length | priority | description | hits | Manage |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 23 | 22 | ppt | 8 | local | ../files/977779806.pptx | 7598 | 0 | 1 | 课件 | 28 | Del Edit |
| 24 | 22 | 首页图 | 1 | local | ../files/97116958.jpg | 115 | 0 | 1 | 首页图片 | 37 | Del Edit |
| 37 | 22 | output20100407b.wmv | 4 | local | ../files/output20100407b.wmv | 71124 | 45 | 1 | output20100407b.wmv | 266 | Del Edit |

图 7.4: 一个视频的相关文件列表。


图 7.5: 显示一个直播中的视频讲座信息的页面。

放器，以便观众能直接在网页上看到视频的内容。在网页播放器无法播放的时候，在播放器下方也提供了视频的实际网络地址，可由观众自行输入到能读取流媒体的播放器上。

### 7.2.3　视频点播功能的实现

对于已经结束的讲座，其视频可以存档到服务器上，使观众可随时收看。通常，需要使用FTP等工具将视频文件传送到服务器上，再对流媒体服务器进行配置，以使观众可远程查看到这一视频。否则只有提供下载功能，待观众下载完整个文件后，再自己选择要收看的部分。这一过程太过耗时且耗流量，非常需要既方便系统管理员上传视频文件并完成相关配置及信息记录，又能使远程观众只选择他所需要的部分观看。Windows Media Service本身的能力为此提供了一些便利。

Windows Media Service有两项基本服务：广播与单播。广播即是一对多的数据传播，服务器在一个指定的点发布数据，来访者只能随着数据发布的进度来查看到视频的进程，适合于现场直播的情况。单播是一对一的数据发布，来访者可以根据自己的需要来读取选定时段的视频，适合于单独播放服务器上已经存在的视频文件[105]。即讲座已经结束，经过整理后的视频纪录文件。在对Windows Media Service进行配置时，可对一个单播的发布点（如：mms://webcast.bao.ac.cn/vod/，vod为单播发布点名称，前面网址为流媒体服务器地址）指定一个文件夹，只要是放到此文件夹内的文件。通过字符串拼接即可知其网络的访问地址。如一个文件为“a.wmv”，可知其远程访问地址为：mms://webcast.bao.ac.cn/vod/a.wmv。利用这一机制，本网页系统通过提供网页界面直接将视频文件上传到服务器。服务端的后台程序会将文件放置到此指定文件夹内，并记录到数据库中（标示文件为可点播类型）。当从数据库中读取到与此视频有关的信息时，网页程序即可自动将视频的访问网址拼接出来，并提供给网页播放器（或流媒体播放器）进行播放。

## 7.3 小结

本章通过对天文科研机构网站的调研，说明了在线直播学术讲座已成趋势。对视频直播系统的硬件、软件架构进行了分析，并以较低成本实现了一个可行的在线直播系统。着重说明了一个对直播信息进行管理的网页系统，包括其数据库及网站结构。

当前互联网上的交流热点已从原来的BBS、博客转移到社交媒体SNS及微博上来。未来在此系统中还将加入对这些新兴媒体的支持，如视频分享等等。加强讲座过程中在网上与讲演者的互动。

此外，在线视频直播系统对于科普也是有益处的。比如对重要天象进行网络直播，提供较权威的对天象的解释。再基于此系统的视频存档、点播功能，在得到授权的情况下，还可以传播一些科学性较强的科普视频、文章，以吸引普通天文爱好者的眼光，达到社会传播的目的。

# 总结与展望

本论文研究了天文资源无缝融合的几个关键技术，包括星表交叉证认技术、天文资源的统一管理平台以及非结构化的天文讲座等视频资源的管理。主要成果与创新点包括以下几项

- 对Jim Gray的星表交叉证认算法——条带算法（Zones Algorithm）——进行了改进。通过将星表天区覆盖信息直接带入到交叉证认过程，并使用同样的索引方式对数据进行组织。使得已经非常高效的Zones Algorithm进一步提速，印证了天区覆盖信息在交叉证认中的价值。

- 快速的星表缺失源检测。通过将星表天区信息与交叉证认相结合，在完成交叉证认之后，可以非常快速地了解两星表间不匹配的数据是由于未在观测区域内异或是因为未达证认匹配半径。这在以往不可行，或需要额外的步骤对数据重新组织。

- 提出光学星表与射电星表进行交叉证认的一种可行方法。基于贝叶斯分析方法，使用直线非对称模型对光学源与射电源的相关性进行评估。以实现对光学与射电星表的交叉证认，为多波段数据融合提出了一种可能的方法。

- 跨平台插件化的天文数据、服务整合平台。通过建立一个平台，将多个技术连接起来，对使用者隐藏计算机技术细节，使得各种天文数据、服务的使用更为自然。

- 低成本的讲座直播、发布方案，更便于使用及部署。天文讲座视频亦是非常重要的天文资源，有效地存档有助于天文工作者更好的学习、工作。这在以往不受重视或难于实现，在此提出的全套低成本方案可使得系统更快得到实现与应用。

星表交叉证认的主要目的是对数据资源本身的整合，可将多个星表的数据合并为同一星表，甚至是将不同波段的数据联系到一起。本论文首先基于Jim

Gray的Zones Algorithm算法，创新地将天区信息直接带入到交叉证认过程。使得交叉证认只在两个星表的重叠区域进行，加快了计算的速度。

关键的天区信息预先已经使用了天区图形操作库（Spherical Library）进行了描述，并发布到天区覆盖图服务（Footprint Service）中。在这里的核心方式是使用Zones Algorithm中的条带来与星表天区覆盖图进行交集运算，取得两者交集，并算出条带的两端边界得到一个条带片段。最终是用条带片段的集合（表）来重新描述天区覆盖图。由于条带片段表中的每个条带片段仅有条带编号、左边界及右边界三个数值，非常便于进行并集、交集等运算，可轻易取得两个星表的重叠区域（的条带片段描述表）。而若直接对两个星表覆盖图做交集运算，复杂度将大大提高，且计算结果难以被复用。尤为重要的是可以使用与星表一样的索引方式来对条带片段进行索引，也即使用条带编号（ZoneID）与赤经（$\alpha$）作为聚集索引：$(\texttt{ZoneID}, \alpha)$。由于计算较复杂，整个使用条带片段来描述星表覆盖图的过程较为缓慢，但是这一计算只需要做一次。所得的新的条带片段描述表可以随意应用到其他场景，只要保持天球的划分方式不变即可。两个星表的条带片段描述表还可以用来生成Zones Algorithm中的邻近条带对比表（ZoneZone），由于数据量比星表小很多，此表的生成过程比原来的Zones Algorithm大大减少：从一分多钟减少到约3秒。

由于条带片段表使用了与星表一样的索引方式，且是聚集索引。保证了与一个条带邻近的条带片段也都保存在邻近物理存储区域上，这与Zones Algorithm中的邻近天体附近的其他天体也都保存在邻近物理存储区域上相一致，系统I/O效率非常高。而通过只在重叠区域的条带片段内对星表进行交叉证认，整个算法得到了加速，在试验中获得了接近20%的速度提升。并且，交叉证认的时间消耗随着两星表重叠区域的减少而同步减少，基本呈线性下降趋势。此算法将应用到美国虚拟天文台的交叉证认服务（Open SkyQuery）新的交叉证认引擎中，并作为中国虚拟天文台未来类似服务的技术储备。

随着天体覆盖图信息的加入，交叉证认还可以判断两个星表（A、B）中未能匹配的天体，是因为一个星表A中天体未在另一个星表B的观测覆盖范围内，异或是在观测范围内但是没有匹配天体。后者的情形即称为星表B的缺失源（Dropout）。在以往的交叉证认过程中，这样的信息是不可获取的，或者是需要使用另一套索引体系（如HTM）来对数据重新进行组织。而在本论文的方法中，这一过程是直接而迅速的。只需要在两个星表覆盖图的重叠区域内检

索出星表A中的所有天体，再简单去掉已经被B匹配的天体即可，在SQL语句中仅仅一个EXCEPT操作就能解决。Dropout检测所得信息非常重要，可用于判断某次观测是否曝光不足、未达观测极限等等。

目前Spherical Library只有Microsoft SQL Server 的SQL 版本，因而只能在SQL Server 中进行上述算法的试验。但是本人已经实现了Spherical Library 的C++ 版本，MySQL等数据库可以通过用户自定义函数UDF 的方式将Spherical Library 引入到MySQL中。未来MySQL 等数据库也将有希望用上Spherical Library 来描述球面图形，并将星表的天区覆盖图带入到交叉证认等过程中去。

基于Zones Algorithm的交叉证认算法主要解决的还是光学星表间的交叉证认问题。而在射电观测中，有些天体带有喷流，在星表中有多个数据对应所观测到的中心以及各喷流瓣。这些喷流瓣远离中心，但是却是源自同一天体。为了与光学星表进行交叉证认，本论文使用了一个直线模型来描述此类喷流的情形。由于无法知道一个光学源A周围的射电源与A的关系，需要遍历所有可能的组合以对应直线模型中的各个组成部分。并通过贝叶斯推断的方式来评估各个组合的概率，最后选择其中可能性最高的一个假设来作为交叉证认结果。通过将程序结果与澳大利亚射电天文学家手工交叉证认的结果进行对比，直线模型对双侧喷流并有中心源的射电天体的交叉证认结果较好，对单侧喷流的证认效果较差，而对无喷流的情形证认效果也不错。在未来，此方法可应用于对SKA等大规模射电观测结果与光学观测结果的数据融合。

另外，贝叶斯推断可以将已经完成的计算作为新的检验的验前概率，因而，实际上还可以加入新的限制条件，对交叉证认结果进行进一步修正。本论文所使用的直线模型也只是一种候选的模型，未来也可以使用其他模型应用此方法对射电星表或其他对同一天体/结构有多个观测结果的情形进行交叉证认，得出一个在该模型下概率较高的结果。

数据融合只是天文资源无缝融合的一个方面，各种天文服务与工具的统一管理也非常重要。本论文通过研究对本地数据文件——主要是FITS文件——的管理，使用虚拟天文台的成熟技术，创新地将本地应用工具与一些天文网络服务有机地结合起来。使得原本各自独立的各种工具和服务都可以在一个平台——天文资源管理器（Astronomical Resource Manager，ARM）上应用，降低了天文学家对各种天文服务的使用难度。ARM 使用了基于Eclipse RCP 框架

的富客户端技术，可实现跨平台及插件化。除了可以有效地通过插件机制稳健地给ARM提供更多的功能，也有助于未来加入更多天文圈内的力量来将此平台的影响力进一步扩大。本论文前半部分论述的星表交叉证认算法，亦可借此落地。在研发相应的在线星表交叉证认服务，丰富中国虚拟天文台应用的同时，在ARM 定制一些专门插件，满足天文学家在小天区范围内进行多星表交叉证认、获取多波段数据等等需求。

最后，本论文讨论了非被足够重视与结构化的天文数据：天文学术讲座视频。通过简单的家用DV及视频采集卡，实现了视频讲座在线直播与过往视频点播的功能，并设计了一个在线系统对视频信息进行统一管理。整个体系由于成本低廉、便携，将有助于更多的讲座能够在网络上传播，达到更好的教育与科学普及效果。当视频资源聚集到一定数量之后，未来可将天文数据与相关视频或教程联结起来，为天文学家使用数据、学习、研究提供更大的便利。

总体上，本论文包含了与天文资源融合有关的三个方面的研究：交叉证认算法、资源融合平台与视频数据的整理、发布系统。这是对天文资源无缝融合的不同方面的探讨，未来还可以把它们进一步整合起来，将整个过程透明化，让天文学家可以一站式访问到所需资源，而无须过多关注具体技术细节。

# 参考文献


[1] Bell, G., Hey, T. & Szalay, A. 2009, Science, 323, 5919, 1297 - 1298

[2] Djorgovski, S. G., Donalek, C., Mahabal, A., et al. 2006, arXiv:astro-ph/0608638

[3] Large Synoptic Survey Telescope, http:// en.wikipedia.org/ wiki/ Large_Synoptic_Survey_Telescope

[4] Becla, J., Hanushevsky, A., Nikolaev, S., et al. 2006, Proceedings of the International Society for Optical Engineering, 6270,

[5] Panoramic Survey Telescope and Rapid Response System, http:// en.wikipedia.org/ wiki/ Pan-STARRS

[6] Data Handling, http:// pan-starrs.ifa.hawaii.edu/ public/ design-features/ data-handling.html

[7] Hanisch, R. J. 2006, Data Science Journal, 5, 168

[8] Taylor, M., Boch, T., Fitzpatrick, M., et al. 2011, arXiv:1110.0528

[9] Kent, B. R., & Plante, R. 2007, Astronomical Society of the Pacific Conference Series, 382, 491

[10] Williams, R., Hanisch, R., Szalay, A., et al. 2008, Simple Cone Search Version 1.03, http:// www.ivoa.net/ documents/ latest/ ConeSearch.html

[11] Tody, D., & Plante, R. 2009, Simple image access specification version 1.0, http:// www.ivoa.net/ documents/ SIA

[12] Dowler, P., Rixon, G., & Tody, D. 2010, Table Access Protocol Version 1.0, http:// www.ivoa.net/ documents/ TAP

[13] Tody, D., Dolensky, M., McDowell, J., et al. 2012, Simple Spectral Access Protocol Version 1.1, http:// www.ivoa.net/ documents/ SSA

[14] Salgado, J., Osuna, P., Guainazzi, M., et al. 2010, Simple Line Access Protocol Version 1.0, http:// www.ivoa.net/ documents/ SLAP

[15] Ortiz, I., Lusted, J., Dowler, P., et al. 2008, IVOA Astronomical Data Query Language, http:// www.ivoa.net/ documents/ latest/ ADQL.html

[16] Salgado, J., Rodrigo, C., Osuna, P., et al. 2012, IVOA Photometry Data Model Version 1.0, http:// www.ivoa.net/ documents/ PHOTDM

[17] McDowell, J., Salgado, J., Blanco, C.R., et al. 2013, IVOA Spectral Data Model, http:// www.ivoa.net/ documents/ SpectralDM

[18] Osuna, P., Guainazzi, M., Salgado, J., et al. 2010, Simple Spectral Lines Data Model Version 1.0, http:// www.ivoa.net/ documents/ SSLDM

[19] Lemson, G., Bourges, L., Cervino, M., et al. 2012, Simulation Data Model Version 1.0, http:// www.ivoa.net/ documents/ SimDM

[20] Rixon, G. & Graham, M. 2008, IVOA Single-Sign-On Profile: Authentication Mechanisms Version 1.01, http:// www.ivoa.net/ documents/ latest/ SSOAuthMech.html

[21] Zwolf, C.M., Harrison, P., & Petit, F.L. 2012, Parameter Description Language Working Draft Version 0.1, http:// www.ivoa.net/ documents/ PDL

[22] Graham, M., Morris, D., Rixon, G., et al. 2012, VOSpace specification Version 2.0, http:// www.ivoa.net/ documents/ VOSpace

[23] Derriere, S., Gray, N., Louys, M., et al. 2013, Units in the VO Version 1.0, http:// www.ivoa.net/ documents/ VOUnits

[24] Derriere, S., Gray, N., Hessman, F.V., et al. 2009, Vocabularies in the virtual observatory, http:// www.ivoa.net/ documents/ latest/ Vocabularies.html

[25] Delmotte, N., Derriere, S., Norman, G., et al. 2006, Maintenance of the list of UCD words Version 1.2, http:// www.ivoa.net/ documents/ latest/ UCDlistMaintenance.html

[26] Derriere, S., Gray, N., Mann, R., et al. 2005, An IVOA Standard for Unified Content Descriptors, http:// www.ivoa.net/ documents/ latest/ UCD.html

[27] Martinez, A.P., Derriere, S., Delmotte, N., et al. 2007,The ucd1+ controlled vocabulary, http:// www.ivoa.net/ documents/ latest/ UCDlist.html

[28] Seaman, R., Williams, R., Allan, A., et al. 2011, Sky event reporting metadata Version 2.0, http:// www.ivoa.net/ documents/ VOEvent

[29] Plante R., Linde T., Williams R., et al. 2007, IVOA Identifiers, http:// www.ivoa.net/ documents/ latest/ IDs.html

[30] Benson, K., Plante, R., Auden, E., et al. 2011, arXiv:1110.0513

[31] Hanisch, R. et al. 2007, Resource Metadata for the Virtual Observatory Version 1.12, http:// www.ivoa.net/ documents/ latest/ RM.html

[32] Ochsenbein, F., Williams, R., Davenhall, C., et al. 2011, arXiv:1110.0524

[33] 崔辰州, 余恒& 卞毓麟 2011, 天文研究与技术：国家天文台台刊, 8, 2, 178 - 184

[34] 赵永恒, 崔辰州 2011, 科研信息化技术与应用, 2, 3, 3

[35] Budavari, T., Szalay, A. S., Fekete, G., Dobos, L., et al. 2007, Astronomical Society of the Pacific Conference Series, 382, 75

[36] 崔辰州, 赵永恒 2004, 天文研究与技术-国家天文台台刊, 1, 3, 203

[37] Budavári, T., Dobos, L., Szalay, A. S., et al. 2007, Astronomical Data Analysis Software and Systems XVI, 376, 559

[38] 孙华平, 崔辰州& 赵永恒 2008, 天文研究与技术：国家天文台台刊, 5, 2, 130

[39] Liu, C., Tian, H.-J., Gao, D., et al. 2008, Publications of the National Astronomical Observatories of China, 5, 145

[40] 刘超, 田海俊, 高丹, 等 2008, 天文研究与技术：国家天文台台刊, 5, 2, 145

[41] 杨阳, 刘超, 田海俊, 等 2008, 天文研究与技术：国家天文台台刊, 5, 3, 234

[42] 路勇, 刘超, 崔辰州, 等 2007, 天文研究与技术：国家天文台台刊, 4, 4, 355

[43] York, D. G., Adelman, J., Anderson, J. E., Jr., et al. 2000, The Astronomical Journal, 120, 1579

[44] Ahn, C. P., Alexandroff, R., Allende Prieto, C., et al. 2012, The Astrophysical Journals, 203, 21

[45] Mazzarella, J. M., Madore, B. F., & Helou, G. 2001, Proceedings of the International Society for Optical Engineering, 4477, 20

[46] VAO Project Execution Plan, 2010, http:// www.usvao.org/ documents/ Virtual_Astronomical_Observatory_Project_Execution_Plan_v1.1.pdf

[47] Budavári, T., & Szalay, A. S. 2008, The Astrophysical Journal, 679, 301

[48] Budavári, T. 2011, Astronomical Data Analysis Software and Systems XX, 442, 79

[49] Górski, K. M., Hivon, E., Banday, A. J., et al. 2005, The Astrophysical Journal, 622, 759

[50] Gray, J., Nieto-Santisteban, M. A., & Szalay, A. S. 2007, arXiv:cs/0701171

[51] Pineau, F.-X., Boch, T., & Derriere, S. 2011, Astronomical Data Analysis Software and Systems XX, 442, 85

[52] 高丹 2008, 海量天文数据融合系统的开发与数据挖掘算法的研究, 博士学位论文, 北京: 中国科学院国家天文台

[53] Gao, D., Zhang, Y. & Zhao, Y. 2006, A system integrated with query, cross-matching and visualization, Astronomical Telescopes and Instrumentation, 627414

[54] Gao, D., Zhang, Y.-X., & Zhao, Y.-H. 2009, Research in Astronomy and Astrophysics, 9, 220

[55] 赵青 2010, 面向海量数据的高效天文交叉证认的研究. 博士学位论文, 天津大学

[56] Zhao, Q., Sun, J., Yu, C., et al. 2009, A paralleled large-scale astronomical cross-matching function, in Algorithms and Architectures for Parallel Processing, pp. 604-614, Springer Berlin Heidelberg

[57] 赵青, 孙济洲, 肖健, 等 2010, 计算机应用研究, 27, 9

[58] Budavári, T., Szalay, A. S., Gray, J., et al. 2004, Astronomical Data Analysis Software and Systems (ADASS) XIII, 314, 177

[59] Budavári, T. 2011, The Astrophysical Journal, 736, 155

[60] Egret, D., & Genova, F. 2001, Proceedings of the International Society for Optical Engineering, 4477, 216

[61] Wells, D. C., Greisen, E. W., & Harten, R. H. 1981, Astronomy & Astrophysics, Supplement, 44, 363

[62] Budavári, T., Szalay, A. S., & Fekete, G. 2010, Publications of the Astronomical Society of the Pacific, 122, 1375

[63] Gray, J., Szalay, A., & Fekete, G. 2007, arXiv:cs/0701163

[64] Szalay, A. S., Gray, J., Thakar, A. R., et al. 2002, arXiv:cs/0202013

[65] Kunszt, P. Z., Szalay, A. S., & Thakar, A. R. 2001, Mining the Sky, 631

[66] Szalay, A. S., Gray, J., Fekete, G., et al. 2007, arXiv:cs/0701164

[67] Fekete, G., Szalay, A. S., & Gray, J. 2004, Astronomical Data Analysis Software and Systems (ADASS) XIII, 314, 289

[68] Gray, J., Szalay, A. S., Thakar, A. R., et al. 2004, arXiv:cs/0408031

[69] Gray, J., Nieto-Santisteban, M. A., & Szalay, A. S. 2007, arXiv:cs/0701171

[70] Fan, D., Budavári, T., Szalay, A. S., Cui, C., & Zhao, Y. 2013, Publications of the Astronomical Society of the Pacific, 125, 218

[71] Martin, D. C., Fanson, J., Schiminovich, D., et al. 2005, The Astrophysical Journall, 619, L1

[72] Morrissey, P., Conrow, T., Barlow, T. A., et al. 2007, The Astrophysical Journals, 173, 682

[73] Adelman-McCarthy, J. K., Agüeros, M. A., Allam, S. S., et al. 2008, The Astrophysical Journals, 175, 297

[74] Fisher, R. 1953, Royal Society of London Proceedings Series A, 217, 295

[75] Breitenberger, E. 1963, Biometrika, 50, 1/2, 81 – 88

[76] 郭瑛, 艾渤, 钟章队 2011, 信息与电子工程, 9, 5, 604

[77] Marshall, A. W. 1954, The use of multi-stage sampling schemes in Monte Carlo computations, No. P-531. RAND CORP SANTA MONICA CALIF

[78] Roweis, S. 1999, Gaussian identities, University of Toronto

[79] Gales, M. J. F. & Airey, S. S. 2006, Computer Speech & Language, 20, 1, 22

[80] Press, W. H., Flannery, B. P., Teukolsky, et al. 1990, Numerical recipes

[81] Golder, E. R., & Settle, J. G. 1976, The Box-Muller method for generating pseudo-random normal deviates, Applied Statistics, 12

[82] 王瑞庆 2008, 电脑学习, 2, 9

[83] Lonsdale, C. J., Smith, H. E., Rowan-Robinson, M., et al. 2003, Publications of the Astronomical Society of the Pacific, 115, 897

[84] Norris, R. P., Afonso, J., Appleton, P. N., et al. 2006, The Astronomical Journal, 132, 2409

[85] Magliocchetti, M., Maddox, S. J., Lahav, O., & Wall, J. V. 1998, Monthly Notices of the Royal Astronomical Society, 300, 257

[86] 樊东卫, 崔辰州, 赵永恒 2011, 天文研究与技术：国家天文台台刊, 8, 3, 306

[87] Cui, C., Fan, D., Zhao, Y., et al. 2012, New Astronomy, 17, 167

[88] Bonnarel, F., Fernique, P., Bienayme, O., et al. 2000, arXiv:astro-ph/0002109

[89] Rubel, D. 2006 The heart of eclipse., Queue 4, no. 8 (2006): 36-44.

[90] Gruber, D., Hargrave, B. J., McAffer, J., et al. 2005, IBM Systems Journal, vol.44, no.2, pp.289,299

[91] Rivieres, J. D., & Beaton, W. 2006, Eclipse platform technical overview, Technical report, IBM and Eclipse Foundation

[92] Gamma, E., & Beck, K. 2004, Contributing to Eclipse: principles, patterns, and plug-ins, Addison-Wesley Professional

[93] FAQ What is the difference between a view and an editor, http:// wiki.eclipse.org/ FAQ_What_is_the_difference_between_a_view_and_an_editor%3F

[94] Hoffmann, M.R. et al., 2006, Eclipse Workbench: Using the Selection Service, Eclipse Corner Article, http:// www.eclipse.org/ articles/ Article-WorkbenchSelections/ article.html

[95] Ochsenbein, F., Williams, R., Davenhall, C., et al. 2004, Toward an International Virtual Observatory, 118

[96] Taylor, M. B. 2006, Astronomical Data Analysis Software and Systems XV, 351, 666

[97] Taylor, M. B. 2005, Astronomical Data Analysis Software and Systems XIV, 347, 29

[98] Etesi, L. I., Csillaghy, A., & Chang, L. C. 2010, In Information Reuse and Integration (IRI), 2010 IEEE International Conference on, pp. 1-6. IEEE

[99] Winer, D. 1999, XML-RPC Specification, http:// xmlrpc.scripting.com/ spec.html

[100] XML-RPC, http:// en.wikipedia.org/ wiki/ XML-RPC

[101] SOAP - Simple Object Access Protocol, http:// en.wikipedia.org/ wiki/ SOAP

[102] Schaaff, A. 2004, Astronomical Data Analysis Software and Systems (ADASS) XIII, 314, 327

[103] Nemiroff, R. J., Bonnell, J., Lowe, S. R., Connelly, P., & Haring, R. 2013, American Astronomical Society Meeting Abstracts, 221, #141.02

[104] Bonnell, J. T., & Nemiroff, R. J. 2000, Bulletin of the American Astronomical Society, 32, 1493

[105] Differences Between Multicast and Unicast, http:// support.microsoft.com/ kb/ 291786/ en-us

# 发表文章目录


[1] **Fan, Dongwei**; Budavári, Tamás; Szalay, Alexander S.; Cui, Chenzhou; Zhao, Yongheng. Efficient Catalog Matching with Dropout Detection. 2013. Publications of the Astronomical Society of the Pacific, Volume 125, issue 924, pp.218-223

[2] Cui, Chenzhou; **Fan, Dongwei**; Zhao, Yongheng; Kembhavi, Ajit; He, Boliang; Cao, Zihuang; Li, Jian; Nandrekar, Deoyani. Enhanced Management of Personal Astronomical Data with FITSManager. 2012. New Astronomy, Volume 17, Issue 2, p. 167-174.

[3] **樊东卫**; 崔辰州; 赵永恒. FITS文件管理器设计与实现, 2011, 天文研究与技术, Volume 8, No.3

[4] 李建; 崔辰州; 何勃亮; 赵永恒; 曹子皇; **樊东卫**; 李长华; 谌悦. 天文数据库回顾与展望. 2013. 天文学进展. Volume 31, No.1

[5] 崔辰州; 李建; 蔡栩; 范玉峰; 王锋; 曹子皇; 苏丽颖; **樊东卫**; 乔翠兰; 何勃亮; 李长华; 赵永恒; 谌悦; 王传军; 辛玉新; 白金明; 季凯帆. 程控自主天文台网络的发展. 2013. 天文学进展. Volume 31, No.2

# 简　　历

## 基本情况

樊东卫，男，壮族，广西壮族自治区忻城县人。1985 年 6 月出生，未婚，中国科学院国家天文台在读博士研究生。

## 教育状况

2003 年 9 月至 2007 年 6 月，广西师范大学计算机科学与信息工程学院，本科，专业：计算机科学与技术。

2008 年 9 月至 2013 年 6 月，中国科学院国家天文台，硕博连读研究生，专业：天文技术与方法。

## 研究兴趣

虚拟天文台，软件工程，天文星表交叉证认，数据挖掘，高性能计算

## 访问学习

2011.09 - 2013.02 in the group of Alexander Szalay, Physics and Astronomy Department, Johns Hopkins University, Baltimore, Maryland, USA.

## 会议

2012.11, ADASS XXII, UIUC, IL, US

2012.05, IVOA Interoperability Meeting, UIUC, IL, US

2012.04, Microsoft Research: Open Data for Open Science Workshop, Redmond, WA, US

2011.12, VAO Group Work Meeting, CalTech, CA, US

2010.12, IVOA Interoperability Meeting, Nara, Japan

2010.11, China-VO 2010, Lijiang, China

2010.03, Astronomical Information Technology Training, NAOC, Beijing,

2009.11, China-VO 2009, Chongqing, China

2009.06, Database Training, Chinese Academy Sciences, Beijing, China

## 联系方式

通讯地址：北京朝阳区大屯路甲20号 中国科学院国家天文台

邮编：100012

E-mail: fandongwei@nao.cas.cn

JOHNS HOPKINS
UNIVERSITY

**The Henry A. Rowland Department of Physics and Astronomy**

Bloomberg Center
3400 N. Charles Street
Baltimore MD 21218-2695
(410) 516-7347 / FAX (410) 516-7239

Assessment Committee
National Astronomical Observatories
Chinese Academy of Sciences

Baltimore, December 17, 2012.

Dear Sirs:

It is a pleasure to write an evaluation letter about the progress of Dongwei Fan's work at The Johns Hopkins University. Since his arrival, Dongwei has been hard at work and already finished several research projects, which cover a wide range of topics related to his PhD studies.

First Dongwei developed a new C++ toolkit that can handle the complex spherical geometries that are used in astronomy, as well as in geospatial information systems. His implementation is online and publicly available for all interested researchers around the World. Building on his strong understanding of the spherical toolkit, Dongwei was able to connect his Java-based FitsManager application to new Virtual Observatory services. The novel plugin looks inside FITS images, extracts the sky coverage information from the World Coordinate System header fields, and uploads them into the NVO Footprint Service.

Next, Dongwei created a new solution that extends and improves upon Jim Gray's Zones Algorithm for crossmatching, which can be applied to the largest astronomical catalogs. Incorporating the footprint information directly into the matching makes the processing significantly faster because one can focus on the overlapping areas only. His algorithmic solutions and their SQL implementations are to appear in the Publication of the Astronomical Society of the Pacific, a high-impact international journal. The manuscript has already received a very favorable referee report and will be published soon.

He has also presented this work at several conferences, he went to a World Wide Telescope meeting at Microsoft, also to various Virtual Observatory meeting, and has also presented a nice poster at the ADASS meeting in Urbane in Nov 2012.

Even before his previous project was completely over, Dongwei started on yet another project on probabilistic cross-identification of radio sources. He quickly learnt and applied the relevant elements of Bayesian statistics and wrote numerical integrator routines that quickly evaluate the likelihood of several scenarios. The code is currently being tested on real catalogs. This work is extremely relevant in the light of the new-generation radio surveys such as the Square Kilometer Array and its currently operating pathfinder telescopes.

In summary, Dongwei's progress at Hopkins has been exceptional, nothing short of spectacular, exceeding all of our high expectations. It was a true pleasure to work with such a talented student, who not only worked hard, produced excellent results, but has been very well liked by everyone in the group.

Sincerely,

Tamas Budavari
Associate Research Professor

Alexander Szalay
Alumni Centennial Professor

JOHNS HOPKINS
UNIVERSITY

The Henry A. Rowland Department of Physics and Astronomy

3400 North Charles Street, Bloomberg 135
Baltimore, MD 21218-2695
(410) 516-7347 Fax (410) 516-7239

February 25, 2013

To Whom It May Concern:

This letter is to verify that Dongwei Fan was a valued, full-time Visiting Ph.D. Graduate Student of the Johns Hopkins University in Baltimore, MD from September 1, 2011 through February 28, 2013. Dongwei worked under the supervision of Professor Alex Szalay in the Department of Physics and Astronomy, in a 100% research based position.

While here, Dongwei conducted research in new algorithms applied to the problem of analyzing and publishing large astronomical catalogues for the International Virtual Observatory. In particular, his research was applied to a project for the Virtual Astronomical Observatory with the goal of creating a permanent facility to integrate all astronomy data resources in the United States. Dongwei's research was also supported by funding to explore the area of data intensive computing through innovative research projects, ranging from large numerical simulations of the cochlea to genomics, astrophysics and fluid dynamics.

If you have any questions or concerns regarding Dongwei's research or status while at the Johns Hopkins University, please don't hesitate to contact me.

Sincerely,

Alexander Szalay, Ph.D.
Professor

# 致　　谢

自1990年秋季入读龙利小学学前班，没跳级，也没留级，一晃就读了23年书。想要感谢的人太多太多。

感谢我的母亲，自1998年9月10日以来，您离开这个世界已近15年。音容渐已模糊，谆谆教诲却不敢忘却。孰料因缘际会，您对孩儿的期望，已然无法达成。儿不孝，望您在另一个世界能平安喜乐。

感谢我的父亲。要供两个大学生读书实属不易。也恭喜您，培养出了村里的第一个大学生，还有第一个研究生。只是，还是要少喝些酒，少吸点烟。少干点活儿，活动活动筋骨就好，别累着了。从初一开始住校至今，离家越来越远，留在家中的时间也越来越短，心中很是不安，希望未来能常偷点闲陪陪您。

感谢赵永恒研究员，崔辰州研究员在五年研究生学习、生活给我的指导与点拨。五年前研究生入学，为寻导师懵懂走入赵老师的办公室，心中惴惴不安。赵老师却只微笑着让我坐下，短短聊上数语，便将我引见给崔老师。事实证明，赵老师果真是个伯乐，慧眼识人。于是，自2008 年我便在China-VO 开始了我的研究生生涯。崔老师亦师亦友，既有学习、工作上的帷幄导引，亦如兄长般关心我的情绪及生活琐事。感激涕零，无以言表。

感谢Johns Hopkins University的Alex Szalay及Tamás Budavári在美国18个月期间给我的指导。Alex身为天文学、计算机双料教授，学识渊博却毫无大师的架子。积极地为大家创造各种机会，有求必应，细心、平易近人得让人感动。Tamás直接指导了我在Hopkins的多项学习及工作，循循善诱，令我深受启发。由于时间关系，在美国尚未完成的项目，未来还将与他继续进行下去，希望能做出一番成绩来。同样需要感谢Ani、Deoyani、Dmitry、Jordan、Lenalee、Margie、Matthias、Tara、王鑫、杨林、叶菁华、郑政等各位同事及朋友的关心、帮助。还有在巴尔的摩遇到的各色人和事，祝各位安好，有机会再见。

感谢LAMOST团组的各位老师、同事、同学，多蒙罗阿理、张彦霞老师的指导及王丹、袁晖的照顾。一分耕耘一分收获，大家都辛苦了，望LAMOST数

据越来越好，科学产出越来越棒。

感谢China-VO全体同事对我的关爱、照顾。感谢曹子皇、谌悦、储王伟、何勃亮、李长华、李建、李正、滕一民、杨阳等等，多年来大家其乐融融的就像是一家人一样。一齐为了共同的目标而努力工作着，各尽其责，相互体谅，彼此热心帮助。

感谢研究生处杜红荣、艾华、马环宇老师对我的帮助，以及对我时有冒失之举的包容与理解。给你们添麻烦了。

感谢龙利小学、古蓬小学、忻城县中学、柳州地区民族高级中学、广西师范大学的各位老师在过去数年来的教导。虽然对其间一些老师的做法不敢苟同，然时过境迁，往事已过眼云烟。多蒙教诲，不胜感激。谨祝身体健康，讲台上站了那么多年，尤其要保护好肠胃和嗓子。

感谢国家留学基金委以及中国驻华盛顿大使馆教育处对我在美国学习、生活提供的资助与便利。彼时不曾想过能到国外走一遭，若无基金委的资助，一介困窘小子实难在外学习如此之久。一年半来，所获匪浅，还望能尽己所能回馈祖国于万一。

感谢一路走来遇到的人和事，或爱或恨，或喜或悲。往事已矣，来时可追。即便是跌跌撞撞，每走一步都是成长。儿时的梦想仍未实现，再接再厉吧。

樊东卫

2013年5月20日