

GNSS 姿态测量算法研究 及软件开发

(申请清华大学工学硕士学位论文)

培 养 单 位 ： 土木工程系

学 科 ： 大地测量学与测量工程

研 究 生 ： 邵 奎

指 导 教 师 ： 白征东 副教授

二〇一五年六月

The Algorithms and Software Development of GNSS Attitude Determination

Thesis Submitted to

Tsinghua University

in partial fulfillment of the requirement

for the degree of

Master of Science

in

Geodesy and Survey Engineering

by

Shao Kui

Thesis Supervisor : Associate Professor
Bai Zhengdong

June, 2015

摘要

姿态测量在航空航天、军事等领域有着广泛应用，早期的解决方案一般应用惯性导航系统（INS），但 INS 因其解算原理和价格因素，在实际应用中受到一定限制。随着全球导航卫星系统（GNSS）的不断发展完善，应用 GNSS 解算姿态成为一个研究热点，在我国大力推广北斗系统的战略背景下，研究更显得重要。

本文主要研究应用 GNSS 数据解算动态载体的姿态信息，重点研究了以下几个方面的问题：

（1）姿态测量原理和算法。对姿态测量中坐标系旋转问题、姿态测量原理和姿态测量算法进行了研究，提出了姿态测量中三维直角坐标旋转顺序的选取原则；

（2）高精度短基线解算。对基线解算原理、短基线误差处理和应用 Kalman 滤波解算 GNSS 基线进行了研究；

（3）姿态测量模糊度解算。对 LAMBDA 算法进行了深入剖析，探索了改进的 LAMBDA 算法和附有限制条件的 LAMBDA 算法。在此基础上，将 M-LAMBDA 与 C-LAMBDA 算法结合起来，提高了模糊度固定的成功率和效率。

在理论研究的基础上，编写了姿态测量软件，软件在 MFC 框架下使用 C/C++ 语言编写，能够使用 BDS 和 GPS 双系统数据解算运动载体三个姿态角，并实时显示解算结果。

最后，设计了测试实验和对比实验。测试实验通过接收载体的运动状态数据，分析所编写软件的精度和稳定性情况，实验表明软件姿态解算稳定度可达到 0.1 度，且具有较高稳定性。对比实验分析了姿态解算中应用不同系统、不同姿态算法的解算差异情况，实验结果显示，BDS 同 GPS 系统均具备独立测姿能力；利用 GNSS（BDS+GPS）数据能够有效提高解算精度和稳定性；不同姿态算法解算结果存在差异，并各自具备优缺点。

关键词：姿态测量；短基线解算；LAMBDA；三维直角坐标转换；GPS/BDS 对比；

Abstract

Attitude measurement is widely applied in aerospace, military and many other fields, Inertial Navigation System (INS) is the first method introduced to solve this problem. However, because of the inherent defects and expensive price, INS has a variety of problems in practical application. With the development of global navigation satellite system (GNSS), using GNSS to calculate the attitude has become a research hotspot, with a view to the booming of BDS System, it turns to be more significative.

The focus of this paper is to calculate attitude information of moving carrier with GNSS data. First of all, research of background and current situation of this field is done, then three important theoretical problems is being discussed in three independent chapter, the three important problems are:

(1) The principle and algorithm of attitude measurement. It includes coordinate transformation and the classical algorithm of attitude measurement, on the basis of which , this paper puts forward the rules of attitude coordinate transformation;

(2) Short baseline solution of high accuracy. It includes the principle of baseline solution, the error processing of short baseline solution and how to imply Kalman filter to solve GNSS baseline;

(3) The constrained OTF integer ambiguity algorithm. It includes detailed study of LAMBDA and modified, constrained algorithm derived from LAMBDA, On the basis of which, this paper combines M-LAMBDA and C-LAMBDA algorithm, by this means, higher fixed rate and efficiency are acquired.

Next, on the work of theory study, a program is developed to solve attitude with GNSS data. The program uses C/C++ language under the MFC frame, it can utilize data of both BDS and GPS and with a graphic interface, results of attitude will be presented synchronously.

At last, this paper designs some experiments, these experiments are divided into two parts, named testing experiment and comparative experiments, testing experiment is to assess the program's precision and accuracy with data gathered from moving carrier, which shows the program has high stability and precision of 0.001 rad degree; Comparative experiments focus on the different calculating results between three

attitude algorithms and two GNSS systems, from which we can conclude that both BDS and GPS possess ability to solve attitude independently with enough precision and stability, using GNSS data will optimize the result effectively; Different attitude algorithms will bring different results and possess respective applicability.

Key words: Attitude Measurement; Short Baseline Solution; LAMBDA; 3D Cartesian Coordinate Rotation; GPS/BDS Comparison;

目 录

第 1 章 绪论	1
1.1 课题背景及意义	1
1.1.1 GNSS 系统概述	1
1.1.2 利用 GNSS 测量姿态的意义	1
1.2 国内外主要研究现状	2
1.2.1 GNSS 姿态测量中整周模糊度的确定	3
1.2.2 测姿算法研究	4
1.2.3 基于 GNSS 的测姿研究	4
1.3 论文研究内容、技术路线与章节安排	5
1.3.1 论文主要研究内容及目标	5
1.3.2 研究技术路线	5
1.3.3 论文章节安排	6
第 2 章 姿态测量原理和方法	8
2.1 姿态测量中三维直角坐标系旋转研究	8
2.1.1 三维直角坐标系旋转变换	8
2.1.2 姿态测量中坐标系旋转顺序的确定	15
2.2 姿态测量模型	16
2.3 姿态测量算法	17
第 3 章 短基线解算及误差分析	21
3.1 GNSS 基线解算模型	21
3.2 短基线解算误差改正	23
3.3 应用 Kalman 滤波解算基线	26
3.3.1 Kalman 滤波基本原理	26
3.3.2 应用 Kalman 滤波求解基线	27
第 4 章 姿态解算中的模糊度求解	32
4.1 常用在航模糊度求解方法	32
4.2 LAMBDA 算法	33
4.2.1 算法准备	34
4.2.2 转换矩阵的构建	35

4.2.3 条件模糊度转换以及搜索过程	39
4.2.4 搜索范围的确定	41
4.3 改进的 LAMBDA 算法 (MODIFIED-LAMBDA)	43
4.3.1 提前置换	43
4.3.2 搜索范围的调整	44
4.4 附有限制条件的 LAMBDA 算法 (CONTROLLED-LAMBDA)	45
4.4.1 直接约束法	45
4.4.2 附有三角约束和基线长度约束的 C-LAMBDA	45
4.4.3 最小二乘约束下的 C-LAMBDA 方法	46
4.5 实例分析	51
第 5 章 程序设计与实现	53
5.1 程序内部解算设计	53
5.1.1 总体技术路线	53
5.1.2 模块技术路线	54
5.2 程序外部框架设计	56
5.2.1 程序需求分析	56
5.2.2 模块功能设计	57
5.2.3 程序界面设计	58
第 6 章 实验研究与分析	62
6.1 实验目的和设计	62
6.2 测试实验	63
6.2.1 准动态实验	64
6.2.2 动态实验	66
6.3 对比实验	69
6.3.1 GPS/BDS 系统对比实验	69
6.3.2 姿态角算法对比实验	72
第 7 章 总结与展望	75
7.1 工作总结	75
7.2 工作展望	76
参考文献	77
致 谢	80

目 录

声 明	81
个人简历、在学期间发表的学术论文与研究成果	82

第1章 绪论

1.1 课题背景及意义

1.1.1 GNSS 系统概述

二十世纪以后，地理信息对于人类发展愈发重要。随着技术的不断发展，导航定位在范围上已经从区域发展到全球，观测方式上从最初的天文观测发展到电磁波观测，精度也从公里级跨向了当前的毫米级。当代全球导航卫星系统（GNSS: Global Navigation Satellite System）发展迅速，正在不断实现由单系统到多系统，地基增强到空基增强的飞跃。

目前，覆盖全球的卫星导航定位系统有四个，分别是美国的 GPS、我国的 BDS、欧盟的 GALILEO 和俄罗斯的 GLONASS。除此之外，区域性的地基和空基增强系统也纷纷建成，如日本的 QZSS（Quasi-Zenith Satellite System）、欧盟的 EGNOS（European Geostationary Navigation Overlay Service），美国的 WAAS（Wide Area Augmentation System）等。结合本文的研究内容，这里简要介绍我国的 BDS 和美国的 GPS。

GPS 是由美国国防制图局发起，喷气推进实验室（JPL）策划执行的全球导航卫星系统。系统建成于 1993 年，在全球布有观测站和注入站，经过二十多年的运行，目前系统状态良好。商业竞争和技术发展使 GPS 硬件接收设备快速发展，随着定轨技术、地基空基增强系统、数学处理算法的不断改进，GPS 可以完成米级精度的单点定位和毫米级精度的相对定位，在众多领域中得到了广泛的应用。

从上世纪八十年代起，我国就开始计划研发自主的北斗卫星导航系统。目前，已经完成北斗三步走战略中的两步，即有源被动北斗一代系统（2003 年），区域性服务的北斗二代系统（2012 年），并计划在 2020 年完成全球性布局。

北斗系统因其特殊的设计，在我国范围内有着更好的信号覆盖和卫星几何图形，具有比 GPS 更大的优势。在应用推广方面，已经有高精度的用户接收机面世，并形成了一定的市场，随着政府大力推广，未来前景广阔。

1.1.2 利用 GNSS 测量姿态的意义

姿态测量在航天航空、航海和雷达测量领域有着广泛应用，其研究历史早于

全球定位系统，但广泛受到重视是在卫星导航系统逐步成熟之后。

姿态测量最早应用在航天领域，作为一种必要的信息获取手段^[1]。如对人造卫星进行姿态测量可以调整卫星姿态，使之改变自身位置朝向，更好的接收太阳能和地面指令信号；又如航空摄影测量中，必须精确获取曝光时刻空中相机的位置和姿态信息，以便为后面的数据处理及成像工作提供所需的足够精度参数；此外，航海领域中也需要对船体测量姿态；军事装备中的飞机、坦克和装甲车，民用中的越野车、地面测量系统等也有相关需求。可以说，随着技术和需求不断发展，姿态测量已逐渐从科技领域和军事领域推广到民用领域，并逐步走进人们生活当中。

早期的姿态测量技术主要应用惯性导航系统（INS），INS 是通过实时获得载体加速度并不断积分来求取载体瞬时的速度和位置信息。惯性导航系统具有以下几个优势^[2]：

- （1）导航信号在时间上具有连续性；
- （2）导航与电磁波无关，可以避免外界电磁干扰；
- （3）不依靠外界的信息，具有隐蔽性。

但 INS 受限于其数学原理，具有误差随观测时间累积的严重问题。同时，高精度 INS 结构复杂，价格昂贵。

近些年来，利用 GPS 载波相位差分技术解算高精度基线得到了较快发展，使得通过 GPS 来搭建姿态测量系统成为可能，并逐渐成为 GPS 领域的又一个研究热点。同传统利用 INS 测姿相比，GPS 测量姿态具有更短的初始时间、更低的价格，在硬件方面还具有无漂移、体积小、重量轻、功耗低等特点。

GNSS 的不断发展使得在很多领域内，利用 GNSS 数据替代单一的 GPS 数据成为更好的选择。在测绘领域，决定测量精度的因素主要有两个，冗余观测数量和观测数据质量。在 GNSS 定位领域，这两个因素直接表现为接收到的卫星数量，卫星数量越多，多余观测量就越充足，同时卫星构成的几何形状也越好，从而几何精度因子（GDOP: Geometric Dilution of Precision）越小，观测数据质量也就越好。所以，一般情况下，观测卫星数量越充足，得到的测量结果精度越高，利用多系统 GNSS 作为数据源，一方面能够显著提高卫星数量，得到更好的计算结果和更高的稳定性。另一方面，北斗因其特殊设计在我国地区有着更好的信号覆盖，且地理信息数据对于国防安全，国民经济和建设至关重要，故利用北斗系统进行导航和姿态测量是必然趋势。

1.2 国内外主要研究现状

GPS 最初主要被用来测量时间和位置,但很早就有学者提出可以利用它来测量运动载体姿态^[3, 4]。在最初阶段, GPS 硬件发展并不迅速,而且价格相对惯导系统并无优势,因此那时利用 GPS 进行姿态测量仅限于仿真研究^[5, 6]。在 1986 年, Evans 利用实测 GPS 载波相位观测值计算得到了载体平台的姿态值,但仅做了简单探索^[7]。1988 年,首台利用 GPS 测量姿态的接收机制造成功,并在海上进行了动态测试,测试天线间距为 40-60cm,获得了较理想的精度^[8]。清华大学利用 GPS 进行姿态测量始于 1999 年,主要是航天器的姿态确定,详细见文章^[9, 10]。

利用 GNSS 数据求解姿态角,需要解决的核心问题有三个,第一个为超短基线的高精度解算,其核心在于在航模糊度求解;第二个为姿态角算法研究,其核心在于算法模型;第三个为多系统数据处理。

下面将针对这三方面分别阐述研究现状。

1.2.1 GNSS 姿态测量中整周模糊度的确定

GNSS 姿态解算的核心之一是短基线的实时解算,而短基线解算的核心是模糊度求解。按接收机状态划分,模糊度解算可以分为静态模糊度解算和动态模糊度解算,相对于静态解算,动态解算中位置参数同模糊度参数高度相关,无法有效剥离,并且由于载体运动,难以通过平滑方法消除环境误差。在 GNSS 测姿领域,模糊度解算是重点也是热点。许多学者都进行了相关研究,如 Y. Li 等提出了一种单差结合双差的算法解算整周模糊度,研究了单差无法解求整周模糊度,而双差由相关性产生的噪声太大的问题^[11],但计算过程较为繁琐。Chansik Park 等利用限制条件改进 LAMBDA 方法,提高了姿态解算精度^[12, 13]。Teunissen 亦应用附有约束条件的 LABMDA 方法解算整周模糊度,并在此基础上进行姿态计算,他分别设置静态试验,机载试验对算法进行验证,结果表明,附有约束的 LAMBDA 算法在解算稳定性和成功率方面存在优势^[14]。Giorgi 等对有限制条件的最小二乘法作了研究,并指出基于单频的模糊度固定在可靠性和解算速度上有特殊困难,文章通过迭代收缩的方法,有效收缩了整周模糊度搜索空间,提高了搜索效率^[15]。Ronald 和 Brown 同样尝试将已知基线约束引入到姿态角求取中,以进行连续时刻的姿态角求解^[16],但该方法需要已知多普勒观测值。Crassidis 提出了一种在没有先验精度情况下求解整周模糊度的方法,通过实验验证,解算结果良好^[17]。国内方面,Bo Wang 提出一种利用姿态测量中接收机特殊形状约束来辅助整周模糊度的解算方法,并将其应用到 LAMBDA 算法的改进中^[12]。刘根友也对 LABMDA 算法进行改进,提出具有坐标函数约束的单历元阻尼 LABMDA 算法^[18]。郑庆辉博士对基于白化分解和遗传算法的整周模糊度算法进

行了研究^[19]，但实际应用中不能保证足够的稳定性。

1.2.2 测姿算法研究

受载体限制，姿态解算中基线一般不会很长，在 GNSS 处理上，一般认为属于超短基线，超短基线处理中，基线两端的接收机在接收卫星数量及所受误差影响上都呈现相似性，这使得各种误差通过差分方法能够得到很好消除，从而提高基线解算精度。姿态角的解算将在基线求解的基础上进行，受基线精度影响较大。

姿态角的解算算法大致可以分为以下几种^[19]：

(1)直接法。此方法不需要已知载体基线长度，可直接利用观测值向量进行求解。直接法受基线布局限制，不能同时利用所有观测基线的先验信息^[20]。

(2)最优估计方法。这是一类方法的总称，针对 Wahba 问题^[21]提出，一般需要迭代求取最优解，也可通过非迭代方法求次优解。当观测量足够大，通过建立一种代价函数，实现姿态元素的最优估计。这类方法包括 QUEST 算法，FOMA 算法，Euler-q 算法，EAA 算法，REQUEST 算法等^[22, 23]。

(3)最小二乘法。采用经典最小二乘理论来求解姿态矩阵或直接求解姿态角。如姿态矩阵最小二乘法，姿态角最小二乘法。Gang Lu 博士论文中对应用最小二乘法求解姿态进行了研究^[24]。

(4)滤波算法研究。通过卡尔曼滤波求解姿态角，数学本质同最小二乘等价。王潜心提出了一种基于自适应抗差卡尔曼滤波的 GPS 定姿方法^[25]。郑庆辉博士对基于白化分解和遗传算法的整周模糊度算法进行了研究，并使用卡尔曼滤波进行姿态解算^[19]。

1.2.3 基于 GNSS 的测姿研究

现有的姿态测量系统和研究基本上以 GPS 数据为数据源。本文研究利用 BDS 和 GPS 双系统数据进行姿态测量，即基于 GNSS 数据的姿态测量。北斗作为我国独立自主的导航系统，利用北斗数据作为解算数据源将成为一个必然趋势。目前，使用北斗数据测姿还未得到充分研究，其精度和稳定性都还待评估。周健等使用北斗数据在船舶上进行了实验^[26]，但只进行了简单性的尝试。孟骞等使用卫星工具包对北斗单天线进行了仿真测量^[27]。Teunissen 对三系统（BDS、GALILEO 和 GPS）组合数据测姿进行了仿真研究，得到了较为稳定的预期结果^[28]。清华大学的王园、元荣等对二代北斗的基线解算进行了研究，研究表明北斗系统有着较高精度，具有实际应用能力^[29]。

1.3 论文研究内容、技术路线与章节安排

在既有的研究中,研究的内容主要为测姿产品的设计与制造,偏于硬件方面,而对理论和算法等方面缺少系统的论述,本文尝试对 GNSS 测姿理论进行研究和梳理,包括附有限制条件的模糊度求解、姿态测量中的三维直角坐标系旋转选择问题等,然后在此基础上,编写一个姿态测量程序。同时,既有研究中使用的数据主要为 GPS,而利用 BDS 数据求解姿态的研究很少,本文将利用 BDS 数据解算姿态,并将两系统的解算结果进行对比研究,以分析 BDS 在测姿应用中的精度和稳定性情况,同时,研究还会探索 GNSS (BDS+GPS) 对姿态解算的影响。针对姿态测量的特殊性,在对现有整周模糊度算法充分研究后,本文结合使用了改进的 LAMBDA 算法 (M-LAMBDA) 和附有限制条件的 LAMBDA 算法 (C-LAMBDA) 解算模糊度,从而提高整周模糊度固定的成功率和效率。

1.3.1 论文主要研究内容及目标

本论文将着重于测姿系统算法和软件方面的研究,对姿态测量中的整周模糊度求解、测姿算法等核心问题进行探索,旨在研究利用 BDS / GPS 数据求解运动载体姿态角,并形成一套利用 BDS / GPS 双系统的高精度姿态测量软件。

论文主要研究内容如下:

(1) 将对姿态测量中的三维直角坐标转换进行深入研究,包括分析常见的转换方式和研究常见方法的衍生转换方式,然后在此基础上,探索姿态测量中坐标旋转的选取原则。

(2) 将研究直接法、矩阵最小二乘法和 TRIAD 三种姿态算法,并对这几种算法进行了实验对比分析。

(3) 将研究短基线解算模型和误差消除,并探讨应用 Kalman 滤波求解基线的原理与模型;

(4) 将深入研究 LAMBDA 算法,并针对姿态测量的特殊条件,有效结合 M-LAMBDA 和 C-LAMBDA 算法,提高了模糊度固定的成功率和效率。

(5) 将对比分析 BDS 和 GPS 双系统数据和多种姿态算法在姿态测量上的应用情况,并探索 GNSS 对姿态解算的影响。

(6) 将设计实验分析所编写软件的精度和稳定性情况。

1.3.2 研究技术路线

针对本文的研究内容,拟定如下研究方案,见图 1-1:

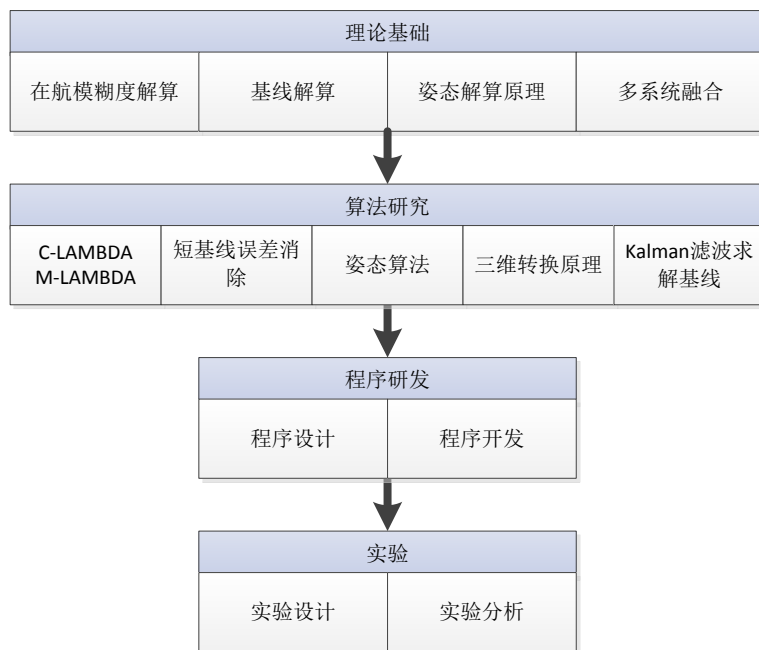


图 1-1 研究技术路线图

首先，对姿态测量的理论基础进行了研究，研究从基线解算和姿态测量原理两方面入手，前者涉及到三个问题需要理清，分别是在航模糊度解算、基线误差消除和双系统组合解算，后者需要解决两个问题，即测姿原理和测姿算法，研究主要通过阅读文献和开源代码入手，在研究过程中，逐步遴选出需要着重解决的问题。随后，对其中的几个具体问题进行深入研究，这几个问题包括附有限制条件的 LAMBDA 算法、姿态三维旋转变换等。其次，基于前面的研究内容，编写了测姿软件。最后，针对研究内容设计多个实验对软件和所研究的问题进行测试，并对实验结果进行分析和总结。

1.3.3 论文章节安排

根据研究内容，论文共分七章。

第 1 章对课题研究背景及意义、国内外研究现状、研究目标和内容进行介绍。

第 2 章介绍姿态测量原理、姿态测量算法和姿态测量中坐标变换选取原则。

第 3 章对基线解算模型和超短基线的误差消除进行了研究，并对应用 Kalman 滤波解算动态基线作了较为详细的论述。

第 4 章重点研究了 LAMBDA 及其改进算法，在此基础上，探索了融合 M-LAMBDA 和 C-LAMBDA 的整周模糊度固定方式。

第 5 章介绍了“GNSS 姿态测量软件”的设计方案以及主要功能。

第 6 章设计了多个实验，探讨了 BDS 和 GPS 数据在姿态测量中的表现以及不同测姿算法在解算上的差异。

第 7 章总结了全文研究成果，并对未来的研究进行展望。

第2章 姿态测量原理和方法

利用 GNSS 求解姿态需要涉及到不同三维直角坐标系的旋转变换，三维直角坐标系间的转换方法有多种，本章在分析这些转换方法的基础上，提出了针对姿态测量的具体情况来选择旋转方式的几条原则。除此之外，本章还对三天线 GNSS 姿态测量模型和一些经典的姿态测量算法进行了论述。

2.1 姿态测量中三维直角坐标系旋转研究

2.1.1 三维直角坐标系旋转变换

GNSS 直接测量所得基线属于地心坐标系，若要求解姿态角，首先要将其旋转到站心直角坐标系下，然后再完成由站心到载体坐标系的旋转，转换后的站心坐标系和载体坐标系重合。

通常情况下，姿态测量所使用的三维直角坐标系都是右手坐标系。假设有两个三轴指向均不同的三维直角坐标系，需要通过旋转变换来使他们相互重合，如图 2-1 所示，旋转过程中的旋转角度就是两个坐标系的姿态差异，也就是姿态测量中需要求解的姿态角。但并不是所有能够使两坐标系重合的旋转都能直接反映姿态信息，只有依次绕三个旋转轴进行旋转才能反映两个坐标系之间的姿态信息。实际上，使两个三维直角坐标系重合到一起，并不需要一定借助三轴的依次旋转，这种情况会在下面详细叙述。同时，由于坐标轴有对称轮换性，通过一种绕轴顺序进行旋转变换，也一定能对称地利用其它旋转轴达到这个效果，这就导致旋转顺序上我们有多种选择，需要综合考虑各种因素来选择适合姿态求解的坐标轴旋转顺序。下面首先对三维直角坐标系的变换问题进行研究。

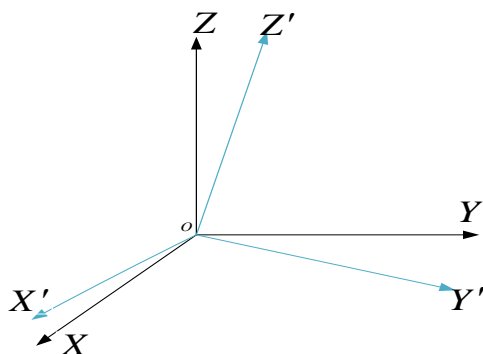


图 2-1 三轴指向不同的两三维直角坐标系

如图 2-1, 设有两个空间直角坐标系 $O-X'Y'Z'$ 和 $O-XYZ$, 它们的坐标原点 O 重合, 并均为右手坐标系, 但相应的坐标轴指向不一致。为了将 $O-X'Y'Z'$ 坐标系旋转至同 $O-XYZ$ 坐标系重合, 需依次绕 $O-X'Y'Z'$ 的某一坐标轴旋转, 旋转过程中其余两轴会随着这个旋转轴旋转一定角度, 通过三次的绕轴旋转, 才能够使两坐标系的三个坐标轴指向相同。

为了方便说明, 本文约定俯视旋转轴时逆时针方向旋转为正, 顺时针方向为负。下文将介绍两类方法将两个不重合的三维直角坐标系旋转重合, 然后根据对称轮换性讨论两类方法的等效旋转方式, 在此基础上, 将提出姿态测量中三维直角坐标系旋转方式的选取原则。

在三维直角坐标系下, 绕坐标轴的旋转可以通过与对应的旋转矩阵相乘实现。绕 X, Y, Z 的旋转矩阵分别表示为,

$$\begin{aligned} R_x &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \delta_x & \sin \delta_x \\ 0 & -\sin \delta_x & \cos \delta_x \end{pmatrix} \\ R_y &= \begin{pmatrix} \cos \delta_y & 0 & -\sin \delta_y \\ 0 & 1 & 0 \\ \sin \delta_y & 0 & \cos \delta_y \end{pmatrix} \\ R_z &= \begin{pmatrix} \cos \delta_z & \sin \delta_z & 0 \\ -\sin \delta_z & \cos \delta_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \quad (2.1)$$

δ_x 为绕 X 轴的旋转角, R_x 为绕 X 轴的旋转矩阵, 其他相似符号可以此规则类推。

有两种方式能够实现这种旋转^[30], 这两种方式具有不同的绕轴顺序, 但具有相同的旋转效果。

方式一: ZXZ 式。 ZXZ 分别表示绕对应的坐标轴旋转, 也就是通过依次绕坐标轴 Z' 、 X' 和 Z' 进行旋转, 使 $O-X'Y'Z'$ 和 $O-XYZ$ 三轴指向一致。

取 XOY 和 $X'OY'$ 两坐标平面的交线 OX'' , 如图 2-2, 它与 OX' 轴的夹角为 δ_{z1} , 首先以 Z' 为旋转轴旋转 δ_{z1} , 使 OX' 轴旋转至 OX'' , 这时 OY' 轴旋转至 OY'' 处。如图 2-3。

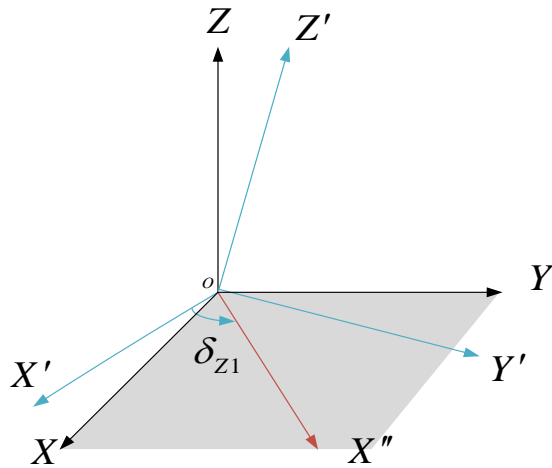


图 2-2 方式一的第一次旋转

OX'' 同时处于 XOY 和 $X'OY'$ 平面，故其同时垂直于 OZ' 和 OZ ，也就垂直于 OZ' 和 OZ 构成的平面，以 OX'' 为轴，旋转角度 δ_x ，使 OZ' 和 OZ 重合，此时 OY'' 轴再次旋转至 OY''' 。如图 2-4。

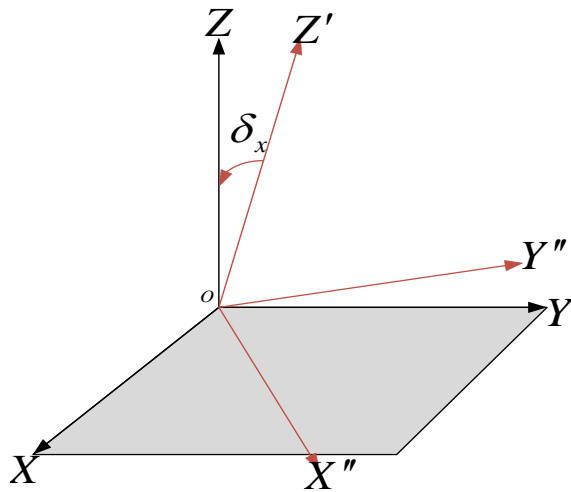


图 2-3 方式一的第二次旋转

此时 OZ' 和 OZ 均与平面 XOY 和 $X''OY'''$ 垂直，则有 XOY 和 $X''OY'''$ 共面，此时绕 OZ' （也即 OZ ）轴旋转角度 $-\delta_{z2}$ ，使 OX'' 同 OX 重合，那么 OY''' 也必定重合与 OY 。如图 2-4，

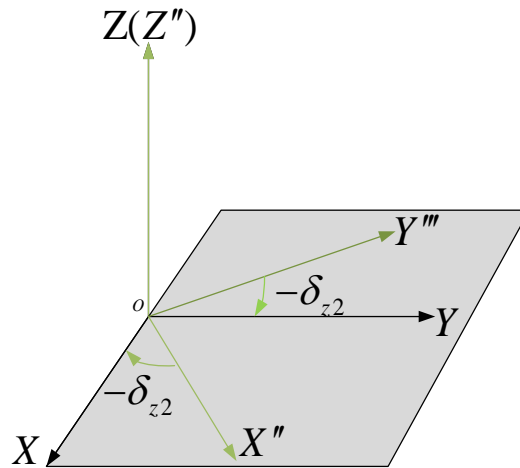


图 2-4 方式一的第三次旋转

上述过程可用式 (2.2) 表达,

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R_z(\delta_{z2})R_x(\delta_x)R_z(\delta_{z1})\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} \quad (2.2)$$

方式二: XYZ 式。第一种方式从 XOY 和 $X'OY'$ 两坐标平面的交线入手, 通过绕 X 轴的一次旋转和绕 Z 轴的两次旋转使两坐标轴重合。接下来的这种方式首先绕从 OX' 轴着手, 如图 2-5, 设 $Y'OZ'$ 与 XOY 平面相交于 OY'' , 首先绕 OX' 旋转 δ_x 将 OY' 旋转至 OY'' , 此时 OZ' 旋转至 OZ'' , 如图 2-6。

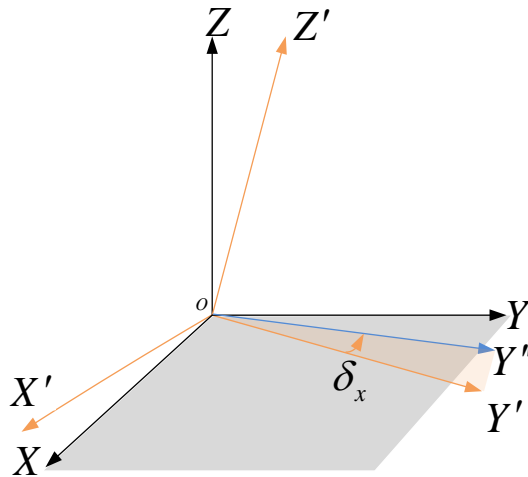


图 2-5 方式二的第一次旋转

旋转后 OY'' 同时处于 XOY 和 $X'OY'$ 平面, 故其同时垂直于 OZ'' 和 OZ , 也就垂直于 OZ'' 和 OZ 构成的平面 ZOZ'' , 如图 2-6, 以 OY'' 为轴, 旋转角度 δ_y , 使

OZ'' 和 OZ 重合, 此时 OX' 轴旋转至 OX'' 位置, 如图 2-7。

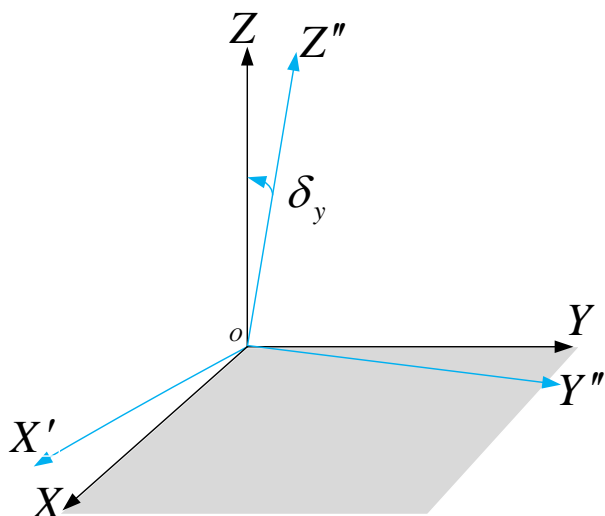


图 2-6 方式二的第二次旋转

由于 OZ'' 同 OZ 重合, 可以断定此时 $X''OY''$ 同 XOY 平面重合, 以 OZ'' (即 OZ) 为轴, 旋转角度 δ_z , 使 OX'' 重合于 OX , 则此时 OY'' 必定重合于 OY 轴, 如图 2-7。

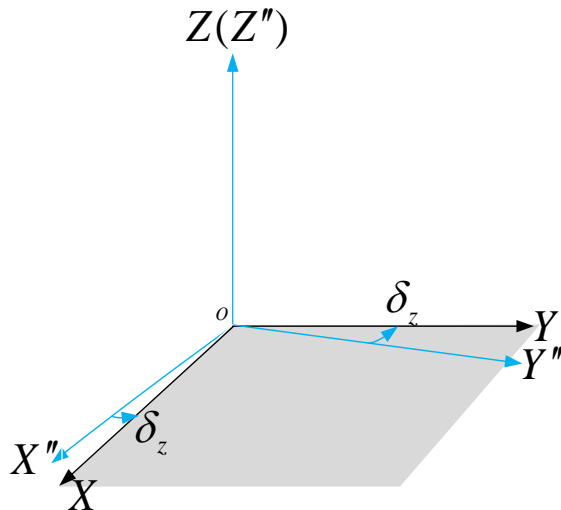


图 2-7 方式二的第三次旋转

上述过程可以表达为,

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = R_z(\delta_z)R_y(\delta_y)R_x(\delta_x) \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} \quad (2.3)$$

需要指出的是，无论是方法一还是方法二，旋转的角度都不是两坐标系对应坐标轴之间的夹角，一般情况下，旋转角小于对应的坐标轴夹角。在姿态测量中，我们并不需要知道坐标轴之间的夹角，姿态角被认为是旋转角。

令 α 为 OX' 和 OX 坐标轴之间的夹角， β 为 OY' 和 OY 坐标轴之间的夹角， γ 为 OZ' 和 OZ 坐标轴之间的夹角。

则方法一中，坐标轴夹角同旋转角的关系可以表示为^[30]，

$$\begin{aligned}\cos \alpha &= \cos \delta_{z1} \cos \delta_{z2} + \sin \delta_{z1} \sin \delta_{z2} \cos \delta_x \\ \cos \beta &= \sin \delta_{z1} \sin \delta_{z2} + \cos \delta_{z1} \cos \delta_{z2} \cos \delta_x \\ \cos \gamma &= \cos \delta_x\end{aligned}\quad (2.4)$$

在方法二中，坐标轴夹角同旋转角的关系可以表示为，

$$\begin{aligned}\cos \alpha &= \cos \delta_y \cos \delta_z \\ \cos \beta &= \cos \delta_x \cos \delta_z + \sin \delta_x \sin \delta_y \sin \delta_z \\ \cos \gamma &= \cos \delta_x \cos \delta_y\end{aligned}\quad (2.5)$$

接下来讨论坐标旋转的对称轮换性问题，在方法一种，我们首先以 OZ' 为轴，选取 XOY 和 $X'OY'$ 的交线 OX'' ，将 OX' 旋转至 OX'' ，但如果将 OY' 而不是 OX' 旋转至 OX'' ，也可以得到相似的旋转结果，接下来，依然可以通过同样的旋转实现两坐标系的重合，现将这种方式称为方法一的替换方法。

无论是方法一还是其替换方法，我们都是首先以 OZ' 为轴，寻找 XOY 和 $X'OY'$ 的交线，然后进行旋转，根据对称轮换性，我们还可以首先以 OX' 为轴，寻找 YOZ 和 $Y'OZ'$ 的交线进行旋转，这种旋转同方法一相似，也有两种方式，即将 OY' 或 OZ' 旋转至 YOZ 和 $Y'OZ'$ 的交线处。同样的，也可以首先选择以 OY' 为轴，构造出两种旋转方式。

如上，方法一共有六种变换方式，如果以旋转轴为顺序，这六种方式可以表示为，

$$\begin{aligned}OZ' - OX' - OZ' \\ OZ' - OY' - OZ' \\ OX' - OZ' - OX' \\ OX' - OY' - OX' \\ OY' - OX' - OY' \\ OY' - OZ' - OY'\end{aligned}\quad (2.6)$$

在方法二中，我们首先以 OX' 为轴，选取 XOY 和 $Y'OZ'$ 的交线 OY'' ，将 OY'

旋转至 OY'' ，但如果我们选取 XOZ 和 $Y'OZ'$ 的交线 OZ'' ，将 OZ' 旋转至 OZ'' ，接下来再以 OZ'' 为轴旋转，使 OY'' 和 OY 重合，最后再以 OY'' 为轴旋转，使三轴指向一致，这种方式依然可以实现两坐标系的重合，现将这种方式称为方法二的替换方法。

方法二及其替换方法，都是首先以 OX' 为轴，寻找 XOY 或 XOZ 与 $Y'OZ'$ 的交线，然后进行旋转的，根据对称轮换性，我们还可以首先以 OY' ，寻找 YOZ 或 XOY 与 $X'OZ'$ 的交线进行旋转，这样也可以有两种方式，即将 OX' 或 OZ' 旋转至 XOY 与 $X'OZ'$ 或 YOZ 与 $X'OZ'$ 的交线处。类似的，也可以首先选择以 OZ' 为轴，构造出两种旋转方式。

这样，我们又得到六种旋转方式和它们对应的旋转矩阵。为方便后面分析，下面直接以 y, p, r 表示绕 OZ' 、 OX' 和 OY' 的旋转角度，构造出六种方式及其对应的旋转矩阵，在后面叙述中，可知 y, p, r 即为偏航角、俯仰角和翻滚角，六种旋转方式和其对应的旋转矩阵分别是，

$OX' - OY' - OZ'$

$$R_{xyz} = \begin{pmatrix} \cos(r)*\cos(y) & \cos(p)*\sin(y) + \cos(y)*\sin(p)*\sin(r) & \sin(p)*\sin(y) - \cos(p)*\cos(y)*\sin(r) \\ -\cos(r)*\sin(y) & \cos(p)*\cos(y) - \sin(p)*\sin(r)*\sin(y) & \cos(y)*\sin(p) + \cos(p)*\sin(r)*\sin(y) \\ \sin(r) & -\cos(r)*\sin(p) & \cos(p)*\cos(r) \end{pmatrix}$$

$OX' - OZ' - OY'$

$$R_{xzy} = \begin{pmatrix} \cos(r)*\cos(y) & \sin(p)*\sin(r) + \cos(p)*\cos(r)*\sin(y) & \cos(r)*\sin(p)*\sin(y) - \cos(p)*\sin(r) \\ -\sin(y) & \cos(p)*\cos(y) & \cos(y)*\sin(p) \\ \cos(y)*\sin(r) & \cos(p)*\sin(r)*\sin(y) - \cos(r)*\sin(p) & \cos(p)*\cos(r) + \sin(p)*\sin(r)*\sin(y) \end{pmatrix}$$

$OY' - OX' - OZ'$

$$R_{yxz} = \begin{pmatrix} \cos(r)*\cos(y) + \sin(p)*\sin(r)*\sin(y) & \cos(p)*\sin(y) & \cos(r)*\sin(p)*\sin(y) - \cos(y)*\sin(r) \\ \cos(y)*\sin(p)*\sin(r) - \cos(r)*\sin(y) & \cos(p)*\cos(y) & \sin(r)*\sin(y) + \cos(r)*\cos(y)*\sin(p) \\ \cos(p)*\sin(r) & -\sin(p) & \cos(p)*\cos(r) \end{pmatrix}$$

$OY' - OZ' - OX'$

$$R_{xzy} = \begin{pmatrix} \cos(r)*\cos(y) & \sin(y) & -\cos(y)*\sin(r) \\ \sin(p)*\sin(r) - \cos(p)*\cos(r)*\sin(y) & \cos(p)*\cos(y) & \cos(r)*\sin(p) + \cos(p)*\sin(r)*\sin(y) \\ \cos(p)*\sin(r) + \cos(r)*\sin(p)*\sin(y), & -\cos(y)*\sin(p) & \cos(p)*\cos(r) - \sin(p)*\sin(r)*\sin(y) \end{pmatrix} \quad (2.7)$$

$$OZ' - OX' - OY'$$

$$R_{zxy} = \begin{pmatrix} \cos(r)*\cos(y) - \sin(p)*\sin(r)*\sin(y) & \cos(r)*\sin(y) + \cos(y)*\sin(p)*\sin(r) & -\cos(p)*\sin(r) \\ -\cos(p)*\sin(y) & \cos(p)*\cos(y) & \sin(p) \\ \cos(y)*\sin(r) + \cos(r)*\sin(p)*\sin(y) & \sin(r)*\sin(y) - \cos(r)*\cos(y)*\sin(p) & \cos(p)*\cos(r) \end{pmatrix}$$

$$OZ' - OY' - OX'$$

$$R_{zyx} = \begin{pmatrix} \cos(r)*\cos(y) & \cos(r)*\sin(y) & -\sin(r) \\ \cos(y)*\sin(p)*\sin(r) - \cos(p)*\sin(y) & \cos(p)*\cos(y) + \sin(p)*\sin(r)*\sin(y) & \cos(r)*\sin(p) \\ \sin(p)*\sin(y) + \cos(p)*\cos(y)*\sin(r) & \cos(p)*\sin(r)*\sin(y) - \cos(y)*\sin(p) & \cos(p)*\cos(r) \end{pmatrix}$$

续(2.7)

2.1.2 姿态测量中坐标系旋转顺序的确定

通过上面分析可知，将三轴指向不同的两个三维直角坐标系旋转至指向相同，共有十二种方式，不同的旋转方式所对应的旋转角度存在差异，但差异并不大，在既有的研究中，并没有统一形式。

为了选择一个最适合测姿的旋转方式，首先应该明确测姿对于矩阵旋转的要求，然后根据这些要求选择旋转方式。本文中，提出以下三个原则来确定姿态求解中三维直角坐标系旋转方式，

(1)因为需要测量三个姿态角，旋转过程需分别绕三轴旋转，由此第一种方式及其可替换方式都不符合要求，即式(2.6)不符合要求；

(2)依据接收机的部署，旋转过程需要选择数学上容易解算的旋转方式。如图2-8，本文中使用三台接收机部署测姿系统，一般在 OY 轴部署一条基线，这条基线的载体坐标可以表示为 $(0 \ Y_{b1} \ 0)^T$ ，结合式(2.10)，不难发现如果要简洁的解算出姿态角，式(2.7)中旋转矩阵的第二列或第二行要尽可能简单，这样与 $(0 \ Y_{b1} \ 0)^T$ 相乘后，方程才易于解算。式(2.7)中有四种旋转方式符合，分别是，

$$\begin{aligned} & OX' - OZ' - OY' \\ & OY' - OX' - OZ' \\ & OY' - OZ' - OX' \\ & OZ' - OX' - OY' \end{aligned} \tag{2.8}$$

(3)旋转的过程需要符合人的一般认知过程，大多数时候，会把天顶方向定为 Z 方向，载体的前进方向设定为 Y 方向， X 方向与两轴成右手坐标系，无论是在地图中还是对移动载体的跟踪过程中，一般会首先选择天顶方向观察载体，也

就是从 Z 方向观察载体，其次会关心载体前进方向的俯仰情况，也就是从 X 方向观察载体，最后我们会关心载体侧向的起伏，也就是 Y 方向的旋转角度。对应这样一个认知过程，通常我们选择 $OZ'-OX'-OY'$ 作为旋转顺序，它对应的旋转矩阵是，

$$R_{Zxy} = \begin{pmatrix} \cos(r)*\cos(y) - \sin(p)*\sin(r)*\sin(y) & \cos(r)*\sin(y) + \cos(y)*\sin(p)*\sin(r) & -\cos(p)*\sin(r) \\ -\cos(p)*\sin(y) & \cos(p)*\cos(y) & \sin(p) \\ \cos(y)*\sin(r) + \cos(r)*\sin(p)*\sin(y) & \sin(r)*\sin(y) - \cos(r)*\cos(y)*\sin(p) & \cos(p)*\cos(r) \end{pmatrix} \quad (2.9)$$

2.2 姿态测量模型

姿态测量是对载体的角度测量，一般是在笛卡尔三维直角坐标系下，通过测量相对于三个直角坐标轴的旋转角度来确定载体的姿态。通常以载体几何中心为原点建立一个指向不变的三维笛卡尔坐标系，则载体的偏航角、俯仰角和翻滚角都可以参照这个笛卡尔坐标系计算出来。

GNSS 姿态系统可以使用不同数量的接收机和不同的布局搭建，实际中可根据具体情况选择不同的接收机布局，不同的布局选择将直接影响到计算模型和求解公式。本文选用三台接收机构成双基线姿态测量系统，主接收机位于载体坐标系原点，从一接收机位于载体坐标系 Y 轴，从二接收机位于 XOY 平面，与 Y 轴所夹角为 θ ，如图 2-8。一般情况下，载体坐标系原点与运动载体的相对位置保持不变， Y 轴与行进方向一致， Z 轴指向天顶， X 轴与 YZ 轴组成右手坐标系。

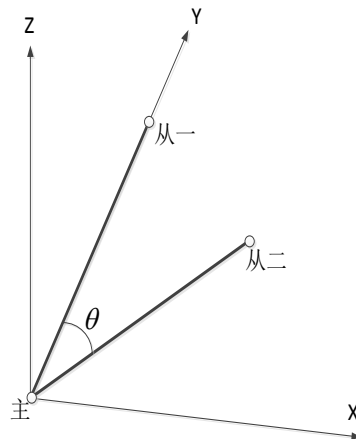


图 2-8 接收机布局

通过 GNSS 直接测量所得基线属于地心坐标系，若要求解姿态角，首先要将

其旋转到站心直角坐标系下，这一步需要首先求取主站的大地坐标，具体可参照^[20]。然后再完成由站心到载体坐标系的旋转，第二次旋转将通过绕坐标轴的三次旋转实现，三次的旋转角分别为偏航角 *YAW*，俯仰角 *PITCH* 和翻滚角 *ROLL*，转换后的站心直角坐标系和载体坐标系重合。

载体到站心坐标系的旋转可用如式 (2.10) 表示：

$$X_l = R_{b2l} \times X_b \quad (2.10)$$

其中， R_{b2l} 为 3×3 转换矩阵， X_l 为站心坐标系坐标，由地心空间直角坐标旋转而来。 X_b 为载体坐标系，因接收机固定在载体上，其值已知且不随时间改变，根据上文描述， R_{b2l} 、 X_l 和 X_b 具体形式如下^[31]。

式(2.10)也可以写成 $X_b = R_{l2b} \times X_l$ ，其中 R_{l2b} 和 R_{b2l} 互为转置。在实际解算中，使用这两种形式解算结果相同。

完成基线解算后， X_l 变为已知，在式 (2.10) 中未知量只剩下 R_{b2l} 中的三个姿态角 y, p, r 了，可以通过数学方法将它们解算出。

2.3 姿态测量算法

目前的姿态算法一般基于姿态矩阵求解，可以分为直接法、基于 Wahba 问题^[21]的最优化方法和最小二乘方法等。下文将对 GNSS 姿态求解的主要算法予以介绍，其中直接法、TRIAD 算法为基于姿态矩阵的算法；QUEST 为基于 Wahba 问题的算法；姿态矩阵最小二乘法和姿态角最小二乘法为基于最小二乘思想的算法。

(1) 直接法^[20]。此方法优点是不需要先验载体基线长度，可直接利用观测值向量进行求解，模型简单，计算量小。缺点是方法受基线布局限制，且不能同时利用所有观测基线的先验信息。

将上文式 (2.11) 中的 R_{b2l} 、 X_l 和 X_b 分别代入式 (2.10) 中，即可得三姿态角计算公式^[33]，

$$\begin{aligned}
 y &= -\arctan\left(\frac{X_{l1}}{Y_{l1}}\right) \\
 p &= \arctan\left(\frac{Z_{l1}}{\sqrt{X_{l1}^2 + Y_{l1}^2}}\right) \\
 r &= -\arctan\left(\frac{X_{l2} \sin(y) \sin(p) - Y_{l2} \cos(y) \sin(p) + Z_{l2} \cos(p)}{X_{l2} \cos(y) + Y_{l2} \sin(y)}\right)
 \end{aligned} \tag{2.12}$$

(2) TRIAD 算法^[34]。此方法由 Black 于 1964 年提出，后被应用到 GPS 领域^[35]。此方法能够利用多条基线的解算数据，但具有主轴敏感性。

由向量运算可知，确定坐标系的三个垂直坐标轴只需要在该坐标系下的两个不平行基线。设坐标系下两个不平行的向量为 u 和 v ，则三坐标轴指向如下^[34]，

$$\begin{aligned}
 i &= u / |u| \\
 j &= u \times v / |u \times v| \\
 k &= i \times j / |i \times j|
 \end{aligned} \tag{2.13}$$

根据式 (2.13)，通过先验测量和实时基线测量可以分别确定载体坐标系和站心坐标系指向，令其分别为 B_i, B_j, B_k 和 L_i, L_j, L_k 。

根据式 (2.10)，旋转矩阵 R_{b2l} 可由式 (2.14) 计算，

$$R_{b2l} = [B_i, B_j, B_k]^{-1} \times [L_i, L_j, L_k] \tag{2.14}$$

姿态角可由 R_{b2l} 确定求出。

(3) QUEST 算法^[36]，QUEST 方法属于最优估计方法，由 Shuster 在 1981 年提出^[37]。此算法计算比较繁琐。

其核心思想是利用四元数将求解 R_{b2l} 的过程转化为求代价函数最小值的过程^[36, 38]。设有代价函数，

$$L(R_{b2l}) = \frac{1}{2} \sum_{i=1}^n r_i |L_i - R_{b2l} B_i|^2 \tag{2.15}$$

其中 r_i 满足 $\sum_{i=1}^n r_i = 1$ 。

需要找出一个合适的 R_{b2l} 使 $L(R_{b2l})$ 值最小，于是式 (2.15) 可以转换为寻找一个最优四元数，然后根据四元数计算最优姿态角。

(4) 姿态矩阵最小二乘法^[25]。此方法能够充分减少随机误差，但在实践中

求解不稳定。

站心坐标系到载体坐标系的旋转可以由式 (2.16) 表达^[25],

$$\begin{aligned} X_{b1} &= R_{l2b} * X_{l1} \\ X_{b2} &= R_{l2b} * X_{l2} \\ [X_{b1} \ X_{b2}] &= R_{l2b} * [X_{l1} \ X_{l2}] \end{aligned} \quad (2.16)$$

考虑到 $R_{l2b} = R_{b2l}^{-1}$, 故 $[X_{l1} \ X_{l2}] = R_{b2l} * [X_{b1} \ X_{b2}]$ 也同时成立。 X_{l1} 和 X_{l2} 分别表示三接收机姿态测量系统中的两条独立基线。

因为式 (2.16) 中未知参数只有三个姿态角, 而两条基线构成的方程数有六个, 则根据矩阵最小二乘原理, 有,

$$R_{l2b} = (X_b * X_l^T) * (X_l * X_l^T)^{-1} \quad (2.17)$$

在只有两条基线的情况下, X_l 和 X_b 均为 3×2 的矩阵, $(X_l * X_l^T)$ 的秩为 2, 故只能对 $(X_l * X_l^T)^{-1}$ 求解广义逆, 根据 MP 广义逆定义, 有,

$$(X_l * X_l^T)^{-1} = X_l * (X_l^{-1} * X_l) * (X_l^{-1} * X_l) * X_l^T \quad (2.18)$$

则有,

$$R_{l2b} = X_b * (X_l^{-1} * X_l) * X_l^T \quad (2.19)$$

根据 R_{l2b} 的表达式, 可求,

$$\begin{aligned} y &= -\arctan(R_{l2b}(2,1) / R_{l2b}(2,2)) \\ p &= \arcsin R_{l2b}(2,3) \\ r &= \arcsin(R_{l2b}(1,3) / \cos(p)) \end{aligned} \quad (2.20)$$

(5) 姿态角最小二乘法。此种方法需要已知姿态角的初值, 其核心思想是误差理论中的条件平差^[24]。此种方法在应用中常有不收敛情况。

将式 (2.10) 含有误差部分展开, 可以变化为经典条件平差表达式,

$$A_i \delta + (B_i, I) * \begin{pmatrix} \delta u_i \\ \delta b \end{pmatrix} + w_i = 0 \quad (2.21)$$

式 (2.21) 中,

$$A_i = \left(\frac{\partial R}{\partial y} u_i, \frac{\partial R}{\partial p} u_i, \frac{\partial R}{\partial r} u_i \right)_{3 \times 3}$$

$$\delta = (\delta y, \delta p, \delta r)^T$$

$$B_i = R(y_0, p_0, r_0)$$

$$w_i = R(y_0, p_0, r_0) * u_i - b_i$$

I为3x3单位矩阵

根据条件平差算法，可得，

$$\delta = N^{-1} * U = - \left[\sum_{i=1}^n A_i^T (B_i^T C u_i B_i + C b_i)^{-1} A_i \right]^{-1} * \left[\sum_{i=1}^n A_i^T (B_i^T C u_i B_i + C b_i)^{-1} w_i \right] \quad (2.22)$$

最终姿态角为起算值加上改正值，

$$\begin{pmatrix} y \\ p \\ r \end{pmatrix} = \begin{pmatrix} y_0 \\ p_0 \\ r_0 \end{pmatrix} + \begin{pmatrix} \delta y \\ \delta p \\ \delta r \end{pmatrix} \quad (2.23)$$

此方法在应用中一般采用迭代方式进行。

基于姿态矩阵的姿态角算法是从数学角度出发，通过计算三维直角坐标系的旋转角来求解姿态角；基于 **Wahba** 问题的姿态角算法本质上是一种加权求解思想，它通过赋予观测值以不同权值来使目标函数达到最小，求解过程稍显繁琐；基于最小二乘原理的姿态算法是从误差处理角度入手，通过最小二乘思想来配置误差，然后通过间接平差和直接平差等误差处理模型来求解姿态角。在保证足够精度情况下，考虑到效率性和简洁性，在所编写的程序中，本文实现了直接法、**TRIAD** 算法和姿态矩阵最小二乘法，并通过实验对三种算法进行了对比分析，具体见 6.3.2 节。

第3章 短基线解算及误差分析

载波相位测量能够提供比码测量更高的精度，故高精度定位一般采用载波进行解算。本文研究的姿态测量，需要以高精度短基线向量为基础，基线求解的精度决定着姿态测量精度，故基线解算非常重要。为了提高精度，基线解算需要考虑多种误差，本章将首先论述 GNSS 基线解算模型，然后讨论短基线解算中误差的消除问题，最后将对应用 Kalman 滤波求解基线进行较为详细的介绍。

3.1 GNSS 基线解算模型

卫星发射的信号将沿着各个方向传播，穿过两万公里的真空层和大气层到达地面，地面接收机将识别并持续跟踪这些信号，从信号中获得载波的累积整周数和不满一周的相位值。当有多台接收机同时对相同卫星跟踪时，就可以通过观测到的载波相位值确定接收机之间的相对位置关系^[39]。

这一节将讨论利用载波求解基线的基本模型和差分算法。载波相位测量值 φ 、整周模糊度 N 、载波波长 λ 和卫地距 ρ 间有以下数学关系，

$$\rho = (\varphi + N)\lambda \quad (3.1)$$

信号在传播过程中受到多种误差的干扰，根据干扰大小和范围，保留电离层、对流层等影响较大误差，忽略诸如固体潮等较长周期误差，将式（3.1）变形，载波观测方程可以写为，

$$\varphi_i \lambda = \sqrt{(X^i - X)^2 + (Y^i - Y)^2 + (Z^i - Z)^2} + cV_{t_R} - cV_{t_i^S} + N_i \lambda - V_{ion}^i + V_{trop}^i + \sigma_\rho^i + \sigma_{mp}^i + \varepsilon_i \quad (3.2)$$

式（3.2）中， σ_ρ^i 为为标定的相位缠绕误差， σ_{mp}^i 为多路径误差， ε_i 为观测噪声。 σ_ρ^i 、 σ_{mp}^i 、 ε_i 一般不能准确知道，通常采用一定办法对这些误差进行限制以达到既定精度要求。所以，一般将载波观测方程用如下公式表达：

$$\varphi_i \lambda = \sqrt{(X^i - X)^2 + (Y^i - Y)^2 + (Z^i - Z)^2} + cV_{t_R} - cV_{t_i^S} + N_i \lambda - V_{ion}^i + V_{trop}^i \quad (3.3)$$

同一历元，接收机可以接收多个卫星信号，而一颗卫星也可以被多个接收机

接收到，由此接收机可以构成多个观测方程，相同时刻不同的观测方程中通常含有相同或相近的参数，如多台接收机同时对相同卫星进行观测，它们观测方程中的卫星钟差项就相同；两台相近的接收机对相同卫星进行观测，观测方程中的电离层和对流层误差改正就相近。根据这种情况，实际数据处理常常使用作差法，将含有相同或者相近项的观测方程进行做差，以消除或消弱这些误差项。

通常采用的做差方式有接收机间求差（单差）、卫星间求差（双差）和不同历元间求差（三差）。为了方便说明，一般规定接收机间求差为一次差，在一次差基础上进行卫星间求差为二次差，在二次差基础上对不同历元间求差为三次差 [40]。

（1）用不同接收机观测方程构建一次差

假设在 t_1 时刻接收机 i, j 同时对卫星 p 进行观测，获得了载波相位观测值，考虑到 $c = \lambda f$ ，可以得到：

$$\begin{aligned}\varphi_i^p(t_1) &= \frac{f}{c} \rho_i^p(t_1) + fV_{t_i}(t_1) - fV_{t_i^p}(t_1) + N_i^p - \frac{f}{c} (V_{ion}^{t_1})_i^p + \frac{f}{c} (V_{trop}^{t_1})_i^p \\ \varphi_j^p(t_1) &= \frac{f}{c} \rho_j^p(t_1) + fV_{t_j}(t_1) - fV_{t_j^p}(t_1) + N_j^p - \frac{f}{c} (V_{ion}^{t_1})_j^p + \frac{f}{c} (V_{trop}^{t_1})_j^p\end{aligned}\quad (3.4)$$

做差化简后，可以得到：

$$\Delta\varphi_{ij}^p(t_1) = \frac{f}{c} \Delta\rho_{ij}^p(t_1) + f\Delta V_{t_{ij}}(t_1) + \Delta N_{ij}^p - \frac{f}{c} \Delta(V_{ion}^{t_1})_{ij}^p + \frac{f}{c} \Delta(V_{trop}^{t_1})_{ij}^p \quad (3.5)$$

经过做单差，原来包含在观测方程中的卫星钟差 $V_{t_i^p}(t_1)$ 被消除，卫星星历误差，电离层误差和对流层误差等影响也被消弱，由于姿态测量中的基线都为短基线，其消弱异常明显。

（2）在卫星间做二次差

假设 i, j 两个测站同时追踪 p, q 两颗卫星，则可以分别对 p, q 两颗卫星列出单差观测方程，

$$\begin{aligned}\Delta\varphi_{ij}^p(t_1) &= \frac{f}{c} \Delta\rho_{ij}^p(t_1) + f\Delta V_{t_{ij}}(t_1) + \Delta N_{ij}^p - \frac{f}{c} \Delta(V_{ion}^{t_1})_{ij}^p + \frac{f}{c} \Delta(V_{trop}^{t_1})_{ij}^p \\ \Delta\varphi_{ij}^q(t_1) &= \frac{f}{c} \Delta\rho_{ij}^q(t_1) + f\Delta V_{t_{ij}}(t_1) + \Delta N_{ij}^q - \frac{f}{c} \Delta(V_{ion}^{t_1})_{ij}^q + \frac{f}{c} \Delta(V_{trop}^{t_1})_{ij}^q\end{aligned}\quad (3.6)$$

将上述两个方程做差并简化，如下：

$$\Delta\nabla\varphi_{ij}^{pq}(t_1) = \frac{f}{c} \Delta\nabla\rho_{ij}^{pq}(t_1) + \Delta\nabla N_{ij}^{pq} - \frac{f}{c} \Delta\nabla(V_{ion}^{t_1})_{ij}^{pq} + \frac{f}{c} \Delta\nabla(V_{trop}^{t_1})_{ij}^{pq} \quad (3.7)$$

可以看到, 经过第二次做差, 观测方程中的接收机钟差 $\Delta V_{t_{ij}}(t_1)$ 被差分掉, 电离层和对流层误差等影响被再一次削弱。

在实际操作中, 计算双差方程所使用的参考星并无严格规定, 一般采用视野中观测时间较长、高度角较大的卫星作为参考星, 然后将其他卫星的单差方程与参考星的双差方程相减作差, 由此组建双差方程。

接下来探讨求解基线所需最少历元的问题。按照接收机位置是否移动, 相对定位可以分为静态相对定位和动态相对定位。顾名思义, 静态相对定位时两接收机位置保持不变, 而动态相对定位时, 至少有一台接收机的位置是随着历元而改变的。

令 n_s, n_t 分别表示接收到的卫星数量和接收历元数, 对于静态定位, 在双差情况下, n_s, n_t 需要满足的关系是:

$$\begin{aligned} (n_s - 1)n_t &\geq 3 + n_s - 1 \\ n_t &\geq \frac{n_s + 2}{n_s - 1} \\ n_s &\geq \frac{n_t + 2}{n_t - 1} \end{aligned} \quad (3.8)$$

双差情况下, 要想观测方程不秩亏, 需满足, $n_s = 2$ 时, $n_t \geq 4$ 或者 $n_t = 2$ 时, $n_s \geq 4$ 。

对于动态相对定位, 当两台接收机同时移动时, 每个历元坐标都会产生变化, 这会导致解算未知数过多, 方程产生秩亏, 无法求解基线向量。所以, 必须至少确定一台接收机坐标, 才能够求解基线。在姿态测量的短基线解算中, 一般通过单点定位确定一台接收机坐标, 并在此基础上求解基线。

由于动态情况下, 接收机位置会产生变化, 所以必须在一个历元情况下进行求解, 即 $n_t = 1$, 但当 $n_t = 1$, 无论单差还是双差, 都不能提供有效的解。所以, 对于动态情况, 只有在模糊度已知情况下才能求解基线向量, 而当得到模糊度并固定以后, n_s, n_t 需要满足关系变为:

$$\begin{aligned} (n_s - 1)n_t &\geq 3n_t \\ n_s &\geq 4 \end{aligned} \quad (3.9)$$

由上可知, 无论静态还是动态, 要想实时解算基线向量, 均需满足 $n_s \geq 4$ 。

3.2 短基线解算误差改正

受信号传播路径、硬件、相对论效应等因素影响,测量过程会出现误差,要想得到真实的卫地距,需要对观测值进行必要的改正。随着基线长度,定位方式以及观测条件的不同,相应误差的改正方式也不同。一般 GNSS 基线处理中,常根据具体情况采用相应方法。

下面将 GNSS 数据处理中常见的误差改正梳理成表,其中归纳了误差项的成因、误差项的数量级和常见改正方式等,然后结合姿态测量中的短基线解算,说明本文在短基线解算中如何进行误差的改正^[41]。

姿态求解的核心是解算超短基线,一般采用双差求解。观测方程经过双差后,部分误差被完全消除,如卫星钟差、接收机钟差等;部分误差被大幅削弱,如卫星星历误差、大气延迟误差和接收机噪声误差等,由于姿态测量中基线一般非常短,经过双差处理后,可认为这部分误差被完全消除了。另外,本文姿态解算针对动态载体,要求求解单历元数据,这就需要在单个历元内实时得到解算结果,所以固体潮改正、海洋潮汐改正等长周期误差可以忽略^[42]。

本文中短基线误差的消除方式,请参看表 3-1:

表 3-1 基线解算中的误差分析

	误差项	成因	量级	常用改正方式	本文处理方式
与接收机和测站有关的误差	接收机钟差	接收机内部晶体振荡器的频率改变	km	差分消除;作为待定参数解算;对不同天线使用相同时频系统	双差消除
	接收机天线相位偏差	理论相位中心同实际相位中心未对齐	cm	利用接收机标定参数改正;忽略	利用接收机标定值改正或忽略
	固体潮改正	摄动天体引力导致地球表面产生周期性涨落	cm	周期项能经由长时间观测去除;通过模型改正;通过差分消除;忽略	忽略
	海洋潮汐改正	由海洋潮汐涨落引起	cm	误差与离海岸远近有关;周期项可通过长时间观测消除;通过模型消除;差分消除;忽略	忽略
	地球自转改正	由非惯性坐标系和地球自转引起	m	通过公式对卫星坐标和卫地距进行改正	公示改正

续表 3-2 基线解算中的误差分析

误差项		成因	量级	改正方式	本文处理方式
与卫星有关的误差	卫星钟差	卫星钟频率漂移引起	km	通过广播星历信息改正；通过精密星历进行内插；差分消除	双差消除
	卫星轨道误差	受轨道计算模型、定轨软件、地面跟踪网、跟踪站分布影响，卫星位置计算不准确	m	利用精密星历内插；利用差分削弱	双差消除
与卫星有关的误差	卫星天线相位偏差	卫星质心同相位中心不重合	cm	改正值在卫星制造时已经进行标定	根据卫星标定值修正
	相位缠绕改正	卫星发射右旋极化信号，接收机同卫星相互位置关系会使接收信号发生缠绕	cm	短基线可通过差分消除；利用 IGS 精密产品减弱；动态观测可以将其混入到接收机钟差项中求解；通过模型改正	差分后可忽略
	卫星钟相对论效应	卫星钟和接收机钟运动速度和重力位不同而引起的相对钟差现象	cm	卫星出厂时已对主要项进行标定，扰动项通过模型改正；忽略	模型改正
与传播路径有关的误差	对流层延迟误差	大气对流层对电磁波信号的折射引起 ^[40]	m	利用模型改正；通过差分减弱；长基线可作为待估参数	主接收机单点定位采用模型改正，基线解使用双差消除
	电离层延迟误差	电离层区域对电磁波传播速度产生影响	m	通过模型改正；通过差分减弱；通过多频组合减弱或消除；长基线可作为待估参数	主接收机单点定位采用模型改正，基线解使用双差消除
	多路径效应	卫星信号经反射后被接收机接收所产生的干扰	m	增强硬件防护；改善算法；建模消除；忽略	忽略

在多路径消除上，因为本文研究针对运动载体，载体运动过程中周围环境会因运动而改变，无法对多路径效应进行有效消除，在多路径解决方法上，本文选择忽略。

3.3 应用 Kalman 滤波解算基线

3.3.1 Kalman 滤波基本原理

Kalman 滤波在二十世纪六十年代被提出，它是一种最优化自回归数据处理方法，广泛应用于工程中已经四十余年，其应用领域包括航天轨道预测、机器人导航与控制，雷达系统与导航跟踪、多系统数据处理融合(例如 INS 和 GPS)，信号处理和计算机图像处理等。

Kalman 滤波解决的主要问题是如何在已有观测值的基础上尽可能合理的对下一个历元值进行预估。它的使用必须满足以下三个条件：

- (1) 系统模型必须是线性的；
- (2) 影响过程或是测量的噪声必须是白噪声（与时间无关）；
- (3) 噪声必须服从高斯分布。

Kalman 滤波的核心为五个线性方程构成的方程组，无论是单一数据源还是集成数据源，只要将它们以这五个方程建模，就可以应用 Kalman 滤波。这五个方程应按照先后顺序分别使用，以达到预测和更新功能。以下公式中使用 k 表示下一时刻，使用 $k-1$ 表示上一时刻，分别对这五个公式进行说明。

$$(1) \quad \hat{x}_k^- = F\hat{x}_{k-1} + Bu_{k-1} + w_{k-1}$$

这个方程通过上一个时刻状态值预测下一个时刻状态值估值。 \hat{x}_{k-1} 是上一时刻的 n 阶状态值矩阵。 F 是 $n \times n$ 的传递矩阵， B 是 $n \times c$ 的控制转移矩阵， u_{k-1} 是人为施加的 $c \times 1$ 控制变量， w_{k-1} 是符合高斯正态分布的过程噪声，它具有 n 阶的协方差矩阵 Q_{k-1} ，一般情况下，认为 Q_{k-1} 不随时间改变， u_{k-1} 代表外加变量，如果系统不施加外部影响，就可以忽略 Bu_{k-1} 项。

实际情况中，可直接得到的测量值 z_k 可能是也可能不是状态矩阵 \hat{x}_k ，如果不是，需要通过使用 $z_k = H_k \hat{x}_k + v_k$ 将 z_k 转换为 \hat{x}_k 。 z_k 是 $m \times 1$ 的测量值矩阵， H_k 是 $m \times n$ 的状态转移矩阵， v_k 是符合高斯正态分布的测量误差，它的方差阵为 R_k 。这样就可以将观测值 z_k 转换为状态值 \hat{x}_k ，很多时候都需要这种转换，如在 GNSS 数据处理中，测量值是卫地距，而最终想要的结果是接收机的三维坐标，就需要使用 $z_k = H_k \hat{x}_k + v_k$ 将测量值 z_k 转换为目标向量 \hat{x}_k 。

$$(2) \quad P_k^- = FP_{k-1}F^T + Q_{k-1}$$

P_{k-1} 表示 $k-1$ 时刻状态值方差阵， P_k^- 表示预估值 \hat{x}_k^- 的方差阵。此式是对预测值 \hat{x}_k^- 误差的协方差阵的预先估计，能够确定预测值的准确性，将会改变预测值与更新值间的加权关系。

$$(3) \quad K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$$

在这个公式中， P_k^- 代表预测估值的方差协方差阵， R_k 是 k 时刻测量值的误差阵， H_k 阵是 k 时刻测量值与状态矩阵 x_k 的转换矩阵，很容易就可以看出，这个方程中融合了预测和新测数据的方差矩阵，确定了他们之间的权重关系，所以 K_k 系数为预测值 x_k^- 同新测值 z_k 之间的权重比值，关于 K_k 公式的推导可以参考王惠南编著的《GPS 导航原理与应用》，见文献^[43]。

$$(4) \quad x_k = x_k^- + K_k (z_k - H_k x_k^-)$$

x_k 为 x_{k-1} 和 z_k 的加权估值，是 Kalman 滤波在当前历元下的估计结果。公式 (1) 中用 $k-1$ 时刻的状态值 x_{k-1} 预测 k 时刻的状态值 x_k^- ，这个过程伴随着误差 $\hat{x}_k = x_k - x_k^-$ ，我们需要对这一误差进行修正。考虑到 k 时刻的测量值 z_k 已知，由公式 $z_k = H_k x_k + v_k$ 可得 $\hat{z}_k = z_k - z_k^- = H_k (x_k - x_k^-) = H_k \hat{x}_k$ ，由此便可估计 \hat{x}_k ，结合公式 (3)，有 $\hat{x}_k = K_k (z_k - H_k x_k^-)$ ，使用 \hat{x}_k 对 x_k^- 进行更新，则有公式 $x_k = x_k^- + K_k (z_k - H_k x_k^-)$ 。

$$(5) \quad P_k = (I - K_k H_k) P_k^-$$

最后一步需要对估计 x_k 的方差阵 P_k ，以准备 $k+1$ 时刻的估计。

使用 Kalman 滤波，首先需要确定三个变量的初值，即 x_k 、 P_k 和 Q_k 的初值 x_0 、 P_0 和 Q_0 ，它们的计算可以参考文献^[43]。

3.3.2 应用 Kalman 滤波求解基线

Kalman 滤波只针对线性系统，由于 GNSS 观测方程为非线性，无论是利用载波还是码伪距来解算位置，都不适用，如果要把它引进 GNSS 基线解算中，需要把 GNSS 基线解算模型转换为线性形式。最简单的方式就是将观测方程用级数展开，舍去高阶项，只留下一阶项来形成近似线性形式。通常把这种经过线性展开后的 Kalman 滤波称为扩展 Kalman 滤波^[43]。为了方便说明，将上文中的五个 Kalman 滤波公式简单变形，写成如下形式^[44]。

$$\begin{aligned} x_k^- &= F_{k-1}^k x_{k-1} \\ P_k^- &= F_{k-1}^k P_{k-1} F_{k-1}^{kT} + Q_{k-1}^k \\ K_k &= P_k^- H \left(\hat{x}_k^- \right) \left(H \left(\hat{x}_k^- \right) P_k^- H \left(\hat{x}_k^- \right)^T + R_k \right)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k \left(y_k - h \left(\hat{x}_k^- \right) \right) \\ P_k &= \left(I - K_k H \left(\hat{x}_k^- \right) \right) P_k^- \end{aligned} \quad (3.10)$$

在对上面公式和公式中的未知量进行解释之前，首先研究一下 GNSS 观测模

型，下面两个公式分别是双差载波观测方程和双差码观测方程。

$$\begin{aligned}\Phi_{rb,k}^{ij} &= \lambda_k \varphi_{rb,k}^{ij} = \rho_{rb}^{ij} - I_{rb,k}^{ij} + T_{rb}^{ij} + \lambda_k (N_{rb,k}^i - N_{rb,k}^j) + \varepsilon_\Phi \\ P_{rb,k}^{ij} &= \rho_{rb}^{ij} - I_{rb,k}^{ij} + T_{rb}^{ij} + \varepsilon_P\end{aligned}\quad (3.11)$$

其中， $()^{ij}$ 表示卫星间单差；

$()_{rb}$ 表示主站和从站间的单差；

φ_r^i 为以周为单位的载波测量值；

P_r^i 为码观测值；

ρ_r^i 为卫地距；

λ 为载波波长；

N_{rb}^i 为单差载波方程模糊度；

$\varepsilon_\varphi, \varepsilon_P$ 为载波和码伪距测量误差；

I_r^i 为电离层误差；

T_r^i 为对流层误差；

对短基线，对流层误差和电离层误差可以通过双差很好的消除，模型可以简化为，

$$\begin{aligned}\Phi_{rb,k}^{ij} &= \lambda_k \varphi_{rb,k}^{ij} = \rho_{rb}^{ij} + \lambda_k (N_{rb,k}^i - N_{rb,k}^j) + \varepsilon_\Phi \\ P_{rb,k}^{ij} &= \rho_{rb}^{ij} + \varepsilon_P\end{aligned}\quad (3.12)$$

在双频情况下，我们假设主站坐标已知，从站坐标未知，则滤波方程中 x 按如下定义：

$$\mathbf{x} = (r_r^T, v_r^T, N_{L1}^T, N_{L2}^T)^T \quad (3.13)$$

r_r 为从站在 ECEF 坐标系下的坐标；

v_r 为从站在 ECEF 坐标系下的速度；

$N_{Lj} = (N_{rb,Lj}^1, N_{rb,Lj}^2, \dots, N_{rb,Lj}^m)^T$ 为以周为单位的站间单差模糊度， m 为接收卫星数量；这里之所以不直接求解双差模糊度，是考虑到双差可能会遇到换参考星问题，导致模型变得复杂。

y_k 是对双差观测方程等式左边的简写，为观测值，其定义如下。

$$y_k = (\Phi_{L1}^T, \Phi_{L2}^T, P_{L1}^T, P_{L2}^T)^T \quad (3.14)$$

$\Phi_{Li} = (\Phi_{rb,Li}^{12}, \Phi_{rb,Li}^{13}, \Phi_{rb,Li}^{14}, \dots, \Phi_{rb,Li}^{1m})^T$ 是双差载波观测值；

$P_{Li} = (P_{rb,Li}^{12}, P_{rb,Li}^{13}, P_{rb,Li}^{14}, \dots, P_{rb,Li}^{1m})^T$ 是双差码伪距观测值，此时参考星为第 1 颗；

$h(\hat{x})$ 是对观测方程右侧的简写，其定义如下。

$$h(\hat{x}) = (h_{\Phi,L1}^T, h_{\Phi,L2}^T, h_{P,L1}^T, h_{P,L2}^T)^T \quad (3.15)$$

$$h_{\Phi,Lj} = \begin{pmatrix} \rho_{rb}^{12} + \lambda_{Lj} (\hat{N}_{rb}^1 - \hat{N}_{rb}^2) \\ \rho_{rb}^{13} + \lambda_{Lj} (\hat{N}_{rb}^1 - \hat{N}_{rb}^3) \\ \vdots \\ \rho_{rb}^{1m} + \lambda_{Lj} (\hat{N}_{rb}^1 - \hat{N}_{rb}^m) \end{pmatrix}, h_{P,Lj} = \begin{pmatrix} \rho_{rb}^{12} \\ \rho_{rb}^{13} \\ \vdots \\ \rho_{rb}^{1m} \end{pmatrix} \quad (3.16)$$

$\rho_r^i = \|\hat{r}_r - r^i\|$, $\rho_b^i = \|r_b - r^i\|$ 分别是主观测站和从观测站站卫地距；

\hat{r}_r 是从站的位置坐标；

r_b 是基准站的位置坐标；

r^i 是卫星 i 的坐标

$H(\hat{x})$ 矩阵是 $h(\hat{x})$ 对 \hat{x} 的偏导矩阵，它具有如下形式，

$$H(\hat{x}) = \frac{\partial h(x)}{\partial x} \Big|_{x=\hat{x}} = \begin{pmatrix} -DE & 0 & \lambda_{L1}D & 0 \\ -DE & 0 & 0 & \lambda_{L2}D \\ -DE & 0 & 0 & 0 \\ -DE & 0 & 0 & 0 \end{pmatrix} \quad (3.17)$$

$E = (e_r^1, e_r^2, e_r^3 \dots e_r^m)^T$ 是从站上各个卫星的方向矢量， e_r^i 为第 i 颗卫星的视线方向向量。

$$D = \begin{pmatrix} 1 & -1 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & -1 \end{pmatrix} \text{ 为单差矩阵，通过这个矩阵，可以把站间单差模糊度}$$

$N_{rb,Li}^i$ 变为双差模糊度。

R_k 是双差测量误差的协方差阵，由下式 (3.18) 计算。

$$R_k = \begin{pmatrix} DR_{\phi,L1}D^T & & & \\ & DR_{\phi,L2}D^T & & \\ & & DR_{P,L1}D^T & \\ & & & DR_{P,L2}D^T \end{pmatrix} \quad (3.18)$$

其中，

$$\begin{aligned} R_{\phi,Lj} &= \text{diag}(2\sigma_{\phi,Lj}^{1\ 2}, 2\sigma_{\phi,Lj}^{2\ 2}, \dots, 2\sigma_{\phi,Lj}^{m\ 2}) \\ R_{p,Lj} &= \text{diag}(2\sigma_{p,Lj}^{1\ 2}, 2\sigma_{p,Lj}^{2\ 2}, \dots, 2\sigma_{p,Lj}^{m\ 2}) \end{aligned} \quad (3.19)$$

$R_{\phi,Li}$ 是载波单差方差向量, $\sigma_{\phi,Li}^i$ 是 L_j 载波的标准差。

$R_{p,Li}$ 是码伪距单差的方差向量, $\sigma_{p,Li}^i$ 是 L_j 载波上的码伪距的误差标准差。

F_{k-1}^k 是 Kalman 滤波中的状态转移矩阵, Q_{k-1}^k 是过程噪声矩阵, 它们按如式 (3.20) 构造。

$$F_{k-1}^k = \begin{pmatrix} I_3 & I_3 \tau_r & & \\ & I_3 & & \\ & & I_{m-1} & \\ & & & I_{m-1} \end{pmatrix}, Q_{k-1}^k = \begin{pmatrix} 0 & & & \\ & q_v & & \\ & & 0 & \\ & & & 0 \end{pmatrix} \quad (3.20)$$

其中 $q_v = E_{enu-ecef} \text{diag}(\sigma_{ve}^2 \tau_t, \sigma_{vn}^2 \tau_t, \sigma_{vu}^2 \tau_t) E_{enu-ecef}^T$, $E_{enu-ecef}$ 是站心坐标系 ENU 到地心坐标系 ECEF 的转换矩阵;

$\tau_t = t_k - t_{k-1}$ 是 GNSS 接收机的采样间隔时间;

$\sigma_{ve}^2, \sigma_{vn}^2, \sigma_{vu}^2$ 分别是站在东 (E) 方向、北 (N) 方向和天顶方向 (U) 速度噪声的方差;

求解上述扩展 Kalman 滤波方程后, 就可以估计出从站到主站的基线向量、速度和整周模糊度。

由上述方程所解出的基线解, 是在整周模糊度未固定情况下所得到的, 为浮点解, 要想获得更高精度的基线解, 需要将模糊度固定, 也就是解算固定解。现阶段应用最广泛最稳定的 OTF (在航) 算法是 LAMBDA 算法, 而 LAMBDA 算法对双差模糊度应用效果最优, 为了使用 LAMBDA 算法固定模糊度, 我们需要将上文求解出的单差模糊度转换为双差模糊度, 双差模糊度和其对应的方差协方差阵 P_k^d 可以使用状态解 \hat{x}_k 和它所对应的协方差阵 P_k 与转换矩阵 G 相乘求取, 如式 (3.21) 所示。

$$\begin{aligned} \hat{x}_k^d &= G \hat{x}_k = (\hat{r}_r^T, \hat{v}_r^T, \hat{N}^T)^T \\ P_k^d &= G P_k G^T = \begin{pmatrix} Q_R & Q_{NR} \\ Q_{RN} & Q_N \end{pmatrix} \\ G &= \begin{pmatrix} I_6 & & \\ & D & \\ & & D \end{pmatrix} \end{aligned} \quad (3.21)$$

利用上面方程，可以得到双差模糊度的浮点解 \hat{N} 和其对应的方差协方差阵 P_k^d ，接下来可利用 LAMBDA 方法对 \hat{N} 向量进行固定，以求取固定解。

LAMBDA 算法的核心为最小二乘思想，在其约束下进行模糊度的整数化。

$$\tilde{N} = \operatorname{argmin} \left[\left(N - \hat{N} \right)^T Q_N^{-1} \left(N - \hat{N} \right) \right], N \in Z \quad (3.22)$$

上式中， \tilde{N} 为模糊度的固定解。当 \tilde{N} 解算成功时，就可以在 \tilde{N} 约束下求解状态向量的固定解了，固定解可按式 (3.23) 求解，

$$\begin{pmatrix} \tilde{r}_r \\ \tilde{v}_r \end{pmatrix} = \begin{pmatrix} \hat{r}_r \\ \hat{v}_r \end{pmatrix} - Q_{RN} Q_N^{-1} (\hat{N} - \tilde{N}) \quad (3.23)$$

LAMBDA 方法的计算过程将在第 4 章有详细的论述。

第4章 姿态解算中的模糊度求解

短基线解算的核心是模糊度求解，模糊度能否固定将决定基线解算精度，并间接地决定姿态求解精度。姿态解算因为已知先验基线长度，如果能够利用这个已知信息，就能够对解算过程添加约束，从而提高成功率。考虑到 LAMBDA 的理论体系最完整，在实际应用中解算成功率最高，而且能够在载体运动情况下进行解算，本文研究基于 LAMBDA 算法求解整周模糊度。本章首先对常用模糊度求解方法进行叙述，然后将详细讨论 LAMBDA 算法，在此基础上，进一步研究改进的 M-LAMBDA 算法和附有约束条件的 C-LAMBDA 算法，提出结合使用 M-LAMBDA 和 C-LAMBDA 算法，利用两种算法的优势提高模糊度固定的效率和成功率。

4.1 常用在航模糊度求解方法

对于姿态求解，只能使用 OTF (On The Fly, 即在航) 模糊度解算方法。经典的 OTF 解算方法有模糊函数法、整周模糊值快速逼近技术、最小二乘搜索等。

(1) 模糊函数法 (AFM) 由 Counselman 于 1981 年提出^[45]，Remondi 将它引入到 GPS 处理中^[46]，这种方法的优势是对周跳不敏感，缺点是需要较长的固定时间，所以不适用于姿态测量系统；

(2) 宽窄巷技术由 Wubben 于 1989 年提出，主要用于双频 P 码接收机。姿态测量为节省成本，常采用单频接收机，故不适用^[47, 48]。

(3) 最小二乘搜索法 (LAST) 由 Hatch 提出，这种方法的主要思想是仅对四颗卫星进行处理，以便减小搜索空间，从而提高搜索效率^[49]；

(4) 整周模糊度快速逼近技术 (FARA 法) 由 Frei 等人提出，这种方法以统计理论为基础，在模糊度估值及其方差所囊括的搜索范围内搜索一组使残差和达到最小的似然模糊度解组合^[50]；

(5) 最小二乘降相关平差 (LAMBDA)^[51] 由荷兰 DELFT 科技大学的 Teunissen 教授提出，算法的核心是最小二乘原理，算法特点是适用性强，能够对单频、双频、非差、单差、双差、多卫星、宽巷 (线性组合模糊度) 进行解算^[51]。在应用上，能够实现只通过较短历元的连续观测就可以固定模糊度，从而获得精确解。

在姿态解算领域，LAMBDA 方法应用的最多，在不断的发展过程中，已有改进版本的 M-LAMBDA 算法出现，针对姿态解算中基线长已知的特点，近年来又提出了附加约束条件的 C-LABMDA 方法。

4.2 LAMBDA 算法

LAMBDA 算法考虑到模糊度的相关特性，通过整数线性变换，逐渐改变模糊度方差阵的相关性，使得方差阵的非对角线元素值减小，达到主角占优，进而减弱模糊度间的相关性，在几何上，整周模糊度搜索空间由极度拉伸的不规则超椭球逐渐转换为各向半径相近的类超球，从而提高模糊度解算效率。这种方法经 Teunissen 及许多学者的多年共同努力，已经在世界范围内成为数学逻辑最严密、模糊度固定率最高的算法。

短时间内的连续观测值，由于时间跨度小，卫星同接收机之间几何图形变化不大，导致解算时候基线向量解和模糊度无法有效剥离，这使得模糊度的方差协方差阵高度关联，将导致最终的搜索空间畸形，降低搜索效率和搜索准确性。

假设模糊度的方差协方差阵 Q_a 的秩为 n ，可以将其每一行看作是 n 维向量空间的一个向量，根据矩阵理论，可以使用这 n 个向量构造向量空间的一组基，在数学上，这组基向量间互为正交，也就是不存在相关性^[52]。条件最小二乘平差就是基于这一思想，通过 Cholesky 构建了 Q_a 的 LDL^T 分解，达到消除原始模糊度间相关性的目的。其中 D 为对角阵，是变换后条件模糊度的误差阵，条件分解能够消除 Q_a 的相关性。

但实际中，条件分解不具有可行性。根据误差理论，必要观测数等于未知参数的个数，对定位来讲，未知量为待定点的三维坐标或者三维向量基线，也就是说，只需要三个测量值就能够确定未知量，多余的测量值则为冗余测量值。这就造成了 D 阵中前三个对角线元素值非常大，而 4 到 n 的对角线元素则小很多，以至形成了 D 阵谱的断裂性。在模糊度搜索中，搜索空间的大小基本上由 D 阵对角线元素决定，这就导致在进行第四个模糊度搜索时，由于搜索空间太小而被迫中断，最终导致无法继续求解模糊度。

下面通过模拟简单的再现这个问题。 Q_a 可按式 (4.1) 简单模拟，

$$Q_a = \begin{pmatrix} \delta_{a(1,1)} & \cdots & \delta_{a(1,n)} \\ \vdots & \ddots & \vdots \\ \delta_{a(n,1)} & \cdots & \delta_{a(n,n)} \end{pmatrix} = \delta^2 \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{pmatrix} + \begin{pmatrix} \beta_{11} & \beta_{12} & \beta_{13} \\ \vdots & \vdots & \vdots \\ \beta_{n1} & \beta_{n2} & \beta_{n3} \end{pmatrix} \begin{pmatrix} \beta_{11} & \beta_{12} & \beta_{13} \\ \vdots & \vdots & \vdots \\ \beta_{n1} & \beta_{n2} & \beta_{n3} \end{pmatrix}^T \quad (4.1)$$

Q_a 中融合了载波和未知基线的误差。式(4.1)中, T 代表矩阵转置, δ 代表载波误差, β 代表基线向量误差, 它的秩为 3。根据实际情况, 载波误差较小, 有 δ^2 远小于 β , 由式(4.1)构成的方差协方差阵经过条件变换后所得到的对角矩阵 D , 它的对角线前三个值远大于后面的值, 将之称为谱断裂性。

LAMBDA 算法解决了 D 阵的谱断裂问题。它通过构建一个 Z 阵实现 n 维空间变换, Z 阵的构建实际上经历了两个过程, 即整数高斯消除和条件方差置换, 前者实现模糊度去相关, 后者实现消去谱断裂性。原始模糊度向量经过与 Z 阵相乘, 得到了一组新的模糊度, 新模糊度的条件方差将会在数值上相当, 从而消除断裂性。在几何表达上, 模糊度空间椭球由转换前的极度拉伸状态, 到转换后的圆化状态, 极大的提高了搜索的效率和准确性。

整周模糊度具有整数特性, 为了不将正确解丢失, 变换后的超椭球体积应与之前超椭球体积相等。所以, Z 矩阵应具有保整特性和保积特性。同时也需能够保证转换后和转换前模糊度一一对应, 以确保转换后的模糊度可以通过乘以 Z 阵的逆阵实现还原。

整个算法由三部分组成, 第一部分构建 Z 矩阵, 第二部分确定 χ^2 (搜索空间), 第三部分进行模糊度搜索。

4.2.1 算法准备

LAMBDA 算法需要输入 \hat{a} 和 Q_a 两个参数, \hat{a} 为浮点模糊度向量, Q_a 为其方差阵, 因为待求的模糊度加上基线向量后数量大于观测方程数量, 未知数数量大于观测值, 一个历元无法同时求取基线向量和模糊度向量。一般通过几个历元的数据求出 \hat{a} 和 Q_a 的初值, 也可以先通过单点定位求解基线向量的初始值, 然后再求解模糊度浮点解。

原始的观测方程可以表示为式(3.2), 稍作改变可以表示为式(4.2),

$$\phi = \|R - r\| - c(dt - dT) + \lambda N + \varepsilon \quad (4.2)$$

双差并简化后可以表达为,

$$\Delta \nabla \phi = \Delta \nabla \|r - R\| + \lambda \Delta \nabla N + \Delta \nabla \varepsilon \quad (4.3)$$

式(4.3)进一步简化为,

$$y = Aa + Bb + e \quad (4.4)$$

其中, a 是双差模糊度;

b 是未知基线向量;

A, B 分别对应 a, b 的系数;

e 是未消除的残差;

对式 (4.4) 应用最小二乘的原理, 在残差 e 平方和最小条件下求解整周模糊度, 可列式 (4.5),

$$\min_{a, b} \|y - Aa - Bb\|_{Q_y}^2, a \in Z^n, b \in R^3 \quad (4.5)$$

进一步推导, 可以得到式 (4.6),

$$\begin{aligned} & \|y - Aa - Bb\|_{Q_y}^2 \\ &= \|y - A\hat{a} - B\hat{b} | a - \hat{e} - Aa - Bb + A\hat{a} + B\hat{b} | a + \hat{e}\|_{Q_y}^2 \\ &= \|\hat{e}\|_{Q_y}^2 + \|\hat{b} | a - b\|_{Q_{b|a}}^2 + \|\hat{a} - a\|_{Q_{\hat{a}}}^2 \end{aligned} \quad (4.6)$$

$\hat{b} | a$ 为以模糊度为条件的条件基线向量值, 当 a 固定后, $\hat{b} | a$ 即可确定, 所以, 式 (4.6) 可进一步简化为,

$$\min_a (\|\hat{e}\|_{Q_y}^2 + \|\hat{a} - a\|_{Q_{\hat{a}}}^2) \quad (4.7)$$

\hat{e} 为模糊度固定下来后, 方程的残差值, 也可由 a 固定后确定, 故式 (4.7) 可进一步简化为,

$$\min_a \|\hat{a} - a\|_{Q_{\hat{a}}}^2 = \min_a (\hat{a} - a) Q_{\hat{a}}^{-1} (\hat{a} - a), a \in Z^n \quad (4.8)$$

4.2.2 转换矩阵的构建

Z 阵 (转换矩阵) 的构建是为了消除原始模糊度向量之间的关联性。 Z 阵需要满足保整特性, 即整数矩阵与 Z 阵相乘后, 依然为整数矩阵, 这个特性在数学上可以用条件 $\det(Z) = 1$ 来 $\det(Z) = 1$ 约束。同时, Z 阵的构建需要使随后的搜索效率最大化, 这就需要 Z 阵对应的协方差阵 Q_z 满足如下两个条件^[53],

(1) Q_z 的非对角线元素要尽量的小, 与 0 越接近越好, 尽量消除彼此相关性;

(2) Q_z 中对角线元素应按照数值大小依次排列, 以便减少搜索次数;

Teunissen 在其文章中提出了 Z 阵构建原理和方法, 但在程序实现时, 一般不采用这种方法, 以下部分将分别介绍 Teunissen 的原理方法和实际程序中所采用的方法, 其中, 方法一为 Teunissen 在其原始文章中提出的方法, 方法二为实

际应用中使用的方法。

(1) 方法一

未固定的模糊度向量中，任意两个相邻的模糊度 $\hat{a}_{(i,i)}, \hat{a}_{(i+1,i+1)}$ 存在相关性，这

种相关性表现为其误差阵 $\begin{pmatrix} \delta\hat{a}_{(i,i)} & \delta\hat{a}_{(i,i+1)} \\ \delta\hat{a}_{(i+1,i)} & \delta\hat{a}_{(i+1,i+1)} \end{pmatrix}$ 的非对角线元素不为 0，根据向量理

论，可以通过构建二维空间的一组基将其完全去相关，如式 (4.9)。为了方便表示，就以第 1 个和第 2 个相邻模糊度为例，

$$\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2||} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\delta_{\hat{a}(2,1)}\delta_{\hat{a}(1,1)}^{-1} & 1 \end{pmatrix} \begin{pmatrix} \hat{a}_1 \\ \hat{a}_2 \end{pmatrix} \quad (4.9)$$

但 $-\delta_{\hat{a}(2,1)}\delta_{\hat{a}(1,1)}^{-1}$ 不是整数，不能保证整数变换，所以用式 (4.10) 代替，

$$\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2'} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\lceil \delta_{\hat{a}(2,1)}\delta_{\hat{a}(1,1)}^{-1} \rceil & 1 \end{pmatrix} \begin{pmatrix} \hat{a}_1 \\ \hat{a}_2 \end{pmatrix} \quad (4.10)$$

$-\lceil \delta_{\hat{a}(2,1)}\delta_{\hat{a}(1,1)}^{-1} \rceil$ 表示取最邻近整数，但这样一来， $\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2'} \end{pmatrix}$ 就不是完全去相关的向

量，于是有式 (4.11)，

$$\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2'} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \varepsilon & 1 \end{pmatrix} \begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2||} \end{pmatrix}, \varepsilon \leq 1/2 \quad (4.11)$$

ε 为向量 $\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2'} \end{pmatrix}$ 和 $\begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2||} \end{pmatrix}$ 的调整参数。

上面过程是以 \hat{a}_1 为条件求 $\hat{a}_{2'}$ ，为了进一步削弱相关性，下一步求以 $\hat{a}_{2'}$ 为条件的 $\hat{a}_{1'}$ ，

$$\begin{pmatrix} \hat{a}_{1'} \\ \hat{a}_{2'} \end{pmatrix} = \begin{pmatrix} 1 & -\lceil \delta_{\hat{a}(1,2')} \delta_{\hat{a}(2',2')}^{-1} \rceil \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \hat{a}_1 \\ \hat{a}_{2'} \end{pmatrix} \quad (4.12)$$

将式 (4.10) 和式 (4.12) 式一直循环下去，令式 (4.10) 中二阶矩阵为 \mathbf{U} ，式 (4.12) 中二阶矩阵为 \mathbf{T} ，不断的使 \mathbf{UT} 互乘，每一个相乘的矩阵都能够实现去相关效果，直到一个相乘的矩阵 \mathbf{U} 或 \mathbf{T} 成为单位阵， \mathbf{TU} 不断相乘的结果就是 \mathbf{Z} 矩阵，同时，我们可以得到转换后的模糊度 \hat{z}_1, \hat{z}_2 。从几何上，不断循环的过程是逐渐圆化超椭球的过程。

对于 n 维情况，需对每两个相邻的模糊度做上述转换，转换后的模糊度向量将实现去相关。

(2) 方法二

实际的程序实现中，一般使用两步来构造 Z 阵，即整数高斯变换过程和置换过程。

用 e_i 表示单位阵 I 的第 i 个纵向量。针对任意两个模糊度 i 和 j ，令 $\mu = [l_{ij}]$ ， l_{ij} 是对 Q_a 进行 LDL^T 分解后， L 矩阵的第 i 行第 j 列元素。算法初始将 Z 阵设定为单位矩阵，根据矩阵原理，则一次整数高斯变化可以表示为 $Z_{ij} = I - \mu e_i e_j^T$ ，那么 $Z_{ij}^{-1} = I + \mu e_i e_j^T$ ，将 Z_{ij} 作用于 L 阵可得，

$$\bar{L} = LZ_{ij} = L - \mu L e_i e_j^T \quad (4.13)$$

\bar{L} 与 L 唯一不同的地方在于 $\bar{l}_{kj} = l_{kj} - \mu l_{ki}, k = i, \dots, n$ ，为了 \bar{l}_{ij} 尽可能小，就出现了上文中的 $\mu = [l_{ij}]$ ，以保证 $\bar{l}_{ij} \leq 1/2, i < j$ 。由式 (4.8)，当对 L 应用 Z_{ij} 时，应同时对 \hat{a} 应用 Z_{ij}^T ，即 $z = Z_{ij}^T \hat{a}$ 。

为使 L 中的所有元素都小于 $1/2$ ，需要对 L 进行很多次的 Z_{ij} 运算，所有的 Z_{ij} 运算构成了最终的 Z 。

每次将 L 矩阵同 Z 矩阵相乘后，需要对 Q_a 和对应的 L 进行更新，令

$$L_{22} = \begin{pmatrix} 1 & \\ & 1 \end{pmatrix}, \quad L_{32} = \begin{pmatrix} l_{i+2,i} & l_{i+2,i+1} \\ \vdots & \vdots \\ l_{n,i} & l_{n,i+1} \end{pmatrix}, \quad L_{21} = \begin{pmatrix} l_{i,1} & \cdots & l_{i,i-1} \\ l_{i+1,1} & \cdots & l_{i+1,i-1} \end{pmatrix}, \quad \text{更新情况如式}$$

(4.14) 所示，

$$\begin{aligned} Q_z &= Z^T Q_a Z = Z^T L^T D L Z = \bar{L}^T \bar{D} \bar{L} \\ \bar{L}_{32} &= L_{32} Z_{22} \\ \bar{L}_{22}^T D_{22} \bar{L}_{21} &= Z_{22}^T L_{22}^T D_{22} L_{21} \\ \bar{L}_{21} &= L_{22} (L_{22} Z_{22})^{-1} L_{21} \end{aligned} \quad (4.14)$$

将 Z_{22} 代入式 (4.14)，得，

$$\begin{aligned}
 \bar{L}_{32} &= \begin{pmatrix} l_{i+2,i} + \partial l_{i+2,i+1} & l_{i+2,i+1} \\ l_{i+3,i} + \partial l_{i+3,i+1} & l_{i+3,i+1} \\ \vdots & \vdots \\ l_{n,i} + \partial l_{n,i+1} & l_{i+n,i+1} \end{pmatrix} \\
 \bar{L}_{22} &= \begin{pmatrix} 1 & 0 \\ l_{i+1,i} + \alpha & 1 \end{pmatrix} \\
 \bar{L}_{21} &= L_{21} \\
 \bar{D}_{22} &= D_{22}
 \end{aligned} \tag{4.15}$$

由式(4.15)最后一行所示, 经过变换, 因为只使用下三角矩阵做乘法, 不会改变 D 中元素的大小, 也就是说没有解决 D 谱的断裂性问题。为了解决谱的断裂性问题, 程序中使用了重新排序的方法, 通过使用一个矩阵 $P = P^T = \begin{pmatrix} & 1 \\ 1 & \end{pmatrix}$

与 $L^T D L$ 做乘法, 即 $P^T L_{22}^T D_{22} L_{22} P$, 实现 D_{22} 中两对角元素的互换, 从而达到目的。

但这种交换不是每次都进行, 只有当做 $P^T L_{22}^T D_{22} L_{22} P$ 变换后, 新的 $d'_{i+1} < d_{i+1}$ 时候才进行。当然, 通过交换改变 D_{22} 中元素的顺序, 同时也要更新 L 、 Z 和 \hat{a} 。

变换后的 L, D 阵中元素变化如下,

$$\begin{aligned}
 d'_{i+1} &= d_i + l_{i+1,i}^2 d_{i+1} \\
 d'_i &= \frac{d_i}{d'_{i+1}} d_{i+1} \\
 l'_{i+1,i} &= \frac{d_{i+1}}{d'_{i+1}} l_{i+1,i} \\
 L_{21} &= \begin{pmatrix} -l_{i+1,i} & 1 \\ \frac{d_i}{d'_{i+1}} & l'_{i+1,i} \end{pmatrix}
 \end{aligned} \tag{4.16}$$

之所以交换 D_{22} 中元素会达到降相关的效果, 可以从条件方差的角度分析, 程序中降相关的顺序为从后向前, 即从第 n 个模糊度开始倒序进行, 就条件方差而言, 第 i 个条件方差以 $i+1 \sim n$ 模糊度为条件求取, 交换后, 第 $i+1$ 个条件模糊度改变了, 则自然第 i 个条件模糊度也会随之改变, 当满足改变后的模糊度 $d'_{i+1} < d_{i+1}$, 则改变后的 d'_i 会变些大些, 交换后相邻的条件方差 d'_i 和 d'_{i+1} 差距会减小, 这样相当于减少了原始 D 阵中的谱断裂性, 也就达到了最初的目的。

当然置换后，并不能一定保证 $\bar{d}_k > \bar{d}_{k+1}$ ，但可以证明，置换后 \bar{d}_k, \bar{d}_{k+1} 在数值上的差异一定小于原始的 d_k, d_{k+1} 。需要注意的是置换后 $l_{i+1,i}$ 的更新值

$$l'_{i+1,i} = \frac{d_{i+1}}{d'_{i+1}} l_{i+1,i}, \text{ 之前小于 } 1/2 \text{ 的 } l_{i+1,i} \text{ 更新后有可能会再次大于 } 1/2, \text{ 为了使 } L \text{ 中元}$$

素全部小于 $1/2$ ，程序中需要进行多次迭代。

如上所述，乘以下三角矩阵 Z_{22} 消除了相邻模糊度的相关性，但是并没有对条件方差产生影响，而交换条件方差后对原始条件方差产生了影响，逐渐消除了谱的断裂性。需要注意的是，后者需要前者配合才能达到预期的效果，否则如果只做交换，谱的断裂性不会改善。

当然，无论是与 Z_{22} 相乘还是与 P 相乘， L 均产生变化，对应的 L_{21}, L_{22}, L_{32} 值会改变，在程序实现时均需要进行不断更新。

上面所说的两个过程，去相关和去断裂性，均是在相邻两模糊度之间使用二阶矩阵不断运算，并都会对 Z 产生影响。两个过程最终目的是要得到 Z 阵，也就是模糊度转换矩阵。

整个程序的大致过程如下，算法从 n 到 1 依次进行，首先进行去相关，然后在每个循环 i 中检查是否有 $d'_{i+1} < d_i$ ，如果满足，则进行交换，也就是做 $P^T L_{22}^T D_{22} L_{22} P$ 运算，每次交换后，搜索重新由 n 开始，如果从 n 到 1 没有依次交换发生，则算法终止，这个过程中需要 $l_{i+1,i}$ 尽可能小，因此需要 L 的非对角线元素小于 0.5。

4.2.3 条件模糊度转换以及搜索过程

经过上节的 Z 阵求解，原始模糊度向量 \hat{a} 和其对应协方差矩阵 $Q_{\hat{a}}$ 与 Z 阵相乘，得到新的模糊度向量 \hat{z} 及其对应的协方差阵 $Q_{\hat{z}}$ ，尽管很大程度的消除了相关性，但 $Q_{\hat{z}}$ 依然为非对角阵，相对于 $Q_{\hat{a}}$ ， $Q_{\hat{z}}$ 只是非对角线的元素在数值上减少了很多，其值均在 0.5 以下，但 $Q_{\hat{z}}$ 仍不为对角阵，式 (4.8) 还是不可以简单展开成一组平方项的和，但对 $Q_{\hat{z}}$ 继续求取条件模糊度后，式 (4.8) 就可以将之展开成平方项和的形式，从而有效进行搜索。在数学上，求取条件模糊度过程等同于对 $Q_{\hat{z}}$ 进行 LDL^T 分解， D 矩阵即为条件模糊度 $\hat{z}_{i|1}$ 的方差阵，由 n 到 1 对 $Q_{\hat{z}}$ 进行条件分解的过程可由式 (4.17) 表达为^[54]，

$$\begin{cases} \hat{z}_n = \hat{z}_n \\ \hat{z}_{n-1|n} = \hat{z}_n - \delta_{\hat{z}(n,n-1)} \delta_{\hat{z}(n,n)}^{-1} (\hat{z}_n - z_n) \\ \hat{z}_{n-2|n-1,n} = \hat{z}_{n-2} - \delta_{\hat{z}(n,n-2)} \delta_{\hat{z}(n,n)}^{-1} (\hat{z}_n - z_n) - \delta_{\hat{z}(n-2,n-1|n)} \delta_{\hat{z}(n-1|n,n-1|n)}^{-1} (\hat{z}_{n-1|n} - z_{n-1}) \\ \vdots \\ \hat{z}_{i|I} = \hat{z}_i - \sum_{j=i+1}^n \delta_{\hat{z}(j|I,i)} \delta_{\hat{z}(j|I,j|I)}^{-1} (\hat{z}_{j|I} - z_j) \\ I = i+1, i+2, \dots, n \end{cases} \quad (4.16)$$

将式 (4.17) 最后一项进行变换, 就可进一步得到式 (4.18),

$$\begin{cases} (\hat{z}_i - z_i) = (\hat{z}_{i|I} - z_i) + \sum_{j=i+1}^n \delta_{\hat{z}(j|I,i)} \delta_{\hat{z}(j|I,j|I)}^{-1} (\hat{z}_{j|I} - z_j) \\ (\hat{z} - z) = L(\hat{d} - z), Q_{\hat{z}} = LDL^* \end{cases} \quad (4.17)$$

其中,

$$\begin{aligned} \hat{d} &= (\hat{z}_1, \hat{z}_{2|1}, \hat{z}_{3|2,1}, \dots, \hat{z}_{n|N})^* \\ \begin{cases} L(i, j) = 0, 1 \leq i < j \leq n \\ L(i, j) = 1, i = j \\ L(i, j) = \delta_{\hat{z}(j|I,i)} \delta_{\hat{z}(j|I,j|I)}^{-1}, 1 \leq j < i \leq n \end{cases} \\ \hat{z}_{i|I} &= \hat{z}_{i|I-1, i-2, \dots, 1} \end{aligned}$$

求取条件模糊度后, 式 (4.8) 可以转换为,

$$\min_{z_1, \dots, z_n \in Z} \sum_{i=1}^n (\hat{z}_{i|I} - z_i)^2 / \delta_{\hat{z}(i|I, i|I)} \quad (4.18)$$

设定 χ^2 为不等式边界条件, 则式 (4.19) 表达为,

$$\sum_{i=1}^n (\hat{z}_{i|I} - z_i)^2 / \delta_{\hat{z}(i|I, i|I)} < \chi^2 \quad (4.19)$$

χ^2 决定着模糊度搜索范围。这里需要说明的是, 式 (4.19) 中不能简单的将 $\hat{z}_{i|I}$ 取整得到 z_i , 这是因为 $\hat{z}_{i|I}$ 的值依赖于 $z_j, j = i+1, i+2, \dots, n$, 不仅与 z_i 有关, 所以直接取整将没有意义。

为了确定模糊度搜索范围, 需要确定 χ^2 的大小, χ^2 的确定会在后面介绍

到，此处仅将其当做常数。

结合式 (4.17)，式 (4.19) 可转换为，

$$\sum_{i=1}^n d_i \left[(z_i - \hat{z}_i) + \sum_{j=i+1}^n l_{ji} (z_j - \hat{z}_j) \right]^2 \leq \chi^2, d_i = \delta_{\hat{z}(i|I, i|I)} \quad (4.20)$$

因为搜索时顺序为 $n \sim 1$ ，所以，对于第 i 个模糊度，已经经历了从 $n \sim i+1$ 的模糊度搜索。此时 $\hat{z}_{i|I} = \hat{z}_{i|i+1 \dots n}$ 。

由 (4.20)，第 i 个模糊度边界设定为，

$$\underbrace{\left[(z_i - \hat{z}_i) + \sum_{j=i+1}^n l_{ji} (z_j - \hat{z}_j) \right]^2}_{left_i} \leq \frac{\chi^2}{d_i} - \underbrace{\frac{1}{d_i} \sum_{l=i+1}^n d_l \left[(z_l - \hat{z}_l) + \sum_{j=l+1}^n l_{jl} (z_j - \hat{z}_j) \right]^2}_{right_i}, i \in [1, n-1] \quad (4.21)$$

简化式 (4.21)，

$$\hat{z}_i - \sqrt{right_i} - \sum_{j=i+1}^n l_{ji} (z_j - \hat{z}_j) \leq z_i \leq \hat{z}_i + \sqrt{right_i} - \sum_{j=i+1}^n l_{ji} (z_j - \hat{z}_j) \quad (4.22)$$

确定搜索范围后，下一步就是搜索。在第 i 级的搜索上，当选定一个候选值 z_i ，就可以确定 z_{i-1} 的边界，如果由式 (4.22)，没有满足条件的 z_{i-1} ，则返回 z_i ，尝试使用 z_i 的下一个候选值来继续确定 z_{i-1} 边界。当搜索进行到第 1 级上，找到了满足搜索范围条件的 z_1 ，则一组模糊度候选值 z 就算确定下来了。当所有满足边界条件的 z 都搜索出来，搜索过程结束。

搜索过程有两种，逐个搜索和逐渐收缩式搜索。

(1) 逐个搜索。即对式 (4.22) 中每个模糊度的搜索范围内的整数从左到右依次尝试，得到所有的候选值。当需要得到所有满足条件候选值的时候，可以使用这种方法。

(2) 也可以从搜索区间的中间值开始，也就是从最接近 \hat{z}_i 的整数开始，依次向左向右搜索，直到搜索到满足个数的候选值，当只需少数个候选值时，这种方式更有效。当然，如果找到一个候选值向量，可以使用这个候选值计算出 $\|\hat{z} - z\|_{Q_i}^2$

作为新的 χ^2 ，并在继续的搜索中使用新的 χ^2 ，从而起到收缩搜索空间的作用。

4.2.4 搜索范围的确定

接下来一个很重要的问题是如何确定 χ^2 ，它决定着搜索范围的大小， χ^2 选择太小，无法保证正确的模糊度组合在搜索范围内， χ^2 如果选择过大，搜索效率就会变低，同时， χ^2 的选择还需满足对候选值个数的要求，它控制着搜索椭球的大小，包含 10 个候选值和包含 2 个候选值的 χ^2 大小会存在不同， χ^2 与要获得的候选值个数有关。

有两种方法可以确定 χ^2 ，下面分别加以介绍。

(1) 通过超椭球体积估算

$$k = \text{round}(E), V_n = E^n \quad (4.23)$$

式 (4.24) 中，

k 为候选值个数；

V_n 为搜索空间超椭球体积，它与 χ^2 相关联，这里的 E 相当于超椭球的半径则 χ^2 可通过 V_n 估算；

(2) 通过数值方式计算 χ^2

下文中以候选值为 2 时做代表。

式 (4.21) 左边可写为，

$$t_a = \sum_{i=1}^n d_i \left[\sum_{j=i}^n l_{ji} (a_j - \hat{a}_j) \right]^2 = \sum_{i=1}^n d_i e_i^2 \quad (4.24)$$

如果每个 a_i 都取与 \hat{a}_i 最接近的整数值，计算出的 $t_a = \chi^2$ 定能保证这个“最近”解在椭球内。如果想要保证还可以得到另一个解，可以将第 k 个模糊度人为加上整数 Δ ，这将得到新的一组模糊度值，使用这个值计算出的 χ^2 将至少能够保证 2 组解。

新模糊度表示为，

$$a_{\text{new}_k} = a + c_k \Delta, c_k = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 \sim k-1 & & & k+1 \sim n & \end{pmatrix} \quad (4.25)$$

a_{new_k} 对应的 $t_{a_{\text{new}_k}}$ 为，

$$t_{a_{\text{new}_k}} = \sum_{i=1}^k d_i (e_i + l_{ki} \Delta)^2 + \sum_{i=k+1}^n d_i e_i^2 = t_a + \sum_{i=1}^k d_i (2e_i l_{ki} \Delta + l_{ki}^2 \Delta^2) \quad (4.26)$$

当 k 从 1 变换到 n 时，将得到 n 个 $t_{a_{\text{new}_k}}$ ，结合最初的 t_a ，共有 $n+1$ 个 t ，将这 $n+1$ 个值进行从小到大排序，选择第二个值作为 χ^2 ，则这个 χ^2 一定能够保证

有两个候选值在椭球内。

4.3 改进的 LAMBDA 算法 (MODIFIED-LAMBDA) ^[55]

传统的 LAMBDA 虽能够有效求解模糊度，但因其计算较为复杂，实际应用中会占用大量计算资源。在后续不断发展中，有学者提出了改进的 LAMBDA 算法，即 M-LAMBDA 算法，M-LAMBDA 算法通过提前置换和动态调整搜索范围，能够有效提高传统 LAMBDA 的计算效率，且不会对模糊度固定成功率造成影响，考虑到效率因素，本文使用了 M-LABMDA 算法求解模糊度。图 4-1 为在不同维度下，M-LAMBDA 和 LAMBDA 算法的效率对比。

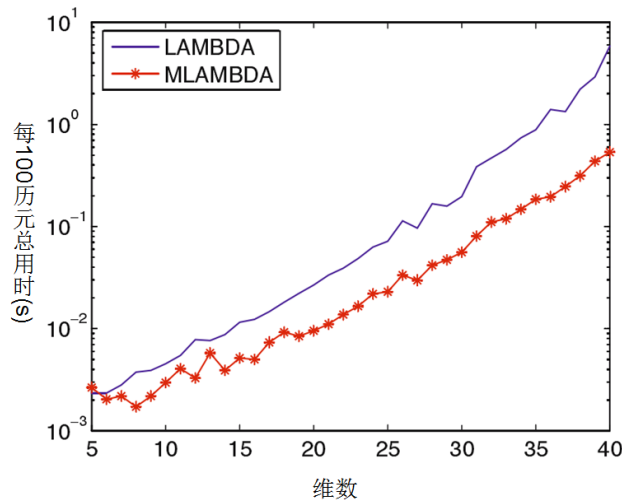


图 4-1 M-LAMBDA 和 LAMBDA 算法的效率对比^[55]

4.3.1 提前置换

转换矩阵 Z 阵的构建为的是去除原始模糊度之间的关联性，从而改善搜索空间，提高搜索速度。转换后 Z 阵对应的误差阵 Q_z 应满足如下两个条件^[56]：

- (1) Q_z 的非对角线元素要尽可能的小，要求 1/2 以下，与 0 越接近越好；
- (2) Q_z 中对角线元素应按照顺序排列，一般从 1 到 n 降序排列；

在具体实现中，置换过程非常耗时，往往牵一发而动全身，如置换第 i 个和第 $i+1$ 个模糊度，就会导致第 $i+2$ 到第 n 个模糊度都要重新置换。这个问题可以通过适当方法加以缓解。如上文所述， Q_z 对应的条件方差 \bar{D} 中对角线元素是需要按照从 1 到 n 降序排列，如果将最初的 Q_a 对角线元素提前提前降序排列，就会减少后面构建 Z 时候置换的次数，排列算法如下。

任一个对称正定的矩阵 Q 可以表达为如下形式，

$$Q = \begin{pmatrix} \tilde{Q} & q \\ q^T & q_{nn} \end{pmatrix} = \begin{pmatrix} \tilde{L}^T & l \\ & 1 \end{pmatrix} \begin{pmatrix} \tilde{D} & \\ & d_n \end{pmatrix} \begin{pmatrix} \tilde{L} & \\ l^T & 1 \end{pmatrix} \quad (4.27)$$

式(4.28)中, 有, $d_n = q_{nn}$, $l = \sqrt{q/d_n}$, $\tilde{Q} - ld_n l^T = \tilde{L}^T \tilde{D} \tilde{L}$ 。

首先, 找到 Q 阵对角线中的最小元素, 把它置换到 $Q(n,n)$ 位置上, 置换过程通过使用置换矩阵同 Q 相乘实现,

$$P_1^T Q P_1 = \begin{pmatrix} \tilde{Q} & q \\ q^T & q_{nn} \end{pmatrix} \quad (4.28)$$

令 $Q_{n-1} = \tilde{Q} - ld_n l^T$, 同样使用置换矩阵找到对角线中的最小值, 将其置于 $(n-1, n-1)$ 位置上, 置换矩阵为 P_2 。重复上述的过程, 就能够对 Q 对角线元素进行排列。这时将得到一个置换矩阵 P , P 是所有置换矩阵的乘积。

4.3.2 搜索范围的调整

如前所式, χ^2 的大小极大地影响着搜索效率, 在 Teunissen 原文中, 提到了一种能够在搜索过程中动态改变 χ^2 数值以减少模糊度搜索空间的方法, 这种方法的基本思想是, 当找到了一个符合条件的模糊度组合, 就使用这个组合重新计算一个新的 χ^2 以代替原始 χ^2 , 这会使搜索的超椭圆空间变小, 然后使余下的模糊度搜索空间变小, 从而提高搜索效率。但因为这个方法不断地以更小的 χ^2 确定搜索范围, 不能保证一定有两组以上的模糊度解, 而通常的模糊度解算最少需要保留两个模糊度解组合, 即最优解和次优解, 以便计算 ratio 以对模糊度有效性进行检核。这就需要对 Teunissen 原文中的收缩办法进行调整, 以使变换后的 χ^2 值依然可以满足最少有两个模糊度组合解。

假定我们需要求解 p 个模糊度组合候选值, 令 $\hat{z}_{i|l} = \bar{z}_i$, 由式(4.20), 最有可能的候选值为,

$$z^1 = ([\bar{z}_1] \quad [\bar{z}_2] \quad \cdots \quad [\bar{z}_n])^T \quad (4.29)$$

令 $z^2 = ([[\bar{z}_1]] \quad [\bar{z}_2] \quad \cdots \quad [\bar{z}_n])^T$, 其中 $[[\bar{z}_1]]$ 表示与 \bar{z}_1 第二近的整数, 从 z^1 到 z^2 只进行第一个元素值的改变。

由此类推, z^p 的第一个元素为与 \bar{z}_1 第 p 近的整数。之所以选择只改变第一个元素值, 是因为经过 Z 变换之后, \bar{D} 阵对角线元素按照从上到下递减排列,

$\delta_{\bar{z}(1,1)}$ 值为最大, 变换第一个元素值对 $\sum_{i=1}^n (\bar{z}_i - z_i)^2 / \delta_{\bar{z}(i,i)}$ 影响最小。

令 $f(\bar{z}) = \sum_{i=1}^n (\bar{z}_i - z_i)^2 / \delta_{\bar{z}(i,i)}$, 当搜索到一组整周模糊度组合时, 让 $\chi^2 = f(z^j)$, $j = \arg \min_{1 \leq i \leq p} (f(z^i))$, 即选定 $f(z^1)$ 到 $f(z^n)$ 的最小值作为新的 χ^2 , 这个 χ^2 就一定能够保证搜索范围内至少有 p 个候选值。

4.4 附有限制条件的 LAMBDA 算法 (CONTROLLED-LAMBDA)

对于基线求解而言, 如果已知基线长度, 那么可以充分利用这个已知条件提高模糊度固定效率。同样的, 对于姿态解算而言, 一般将接收机布置在刚体上面, 它们彼此之间的长度已知, 一个合理的能够利用这个已知条件的模糊度解算方法将更高地提高解算成功率。

4.4.1 直接约束法

目前, 已有学者已经提出一些方法利用已知的基线长度, 其中最简单的一个方式就是将它用在有效性的检测中, 如式 (4.30) 所示,

$$l - \delta l \leq \|\tilde{b}(a)\| \leq l + \delta l \quad (4.30)$$

δl 是一个较小的值, 它决定了 l 的变化范围, 当整周模糊度被固定下来, 就可以以它为条件求解基线向量 $\tilde{b}(a)$, 然后将其长度与 l 作比较, 当它们之差的绝对值超过 δl 时, 就认为这组模糊度不正确, 否则就认为这组模糊度正确。

但实际上, 构造一个合适的 δl 并不容易, 因为在已知模糊度情况下求解 $\tilde{b}(a)$, 会受噪声干扰, 这使得原本正确模糊度遭到拒绝, 而错误的模糊度解可能会被接受。

4.4.2 附有三角约束和基线长度约束的 C-LAMBDA^[12]

在搜索椭球的确定上, 按照式 (4.19), 将每个模糊度直接取整计算出的 χ^2 , 受误差的影响, 可能会漏掉正确的模糊度解, 为了保证椭球能够包含正确解, 需要增大椭球体积。增大体积可以选用式 (4.25), Δ 一般选择 1。这样做虽然确保搜索空间包含正确解, 但也扩大了搜索空间, 从而牺牲了搜索效率。此方法利用先验已知信息, 通过三角限定和基线长限定缩小搜索范围, 从而提高搜索效率。

此方法的基本步骤如下:

- 1) 找到 χ^2 使正确解包含在搜索范围内;
- 2) 在进行 LAMBDA 方法的 Z 变换后, 解算三角限制和基线长度限制的模糊度边界;
- 3) 确定每个模糊度搜索上界和下界, 上界和下界分别选择三角限定、基线长限定和传统 LAMBDA 算法边界限定中的最小值和最大值;
- 4) 搜索模糊度, 解算候选模糊度对应的基线向量长度 $\|\tilde{b}\|$;
- 5) 使用基线长度约束进行检验;
- 6) 如果有较多候选值满足条件, 则选择其中使双差观测值残差较小的两组, 进行最优和次最优候选值 ratio 检验, 通过检验的模糊度组合为最终解。

4.4.3 最小二乘约束下的 C-LAMBDA 方法^[13]

针对如何利用已知基线长度信息, 已有学者提出多种方法, 在综合分析多种方法的基础上, 本文认为 Chansik Park 等提出的 C-LAMBDA 算法更适用姿态解算, 原因如下:

- (1) 采用最小二乘思想对基线长进行约束, 算法严密, 且与 LAMBDA 算法相一致;
- (2) 通过实验验证, 表明 C-LAMBDA 能够有效提高短基线模糊度固定成功率, 见图 4-2。

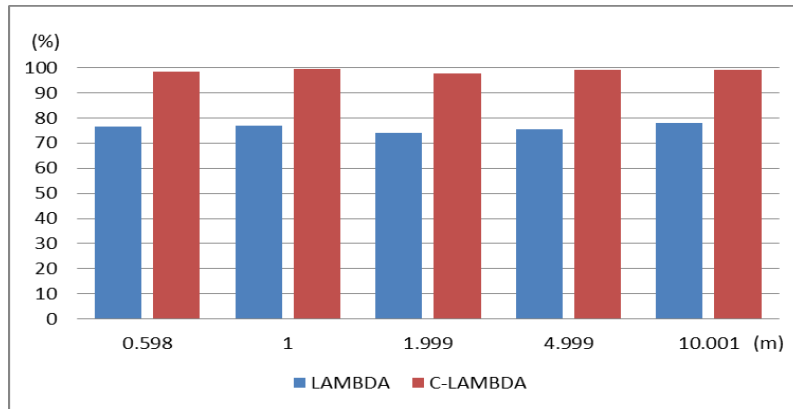


图 4-2 不同基线长度下 C-LAMBDA 与 LAMBDA 固定成功率对比^[14]

以下将对这种约束算法进行详细介绍。

已知的基线约束可以表达为 $\|b\|_l^2 = l^2$, 现在需要将之约束到最小二乘中, 根据原始最小二乘约束, 式 (4.5) 可以转化为,

$$\min_{a \in Z^3, b \in R^3} (\|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - b\|_{Q_{b(a)}}^2) \quad (4.31)$$

式 (4.31) 进一步转换为,

$$\begin{aligned} \min_{a \in Z^3, b \in R^3} (\|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2) \\ \tilde{b}(a) = \operatorname{argmin}_{\hat{b} \in R^3, \|b\|=l} (\|\hat{b}(a) - b\|_{Q_{\hat{b}(a)}}^2) \end{aligned} \quad (4.32)$$

这样, 搜索空间被定义为,

$$\|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2 \leq \chi^2 \quad (4.33)$$

对于直接的 LAMBDA 算法, χ^2 可以通过多种方法确定, 但它针对不等式中只有 $\|\hat{a} - a\|_{Q_a}^2$ 项的情况, 对于式 (4.33), 能够保证 $\|\hat{a} - a\|_{Q_a}^2$ 最小的 χ^2 不一定能够保证 $\|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2$ 在搜索范围内, 但如果使用一个较大的 χ^2 , 会导致搜索空间过大, 造成搜索困难。

一个解决的办法是首先采取一个较小的 χ_0^2 进行代入, 如果这个值不能满足 $\|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2$ 有候选值, 那么我们就逐渐增大 χ_0^2 的值, 直到出现候选值为止。

通常情况下, 使用 $\chi_0^2 = \|\hat{a} - a_{\text{ISE}}\|_{Q_a}^2$ 作为初值, a_{ISE} 为传统 LAMBDA 方法的解算值。

由此, 搜索过程可分为两步,

- (1) 使用传统 LAMBDA 算法求解满足的所有的模糊度向量候选值;
- (2) 对于每一组模糊度候选值, 计算对应的 $\hat{b}(a)$ 和 $\tilde{b}(a)$, $\hat{b}(a)$ 可按式 (3.23) 计算, $\tilde{b}(a)$ 按下文方式计算。

由以上两步可能得到不止一组的 $(a, \tilde{b}(a))$ 值, 对于这些值, 分别代入式 (4.33) 左边进行求解, 使等式左边取得最小值的即为最终结果。

接下来需要考虑如何求解 $\tilde{b}(a)$ 的值。

$\tilde{b}(a)$ 为在 $\|b\|_l^2 = l^2$ 约束下的解, 可以通过建模在三维直角坐标系下进行求解。

令 $\hat{\beta}(a) = Q_{\hat{b}(a)}^{-1/2} \hat{b}(a)$, $\beta = Q_{\hat{b}(a)}^{-1/2} b$, 则 $\min_{b \in R^3} \|\hat{b}(a) - b\|_{Q_{\hat{b}(a)}}^2$ 可以转变为 $\min_{\beta \in R^3} \|\hat{\beta}(a) - \beta\|_I^2$ 。 β 满足 $\beta^T Q_{\hat{b}(a)} \beta = l^2$, 不难看出它在几何上所表示的是一个椭圆, 则问题转换为在椭圆 $\beta^T Q_{\hat{b}(a)} \beta = l^2$ 上找到一点 $\check{\beta}$ 使 $\min_{\beta \in R^3} \|\hat{\beta}(a) - \beta\|_I^2$ 最小, 如图 4-3。

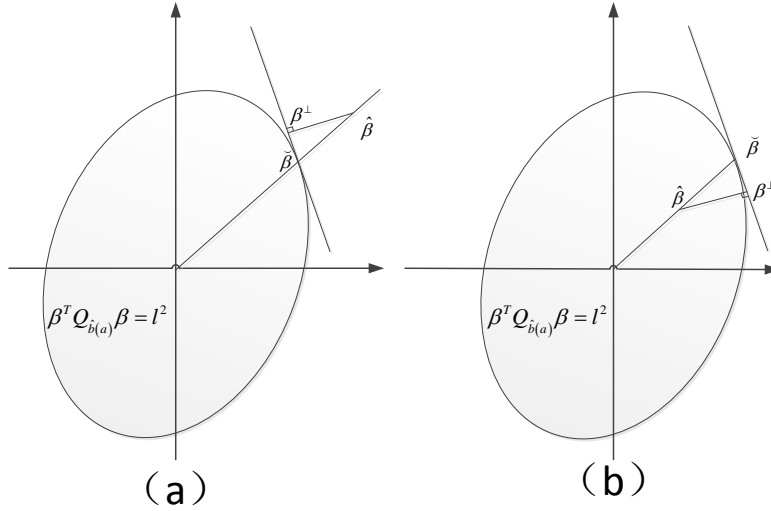


图 4-3 利用椭圆求解 $\check{\beta}$, (a) 中浮点解 $\hat{\beta}$ 在椭圆外, (b) 中浮点解 $\hat{\beta}$ 在椭圆内^[13]

$\hat{\beta}(a)$ 是超椭圆空间的一个点, 它可能在椭圆外, 也可能在椭圆内。在椭圆上一点 $\check{\beta}$ 做切平面, 过 $\hat{\beta}(a)$ 做切平面的垂线交切平面于 β^\perp 点, 在切平面上给定任意两个不重合的点 β_1 和 β_2 , 有式 (4.34) 成立,

$$\begin{aligned} \check{\beta}^T Q_{\hat{b}(a)} (\beta^\perp - \check{\beta}) &= 0 \\ (\beta_1 - \check{\beta})^T (\beta^\perp - \hat{\beta}(a)) &= 0 \\ (\beta_2 - \check{\beta})^T (\beta^\perp - \hat{\beta}(a)) &= 0 \end{aligned} \quad (4.34)$$

$\check{\beta}^T Q_{\hat{b}(a)}$ 为椭圆上 β^\perp 点处的法向量。

由式 (4.34) , 可求得,

$$\beta^\perp = \begin{pmatrix} \tilde{\beta}^T Q_{\hat{b}(a)} \\ (\beta_1 - \tilde{\beta})^T \\ (\beta_2 - \tilde{\beta})^T \end{pmatrix}^T \begin{pmatrix} \tilde{\beta}^T Q_{\hat{b}(a)} \tilde{\beta} \\ (\beta_1 - \tilde{\beta})^T \hat{b}(a) \\ (\beta_2 - \tilde{\beta})^T \hat{b}(a) \end{pmatrix} \quad (4.35)$$

式(4.35)中, β^\perp 和 $\tilde{\beta}$ 均为未知量, 而 $\tilde{\beta}$ 是想要得到的值。可以通过迭代法对式(4.35)进行求解, 迭代可以按照下面过程进行,

(1) 给出的初值, $\tilde{\beta} = \frac{l\hat{b}(a)}{\hat{b}(a)^T Q_{\hat{b}(a)} \hat{b}(a)};$

(2) 由特例法计算 β_1 和 β_2 的值。以 β_1 为例, 利用 1) 中给出的 $\tilde{\beta}$ 计算 $\tilde{\beta}^T Q_{\hat{b}(a)} = (\alpha_x \quad \alpha_y \quad \alpha_z)^T$, 考虑到 $\tilde{\beta}^T Q_{\hat{b}(a)} (\beta_1 - \tilde{\beta}) = 0$, 用特殊值法, 取为切平面与 Z 轴的交点, 则可以求出 $\beta_1 = (0 \quad 0 \quad \beta_{z1})^T$, $\beta_{z1} = \frac{\tilde{\beta}^T Q_{\hat{b}(a)} \tilde{\beta}}{\alpha_z}$ 。同理, 可以得

到 $\beta_2 = (0 \quad \beta_{y2} \quad 0)^T$, $\beta_{y2} = \frac{\tilde{\beta}^T Q_{\hat{b}(a)} \tilde{\beta}}{\alpha_y}$, β_2 为切平面于 Y 轴的交点。

(3) 计算。 $\tilde{\beta} = \frac{l\beta^\perp}{(\beta^\perp)^T Q_{\hat{b}(a)} \beta^\perp};$

(4) 计算。 $\delta l = l_{new} - l_{old}$, $l_{new} = \sqrt{\tilde{\beta}^T Q_{\hat{b}(a)} \tilde{\beta}}$, 而 l_{old} 是通过上次迭代得到的值。当 δl 的值小于预先设定的阈值时, 认为所得到的 $\tilde{\beta}$ 就是最终结果。

(5) 计算 $\tilde{b}(a)$ 。 $\tilde{b}(a) = (Q_{\hat{b}(a)}^{-1/2})^{-1} \tilde{\beta}$ 。

上面的计算量较大, 下面说明如何加速上文搜索速度。

对于每一个模糊度候选向量 a , 我们要求解其对应的 $\tilde{b}(a)$, 经过进行

$\|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2 \leq \chi^2 - \|\hat{a} - a\|_{Q_a}^2$ 的验证, 才能得到 $(a, \tilde{b}(a))$ 的全部候选值。但并不

是每一个 a 所对应的都能满足验证条件, 这样就会有很多冗余计算, 可通过一种简单方式减少冗余计算发生的概率。

数学上有式(4.36)成立,

$$\frac{(\|\hat{b}(a)\| - l)^2}{\lambda_{\max}} \leq \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2 \leq \frac{(\|\hat{b}(a)\| - l)^2}{\lambda_{\min}} \quad (4.36)$$

λ_{\max} 和 λ_{\min} 分别是 $Q_{\hat{b}(a)}$ 的最大和最小特征值, l 是已知基线的长度。

由式 (4.36) 可知, 当不能满足 $\frac{(\|\hat{b}(a)\| - l)^2}{\lambda_{\max}} \leq \chi^2 - \|\hat{a} - a\|_{Q_a}^2$, 则 $\hat{b}(a)$ 一定不能满足 $\|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2 \leq \chi^2 - \|\hat{a} - a\|_{Q_a}^2$, 这样我们就可以通过一步提前验证检验 $\tilde{b}(a)$ 是否满足最终条件, 如果不满足, 就不再计算 $\tilde{b}(a)$, 由此可以提高计算效率。

整个算法的步骤如下:

- (1) 获得基线向量和模糊度的浮点解 \hat{b} 和 \hat{a} 及其对应的方差协方差阵 $Q_{\hat{b}}$, $Q_{\hat{a}}$, $Q_{\hat{a}\hat{b}}$ 和 $Q_{\hat{b}\hat{a}}$;
 - (2) 使用普通 LAMBDA 算法计算模糊度解 a_{ISE} , 并利用它得到初始的 $\chi_0^2 = \|\hat{a} - a_{\text{ISE}}\|_{Q_a}^2$;
 - (3) 计算所有满足 $\|\hat{a} - a\|_{Q_a}^2 \leq \chi_0^2$ 的 a , 若没有足够的解, 则跳到第 7) 步。
- 计算每个 a 对应的 $\hat{b}(a) = \hat{b} - Q_{\hat{b}\hat{a}} Q_{\hat{a}}^{-1} (\hat{a} - a)$ 。
- (4) 计算 $Q_{\hat{b}(a)}$ 和它的最大特征值 λ_{\max} 。
 - (5) 验证 $\frac{(\|\hat{b}(a)\| - l)^2}{\lambda_{\max}} \leq \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2$, 如果满足这个不等式, 使用上文提到的方法计算 $\tilde{b}(a)$ 。
 - (6) 计算代价函数 $W(a) = \|\hat{a} - a\|_{Q_a}^2 + \|\hat{b}(a) - \tilde{b}(a)\|_{Q_{\hat{b}(a)}}^2$ 。
 - (7) 如果在 χ_0^2 对应的椭球范围内没能找到足够的模糊度解, 则逐步加大搜索范围, 一般令加大步长为 χ^2 , 即 $\chi^2 = \chi^2 + \chi_0^2$ 。
 - (8) 选出使 $W(a)$ 值最小的模糊度 a , 即为搜索结果。

4.5 实例分析

M-LAMBDA 算法能够提高模糊度搜索效率, C-LAMBDA 能够提高模糊度固定成功率, 为综合利用两种算法的优点, 本文提出结合使用 M-LAMBDA 算法和 C-LAMBDA 算法 (为方便, 简称为 MC-LAMBDA 算法) 固定模糊度, 为了验证算法效果, 进行了如下实验。

实验地点选择在清华大学东操场, 采用定制铁环进行, 如图 4-4, 场地周围无遮挡, 观测条件较好。实验共进 45 分钟, 本文选择了中间一段质量较好的数据进行解算, 这部分数据的 GPS 观测卫星数均为 9 颗。分别选择使用 LAMBDA 算法、M-LAMBDA 算法、C-LAMBDA 算法和 MC-LAMBDA 算法对数据进行解算, 解算完成后对模糊度解算时间和固定成功率进行统计, 见表 4-2。

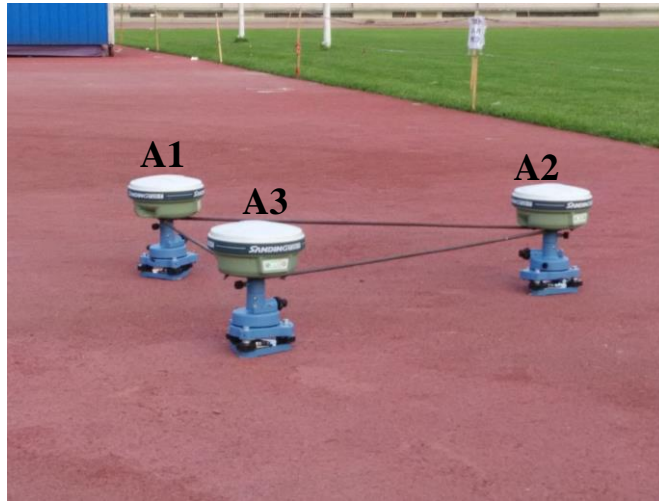


图 4-4 实验现场图

表 4-1 模糊度解算时间和固定成功率对比

基线	算法	每 100 历元所用时间 (ms)	固定率(%)
A1-A2	LAMBDA	1.08	95.6
	M-LAMBDA	0.82	95.6
	C-LAMBDA	2.31	98.1
	MC-LAMBDA	1.98	98.1
A2-A3	LAMBDA	1.42	96.4
	M-LAMBDA	1.09	96.4
	C-LAMBDA	2.93	97.7
	MC-LAMBDA	2.65	97.7

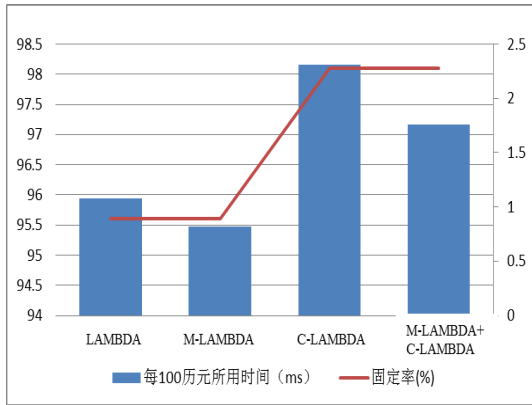


图 4-5 A1-A2 基线模糊度解算对比

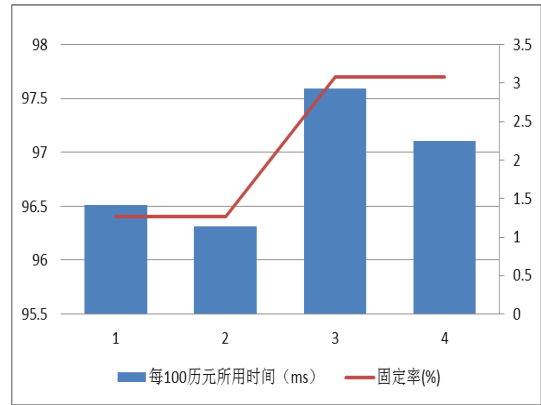


图 4-6 A2-A3 基线模糊度解算对比

分析上面数据可知，M-LAMBDA 算法能够有效提高 LAMBDA 算法的解算效率，C-LAMBDA 算法能够有效提高模糊度固定成功率，但 C-LAMBDA 因为要进行较为复杂的矩阵运算，会降低搜索效率。实验中通过结合 M-LAMBDA 算法和 C-LAMBDA 算法，在提高固定成功率的同时，能够降低因 C-LAMBDA 算法的复杂计算所带来的效率损失。

第5章 程序设计与实现

本文在理论研究的基础上,编写了 GNSS 姿态测量程序,程序在 Windows7 环境下,采用 Visual Studio 2008 编写,使用 MFC 框架,编写语言为 C/C++。

程序分为内部解算部分和外部显示部分,外部通过图形将解算结果实时展示给用户,而内部解算部分通过 DLL(动态链接库)将计算过程封装起来。这样,外部程序只需按照既定格式将参数输入给 DLL,就可以得到姿态解算结果。程序运行时,解算模块会以独立线程执行,不会影响到主程序其他操作。程序可对用户设置的参数进行可行性验证,当输入信息矛盾或者不正确时候,程序会进行相应提示。

本程序在数据读入、误差改正和基线解算方面部分调用了网络开源代码^[57-60],但均根据本文姿态解算的具体情况作了相应改进。

以下将分别从程序的内部设计和外部设计对程序进行介绍,内部设计部分重点介绍程序解算流程,外部设计主要叙述程序的模块设计与功能。

5.1 程序内部解算设计

在程序内部实现上,需要经过观测数据读取、误差改正,基线浮点解、模糊度解算、基线固定解、姿态解算、解算结果保存几个部分。其中核心算法集中在基线求解、模糊度求解和姿态求解上。下面将分别从总体技术路线和重点模块技术路线两方面对程序的内部实现进行叙述,总体技术路线为整个程序内部实现流程,通过它可以对程序解算流程有一个直观的认识;模块技术路线将从基线解算部分和姿态解算部分两个方面对程序的具体实现细节进行叙述。

5.1.1 总体技术路线

启动程序后,首先会先读入观测数据,然后进入解算过程,解算过程的重点是解算模糊度,这部分在第4章已经详细介绍过,模糊度解算成功后,程序会根据已固定的模糊度求解基线,本文使用双基线,在求解过程中,它们将分别进行进行。最后,程序将根据求解的基线解算姿态角。解算成功后,会将结果传给显示模块实时显示出来,并保存在文件中。为了调用方便和增强移植性,程序将读取观测数据部分、基线解算部分和姿态角求解部分封装到一个动态链接库(DLL)中,程序的总体技术路线如图 5-1 所示。

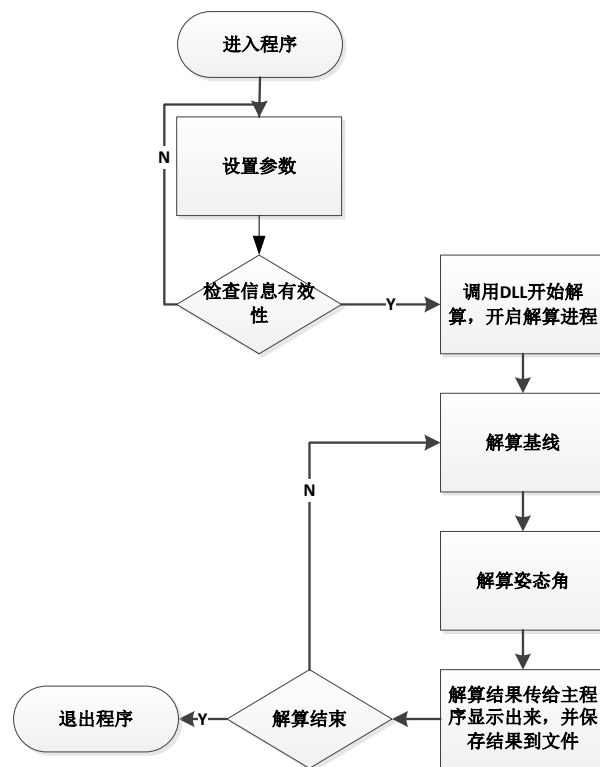


图 5-1 总体技术路线

5.1.2 模块技术路线

前文已经叙述, 对于姿态解算, 最重要的有三部分, 分别是模糊度求解、基线求解和姿态角计算, 由于在第4章已经详细说明过模糊度解算的流程和要点, 在下面部分中, 将分别说明基线解算部分和姿态角解算部分。

(1) 基线解算

基线解算是整个程序的核心部分, 决定着运算结果的精度。程序采用 Kalman 滤波解算基线, 采用 MC-LAMBDA 算法求解整周模糊度, 程序中首先通过 Kalman 滤波求解浮点基线解, 然后调用模糊度解算函数固定模糊度, 当模糊度固定成功后, 会进一步求解基线固定解。Kalman 滤波的求解模型已经在 3.3.2 节中说明, 此处不再叙述。

基线解算流程如图 5-2 所示, 在开始解算前, 要归一化三台接收机数据, 也就是让当前读入的三台接收机数据 UTC 时间相同, 同时会根据程序设置提出质量差和低于截止高度角的卫星数据。

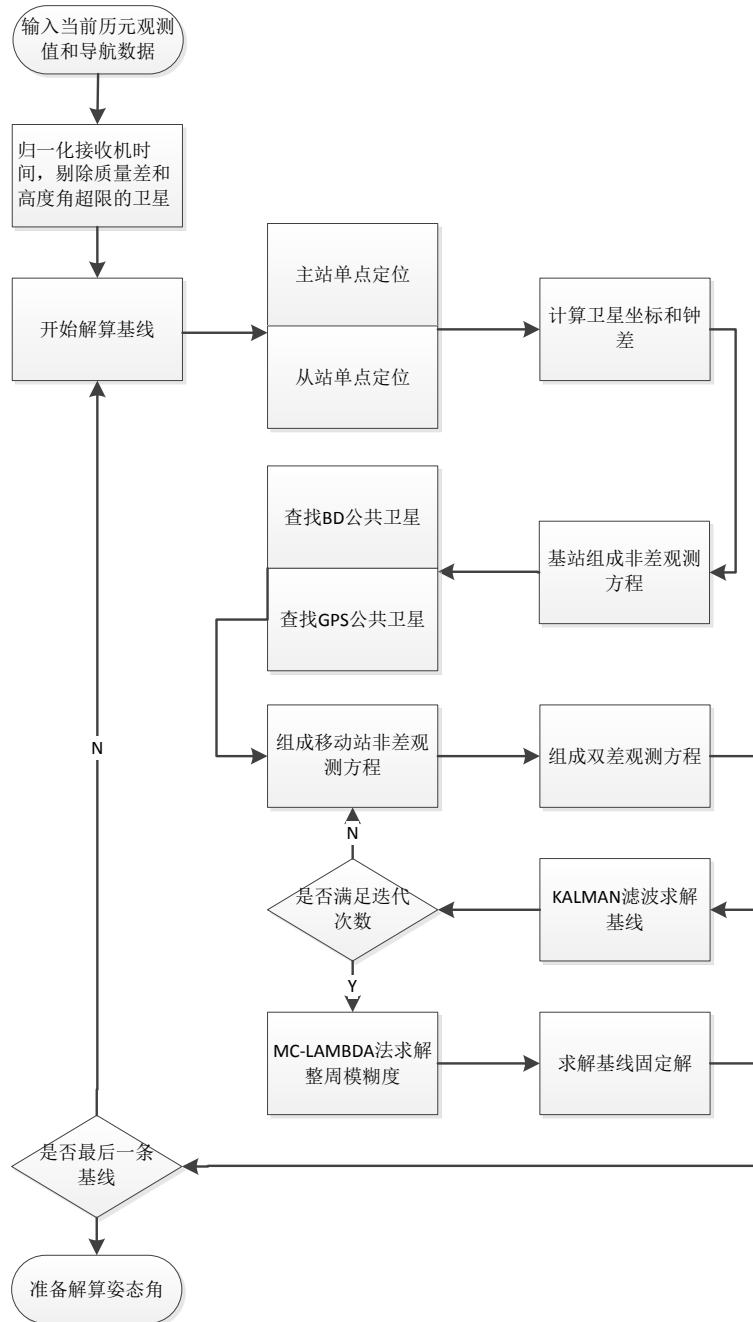


图 5-2 基线解算技术路线

(2) 姿态解算

姿态解算在基线解算成功的基础上进行，本程序提供三种姿态解算算法，分别是直接法、矩阵最小二乘法和 TRIAD 算法，第 3 章对各种算法进行了说明，下文将说明姿态解算流程。

程序中姿态解算流程如图 5-3，

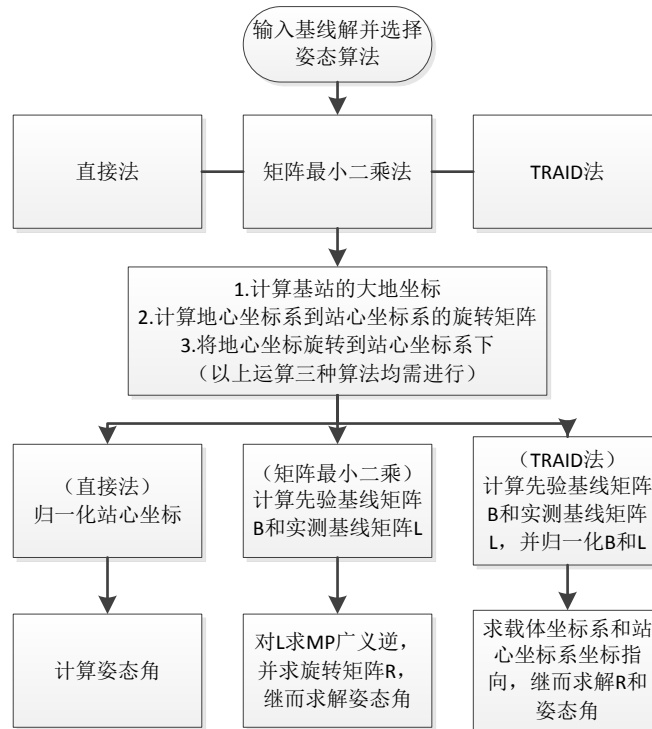


图 5-3 姿态角解算技术路线

除直接法外，其他两种方法需要提前输入先验基线长度信息。

5.2 程序外部框架设计

5.2.1 程序需求分析

(1) 可控性

载体在行进过程中，接收机会面临不同的工作环境的干扰，针对不同情况，程序应能够进行相应设定以获得最高解算精度，这突出表现在程序的设置功能上，比如，程序应能够及时调整截止高度角来适应这种变化；程序也应能对单双频进行设定，以便能同时适应单频和双频的接收机；对于多系统接收机，用户应能够根据具体情况选择卫星系统。

(2) 交互性

在结果呈现上，应该以最简单最直观的方式提供解算结果，在文字，表格，图像等多种表示方式中，以图像表达最为直观。在设计上，程序使用图像仪表盘显示解算结果，但图像只能定性表达姿态角的大小，无法知道确切的数值。而单纯以数值形式表达不够直观。所以，本程序在界面设计上，结合了表盘表达形式和数值表达形式。

(3) 效率性

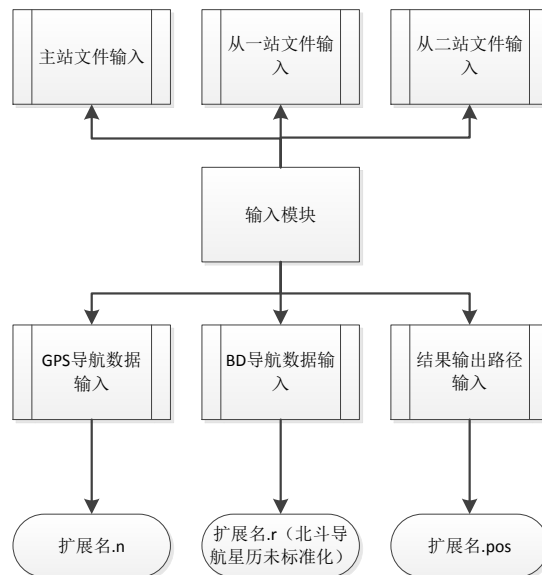
姿态解算需要实时性解算，过程中需要处理误差，解算基线和姿态角，涉及到较多的耗时运算，如矩阵求逆等，所以程序在解算上应足够快。本程序因而采用了效率较高的 C 作为编程语言，在一些地方采用了类，进而混合了 C++ 语言。

5.2.2 模块功能设计

在外部模块设计上，由输入模块，参数设置模块，控制模块组成，输入模块读入观测数据，参数设置模块针对数据源和计算过程进行参数设定，控制模块可在程序执行时的对程序进行控制，这在程序调试过程中非常重要。

(1) 输入模块

输入模块可输入三方面信息，分别是观测数据输入，导航星历文件和解算结果文件路径。其中输入文件包括主站文件、从一站文件和从二站文件，导航文件需输入主站的 GPS 导航文件和 BDS 导航文件，程序中输入模块部分如图 5-4。



(2) 设置模块

设置模块包含程序解算过程中的关键参数设置，可对卫星系统、使用频率、滤波迭代次数和姿态角求解算法等进行设置，也可对先验基线长度、截止高度角等信息进行输入，程序中设置模块部分如图 5-5 所示。

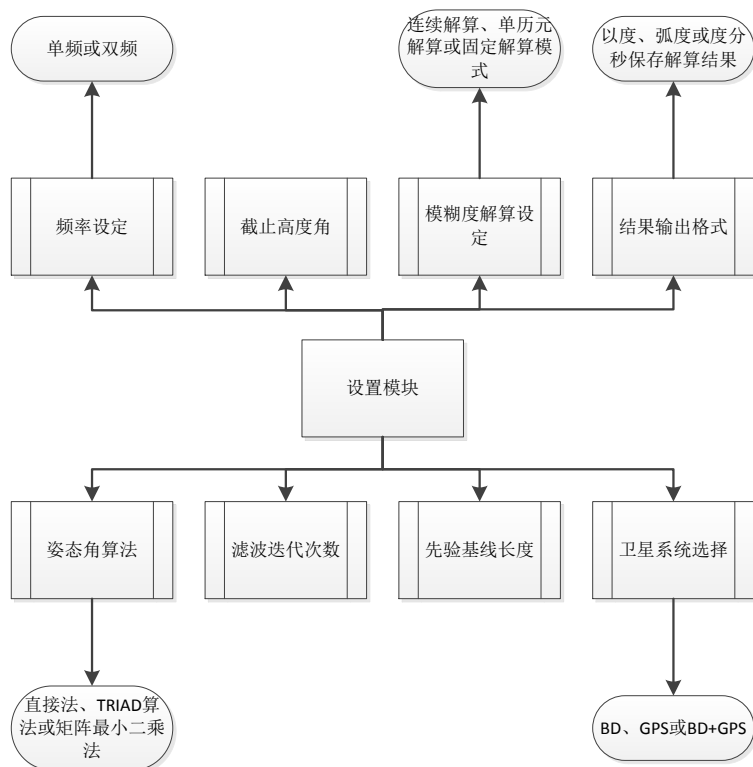


图 5-5 设置模块说明

(3) 控制模块

过程控制模块可以对已经开始的测姿系统进行控制，让其在某个历元暂停或停止，以及暂停后的继续解算，其目的在于动态监控解算结果，方便程序调试，程序中控制模块部分如图 5-6 所示。

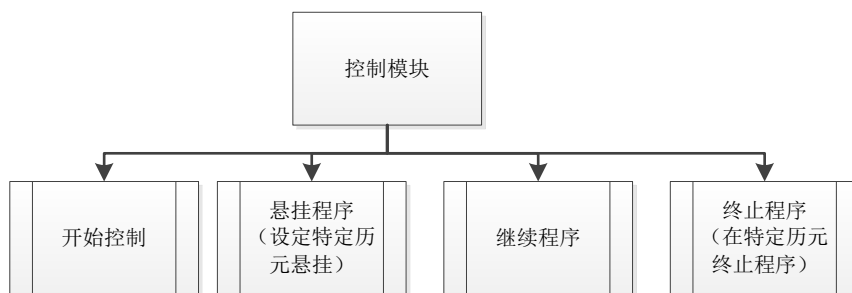


图 5-6 控制模块说明

5.2.3 程序界面设计

程序主界面如图 5-7，共设置了五个按钮，【读入】按钮为读入 Rinex 格式数据，读入之前需要提前指定文件路径。读入文件成功后，单击【开始】按钮开始运行测姿程序。【信息】按钮可以显示或隐藏的下方的解算信息。【控制】按钮

可以设定暂停或者停止历元，使程序在指定历元暂停或者停止。

程序主界面采用三个指针面板显示姿态角解算结果，设置红色指针面板表示偏航角，蓝色指针面板表示俯仰角，绿色指针表示翻滚角，为了准确表达当前姿态角数值，在指针面板上还同时输出姿态角数值，角度值以度为单位显示，以便获得精确解算结果。在主界面的信息板上，程序能够实时显示当前历元的 UTC 时间、模糊度固定情况等信息，某些在解算过程中会产生变化的数值，程序会根据解算结果进行实时更新，如模糊度会根据具体情况将状态显示为“固定解”，“浮点解”和“无解”。

程序主界面如图 5-7 所示，

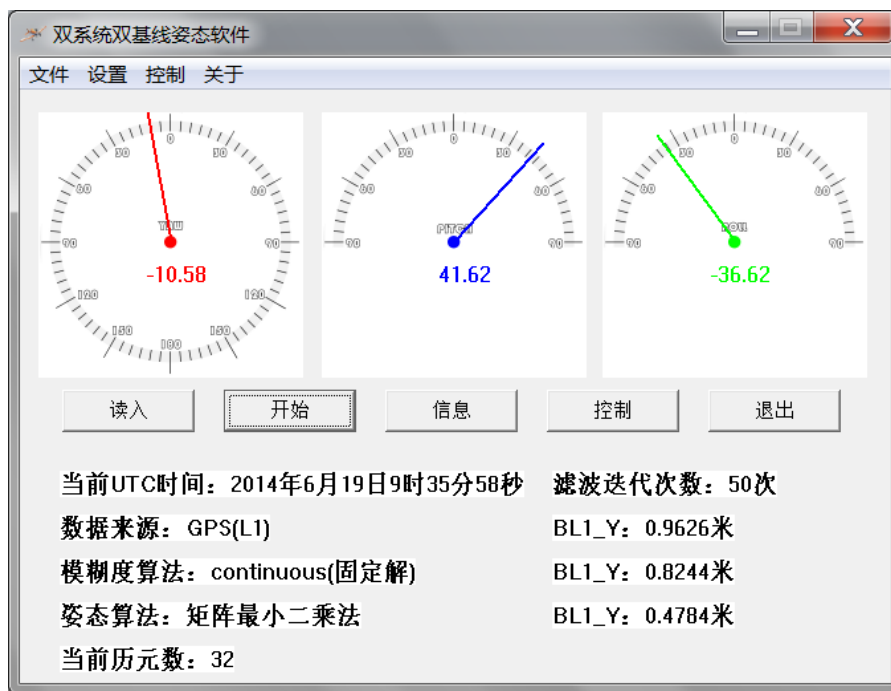


图 5-7 程序主界面

设置对话框允许用户输入解算控制信息，以便控制解算精度和解算结果输出格式。设置参数包括卫星系统、卫星频率、卫星截止高度角、解算结果输出单位和姿态算法等，实际操作中可根据具体情况进行选择。部分姿态算法需要输入已知先验基线信息，本文提供的直接法、TRIAD 和最小二乘矩阵法三种算法解算姿态，除直接法无需输入先验基线信息外，其他姿态算法均需输入先验基线信息。在基线解算过程中，使用了迭代运算，故程序中需要设定迭代次数，一般情况下，迭代次数越多，结果越精确，但运算将更为耗时。需要注意的是，当迭代达到一定次数，增加迭代次数将无法有效提高解算精度，程序设置对话框如图 5-8 所示，

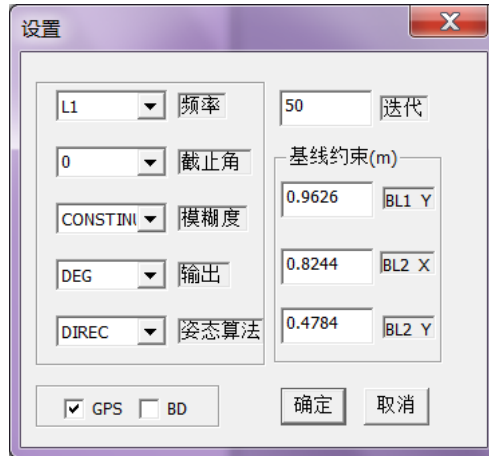


图 5-8 程序设置对话框

输入对话框为程序提供观测值和导航信息，并可设置解算结果文件保存路径。主站接收机为三台接收机中被其他两台共用的接收机，从一站与主站构成的基线为载体前进方向，亦为载体坐标系的Y轴方向。从二站与主站构成的基线位于XOY平面内，与Y轴构成一定角度。程序应当提供三台接收机的观测信息和主站的导航文件信息才能开启，否则，程序将提醒输入信息不足。当只使用BDS或者是GPS单系统解算时，可以只提供单系统观测数据文件和导航信息文件。

输入对话框如图5-9所示，



图 5-9 程序文件对话框

用户可根据需求在特定历元下对程序进程进行控制，使之处于开始、悬挂、继续和停止状态，这在程序调试过程中非常重要。能够及时观察到特定历元的姿态解算情况。图 5-10 为菜单栏中的控制选项，图 5-11 为设置具体暂停和停止历元。

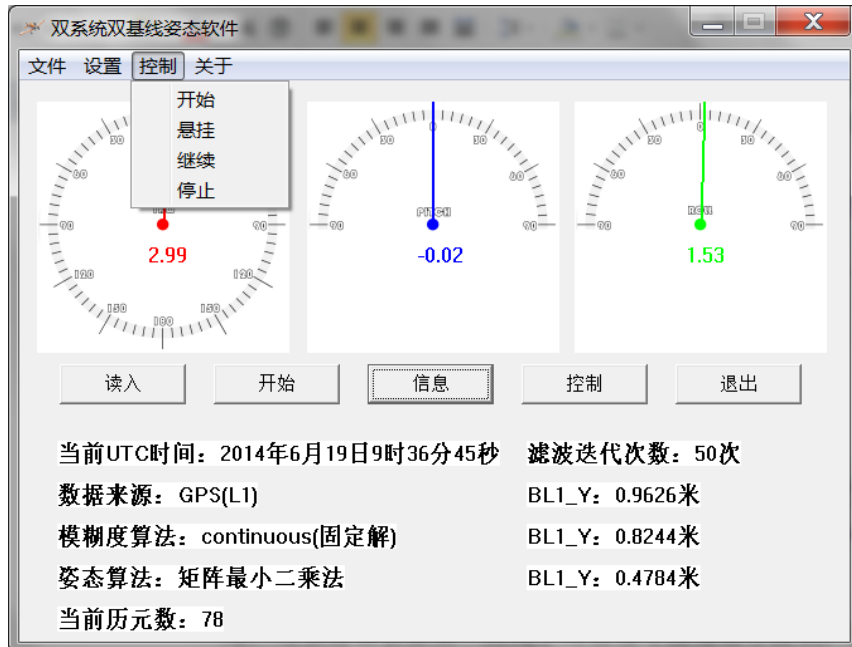


图 5-10 程序控制界面



图 5-11 在特定历元暂停或停止程序

第6章 实验研究与分析

本章实验分为两部分，第一部分为测试实验，目的是对所编写的软件在实际情景中的应用情况进行测试，分为准动态测试和动态测试，准动态测试为在静止条件下阶段性改变系统姿态，以分析系统对载体姿态变化的敏感程度。动态测试为车载实验，在运动情况下接收数据，通过解算数据来分析解算情况。第二部分为对比实验，共进行了2组实验，一组实验用来对比 BDS 和 GPS 系统在姿态解算中的差异，另一组实验用来对比三种姿态算法的解算情况。本章实验采样间隔均为1秒。

6.1 实验目的和设计

图 6.1 和图 6.2 为荷兰著名学者、LAMBDA 算法的提出者 Teunissen 教授对静态和动态情况下所做的姿态解算实验，它们对本文实验有一定启示作用。Teunissen 在两个实验中均使用三台接收机搭建测姿系统，静态实验的接收机使用木条固定以保证相对位置不变，动态机载实验则直接将接收机固定到了机身和机翼上。本文也使用三台接收机构成测姿系统，在固定上，使用特制的钢制模具，下文中会进行描述。

图 6.1 和图 6.2 分别为 Teunissen 教授所做的静态和机载实验。

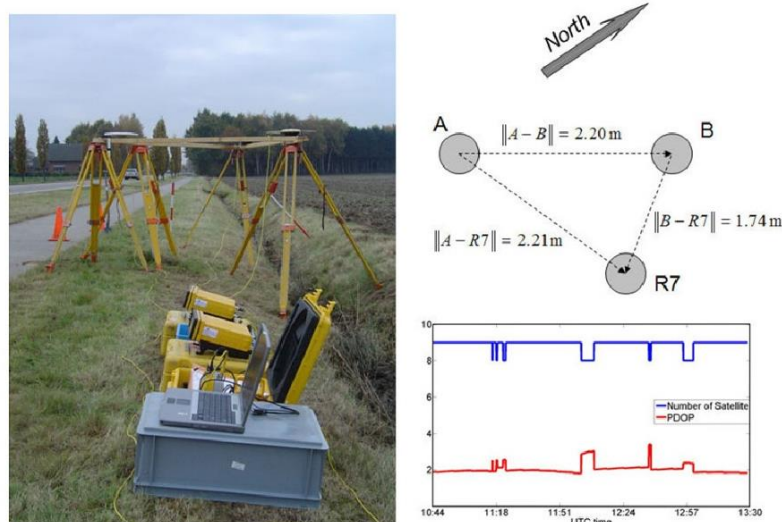


图 6-1 Teunissen 教授进行的静态测姿实验^[14]



图 6-2 Teunissen 教授进行的机载测姿实验^[14]

表 6-1 介绍了本章实验的实验目的和方式。

表 6-1 本章实验说明

类别	实验变量	实验目的	实验方式
测试实验	准动态	分析低速载体的姿态的解算情况	在静态条件下阶段性改变测姿系统姿态，观察并分析结果的稳定性和精度
	动态	分析车载运动下姿态系统的解算情况	置测姿系统与车顶，接收、解算并分析解算结果
对比实验	导航系统	对比 BDS 和 GPS 测姿的精度和稳定性；研究 GNSS 对测姿的影响	对同一时段相同接收机内的不同系统的数据分别解算，并分析解算结果；对同时段数据利用 BDS+GPS 进行解算，并分析解算结果
	姿态算法	对比不同姿态算法的解算情况	使用不同姿态算法解算相同观测数据，并分析解算结果

对实验的解算结果的统计分析使用了一些数学指标，分别是姿态角的平均值、标准差、无效数据剔除率和最大误差，其中，平均值的差异表现出解算结果的系统差异；标准差的对比显示了解算结果的稳定程度；数据剔除率是以所有观测数据为样本，求取中误差，极限误差选取三倍中误差，落在极限误差外的数据视为无效数据，进行淘汰，数据剔除率是淘汰数据占全部数据的比率，显示了测量数据的有效比例；最大误差在一定程度上能够反映不同系统和不同算法带来的粗差情况，它是剔除了无效数据后，有效数据与平均值的最大差值。实验中的统计结果除数据剔除率以百分比率表示外，其余均以度为单位。

6.2 测试实验

测试实验分为两部分，即准动态实验和动态试验，前者是在载体静止条件下

每隔一段时间改变载体姿态角进行观测；后者是将测姿系统安装到车顶，进行车载动态试验。这两部分实验将分别考察本文所编写软件在准动态和动态情况下解算的精度和准确性。

6.2.1 准动态实验

实验采用定制的等边铁环进行，准动态观测为静态观测，但每隔一段时间改变一次铁环姿态。

(1) 实验信息

实验开始时间：北京时间 2014 年 12 月 15 号 15 点 44 分

实验结束时间：北京时间 2014 年 12 月 15 号 16 点 43 分

实验地点：清华大学土木工程系测量所楼顶

仪器型号：国产测量型双频接收机，图 6.3 和图 6.4 为实验现场图。



图 6-3 实验现场图 a



图 6-4 实验现场图 b

(2) 实验结果

解算实验数据，得到三姿态角信息如下，

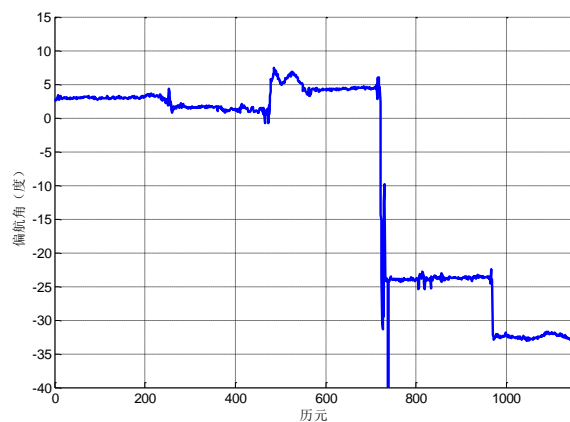


图 6-5 偏航角结果图

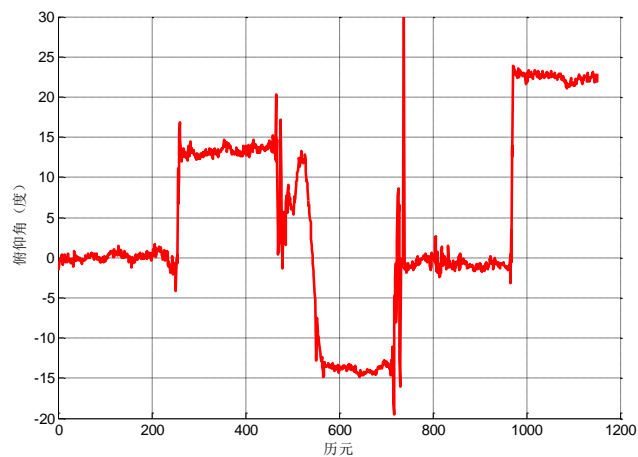


图 6-6 俯仰角结果图

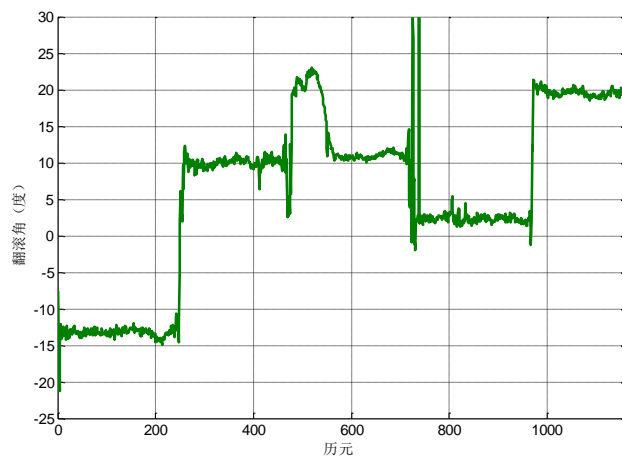


图 6-7 翻滚角结果图

分别截取上图中 5 段观测数据中较为稳定的部分进行统计得到表 6-2。

表 6-2 各段姿态角解算统计信息

观测时段	姿态角	方差	最大误差	对应历元
第 1 段	yaw	0.2115	0.4041	
	pitch	0.7442	1.055	10-110
	roll	0.9685	1.142	
第 2 段	yaw	0.1854	0.3447	
	pitch	0.9854	1.3363	280-320
	roll	1.0227	1.1033	

续表 6-3 各段姿态角解算统计信息

观测时段	姿态角	方差	最大误差	对应历元
第 3 段	yaw	0.2474	0.2554	580-620
	pitch	0.5746	0.6551	
	roll	0.7665	0.8003	
第 4 段	yaw	0.4251	0.5421	900-960
	pitch	0.8742	1.0632	
	roll	1.3551	1.7453	
第 5 段	yaw	0.4623	0.5521	990-1070
	pitch	0.9233	1.1024	
	roll	0.8565	1.0218	

(3) 实验分析

从图中可以看出，姿态角曲线分段明显，能够及时反映姿态角变化。每个分段间能够保证解算结果的稳定性。分段曲线之间会出现短暂的不稳定现象，一般持续十几个历元，这是由于人工调整姿态角时遮挡信号导致的。三个姿态角分段能够严格的对其，表现出对载体姿态变换的同步性。430 到 480 历元间出现了较严重的不稳定情况，这是在更换载体姿态时候遮挡了较长时间的信号，导致模糊度进行了重新固定。可以看出，每段观测值在解算稳定之后，波动不明显，均保持了较高的稳定性。

在实验过程中，因缺乏真值对比，无法确定解算的真实误差，从图中解算结果曲线可以得出，只要模糊度正确解算，就能够得到正确的姿态角，能够及时反映载体姿态的变换，否则，将会出现较大误差。从方差数值上看，除第 4 段数据的 roll 值外，各段数据中的三个姿态角方差均小于 1 度，呈现出较大稳定性。从最大误差方面，各段数据中无较大值出现，表现出较好抗差能力。

实际上，这段实验数据的可用卫星数较少，质量不是很好，但正因此，这段数据具有普遍性，能够代表软件在较差数据下的解算情况。

6.2.2 动态实验

此实验目的为测试软件在动态情况的解算情况。实验以车载形式进行，将定制的铁环安装到汽车车顶，以汽车为载体进行动态测试。实验地点选在北京市西五环的一段路段，为避免交通拥堵，实验时间选择在晚上十点。汽车以中低速运行在道路上，车顶放置的铁环会随着道路的起伏和弯度而呈现不同的姿态，经过

处理可得到汽车在行驶过程中的姿态信息。实验目的在于监测在动态情况下，程序解算变化姿态角的精度和稳定性能力。

(1) 实验信息

实验开始时间：北京时间 2014 年 12 月 29 号 21 点 52 分

实验结束时间：北京时间 2014 年 12 月 29 号 23 点 02 分

实验地点：北京西五环，闵西桥至宛平桥段

实验环境：晴天，车流量偏大。因为要紧贴路边行驶，在车的一侧始终存在较大遮挡角，车的另一侧视野良好。偶尔经过过街天桥会遮挡信号，造成信号间断。整段路线长度为十五公里。

汽车速度：20km/h~60 km/h

仪器型号：国产测量型双频接收机

图 6.8 为实验路线示意图，图 6.9 为车载示意图。

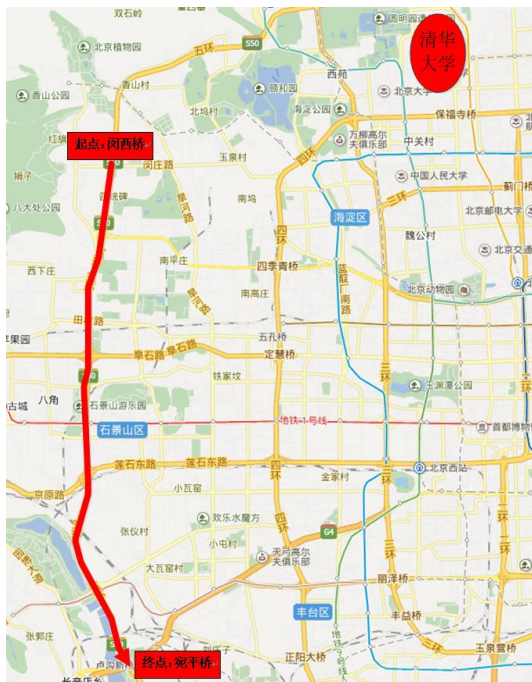


图 6-8 路线示意图

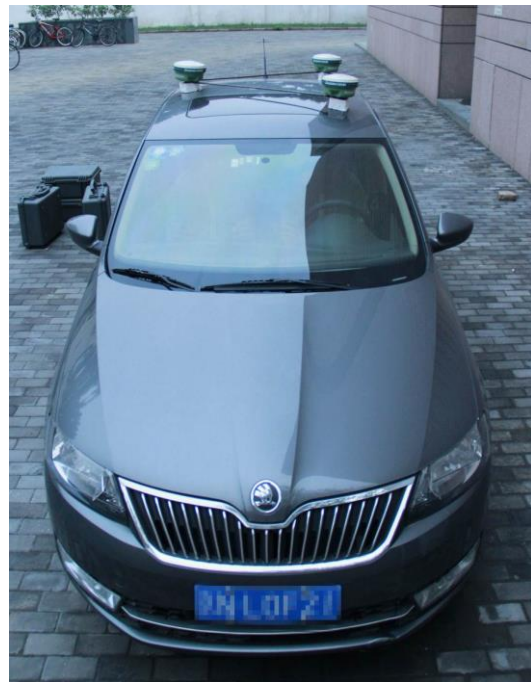


图 6-9 车载示意图

(2) 实验结果

因实验路段存在天桥、路边挡板等遮挡，实验中出现大量无效数据，加上车速比较快，相邻两次遮挡之间时间间隔比较小，而每一次遮挡都要重新解算模糊度，这导致有效数据更加难以获得。图 6.10 到图 6.12 为一段时间的数据解算结果，作为代表。

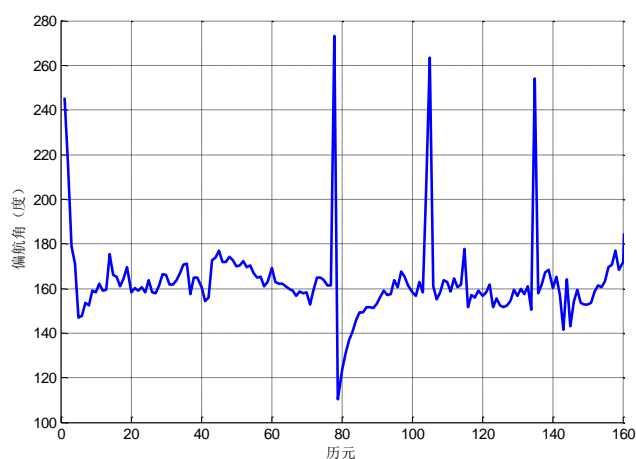


图 6-10 偏航角结果图

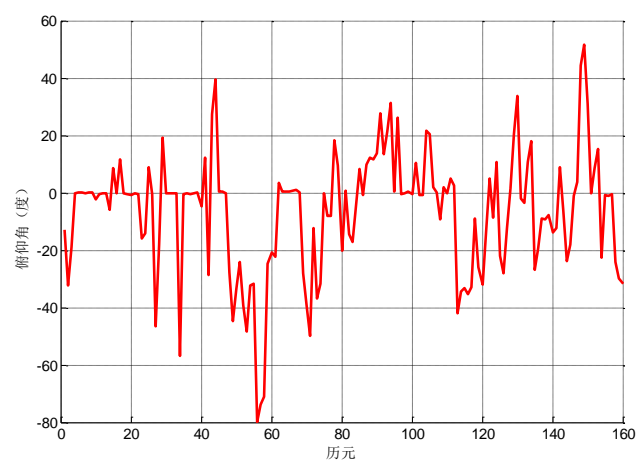


图 6-11 俯仰角结果图

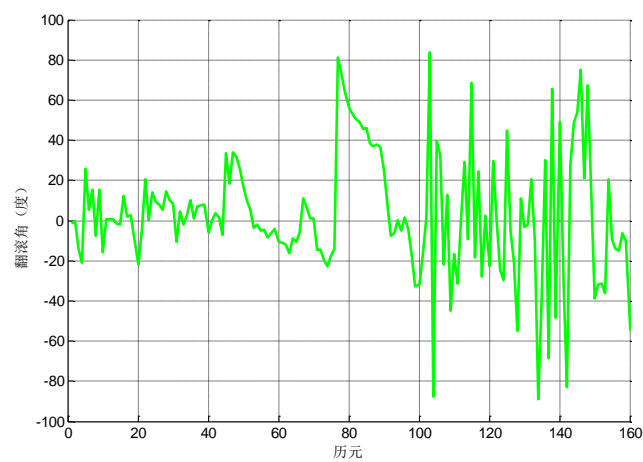


图 6-12 翻滚角角结果图

(3) 实验分析

动态结果受环境影响更大，解算结果远不如静态测量，同时因实验遮挡影响，需要不断重新解算模糊度，无法获得整段很好的观测数据。实验所用这一段解算结果比较典型，能够代表解算情况。数据的前部相对稳定，能够恰当反映当前的姿态角，可以看出偏航角 YAW 在 160 度附近，而俯仰角 PITCH 和翻滚角 ROLL 均在 0 度附近，这与实际情况相符，但数据依然具有较大波动，误差达到 10 度级别，精度较静态相差一百倍数量级。数据的后半部波动更为剧烈，不能反映实际情况，均为无效数据。

在城市中使用 GNSS 定位存在较大干扰，姿态测量需要实时解算模糊度，具备较大困难。

6.3 对比实验

这部分实验均为静态实验，实验中接收机位置保持不变，周边的观测环境也保持不变，能够得到稳定的观测值，在理想的情况下，静态实验中的姿态角度应保持不变。但受到观测噪声和算法的影响，解算结果会呈现出一定的波动和差异，对静态输出结果进行统计分析，可以得到算法的精确度和稳定度等信息。实验设置了 2 组静态观测，分别对不同系统，不同姿态角算法进行了对比分析。

6.3.1 GPS/BDS 系统对比实验

此实验目的之一为对比分析 BDS 和 GPS 的姿态测量情况，目的之二为研究 BDS+GPS 对姿态测量的影响。实验采用自制等边铁环进行，铁环边长约为 1 米。接收机可以同时接收 GPS 和 BDS 双系统数据，采集结束后，分别使用本文所编写软件进行解算，并对比分析解算结果，通过统计分析 BDS 和 GPS 系统在姿态解算上的精度和稳定性，在此基础上，进一步分析 BDS+GPS 对姿态测量的影响^[61]。

(1) 实验基本信息

实验开始时间：北京时间 2015 年 3 月 25 号 9 点 16 分

实验结束时间：北京时间 2015 年 3 月 25 号 10 点 07 分

实验地点：清华大学土木工程系测量所楼顶

仪器型号：国产测量型双频接收机

(2) 实验结果

图 6-13 到图 6-15 中，分别是 BDS 与 GPS 的三姿态角解算结果和它们与 BDS+GPS 解算结果对比。

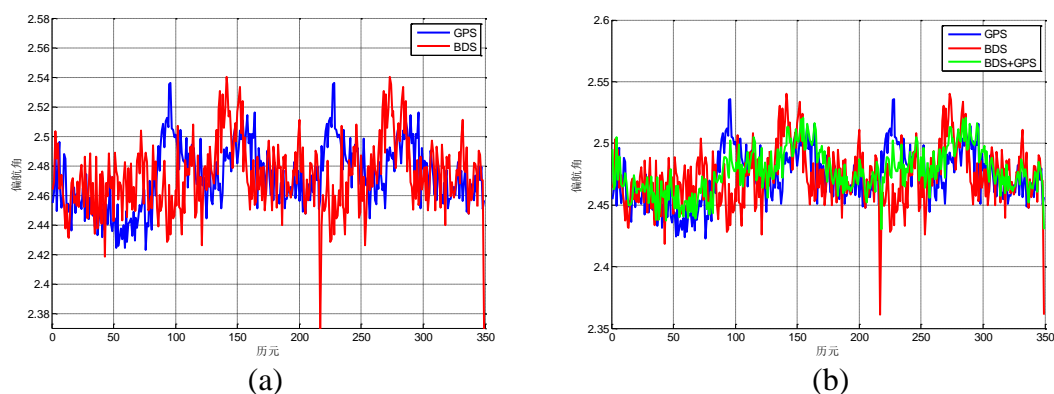


图 6-13 (a)BDS/GPS 偏航角结果对比, (b)BDS/GPS/BDS+GPS 偏航角结果对比

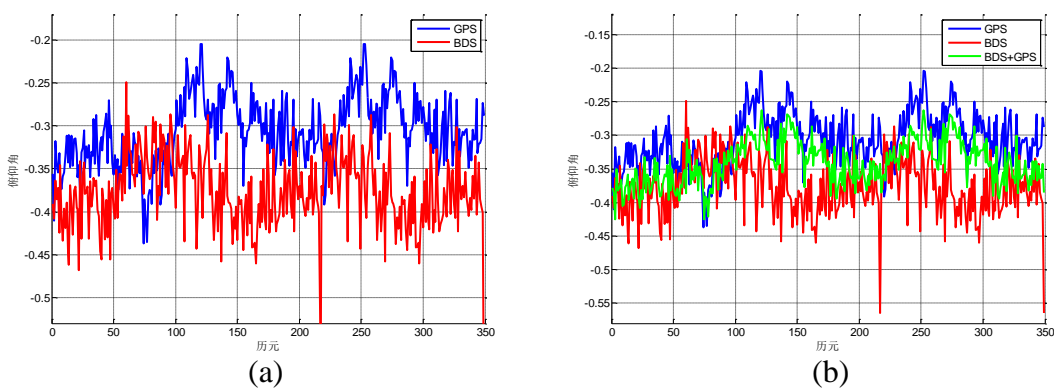


图 6-14 (a)BDS/GPS 俯仰角结果对比, (b)BDS/GPS/BDS+GPS 俯仰角结果对比

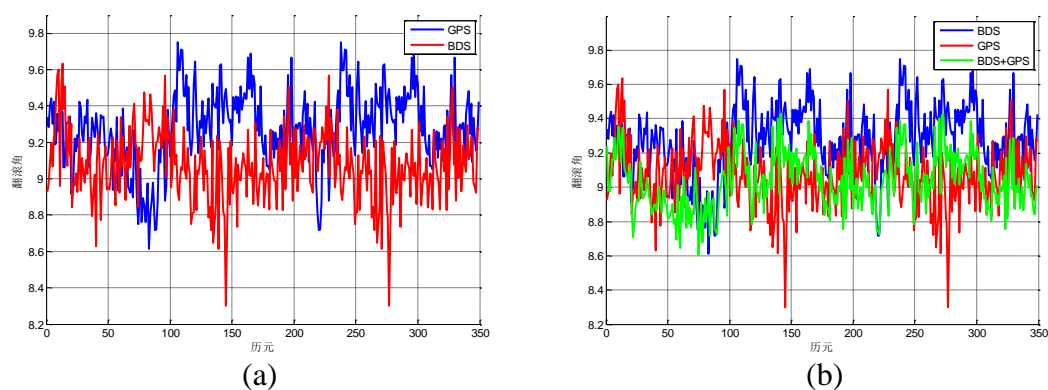


图 6-15 (a)BDS/GPS 翻滚角结果对比, (b)BDS/GPS/BDS+GPS 翻滚角结果对比

解算结果统计信息如表 6-4 所示,表中分别列出测量平均值、标准差、数据剔除率和最大误差。其中最大误差为排除粗差后的统计结果。

表 6-4 实验统计数结果

姿态角	实验组	平均值	方差	最大误差	数据剔除率
YAW	BDS	2.4659	0.0746	0.0825	7.168
	GPS	2.4727	0.0672	0.0803	2.243
	BDS+GPS	2.4692	0.0411	0.0647	4.553
PITCH	BDS	-0.3919	0.1078	0.1623	7.168
	GPS	-0.3187	0.0868	0.1572	3.537
	BDS+GPS	-0.3602	0.0687	0.0933	6.187
ROLL	BDS	9.1179	0.3911	0.8614	10.584
	GPS	9.3198	0.3037	0.8188	5.865
	BDS+GPS	9.1455	0.3011	0.5963	8.325

(3) 实验分析

从图和表中数据上可以得出,两个系统解算结果具有一定系统差异,偏航角、俯仰角和翻滚角的数值,BDS 均比 GPS 解算结果偏小,但差异并不大,可认为这种差异是接收数据的随机性造成的。

在方差方面,除俯仰角外,BDS 数值均大于 GPS,翻滚角方面数值差异比较明显,可认为 BDS 在稳定性方面要差于 GPS 系统。在最大误差方面,BDS 和 GPS 的偏航角和俯仰角差异较小,翻滚角差异较大,可认为 BDS 在控制误差方面不如 GPS。在数据剔除率方面,BDS 明显不如 GPS 系统,差异较明显,这主要是因为 BDS 的数据质量较差,出现了较多历元的不可用数据。从图上可以看出,BDS 解算结果在几个历元出现了明显的跳变,而相同历元处,GPS 解算结果未出现这种情况,也从另一方面说明了 BDS 数据质量较 GPS 差。

从数值上看,两系统的方差和最大误差均在 0.1 度级别,达到了较高的精度和稳定性,可认为两系统均具备独立测姿能力

从表 6-4 也可以看出,当利用 GNSS (BDS+GPS) 数据后,各项统计指标均有改善,方差、最大误差指标变小,而且 BDS 在解算过程中出现的跳变值也被修复,体现出较好的稳定性和控制误差能力,在数据剔除率方面,虽仍然不及 GPS,但较 BDS,已有明显改善。

6.3.2 姿态角算法对比实验

本实验目的为对比不同姿态角算法的解算差异情况。在解算数据时分别使用直接法、矩阵最小二乘法（矩阵法）和 TRIAD 法，并对解算结果进行对比，以分析不同算法下解算结果的差异性。

（1）实验信息

在解算设置上，每组解算除了在解算算法上有所改变，其余相同，另外，矩阵法和 TRIAD 法需要先验基线长度和夹角信息，实验采用定制铁环，其边长信息为已知，如图 6-16 所示。

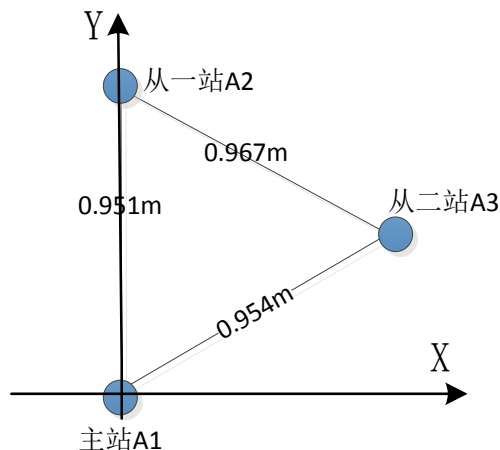


图 6-16 定制铁环说明

表 6-5 载体坐标系先验坐标

基线	X 坐标	Y 坐标	Z 坐标
A1A2	0m	0.951m	0m
A1A3	0.832m	0.461m	0m

（2）实验结果

解算结果如图 6-17、图 6-18 和图 6-19 所示，为方便对比，将偏航角（YAW）、俯仰角（PITCH）和翻滚角（ROLL）在一幅图中表达，三种算法使用不同颜色标示，其中直接法为蓝色，矩阵法为红色，TRIAD 法为绿色。

表 6-6 为直接法、TRIAD 算法和矩阵最小二乘法的结果统计数据，统计参数为方差、最大误差和数据剔除率。

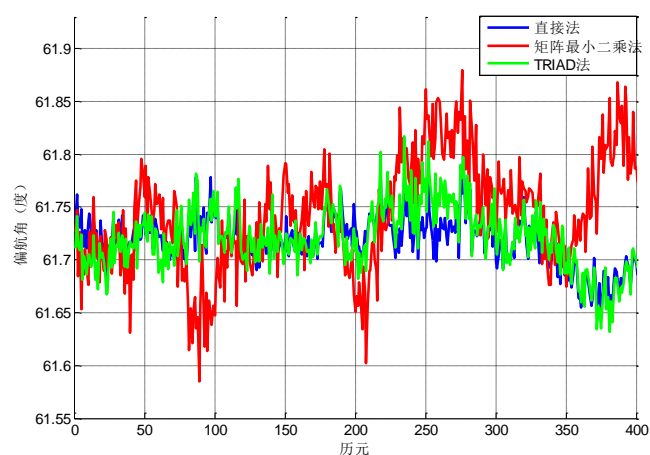


图 6-17 三种算法偏航角对比图

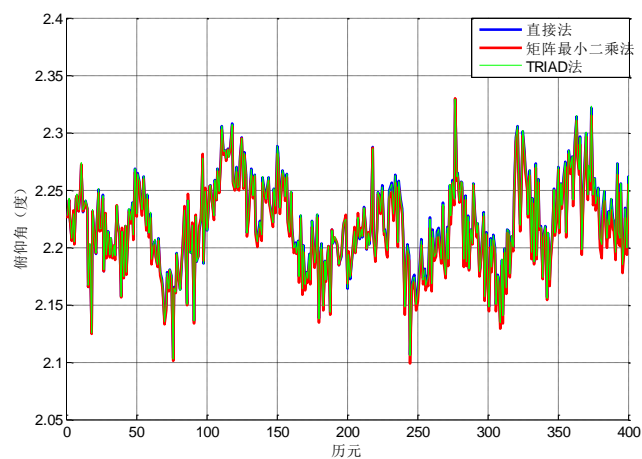


图 6-18 三种算法翻滚角

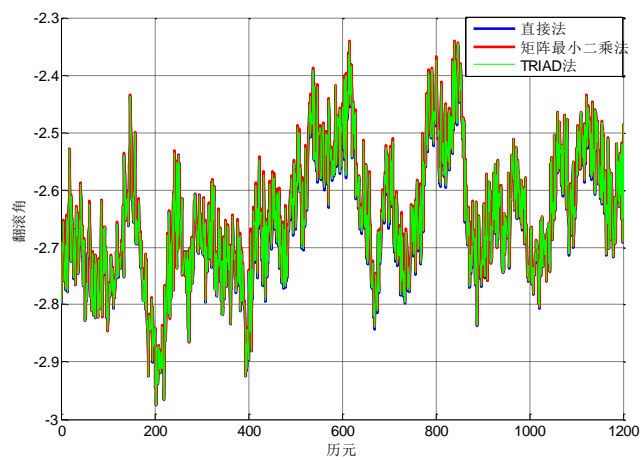


图 6-19 三种算法俯仰角对比图

表 6-6 三种姿态角算法对比统计数据

姿态角	算法	方差	最大误差	数据剔除率
YAW	直接法	0.0535	0.0862	2.364
	矩阵法	0.0717	0.1768	3.221
	TRIAD 法	0.0302	0.1042	2.335
PITCH	直接法	0.0831	0.1347	3.564
	矩阵法	0.0863	0.1402	3.824
	TRIAD 法	0.0816	0.1336	3.569
ROLL	直接法	0.1092	0.3375	4.526
	矩阵法	0.1102	0.3382	4.725
	TRIAD 法	0.0973	0.3311	4.573

(3) 实验分析

理论上, 直接法不需要先验基线信息作为前提条件, 且解算过程不能综合两条基线, 不能充分利用所有信息。矩阵法利用矩阵进行最小二乘运算, 由于矩阵运算的特殊性, 当矩阵元素中一个出现问题时, 可能带来整体的解算偏差, 故其稳定性较差。从表 6-5 中的数据剔除率看, 虽然三种算法的数据剔除率在同一级别, 但矩阵法的剔除率较大, 无效数据较多, 而直接法和 TRIAD 法大致相同。在方差方面, TRIAD 算法数值最小, 表现最为稳定, 直接法次之, 矩阵法最差。

从表 6-5 中数值也可看出, 三种算法在方差、最大误差和数据剔除率上并没有出现较大差异, 可认为三种算法解算结果上无显著差异。

由此, 可以得出, 在外部干扰相对稳定时, 三种算法并无明显优劣, 但基于理论上的分析, 仍不排除在观测环境发生变化时, 不同算法会出现各自问题, 这还需要更多实验才能验证。

第7章 总结与展望

7.1 工作总结

本文研究利用 GNSS 系统求解运动载体姿态角,围绕这个问题,分别在理论研究、编程实现和实验分析三个方面做了工作,其中,理论研究部分又分成了三个方面,这三个方面各作为本文的三个章节进行叙述,即第2章、第3章和第4章,而编程实现和实验研究构成了本文的第5章和第6章。通过这些章节的论述,本文对利用 GNSS 测量载体姿态的原理作了一个比较全面的阐述,并编程实现了一个测姿程序,最后实际采集数据对程序进行检验和分析。

本论文的主要工作有:

(1) 研究了 GNSS 姿态测量的基本原理和关键算法

(a) 对测姿中的坐标转换问题、姿态测量的原理和经典测姿算法进行了研究,提出了姿态测量中坐标转换的选取原则;

(b) 对短基线解算模型、短基线误差处理和利用 Kalman 滤波求解 GNSS 基线进行了论述;

(c) 对 LAMBDA 及其改进方法 M-LAMBDA,附有限制条件的 C-LAMBDA 作了比较全面的研究,并将 M-LAMBDA 与 C-LAMBDA 算法结合起来,提高了模糊度的固定成功率和效率。

(2) 编写了 GNSS 姿态测量程序

虽然这部分内容在论文中只占一章,但事实上其工作量却是最大的。程序能够完成读入 GNSS 数据,误差消除,求解模糊度,解算基线和姿态角解算的全流程计算。通过实验对软件进行分析,结果显示,在数据质量较好情况下,软件稳定性能能够达到 0.1 度。

(3) 设计了并完成了多个实验

这部分设计了两个测试实验,对本文所编写的测姿系统在准动态和动态情况下的解算情况作了探究。其中,实验数据的解算部分由姿态测量程序完成,而统计分析部分则由 Matlab 语言另行编写。另外,实验部分还添加了两个对比实验,分别对 BDS 和 GPS 两个系统和三种姿态求解算法在姿态解算中的表现作了对比分析。结果表明,两系统均具备独立测姿能力,且精度相当,利用 GNSS 数据能够有效优化解算的精度和稳定性。三种姿态算法解算结果具有差异,但精度很稳定

性相当。

本文的创新性的工作包括：

(1) 针对姿态测量坐标转换，本文提出了三个标准来选取合适的坐标系旋转顺序。

(2) 将 M-LAMBDA 算法与 C-LAMBDA 算法相结合，充分利用了姿态测量的先验基线长度信息，在提高了模糊度固定成功率基础上，尽可能小的减少了效率损失。

(3) 设计实验对 BDS 和 GPS 两系统在姿态测量领域的表现进行了对比分析，利用 GNSS (BDS+GPS) 数据解算姿态，提高了姿态解算的精度和稳定性。

7.2 工作展望

本文完成了一定的工作，但在整个姿态测量领域中，研究并不全面，具体表现在：

(1) 本文未涉及到硬件平台搭建，只能进行事后处理，希望在未来的研究中，能够结合硬件，真正的做到实时解算载体姿态角。

(2) 本文研究利用 GNSS 求解姿态，在后续的研究中可以结合 INS 等其他设备，综合不同方法的优势以达到更高精度和稳定性。

(3) 姿态测量领域还存在如何设定最优基线长度，最优基线夹角等最优化问题，希望在后续研究中，能够弥补这部分研究的不足。

参考文献

- [1] Bar-Itzhack I, Montgomery P Y, Garrick J C. Algorithms for attitude determination using GPS. OMANUTH PRESS LTD, 1997.
- [2] 高钟毓. 惯性导航系统技术, 清华大学出版社, 2012.
- [3] Spinney V W. Applications of Global Positioning System as an attitude reference for near earth users. 1976.
- [4] Ellis J F, Creswell G A. Interferometric attitude determination with the Global Positioning System. *Journal of Guidance, Control, and Dynamics*, 1979, 2(6): 522~527.
- [5] Brown A K, Thorvaldsen T P, Bowles W M. Interferometric attitude determination using the global positioning system-A new gyrotheodolite. 1983.
- [6] Hermann B R. A simulation of the navigation and orientation potential of the Ti - Agr. *Marine Geodesy*, 1985, 9(2): 133~143.
- [7] Evans A G. Roll, pitch, and yaw determination using a global positioning system receiver and an antenna periodically moving in a plane. *Marine Geodesy*, 1986, 10(1): 43~52.
- [8] Kruczynski L R, Li P C, Evans A G, et al. Using GPS to determine vehicle attitude. 1989.
- [9] 张宏兵, 葛茂荣. GPS 姿态确定原理及软件设计. *测绘通报*, 1999, (12): 27~29.
- [10] 过静君, 葛茂荣. 适用于航天器的 GPS 姿态测量算法和软件设计. *清华大学学报: 自然科学版*, 2000, 40(6): 99~102.
- [11] Li Y, Zhang K, Roberts C, et al. On-the-fly GPS-based attitude determination using single-and double-differenced carrier phase measurements. *GPS Solutions*, 2004, 8(2): 93~102.
- [12] Wang B, Miao L, Wang S, et al. A constrained LAMBDA method for GPS attitude determination. *GPS solutions*, 2009, 13(2): 97~107.
- [13] Park C, Teunissen P J. Integer least squares with quadratic equality constraints and its application to GNSS attitude determination systems. *International Journal of Control, Automation and Systems*, 2009, 7(4): 566~576.
- [14] Teunissen P, Giorgi G, Buist P J. Testing of a new single-frequency GNSS carrier phase attitude determination method: land, ship and aircraft experiments. *GPS solutions*, 2011, 15(1): 15~28.
- [15] Giorgi G, Teunissen P, Buist P J. A search and shrink approach for the baseline constrained LAMBDA method: experimental results. 2008.
- [16] Brown R A. Instantaneous GPS attitude determination. *IEEE*, 1992.
- [17] Crassidis J L, Markley F L, Lightsey E G. Global positioning system integer ambiguity resolution without attitude knowledge. *Journal of guidance, control, and dynamics*, 1999, 22(2): 212~218.

- [18] 刘根友. 一种 GPS 测定姿态的新方法——具有坐标函数约束的单历元阻尼 LAMBDA 算法. 测绘科学, 2003, 28(3): 36~38.
- [19] 郑庆晖. 基于 GPS 的航天器姿态, 相对姿态确定研究:[长沙: 国防科技大学, 2003.
- [20] 韩慧群. GPS 姿态测量系统研究与开发 [D]:[哈尔滨: 哈尔滨工程大学, 2006.
- [21] Wahba G. A least squares estimate of satellite attitude. SIAM review, 1965, 7(3): 409.
- [22] Farrell J. Aided navigation: GPS with high rate sensors, McGraw-Hill New York, 2008.
- [23] Hofmann-Wellenhof B, Lichtenegger H, Wasle E. GNSS - global navigation satellite systems: GPS, GLONASS, Galileo, and more, Springer, 2007.
- [24] Lu G. Development of a GPS multi-antenna system for attitude determination., Citeseer, 1995.
- [25] 王潜心. 机载 GPS 动态定位定速与定姿理论研究及软件开发 [D]:[中南大学, 2011.
- [26] 周健, 毛刚, 赵李健, et al. BD2/GPS 四频高精度接收机在远望号船姿测量中的应用. 测绘科学技术学报, 2012, 29(3): 171~174.
- [27] 孟骞, 刘建业, 曾庆化. 用卫星工具软件包进行北斗单天线姿态仿真测量. 导航定位学报, 2013, 3: 17.
- [28] Nadarajah N, Teunissen P. Instantaneous GPS/BeiDou/Galileo Attitude Determination: A Single-Frequency Robustness Analysis under Constrained Environments. Proceedings of The Institute of Navigation Pacific PNT, Honolulu, HI, USA, 2013, : 22~25.
- [29] 王园, 白征东, 元荣. 北斗二代卫星厘米级相对定位. 测绘通报, 2014, (07): 11~13.
- [30] 一民. 现代大地控制测量, 测绘出版社, 2008.
- [31] 刘若普. GPS 三维姿态测量技术研究 [D]:[上海交通大学, 2007.
- [32] 黎湧, 吴宏鑫, 刘良栋. 融合 TRIAD 算法用于 GPS 姿态确定 [J]. 中国空间科学技术, 2000, 4(2): 30~36.
- [33] 陈林. GPS 姿态测量系统的研究:[重庆: 重庆大学, 2007.
- [34] 刘晓辉, 党亚民, 王潜心, et al. 双基线姿态确定三种算法的比较分析. 大地测量与地球动力学, 2013, 33(1): 123~127.
- [35] Black H D. A passive system for determining the attitude of a satellite. AIAA Journal, 1964, 2(7): 1350~1351.
- [36] 段志勇, 袁信. 基于四元数方法的 GPS 航姿解算. 南京航空航天大学学报, 1999, 31(5): 492~496.
- [37] Shuster M D, Oh S D. Three-axis attitude determination from vector observations. Journal of Guidance, Control, and Dynamics, 1981, 4(1): 70~77.
- [38] Bar-Itzhack I Y. REQUEST-A recursive QUEST algorithm for sequential attitude determination. Journal of Guidance, Control, and Dynamics, 1996, 19(5): 1034~1038.
- [39] 过静琨, 刘亮, 白征东. GPS 单频载波相位相对定位的 OTF 算法及实现. 测绘通报, 2009, (05): 9~11.
- [40] 李正航, 黄劲松. GPS 测量原理与数据处理: 武汉: 武汉大学出版社, 2010.

- [41] 刘亮. GPS 高精度实时动态定位算法研究及软件开发:[清华大学, 2009.
- [42] Gleason S, Gebre-Egziabher D. GNSS applications and methods, Artech House, 2009.
- [43] 王惠南. GPS 导航原理及应用: 北京: 科学出版社, 2003.
- [44] Takasu T, Yasuda A. Kalman-filter-based integer ambiguity resolution strategy for long-baseline RTK with ionosphere and troposphere estimation. 2010.
- [45] Counselman C C, Gourevitch S A. Miniature interferometer terminals for earth surveying: ambiguity and multipath with Global Positioning System. Geoscience and Remote Sensing, IEEE Transactions on, 1981, (4): 244~252.
- [46] Remondi B W. Using the global positioning system(GPS) phase observable for relative geodesy: Modeling, processing, and results[Ph. D. Thesis]. 1984.
- [47] Wübbena G. The GPS adjustment software package-GEONAP-concepts and models. 1989.
- [48] Seeber G, Wubben G. Kinematic positioning with carrier phases and "on the way" ambiguity solution. 1989.
- [49] Hatch R. Instantaneous ambiguity resolution: Kinematic systems in geodesy, surveying, and remote sensing. Springer, 1991 299~308.
- [50] Frei E, Beutler G. Rapid static positioning based on the fast ambiguity resolution approach FARA: theory and first results. Manuscripta geodetica, 1990, 15(6): 325~356.
- [51] Teunissen P. The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation. Journal of Geodesy, 1995, 70(1-2): 65~82.
- [52] 方保镭, 周继东, 李医民. 矩阵论. 清华大学出版社, 2004.
- [53] 赵蓓, 王飞雪, 孙产富, et al. LAMBDA 整周模糊度解算方法中的整数 Z 变换算法. 弹箭与制导学报, 2008, 28(3): 254~257.
- [54] De Jonge P, Tiberius C. The LAMBDA method for integer ambiguity estimation: implementation aspects. Publications of the Delft Computing Centre, LGR-Series, 1996, 12(12): 1~47.
- [55] Chang X, Yang X, Zhou T. MLAMBDA: a modified LAMBDA method for integer least-squares estimation. Journal of Geodesy, 2005, 79(9): 552~565.
- [56] Chang X, Yang X, Zhou T. On-the-fly GPS-based attitude determination using single- and double-differenced carrier phase measurements. GPS Solutions, 2004, 93~102.
- [57] Takasu T, Yasuda A. Development of the low-cost RTK-GPS receiver with an open source program package RTKLIB. International Convention Centre Jeju, Korea, 2009.
- [58] Takasu T, Yasuda A. Evaluation of RTK-GPS performance with low-cost single-frequency GPS receivers. 2008.
- [59] Takasu T. RTKLIB: An open source program package for GNSS positioning: 2011.
- [60] Takasu T. RTKLIB ver. 2.4. 1 Manual. Last checked, 2011, 9: 2011.
- [61] 白征东, 邵奎. 北斗与 GPS 双系统测姿研究. 测绘通报, 2014, (S2): 219~223.

致 谢

我的导师白征东老师最初是义务指导我的，我后来才成为他的学生。他是一个博学和有耐心的导师，在论文完成的整个生命周期，从选题到答辩，他都给予了巨大的关心、支持和指导，过程中我的错误、遗漏等问题，他都能够充分的理解和指正，在生活方面，他也给予了无私照顾，我非常感激他。

感谢地球空间信息所的赵红蕊老师、刘钊老师和我曾经的导师唐中实老师对我无私的指导和帮助。

感谢已经毕业刘亮、杨聪、朱永辉、罗腾、陈宇、原波、付春浩、任常、元荣、王园、傅罡师兄们，他们所做的深厚知识积淀是完成本论文的基础。感谢在一起共同学习的吴刚祥、高培超、王浩以及实验室的汤小禹、项伟、李帅师弟，他们为本文提出了很多宝贵的意见和帮助。

特别感谢元荣师兄，他为本研究提供了很多帮助和指导。

感谢父母对我的养育之恩，感谢所有支持我的人，谢谢大家！

声 明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签 名：_____日 期：

个人简历、在学期间发表的学术论文与研究成果

个人简历

1989 年 8 月 5 日出生于辽宁省凌源市。

2008 年 9 月考入中南大学地球科学与信息物理工程系，2012 年 7 月本科毕业并获得工学学士学位。

2012 年 9 月考入清华大学土木工程系攻读大地测量学与测量工程硕士至今。

发表的学术论文

- [1] 邵奎, 白征东. 北斗与 GPS 双系统测姿研究. 测绘通报 (已录用, 中文核心期刊)
- [2] 邵奎, 白征东. 基于超宽巷组合的北斗三频 TurboEdit 方法. 测绘通报 (已录用, 中文核心期刊)