

# 两导航模块相对导航的研究<sup>①</sup>

胡绍林<sup>1,2</sup>, 李 萌<sup>1</sup>

<sup>1</sup>(西安理工大学 自动化与信息控制学院, 西安 710048)

<sup>2</sup>(航天器在轨故障诊断与维修重点实验室, 西安 710043)

**摘 要:** 主要研究两个导航模块间的相对导航, 并实现了两导航模块间相对位置的实时测量. 讨论了不同坐标系下定位结果的互相转换. 对导航模块输出数据进行了简要的分析, 叙述了在 Visual Studio 编译环境下利用 C# 语言编写程序实现对定位数据的提取分析, 并对两个模块定位信息处理从而实时计算出两个导航模块的相对位置信息.

**关键词:** 相对导航; 坐标系转换; 定位数据; C# 语言

## Relative Navigation Research for Two Navigation Modules

HU Shao-Lin<sup>1,2</sup>, LI Meng<sup>1</sup>

<sup>1</sup>(Xi'an University of Technology, Xi'an 710048, China)

<sup>2</sup>(Spacecraft Fault Diagnosis and Repair Key Laboratory, Xi'an 710043, China)

**Abstract:** This paper mainly studies the relative navigation between two navigation modules, and realizes the real-time measurement of the relative position between the two navigation module. Location result in different coordinate system conversion was been discussed. The navigation module output data was analyzed. The use of C# language procedures realize the extraction of the positioning data in the Visual Studio compiler environment. The two module positioning information was been processed, it calculates the relative position information of two navigation module.

**Key words:** relative navigation; transfer of axes; locator data; C# language

卫星导航技术应用领域十分广阔. 相对导航技术不仅在国防安全中扮演重要角色, 而且越来越多的渗透到民用行业. 交通运输部门用 GPS 等相关集成技术建立智能交通和城市交通监控系统, 在知道汽车的相对位置后能够更智能和快速的拥挤路段车辆进行调度. 港口船舶利用相对导航技术能实时测量出船只间的相对距离从而避免在船舶进出港口时船只的碰撞. 刑侦部门利用 GPS 相对导航技术实现对目标物体的追踪<sup>[1]</sup>. 不同接收机间的相对导航定位其实是指不同接收机通过无线网络将其定位信息传送到其他接收机模块, 从而得到它们间的相对位置信息. 实现上述功能的基础都是首先获得接收机模块的定位信息, 所以要对导航接收机输出的定位信息进行提取和分析, 而且在进行 GPS 定位计算时还需要进行不同坐标系之间的转换<sup>[2]</sup>.

计算相对位置同样要进行坐标系的转换.

GPS 接收机的型号种类非常多, 但大多数接收机输出数据格式都采用美国国家电子协会为海用电子以 ASCII 码格式输出, GPS 接收机根据 NMEA0183 协议标准规范, 将位置、速度等信息通过串口传送到 PC 机, PC 机通过程序实现对定位信息的提取. 本文以 UBLOX NEO-6M GPS 模块输出的 NMEA0183 协议数据为例, 利用 C# 程序进行对定位信息的提取和定位点坐标系的转换, 从而实现了计算两个导航接收机模块的相对位置信息的功能<sup>[3]</sup>.

### 1 相对导航系统概述

本相对导航系统硬件部分由两个 GPS 导航模块和两个无线 POP2032 无线模块组成, 其中一个 GPS 导航

① 基金项目: 国家自然科学基金(61473222)

收稿时间: 2014-10-09; 收到修改稿时间: 2014-12-22

模块的 Txd 与 POP2032 无线模块的 Rxd 相连, 通过无线模块将定位等信息发送到与 PC 机相连的无线接收模块, 无线模块与 PC 机之间是用 USB 转串口线相连, 另一个 GPS 模块与 PC 机也为 USB 转串口线相连. 最后在电脑上实现解析. 用 5V 电源进行供电. 其中 GPS 导航模块与无线接收模块都用 5V 电源进行供电. 下图 1 是整个相对导航系统的示意图.

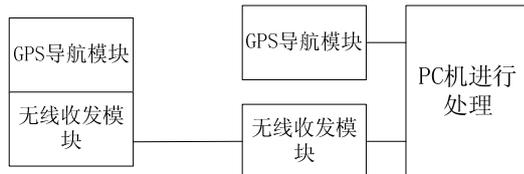


图 1 相对导航系统整体示意图

在远端的 GPS 模块将定位信息通过无线模块发送到连接 PC 机的无线模块, PC 机和无线模块通过 USB 转串口相连, 最终实现在 PC 机上对数据的解析.

## 2 系统的总体设计与实现

### 2.1 对 GPS 模块定位信息的提取

利用 C# 语言实现对 GPS 模块输出数据的处理. C# 是微软公司在 2000 年 7 月发布的一种全新且简单、安全、面向对象的程序设计语言, 是专门为 .NET 的应用而开发的语言. 利用 C# 可以更为简单高效的对 GPS 接收机输出的数据进行解析. 具体的程序流程和思路会在文章第三部分体现<sup>[4]</sup>.

### 2.2 不同坐标系之间的转换

从接收机模块提取出的定位信息是 NMEA0183 协议数据, 该协议数据中包含的定位信息是大地坐标系中的经度、纬度、大地高程信息. GPS 接收机定位给出的用户位置一般表达在 WGS-84 直角坐标系中, 或者等效地表达成 WGS-84 大地坐标系中. 要计算两个模块的相对位置, 必须把大地坐标系中的定位信息转换到地心地固直角坐标系中<sup>[5]</sup>. 从大地坐标 $(\phi, \lambda, h)$ 到地心地固直角坐标系 $(X, Y, Z)$ 的变换公式(1)如下:

$$\begin{aligned} x &= (N+h)\cos\phi\cos\lambda \\ y &= (N+h)\cos\phi\sin\lambda \\ z &= [N(1-e^2)+h]\sin\phi \end{aligned} \quad (1)$$

其中,  $N$  是基准椭球体的卯酉圆曲率半径,  $e$  为椭球体偏心率, 它们用公式(2)和(3)可以获得, 其中的  $f$

可以通过 WGS-84 的基本大地参数值查询到<sup>[6]</sup>. 公式(2)和(3)如下所示:

$$N = \frac{a}{\sqrt{1-e^2\sin^2\phi}} \quad (2)$$

$$e^2 = f(2-f) \quad (3)$$

反过来, 从地心地固直角坐标 $(X, Y, Z)$ 到大地坐标 $(\phi, \lambda, h)$ 的变换公式为:

$$\begin{aligned} \lambda &= \arctan\left(\frac{y}{x}\right) \\ h &= \frac{p}{\cos\phi} - N \\ \phi &= \arctan\left[\frac{z}{p}\left(1 - e^2 \frac{N}{N+h}\right)^{-1}\right] \end{aligned} \quad (4)$$

其中,  $N$  和  $e^2$  可以分别由式(2)和(3)求出, 中间变量  $p$  的计算公式如下:

$$p = \sqrt{x^2 + y^2} \quad (5)$$

因为  $h$  的计算式含有待求的  $\phi$ , 而  $\phi$  的计算式反过来又含有待求的  $h$ , 所以只能借助迭代法来逐次逼近求解  $h$  和  $\phi$  的值. 一般经过 3~4 次循环迭代后就可以计算出结果.

GPS 接收机输出的 NMEA0183 协议中可以直接提取出大地坐标系坐标, 所以在知道经度、纬度、大地高程后, 通过上述的公式可以得到在地心地固坐标系中的坐标 $(X, Y, Z)$ , 这样就可以求出两个位置的相对距离和相对偏角等信息<sup>[7]</sup>.

## 3 相对导航系统的软件实现

### 3.1 对 GPS 模块定位信息的提取

GPS 接收机模块和无线通讯模块都通过串口与计算机相连, 程序对串口进行适合的设置就能实现对 GPS 信号的接收. GPS 信号有二进制和 ASCII 码两种输出形式, 本文接收机模块输出的是后者即 NMEA0183 协议数据. NMEA0183 协议数据以 \$ 为开头, 接着是信息类型, 后面是数据, 以逗号分隔开, 最后以回车和换行符结束的数据<sup>[8]</sup>. 输出的数据包含有 7 中不同类型, \$GPZDA、\$GPGGA、\$GPGLL、\$GPVTG、\$GPGSA、\$GPGSV、\$GPRMC, 其中最重要的是 \$GPGGA 和 \$GPRMC 段数据, 其中包含有时间和定位信息, 对着两个数据段的处理和提取也是本程序的重点<sup>[9]</sup>.

#### 3.1.1 串口通信的实现

为了使计算机能正确通过串口从 GPS 接收机模块

读取数据,需要编写串口通讯程序.首先对串口进行端口号、波特率进行设置,奇偶校验位和停止位分别默认为0和8.然后开始读串口数据.

```
device.SerialPort.PortName = this.txtPortName.Text;
device.SerialPort.BaudRate=int.Parse(cmboBoxBaudRate.SelectedItem.ToString());
device.SerialPort.Open();
readingThread=new Thread(Do_Reading){ IsBackground = true };
readingThread.Start();
```

### 3.1.2 定位信息的提取

在 Do\_Reading 中调用了解析 GPGAA 和 GPRMC 的方法.根据协议可以看出输出数据的基本类型大致相同,可以把串口输出的数据统一转换成一类即 GPS 数据基类.包含数据头、数据部分和校验位.本文所要分析的是 GPGAA 和 GPRMC 数据段,只需要编写相应的方法找寻该字段,并处理该字段包含信息的位,就可以实现对需要的数据进行提取并分析.

一串 GPRMC 数据段为例: \$GPRMC,060215.00,A,3415.43442,N,10859.20707,E,0.393,,030714,,A\*79 以逗号和\*位特征,将数据段分为数据头 \$GPRMC, 数据部分 060215.00, A,3415.43442,N,10859.20707,E, 0.393,,030714,,和校验位 A\*79 三部分.具体程序如下

```
public Message(string message)
{
string[] temp = message.Split(',','*');
this.HeadType=(HeadType)Enum.Parse(typeof(HeadType),temp[0].ToUpper().Substring(1));
string ver = temp[temp.Length - 1];
this.Verification = temp[temp.Length - 1];
this.Data = new string[temp.Length - 2];
Array.Copy(temp, 1, this.Data, 0, this.Data.Length);
}
```

然后根据 NMEA0183 协议,对 GPRMC 字段以逗号进行分割,如要提取出时间就为数据部分第一个字段 060215.00 为 6 时 2 分 15 秒.

```
string time = base.Data[0];
int hour = int.Parse(time[0].ToString() + time[1]);
int min = int.Parse(time[2].ToString() + time[3]);
int sec = int.Parse(time[4].ToString() + time[5]);
```

输出数据段中的 3415.43442,N 为纬度信息,提取

其信息的代码如下:

```
public int LatitudeDegree
{get
{string latitude = this.Latitude;
return int.Parse(latitude[0].ToString() + latitude[1]);
}}
public int LatitudeMin
{get
{string latitude = this.Latitude;
return int.Parse(latitude[2].ToString() + latitude[3]);
}}
public double LatitudeSec
{get
{string latitude = this.Latitude;
return double.Parse("0." + latitude.Substring(5)) * 60;
}}
```

根据协议进行适当的计算就可以得到用户的定位信息为北纬 34 度 15 分 26 秒.其他的日期、经度、大地高程、航向等信息都通过这种方法可以提取出来.因为篇幅有限这里只把程序的关键步骤截取出来.

### 3.2 定位信息所在坐标系的转换

上面已经得到了用户在大地坐标中的  $(\phi, \lambda, h)$ , 然后结合上文中的公式(1)、(2)、(3)就可以直接求解出在地心地固坐标系中的坐标  $(X, Y, Z)$ . 程序中专门编写了个类 GPSMath 中方法对其进行转换<sup>[10]</sup>. 关键代码如下:

```
public class GPSMath
{public const double A = 6378137.0;
public const double F = 1 / 298.257223563;
public static readonly double E_Square;
static GPSMath()
{E_Square = F * (2 - F);}
public static double Get_N(int latitudeDegree, int latitudeMin, double latitudeSec)
{return A/Math.Sqrt(1-E_Square*Math.Pow(Math.Sin(Get_Radian(latitudeDegree, latitudeMin, latitudeSec)), 2));}
public static double Get_Radian(int degree, int min, double sec)
{double d = degree + min / 60 + sec / 3600;
return d * Math.PI / 180;}
```

### 3.3 相对距离的求解

软件正常工作时计算机通过两个串口接收到不同接收机模块的定位信息。利用距离公式就可以计算出两个接收机模块的相对位置。代码部分如下：  

```
public static double RelativeDistance(double x1, double y1, double z1, double x2, double y2, double z2){return Math.Sqrt(Math.Pow(x1 - x2, 2) + Math.Pow(y1 - y2, 2) + Math.Pow(z1 - z2, 2));}
public static double RelativeAngle(double x1, double y1, double x2, double y2){return Math.Atan(Math.Abs(y2 - y1) / Math.Abs(x2 - x1));}
```

为了确保计算相对位置的实时性，对两个导航模块的定位数据时间进行判定，只有时间差小于 10 秒才做相对计算，代码如下：

```
if((list[i].UTCDateTime-list2.UTCDateTime).TotalSeconds < 10), 程序相对数据信息全部存储在 EXCEL 中。
```

### 4 软件测试试验

试验地点选在西安理工大学新学科楼，两个接收机模块分别在 702 教研室窗口和 708 教研室窗口，试验时远端的接收机通过无线模块将定位信息实时发送到 708 教研室的 PC 机，PC 机子在结合与本机相连的另一接收机的定位信息，从而计算出相对位置。统程序的主界面可以显示当前导航模块的定位信息即时间、经度、纬度、海拔高度和在 WGS84 地心地固坐标系中的坐标(X,Y,Z)，显示两个导航模块的相对位置信息，主界面如图 2 所示。



图 2 系统的主界面图

可以看出程序测量出的两个导航模块的相对距离为 37 米左右，而实际距离应为 30 米。这是因为导航模块也存在定位误差所造成的。实验完毕后将两模块的定位数据存储到 EXCEL 中以便于日后的分析处理。

### 5 结论

本文研究和实现对导航模块输出数据的实时接收和处理，不同坐标系下定位信息的互相转换。并利用 C# 语言编写程序实现了对两个导航模块相对位置的计算。分析了导航模块输出的 NMEA0183 协议数据的类型，对其包含的定位信息进行了准确提取。在研究导航模块定位信息的提取和相对导航方面有重要意义。

#### 参考文献

1 晏磊,苗李莉,刘岳峰.关于 GPS 技术在智能交通系统中的应用分析.测绘通报,2005:(8)26-28.

- 2 谢钢.GPS 原理与接收机设计.北京:电子工业出版社,2009.
- 3 李明峰,冯宝红,刘三枝.GPS 定位技术及其应用.北京:国防工业出版社,2006.
- 4 万丽华,魏二虎.导航系统定位精度和 DOP 的仿真研究.测绘地理信息,2013,38:33-37.
- 5 徐绍铨,张华海,杨志强,王泽民.GPS 测量原理及其应用.武汉:武汉大学出版社,2003(1):154-190.
- 6 杨俊.武奇生.GPS 基本原理及其 Matlab 仿真.西安:西安电子科技大学出版社,2006.
- 7 方书山,章传银,秘金钟.NMEA-0183 格式数据流解析的一种实用方法.测绘通报,2013(11):114-116.
- 8 崔先国,苏国猛,李云岭.数字测图与 GIS 数据扩展的探讨.测绘通报,2009(3):5-7.
- 9 王耀华,张信丽.基于规则库的制图数据与 GIS 数据一体化建库方案探讨.测绘标准化,2011,27(4):23-25.
- 10 刘秋香,王云,姜桂洪.Visual C#.NET 程序设计.北京:清华大学出版社,2011.