

# GREAT-FGO 用户指南

# 目录

第一章 概述 .....	1
第二章 因子图优化原理 .....	2
2.1 因子图优化理论 .....	2
2.1.1 状态节点与观测向量 .....	2
2.1.2 滑动窗口与边缘化 .....	2
2.2 基于因子图优化的 RTK 方法 .....	3
2.2.1 概述 .....	3
2.2.2 多历元 GNSS 数据关联方法 .....	3
2.2.3 基于因子图优化的 GNSS RTK 估计器 .....	5
2.2.4 基于边缘化的模糊度传递方法 .....	8
2.3 基于因子图优化的 RTK/INS 方法 .....	10
2.3.1 概述 .....	10
2.3.2 紧耦合系统定义与因子图表达 .....	10
2.3.3 RTK/INS 因子构建 .....	11
第三章 环境要求和许可证 .....	15
3.1 环境要求 .....	15
3.2 许可证 .....	15
3.3 版权 .....	15
第四章 在 windows 下编译与安装 .....	17
第五章 算例数据处理说明 .....	20
5.1 算例文件目录结构 .....	20
5.2 在 VS 中进行参数配置与运行 .....	20
5.3 使用绘图脚本进行结果评估 .....	22
第六章 附录 .....	24
6.1 XML 配置文件说明 .....	24
6.2 结果文件 (**.fgo/**.ins) 说明 .....	32

6.3 轨迹文件 (**.kml) 说明 .....	34
----------------------------	----

# 第一章 概述

GREAT (GNSS+ REsearch, Application and Teaching) 软件由武汉大学测绘学院设计开发，是一个用于空间大地测量数据处理、精密定位和定轨以及多源融合导航的综合性软件平台。软件中，核心计算模块使用 C++语言(C++17)编写，辅助脚本模块使用 Python3 和 C-Shell 语言实现自动化数据处理。其中，所有 C++ 模块的编写都遵循 Google 开源项目代码风格指南，并且使用 GIT 工具进行版本控制。GREAT 软件使用 CMake 工具进行编译管理，用户可以灵活选择 GCC、Clang、MSVC 等主流 C++编译器。目前软件提供了 Windows 和 Linux 平台的命令行应用程序。

GREAT-FGO 是 GREAT 软件中的一个重要模块，主要用于基于因子图优化的 RTK 与 RTK/INS 解算，官方获取链接为：<https://github.com/GREAT-WHU/GREAT-FGO.git>。GREAT-FGO 由 GREAT-PVT (<https://github.com/GREAT-WHU/GREAT-PVT.git>) 与 GREAT-MSF (<https://github.com/GREAT-WHU/GREAT-MSF.git>)扩展而来。软件中，核心计算模块使用 C++语言编写，辅助脚本模块使用 Python3 语言实现结果绘制。GREAT-FGO 软件使用 CMake 工具进行编译管理，用户可以灵活选择 GCC、Clang、MSVC 等主流 C++编译器。GREAT-FGO 由 2 个可移植程序库组成，分别是 LibGREAT 和 LibGnut。除了原 GREAT-PVT 中的 GNSS 定位解决方案，GREAT-MSF 提供的多传感器融合导航功能外，GREAT-FGO 提供了基于因子图优化的 RTK 与 RTK/INS 的算法。本次开源的 GREAT-FGO 软件支持以下主要功能：

- 支持基于因子图优化的 RTK 解算，可实现载波相位模糊度固定
- 支持基于因子图优化的 RTK/INS 紧耦合解算，具备组合系统动态快速初始化能力。
- 支持 GPS、GLONASS、Galileo、BDS-2/3 等卫星导航系统
- 支持自定义 IMU 数据格式、噪声模型
- 支持轨迹动态显示与谷歌地球查看

此外，软件包还提供定位结果绘图脚本，便于用户对数据进行结果分析。

## 第二章 因子图优化原理

### 2.1 因子图优化理论

#### 2.1.1 状态节点与观测向量

因子图优化是一种可以处理大量变量与约束关系的优化方法，其结构包括变量节点、因子节点与边。变量节点代表待优化的变量或参数，因子节点代表与变量相关的函数或概率分布，边连接变量节点和因子节点，表示变量与因子之间的依赖关系。因子图的优化目标是最小化成本函数，即找到一组最优的系统状态变量，使得所有因子的成本（或误差）之和最小。这一目标可以通过以下几个步骤来完成：（1）初始化：为变量节点分配初始值。这些值可以是随机的，也可以是基于某种先验知识的估计。（2）消息传递：在因子图中进行迭代的消息传递过程。每个因子节点向其连接的变量节点发送关于当前变量值的更新信息（即梯度、置信度等），而变量节点则根据接收到的信息调整自己的值。（3）更新变量值：根据因子节点返回的更新信息，变量节点更新自己的值。这一步骤可能涉及求解一个局部优化问题，如最大后验概率估计、最小二乘问题等。（4）迭代收敛：重复上述消息传递和变量更新过程，直到达到某个收敛条件（如变量值的变化小于预设阈值、迭代次数达到上限等）。此时，可以认为已经找到了一个近似最优的解。

#### 2.1.2 滑动窗口与边缘化

因子图优化包括全局优化、滑动窗口优化、增量式平滑 iSAM 优化等方法。其中全局优化由于考虑之前的所有历元，解算速率随着时间增加而显著下降，无法长期运行。增量式优化（如 iSAM 系列方法）通过稀疏性和贝叶斯树等高级数据结构，实现了增量式的求解。当新的数据和状态加入时，iSAM 方法无需重新求解整个问题，而是仅对信息矩阵中受新数据影响的部分进行高效更新，从而降低解算复杂度；滑动窗口的优化方法核心思想是仅保留一个窗口内的状态进行联合优化，而将窗口之前的历史状态从优化变量中移除，因而大大提升了解算速率。在这过程中，窗口滑动与边缘化策略至关重要。

滑动窗口是当窗口内历元已满后，进入下一历元时，新的观测数据被加入窗

口，而窗口内最早历元的数据被剔除。新加入的数据要与窗口内之前历元的数据建立关联；被剔除的数据，其与其余历元数据的关联不能直接丢弃，而是使用边缘化的方法保留其约束。

边缘化是在每次窗口滑动之前，通过舒尔补方法将要被剔除的观测转化为一个关于窗口内剩余状态的先验概率约束。这个先验项封装了被移出状态对所有保留状态的约束信息，从而将历史观测的影响以概率的形式留存下来，有效地保持了窗口外关键信息对当前估计的影响，避免了精度断崖式下降。

## 2.2 基于因子图优化的 RTK 方法

### 2.2.1 概述

基于因子图优化的 RTK 方法主要包括数据输入、数据预处理和因子图优化三个模块。首先对来自基准站和流动站的 GNSS 原始观测值（伪距和载波相位）进行预处理，包括基准站-流动站共视卫星选取、载波相位周跳探测、多历元 GNSS 数据关联以及双差观测值构建。在此基础上进一步构建双差因子用于滑动窗口优化，当窗口大小达到预设的大小时，对窗口内的最老状态和相应观测进行滑动，并基于边缘化方法构建先验模糊度约束用于后续窗口优化。在每次滑窗优化后得到浮点模糊度和相应的方差协方差矩阵，基于 LAMBDA 方法进行整数模糊度固定，最终将固定解输出。

### 2.2.2 多历元 GNSS 数据关联方法

#### 2.2.2.1 基于滑动窗口的多历元 GNSS 数据关联

GNSS 中，完整的载波相位观测可以表达为：

$$\Phi_{r,n}^s = \rho_r^s + t_r - t^s - I_{r,n}^s + Z_r^s + \lambda_n(N_{r,n}^s + B_{r,n} - B_n^s) + \varepsilon_{r,n}^s \quad (2.1)$$

其中， $\rho_r^s$ 表示卫星天线相位中心到接收机天线相位中心的几何距离， $t_r$ 、 $t^s$ 分别代表接收机和卫星钟差， $Z_r^s$ 代表对流层斜延迟， $I_{r,n}^s$ 代表电离层斜延迟， $B_{r,n}$ 和 $B_n^s$ 分别代表接收机端和卫星端的相位延迟， $\varepsilon_{r,n}^s$ 代表相位观测值上的噪声与多路径误差的总和， $N_{r,n}^s$ 表示未知的整周部分。对于载波相位观测，一旦确定了未

知的整周部分 $N_{r,n}^s$ ，就可以获得毫米级精度的载波相位观测值。而确定 $N_{r,n}^s$ 的过程被称为整数模糊度固定 (Integer Ambiguity Resolution, IAR)。此外，一旦接收机成功捕获卫星，并对其保持连续跟踪，未知的整周模糊度 $N_{r,n}^s$ 是一个时不变的常数。基于以上载波相位特性，基于滑动窗口的多历元 GNSS 数据关联方法，充分利用载波相位的连续特性，基于滑窗内的卫星共视关系，构建不同历元之间的模糊度约束。

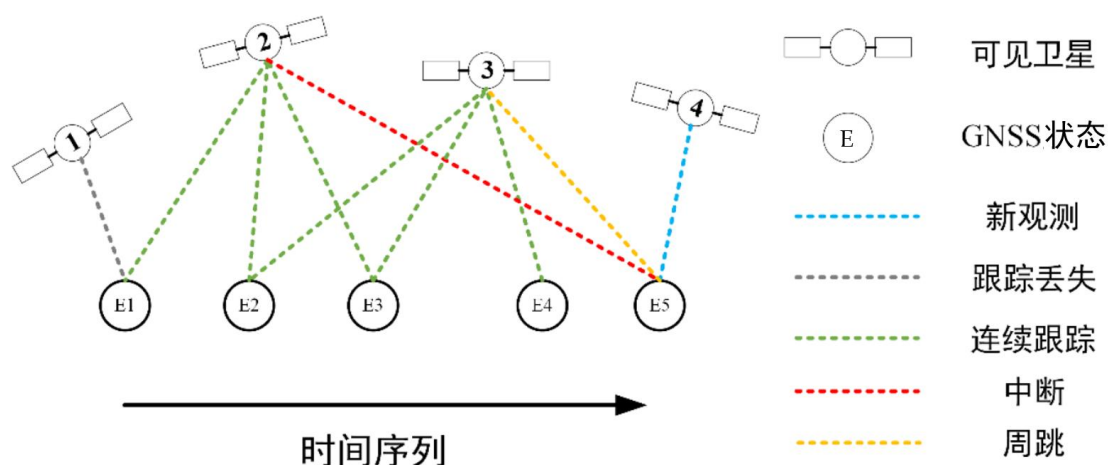


图 2.1 滑动窗口内卫星跟踪示意图

如图 2.1 所示，针对滑动窗口内卫星跟踪的五种情况，代码使用了一套通用的模糊度参数管理机制，用于构建滑动窗口内不同位置节点之间的数据关联，保证形成正确的模糊度约束关系。主要有以下四个步骤：

- 添加：对新跟踪的卫星  $i$ （蓝色），将其载波相位观测值  $\Phi_f^i$  唯一标识并在系统内新增相应的模糊度参数；
- 合并：对于窗口内的同一卫星  $i$ 、同一频率  $f$  的载波相位模糊度进行参数合并；
- 分离：当窗口内某颗卫星  $j$  发生周跳或者中断时（黄色、红色），将  $\Phi_f^j$  分离成两个独立的模糊度参数；
- 删除：对不再被窗口内其他历元观测到的卫星（黑色），即窗口跟踪丢失，对其相应的模糊度参数进行删除；

基于以上步骤，可以通过载波相位模糊度的连续特性对不同历元独立的卫星观测构建时间关联，通过窗口内联合非线性优化实现精度更优、鲁棒性更好的状态估计。

### 2.2.2.2 双差观测值构建

通过基准站和流动站的共视卫星关系，构建多系统双差观测值用于后端优化。假设流动站  $r$  和基准站  $b$  共同观测到卫星  $i, j$ ，其在频率  $f$  上的伪距和载波相位观测值分别为  $(P_b^{i,f}, P_b^{j,f}, P_r^{i,f}, P_r^{j,f})$  和  $(L_b^{i,f}, L_b^{j,f}, L_r^{i,f}, L_r^{j,f})$  通过双差运算可以得到双差观测方程如下：

$$\nabla \Delta P_{br,f}^{ij} = \nabla \Delta \rho_{br,f}^{ij} + \nabla \Delta I_{br,f}^{ij} + \nabla \Delta T_{br}^{ij} + \varepsilon_{\nabla \Delta P_{br,f}^{ij}} \quad (2.2)$$

$$\nabla \Delta L_{br,f}^{ij} = \nabla \Delta \rho_{br,f}^{ij} + \lambda_f \nabla \Delta N_{br,f}^{ij} - \nabla \Delta I_{br,f}^{ij} + \nabla \Delta T_{br}^{ij} + \varepsilon_{\nabla \Delta L_{br,f}^{ij}} \quad (2.3)$$

其中， $\nabla \Delta(\cdot)$  表示双差运算符， $i, j$  分别表示参考星和非参考星。与非差非组合观测相比，双差观测可以消除接收机和卫星的钟差以及硬件延迟，另外还可以大幅削弱对流层和电离层误差的影响，实现厘米级高精度定位。双差电离层和对流层的残余量级主要受到基线距离长短的影响，当基线距离小于 10 km 时（短基线），可以忽略双差电离层延迟和双差对流层延迟。因此，在短基线的情况下，双差观测方程可以简化为：

$$\nabla \Delta P_{br,f}^{ij} = \nabla \Delta \rho_{br,f}^{ij} + \varepsilon_{\nabla \Delta P_{br,f}^{ij}} \quad (2.4)$$

$$\nabla \Delta L_{br,f}^{ij} = \nabla \Delta \rho_{br,f}^{ij} + \lambda_f \nabla \Delta N_{br,f}^{ij} + \varepsilon_{\nabla \Delta L_{br,f}^{ij}} \quad (2.5)$$

## 2.2.3 基于因子图优化的 GNSS RTK 估计器

### 2.2.3.1 滑动窗口模型

本代码使用基于滑动窗口的因子图 RTK 估计器，窗口内包含的待估状态定义为：



$$\begin{aligned}
 \boldsymbol{\chi} &= [\boldsymbol{x}_0 \quad \boldsymbol{x}_1 \quad \cdots \quad \boldsymbol{x}_n \quad \Delta N] \\
 \boldsymbol{x}_k &= [\boldsymbol{p}_{b_k,r}^e, \boldsymbol{p}_{b_k,b}^e], 0 \leq k \leq n \\
 \Delta N &= [\Delta N_{br,f}^{ij,G}, \Delta N_{br,f}^{ij,E}, \Delta N_{br,f}^{ij,C}]
 \end{aligned} \tag{2.6}$$

其中  $\boldsymbol{x}_k$  表示第  $k$  个状态节点包含的流动站和基准站位置，本代码中对于基站坐标进行固定不作估计。 $\Delta N$  表示整个窗口内的单差模糊度参数，其中

$\Delta N_{br,f}^{ij,G}, \Delta N_{br,f}^{ij,E}, \Delta N_{br,f}^{ij,C}$  分别表示不同星座的单差模糊度（注：为了在窗口滑动时更方便地管理模糊度参数，代码维持的是站间单差模糊度），G 表示 GPS，E 表示伽利略，C 表示北斗。

在滑动窗口内，连续跟踪的模糊度用于构建不同位置状态节点之间的数据关联，对于窗口内模糊度的添加、删除，以及出现的中断和周跳等情况基于提出的数据关联方法进行模糊度管理，保证滑动窗口内构建正确的模糊度约束关系。当窗口达到预设大小时，将最老帧移出窗口，实现窗口的滑动。同时使用边缘化将最老帧的观测信息添加至先验因子中去。

### 2.2.3.2 非线性优化

因子图 RTK 优化问题的目标函数可以表达为：

$$\boldsymbol{\chi}^* = \arg \min_{\boldsymbol{\chi}} \left\{ \sum_{i,j,f \in \mathbf{P}} \|r_p(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})\|_{\boldsymbol{\Omega}_p^{s,ij,f}}^2 + \sum_{i,j,f \in \mathbf{L}} \|r_L(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})\|_{\boldsymbol{\Omega}_L^{s,ij,f}}^2 \right\} \tag{2.7}$$

其中  $r_p(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})$  和  $r_L(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})$  分别表示双差伪距  $P_{br}^{ij}$  和双差载波  $L_{br}^{ij}$  关于卫星  $s$  在频率  $f$  上的测量残差。 $\mathbf{P}$  和  $\mathbf{L}$  分别表示滑动窗口内所有关于伪距和载波的测量值。

基于双差观测模型(2.4)和(2.5)，伪距残差  $r_p(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})$  和载波相位残差  $r_L(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi})$  可以表示成：

$$\begin{bmatrix} r_p(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi}) \\ r_L(\hat{\boldsymbol{z}}_{br,f}^{ij,s}, \boldsymbol{\chi}) \end{bmatrix} = \begin{bmatrix} \nabla \Delta P_f^{ij} - \nabla \Delta \rho_0^{ij,f} \\ \nabla \Delta L_f^{ij} - \nabla \Delta \rho_0^{ij,f} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J} & 0 \\ \boldsymbol{J} & \lambda_f \end{bmatrix} \begin{bmatrix} \delta \boldsymbol{p} \\ \nabla \Delta N_f^{ij} \end{bmatrix} + \begin{bmatrix} v_{\nabla \Delta P_f^{ij}} \\ v_{\nabla \Delta L_f^{ij}} \end{bmatrix} \tag{2.8}$$

其中  $\nabla \Delta \rho_0^{ij,f}$  表示基于初始流动站坐标  $\boldsymbol{p}_0 = [x_0, y_0, z_0]^T$ ，基站坐标  $\boldsymbol{p}_b = [x_b, y_b, z_b]^T$ ，

参考卫星  $i$  的位置  $\mathbf{p}_i^s = [x_i^s, y_i^s, z_i^s]^T$  以及非参考卫星  $j$  的位置  $\mathbf{p}_j^s = [x_j^s, y_j^s, z_j^s]^T$  计算得到的卫地距离； $\mathbf{J}$  表示双差观测关于位置向量的雅可比矩阵。

在非线性优化过程中需要对双差观测值赋予初始权重，本代码首先基于高度角定权模型获得非差观测值的先验精度，进一步基于误差传播定律获得双差观测值的先验方差：

假设有  $n$  颗卫星，则站间单差后的观测值方差阵为：

$$\mathbf{C}_{br}^s = \begin{bmatrix} (\sigma_r^1)^2 + (\sigma_b^1)^2 & & & \\ & (\sigma_r^2)^2 + (\sigma_b^2)^2 & & \\ & & \ddots & \\ & & & (\sigma_r^n)^2 + (\sigma_b^n)^2 \end{bmatrix} \quad (2.9)$$

其中  $\sigma$  表示非差观测值（伪距或载波）的先验精度。对于星间差，假设卫星 1 为参考星，则定义星间单差矩阵为：

$$\mathbf{S} = \begin{bmatrix} 1 & -1 & 0 & \cdots & 0 \\ 1 & 0 & -1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix} \quad (2.10)$$

则所有双差观测值的先验方差矩阵可以表示为： $\mathbf{\Omega} = \mathbf{S}\mathbf{C}_{br}^s\mathbf{S}^T$ 。最后，方程（3-11）可以通过非线性优化方法求解。本代码依赖 Ceres 库进行非线性优化，优化流程如图 2.2 所示。

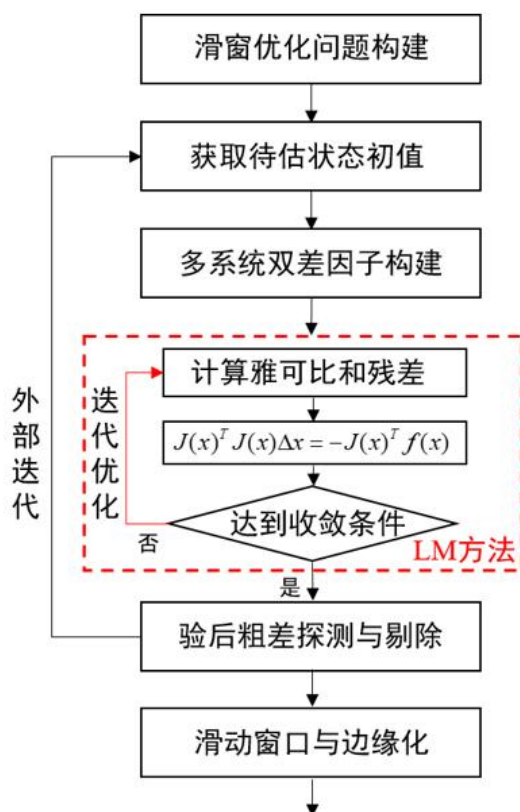


图 2.2 非线性优化流程图

### 2.2.4 基于边缘化的模糊度传递方法

在基于滑动窗口的因子图 RTK 估计器中，通过设置一个最大窗口阈值，来避免窗口长度的无限增长。当窗口大小达到最大阈值时，将最老的位置节点及其对应的 GNSS 观测值一起滑出。在现有的基于因子图的 GNSS 精密定位框架中，通常会选取较大的窗口阈值，以充分利用连续跟踪载波相位观测值构建多个 GNSS 位置节点之间的数据关联。然而，对于载波相位的跟踪，特别是高度角较高的卫星，往往持续多个 GNSS 时段，持续时间远大于窗口长度。因此，在 GNSS 精密定位中仅使用固定大小的窗口而不考虑历史信息传播可能会导致次优的浮点模糊度估计。此外，模糊度固定的可靠性和固定率与浮点模糊度的估计精度密切相关。因此，代码使用了一种基于边缘化的模糊度传递方法，以确保最大程度利用载波相位观测值的连续特性。

边缘化过程的数学表达为：假设  $H\delta x_0 = b$  为最老位置节点对应观测的法方程，其中  $H = J(x)^T \Omega_x^{-1} J(x)$ ， $b = J(x)^T \Omega_x^{-1} l(x)$ 。 $\delta x_0$  可以进一步分为需要边缘

化的状态  $\delta \mathbf{x}_m$  和需要保留的状态  $\delta \mathbf{x}_a$ ，即：

$$\begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{ma} \\ \mathbf{H}_{am} & \mathbf{H}_{aa} \end{bmatrix} \begin{bmatrix} \delta \mathbf{x}_m \\ \delta \mathbf{x}_a \end{bmatrix} = \begin{bmatrix} \mathbf{b}_m \\ \mathbf{b}_a \end{bmatrix} \quad (2.11)$$

本代码方法中  $\delta \mathbf{x}_a$  特指最老位置节点与当前窗口内有共视的模糊度参数，如图 3.6 中绿色状态节点所示，而对只被最老位置节点观测到的卫星，其对应的模糊度参数会被边缘化掉。代码采用舒尔补消元的方法，对式(2.11)构建舒尔补项：

$$\begin{bmatrix} \mathbf{I} & 0 \\ -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{ma} \\ \mathbf{H}_{am} & \mathbf{H}_{aa} \end{bmatrix} \begin{bmatrix} \delta \mathbf{x}_m \\ \delta \mathbf{x}_a \end{bmatrix} = \begin{bmatrix} \mathbf{I} & 0 \\ -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{b}_m \\ \mathbf{b}_a \end{bmatrix} \quad (2.12)$$

进一步可以得到：

$$\begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{ma} \\ 0 & -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{H}_{ma} + \mathbf{H}_{aa} \end{bmatrix} \begin{bmatrix} \delta \mathbf{x}_m \\ \delta \mathbf{x}_a \end{bmatrix} = \begin{bmatrix} \mathbf{b}_m \\ -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{b}_m + \mathbf{b}_a \end{bmatrix} \quad (2.13)$$

基于舒尔补消元实现对  $\delta \mathbf{x}_m$  状态的边缘化操作，其中关于窗口内共视模糊度参数的线性方程为  $(-\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{H}_{ma} + \mathbf{H}_{aa})\delta \mathbf{x}_a = -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{b}_m + \mathbf{b}_a$ ，可以用作模糊度先验信息。通过每次窗口的滑动和边缘化操作可以实现共视模糊度信息的传递，从而实现对连续跟踪载波相位观测值的充分利用。令  $\mathbf{H}_{ap} = -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{H}_{ma} + \mathbf{H}_{aa}$ ， $\mathbf{b}_{ap} = -\mathbf{H}_{am}\mathbf{H}_{mm}^{-1}\mathbf{b}_m + \mathbf{b}_a$ ，则有： $\mathbf{H}_{ap}\delta \mathbf{x}_a = \mathbf{b}_{ap}$ 。最终得到因子图 RTK 状态估计的完整最大后验估计形式：

$$\boldsymbol{\chi}^* = \arg \min_{\boldsymbol{\chi}} \left\{ \|\mathbf{b}_{ap} - \mathbf{H}_{ap}\boldsymbol{\chi}\|_{\boldsymbol{\Sigma}_{ap}}^2 + \sum_{i,j,f \in \mathbf{P}} \|r_p(\hat{\mathbf{z}}_{br,f}^{i,s}, \boldsymbol{\chi})\|_{\boldsymbol{\Sigma}_p^{s,ij,f}}^2 + \sum_{i,j,f \in \mathbf{L}} \|r_L(\hat{\mathbf{z}}_{br,f}^{i,s}, \boldsymbol{\chi})\|_{\boldsymbol{\Sigma}_L^{s,ij,f}}^2 \right\} \quad (2.14)$$

需要注意的是在每次进行窗口边缘化操作的过程中，法方程的构建不仅包含当前窗口与最老位置节点相关的观测，还包括已有的先验项。

## 2.3 基于因子图优化的 RTK/INS 方法

### 2.3.1 概述

在 RTK/INS 导航中，因子图优化算法通过构建包含位置、速度、姿态、IMU 偏差及 GNSS 模糊度参数的概率图模型，在滑动窗口内实现多历元观测信息的联合优化。该方法将系统状态估计问题转化为基于最大后验概率的图优化问题，通过迭代重线性化处理 GNSS 观测方程和 INS 动力学方程中的强非线性项，有效降低了线性化处理带来的精度损失，提升模糊度固定成功率和导航解算的稳定性，特别是在 GNSS 信号短暂中断等挑战环境下表现出优越的容错性能。

### 2.3.2 紧耦合系统定义与因子图表达

本代码中设计的 RTK/INS 系统为观测值级的紧耦合系统，具体定义可见 GREAT-MSF 说明文档，在此不再赘述。

本代码的因子图包括了两类状态节点和四类测量因子，如图所示。状态节点包括动态参数和静态参数，动态参数是随时间变化的，即与惯导相关的状态  $x_{ins}$ 。而静态参数主要是载波相位模糊状态节点  $x_{amb}$ ，如果载波相位锁定不发生跳变或中断，将在整个窗口中保持不变。四类测量因子包括：(a) 边缘化的先验因子；(b) 双差伪距测量因子；(c) 双差载波相位测量因子；(d) IMU 预积分相对因子。图优化问题的目标函数可表示为最小化上述因子的测量残差，如下：

$$\min_x \left\{ \left\| \mathbf{z}_p - \mathbf{H}_p \hat{\mathbf{x}}_{sys} \right\|^2 + \sum_{i \in I} \left\| e_l^i(\mathbf{z}_{i,i+1}^I, \hat{\mathbf{x}}_{sys}) \right\|_{R_l}^2 + \sum_{i \in G} \left\| e_\gamma^i(\mathbf{z}_i^\gamma, \hat{\mathbf{x}}_{sys}) \right\|_{R_\gamma}^2 + \sum_{i \in G} \left\| e_\phi^i(\mathbf{z}_i^\phi, \hat{\mathbf{x}}_{sys}) \right\|_{R_\phi}^2 \right\} \quad (2.15)$$

式中， $e_l$  表示 IMU 预积分相对约束测量残差， $e_\gamma$  和  $e_\phi$  分别表示双差伪距和载波相位测量残差。 $\mathbf{z}_p - \mathbf{H}_p \hat{\mathbf{x}}_{sys}$  是历史信息边缘化的先验值。

图 2.3 展示了基于因子图优化的 GNSS/INS 紧耦合方法架构。该系统主要由 GNSS 模块、IMU 模块、滑动窗口联合优化模块以及整数模糊度固定模块组成。系统工作流程如下：首先，系统在 GNSS 辅助下进行快速位置、速度和姿态的初始化，获取 IMU 偏差的粗略初值。之后，由 IMU 机械编排算法驱动系统进行状态传播和更新。在 GNSS 观测值预处理方面，采用基于 MW 和 GF 组合进行载波相位周跳探测，利用惯导短期高精度的位置信息识别和剔除伪距观测中的显著异常值。同时，利用用户和基准站的观测数据构建双差伪距和载波相位测量因子，将它们与 IMU 相对预积分因子一起加入因子图中。通过紧密耦合所有的测量因子，利用最大后验估计获得系统的最优状态。之后，将后验的浮点模糊度及其协方差矩阵输入整数模糊度固定模块进行整数搜索。随后，将成功固定的整数模糊度作为约束条件反馈至因子图框架，从而获得具有整数特性的厘米级定位精度。此外，估计得到的 IMU 器件偏差将反馈至下一时刻的 INS 状态传播过程。最后，将滑动窗口中最老帧的状态节点和测量因子进行边缘化，并将其作为后续时刻状态估计的先验约束条件，以保持系统的计算效率。

因子图提供了一个模块化的弹性估计框架，能够直观便捷地管理状态节点和测量因子，实现变量和测量信息的灵活增删。并且，由于滑动窗口的设计与测量因子的统一管理，也能够自然地处理多传感器异步、异频采样以及可能存在的测量延迟问题，同时还可以在任意状态节点之间快速建立关联。在本代码中，原始GNSS伪距和载波相位因子是一元绝对测量因子；IMU预积分因子是二元相对测

量因子，它连接了任意两个时刻的状态节点。本节将详细给出几种测量因子的构建方法。

### 2.3.3.1 GNSS 伪距和载波相位因子

GNSS 测量因子包括原始双差伪距测量因子和双差载波相位测量因子。以第  $i$  个伪距和载波相位观测值为例，测量因子可合并表示为

$$\begin{bmatrix} \mathbf{e}_\gamma^i(\mathbf{z}_i^\gamma, \hat{\mathbf{x}}_{\text{sys}}) \\ \mathbf{e}_\varphi^i(\mathbf{z}_i^\varphi, \hat{\mathbf{x}}_{\text{sys}}) \end{bmatrix} = \begin{bmatrix} \gamma_{dd}^i - \gamma_{ins}^i \\ \varphi_{dd}^i - \varphi_{ins}^i \end{bmatrix} = \begin{bmatrix} J_\theta & J_p & \mathbf{0} \\ J_\theta & J_p & J_N \end{bmatrix} \begin{bmatrix} \delta\theta_b^e \\ \delta\mathbf{p}_b^e \\ \delta N_{dd}^i \end{bmatrix} \quad (2.16)$$

式中， $\gamma_{ins}$  和  $\varphi_{ins}$  分别表示由惯导预测得到的双差伪距和相位观测值， $J_\theta$ 、 $J_p$  和  $J_N$  分别表示测量因子相对于姿态、位置和相位模糊度的雅可比系数，式中其他符号定义与前文一致。在每次迭代优化时，这些因子都会在当前估计值  $\hat{\mathbf{x}}_{\text{sys}}$  处进行重新线性化。另外需要注意的是，考虑到因子操作的便捷性和高效的计算，代码忽略了双差测量因子之间的相关性，即假设它们之间的测量噪声是相互独立的。通过遍历窗口内所有时刻的 GNSS 观测值可以构建完整的测量因子，之后将其一同加入因子图中。实际上，在滑动窗口内，对连续跟踪的载波相位观测值仅维护了一个对应的模糊度状态节点。因此，多个历元的同一个相位观测将会关联到相同模糊度节点，这个机制在前文已详细讨论。

### 2.3.3.2 IMU 预积分因子

高频的 IMU 测量不仅能用于连续的航位推算，还可用于计算相邻时刻的相对位置、速度和姿态转换。由于 IMU 数据通常存在较大的随机噪声和系统误差，直接对 IMU 的测量值进行积分容易导致快速误差累积。IMU 预积分是通过将加速度和角速度在一小段时间内积分来估计相对位置、速度和姿态的变化，从而避免连续积分运算导致的位姿漂移。

预积分测量包括了两个测量时刻之间的相对位置、速度和姿态变换信息。根据惯性传感器测量原理， $i$  时刻的姿态、速度和位置信息可以由  $i-1$  时刻的递推得到，即

$$\mathbf{R}_{b,i}^e = \mathbf{R}_{b,i-1}^e \mathbf{R}_{b,i}^{b,i-1} \quad (2.17)$$

$$\mathbf{v}_{b,i}^e = \mathbf{v}_{b,i-1}^e + \mathbf{g}^e \Delta t + \sum_{k=i-1}^i \mathbf{R}_{b,k}^e (\tilde{\mathbf{a}}_{ib}^b - \mathbf{b}_a - \varepsilon_a) \Delta t \quad (2.18)$$

$$\mathbf{p}_{b,i}^e = \mathbf{p}_{b,i-1}^e + \sum_{k=i-1}^i \mathbf{v}_{b,k}^e \Delta t + \frac{1}{2} \mathbf{g}^e \Delta t^2 + \frac{1}{2} \sum_{k=i-1}^i \mathbf{R}_{b,k}^e (\tilde{\mathbf{a}}_{ib}^b - \mathbf{b}_a - \varepsilon_a) \Delta t^2 \quad (2.19)$$

式中， $\mathbf{R}_{b,i-1}^e$ 、 $\mathbf{v}_{b,i-1}^e$ 和 $\mathbf{p}_{b,i-1}^e$ 分别表示  $i-1$  时刻的姿态、速度和位置，其余符号与前文一致。可以看到，递推的状态传递会受到前一时刻状态的影响。因此，可以进一步对上式进行变换，从而导出只依赖 IMU 测量、与之前状态无关的相对信息，即

$$\Delta \mathbf{R}_{b,i}^{b,i-1} = (\mathbf{R}_{b,i-1}^e)^\top \mathbf{R}_{b,i}^e = \prod_{k=i-1}^i (\mathbf{I}_{33} + (\tilde{\boldsymbol{\omega}}_{ib}^b - \mathbf{b}_g - \varepsilon_g) \Delta t) \quad (2.20)$$

$$\Delta \mathbf{v}_{b,i}^{b,i-1} = \mathbf{R}_e^{b,i-1} (\mathbf{v}_{b,i}^e - \mathbf{v}_{b,i-1}^e - \mathbf{g}^e \Delta t) = \sum_{k=i-1}^i \Delta \mathbf{R}_{b,k}^{b,i-1} (\tilde{\mathbf{a}}_{ib}^b - \mathbf{b}_a - \varepsilon_a) \Delta t \quad (2.21)$$

$$\Delta \mathbf{p}_{b,i}^{b,i-1} = \sum_{k=i-1}^i \left( \Delta \mathbf{v}_{b,k}^{b,i-1} \Delta t + \frac{1}{2} \Delta \mathbf{R}_{b,k}^{b,i-1} (\tilde{\mathbf{a}}_{ib}^b - \mathbf{b}_a - \varepsilon_a) \Delta t^2 \right) \quad (2.22)$$

式中， $\Delta \mathbf{R}_{b,i}^{b,i-1}$ 、 $\Delta \mathbf{v}_{b,i}^{b,i-1}$ 和 $\Delta \mathbf{p}_{b,i}^{b,i-1}$ 即为 IMU 预积分的相对姿态、速度和位置测量值。另外，对于四元数的姿态表示，可以得到如下形式，

$$\Delta \mathbf{q}_{b,i}^{b,i-1} = \prod_{k=i-1}^i \left[ \frac{1}{2} (\tilde{\boldsymbol{\omega}}_{ib}^b - \mathbf{b}_g - \varepsilon_g) \Delta t \right] \quad (2.23)$$

式中符号与前文一致。需要注意的是，此处的  $\prod_{k=i-1}^i$  代表四元数的连乘运算。可以看到，IMU 预积分相对因子是在两个惯导状态之间构建的。那么，基于任意两个时刻的预积分测量，可以构建这些测量信息与位置、速度和姿态之间的函数模型。

假设两个时刻之间的预积分时长为  $\Delta t$ ，那么有



$$\mathbf{e}_l^{i-1}(\mathbf{z}_{i-1,i}^l, \hat{\mathbf{x}}_{ins}^i, \hat{\mathbf{x}}_{ins}^{i-1}) = \begin{bmatrix} \hat{\mathbf{R}}_e^{b,i-1} \left( \hat{\mathbf{p}}_{b,i}^e - \hat{\mathbf{p}}_{b,i-1}^e + \mathbf{g}^e \frac{\Delta t^2}{2} - \hat{\mathbf{v}}_{b,i}^e \Delta t \right) - \Delta \mathbf{p}_{b,i}^{b,i-1} \\ \hat{\mathbf{R}}_e^{b,i-1} (\hat{\mathbf{v}}_{b,i}^e - \hat{\mathbf{v}}_{b,i-1}^e + \mathbf{g}^e \Delta t) - \Delta \mathbf{v}_{b,i}^{b,i-1} \\ 2vec(\hat{\mathbf{q}}_e^{b,i-1} \otimes \hat{\mathbf{q}}_{b,i}^e \otimes \Delta \mathbf{q}_{b,i-1}^{b,i}) \\ \hat{\mathbf{b}}_{g,i} - \hat{\mathbf{b}}_{g,i-1} \\ \hat{\mathbf{b}}_{|a,i} - \hat{\mathbf{b}}_{a,i-1} \end{bmatrix} \quad (2.24)$$

式中， $\Delta \mathbf{p}_{b,i}^{b,i-1}$ 、 $\Delta \mathbf{v}_{b,i}^{b,i-1}$ 和 $\Delta \mathbf{q}_{b,i-1}^{b,i}$ 分别表示预积分的位置增量、速度增量和姿态增量，函数运算 $vec(\mathbf{q})$ 表示返回四元数 $\mathbf{q}$ 的虚部矢量。

## 第三章 环境要求和许可证

### 3.1 环境要求

安装包中的 Windows 可执行 CUI 应用程序是由 VS (Microsoft Visual Studio) 在 Windows 11 (64 位)上构建的。文件夹中包含了所有必需的动态链接库。

此外，用户可以使用开源、跨平台的编译工具 CMake 在他们自己的操作系统(Windows)上构建可执行的二进制应用程序。

### 3.2 许可证

GREAT-FGO 是一个开源软件，遵循 GNU 通用公共许可证(版本 3) (<https://www.gnu.org/licenses/gpl-3.0.html>)。

### 3.3 版权

开发人员：

- 武汉大学 GREAT 团队，Wuhan University

三方库：

- GREAT-FGO 使用 G-Nut 库 (<http://www.pecny.cz/>) Copyright (C) 2011-2016  
GOP - Geodetic Observatory Pecny, RIGTC
- GREAT-FGO 使用 pugixml 库 (<http://pugixml.org>) Copyright (C) 2006-2014  
Arseny Kapoulkine
- GREAT-FGO 使用 Newmat 库([http://www.robertnz.net/nm\\_intro.htm](http://www.robertnz.net/nm_intro.htm))Copyright  
(C) 2008: R B Davies
- GREAT-FGO 使用 spdlog 库 (<https://github.com/gabime/spdlog>) Copyright(c)  
2015-present, Gabi Melman & spdlog contributors
- GREAT-FGO 使用 GLFW 库(<https://www.glfw.org/>) Copyright (C) 2002-2006  
Marcus Geelnard, Copyright (C) 2006-2019 Camilla Löwy
- GREAT-FGO 使用 Eigen 库 (<https://eigen.tuxfamily.org>) Copyright (C)  
2008-2011 Gael Guennebaud

- GREAT-FGO 使用 PSINS 库 (<https://psins.org.cn>) Copyright(c) 2015-2025 Gongmin Yan
- GREAT-FGO 使用 Ceres 库(<http://ceres-solver.org/>) Copyright 2023 Google Inc.

## 第四章 在 windows 下编译与安装

该软件包可以通过网站 <https://github.com/GREAT-WHU/GREAT-FGO> 访问。可以通过 git 克隆,也可直接下载将软件包 **GREAT-FGO\_<ver>.zip**。GREAT-FGO 目录结构如下。

表 4.1 GREAT-FGO 软件目录结构

目录/文件	目录/文件	说明
./src		源代码
	./app	GREAT-PVTFGO、 GREAT-GINSFGO、 GREAT-MSF 和 GREAT-PVT 主程序
	./LibGREAT	GNSS 和多传感器融合核心算法库
	./LibGnut	Gnut 库
	./Third-party	第三方库
	CMakeLists.txt	CMakeLists 文件
./ sample_data		算例数据
	./FGO_20211012	GNSS 挑战场景算例
	./FGO_20250928	GNSS 开阔场景算例
./plot		画图脚本
./doc		文档文件
	GREAT-FGO_1.0.pdf	用户指南

以下说明展示了如何在 Windows 上构建 GREAT-FGO 可执行程序。

(1) 通过网站 (<https://cmake.org/download/>) 获取并安装 CMake。请注意, CMake 要求版本为 3.X。本代码已在 3.31.2 版本 CMake 上测试通过。

(2) 执行 cmake-gui。

(3) 执行“Browse Source...”, 然后选择目录 **<install\_dir>/GREAT-FGO\_<ver>/src**, 或者将目录 **<install\_dir>/GREAT-FGO\_<ver>/src** 中的 CMakeList.txt 拖放到 cmake-gui 界面。然后将属性“Where to build the binaries”修改为 **<install\_dir>/GREAT-FGO\_<ver>/build**。

(4) 执行“Configure”并为项目选择集成开发环境（Integrated Development Environment, IDE），这里以选择 Visual Studio 15 2017 为例（仅在第一次单击“Configure”按钮时出现）。

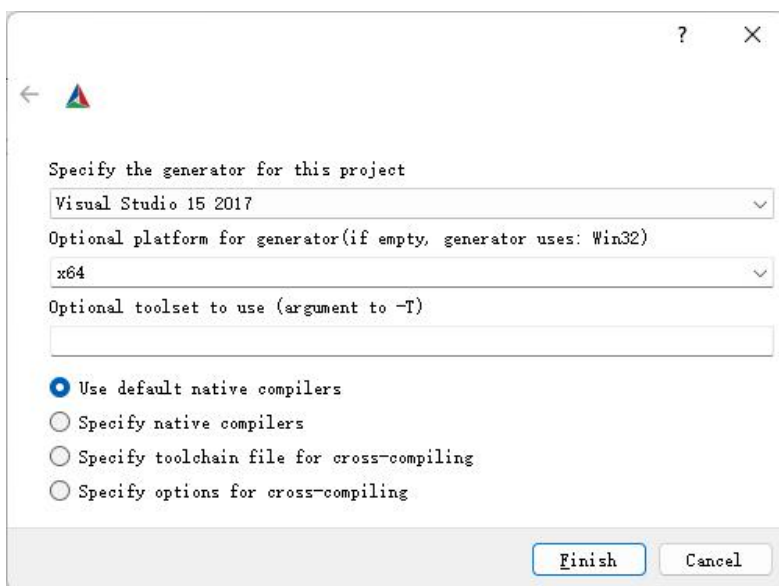


图 4.1 IDE 选择示例

(5) 配置三方库路径。

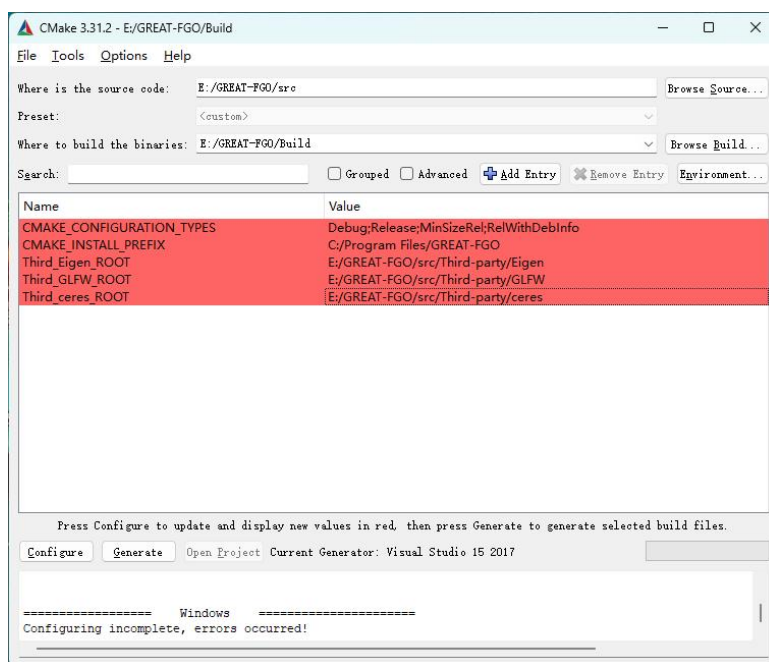
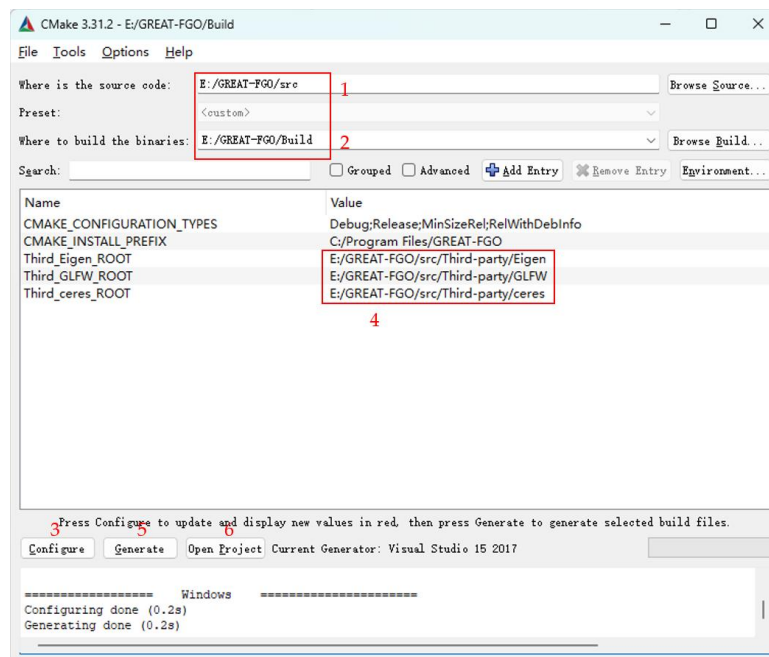


图 4.2 配置三方库路径

(6) 执行“Generate”将构建文件写入 `<install_dir>/GREAT-FGO_<ver>/build`。

(7) 执行“Open Project”，然后在相应的 IDE 中编译源代码。



## 第五章 算例数据处理说明

### 5.1 算例文件目录结构

算例数据在文件夹<install\_dir>/GREAT-FGO\_<ver>/sample\_data 当中。共提供了两组算例数据，一组为 GNSS 开阔场景数据 FGO\_20250928，另一组为 GNSS 挑战场景数据 FGO\_20211012，两组数据均提供了 RTK 和紧组合 RTK/INS 的 XML 配置文件，配置文件说明请参见附录。算例数据文件夹目录结构如下：

表 5.1 算例文件目录结构

目录/文件	说明
./GNSS	GNSS 数据文件夹，包含基站、流动站观测值文件夹和相应的星历、钟差等产品文件夹
./IMU	IMU 数据文件夹
./model	系统模型文件夹
./xml	xml 配置文件夹
./ref	使用 IE 解算出的参考结果文件夹
./result	结果输出文件夹

### 5.2 在 VS 中进行参数配置与运行

- (1) 完成编译后在 Visual Studio 解决方案资源管理器中打开 app 文件夹，右键 GREAT\_PVTFGO，将 GREAT\_PVTFGO 设为启动项目（这里以 RTK 解算为例）。

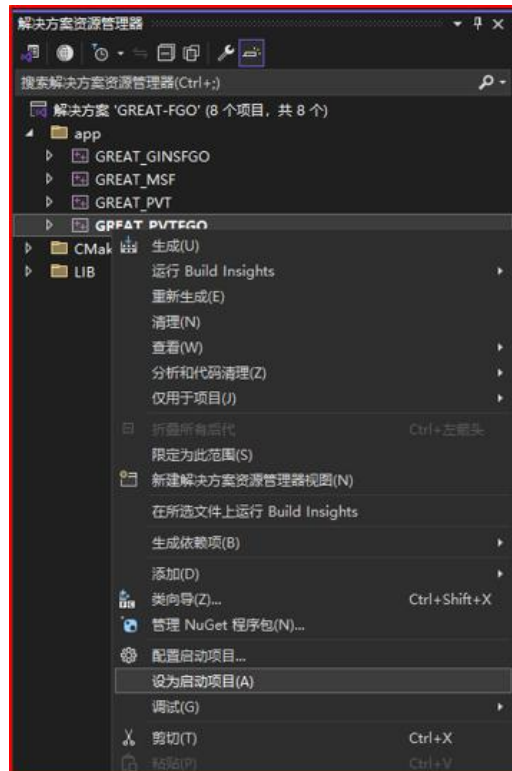


图 5.1 在 VS 中将 GREAT\_PVTFGO 设为启动项目

- (2) 右键 GREAT\_PVTFGO，选择属性，将配置改为 RelWithDebInfo，设置算例所在目录设为工作目录，并设置命令参数。

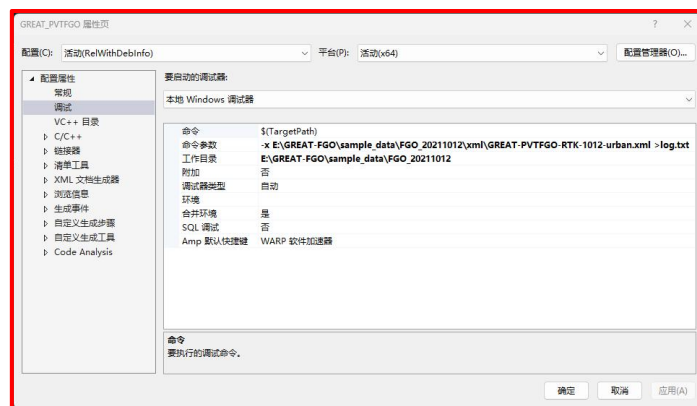


图 5.2 以某算例配置解算命令参数和工作目录设置

- (3) 以 RelWithDebInfo 运行即可

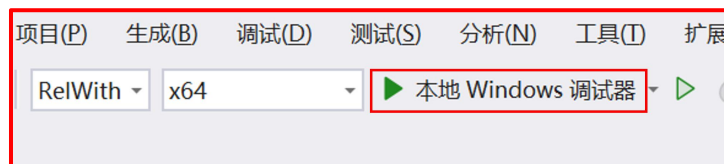


图 5.3 运行 GREAT-PVTFGO



### 5.3 使用绘图脚本进行结果评估

GREAT-FGO 提供了 python 脚本来绘制和分析 RTK 与 RTK/INS 解算结果，这些脚本位于文件夹 **plot** 中。主程序为 **Evaluation.py**，运行程序生成结果后修改如下路径即可运行。这里展示 FGO\_20250928 的 RTK/INS 解算结果，其余数据定位误差序列可查看 **plot** 文件夹下的 **results** 文件夹。

```
# Parameter settings
file_data = r'..\sample_data\FGO_20211012\result\SEPT-RTK-TCI-ADIS-FGO.ins'
file_ref = r'..\sample_data\FGO_20211012\ref\groundtruth_1012_ADIS.txt'
output_dir = r'..\results\20211012'
```

图 5.4 修改 **Evaluation.py** 中的路径

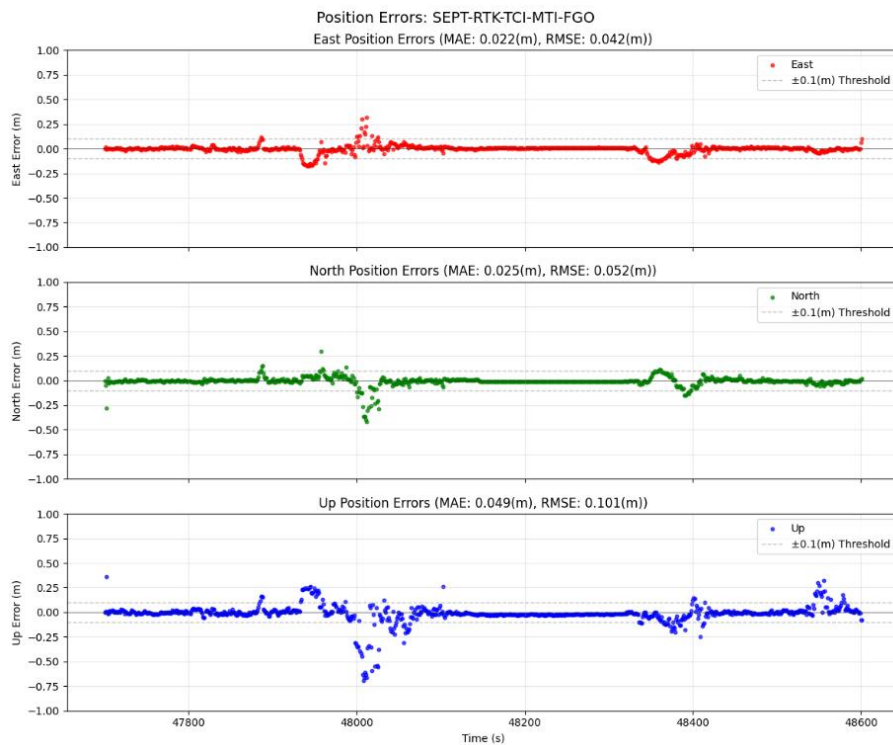


图 5.5 RTK/INS 位置绘图结果(a)

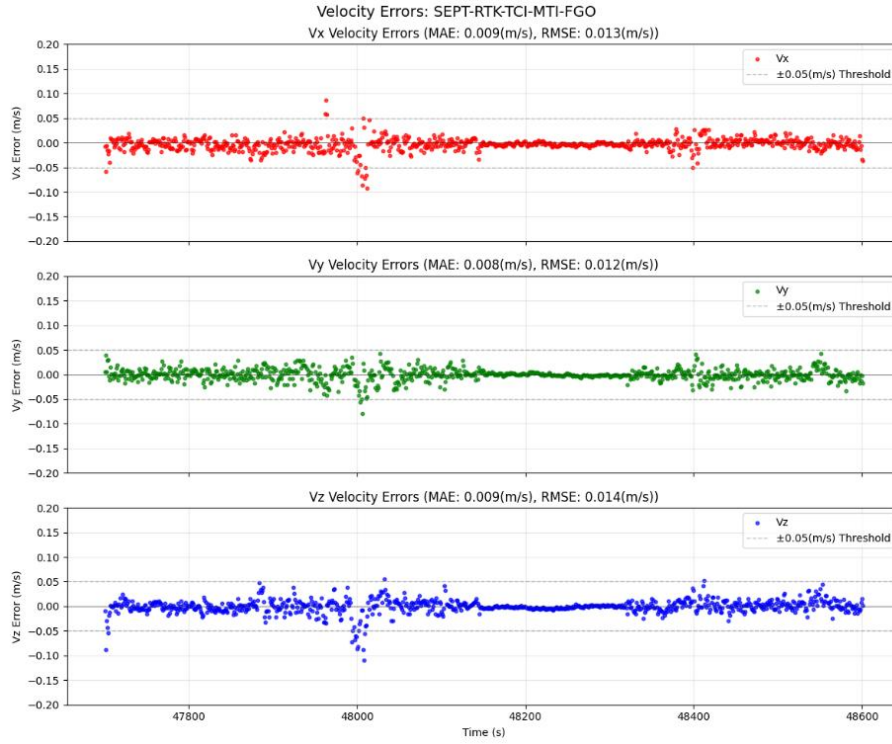


图 5.6 RTK/INS 速度绘图结果(b)

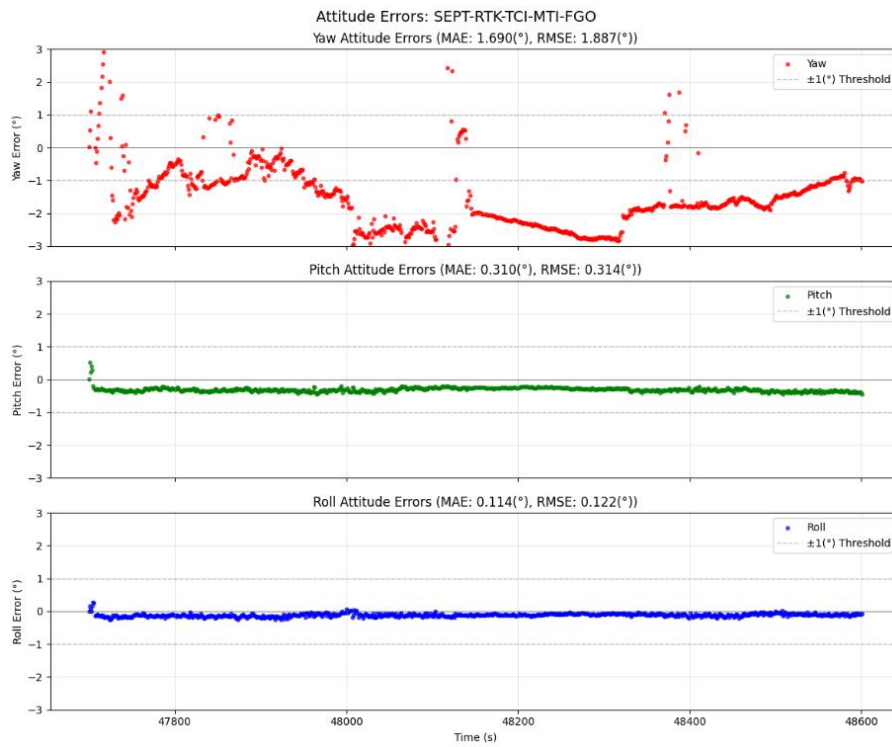


图 5.7 RTK/INS 姿态绘图结果(c)

## 第六章 附录

### 6.1 XML 配置文件说明

配置文件格式为 XML，其中包含了输入输出文件设置、处理方案、解算策略等信息。每行中以"<!--"开始并以"-->"结尾的部分为注释。下表展示了 GREAT-FGO 程序的 XML 文件的格式，其中<fgo>和<fgomsf\_sensors>节点为新增节点，其余含义与 GREAT-MSF 的 XML 配置相同

表 6.1 GREAT-FGO 的 XML 配置文件说明

项目	说明	XML 文件中的 标签或属性
基础设置（一级节点）		<gen>
开始时刻	设置开始时刻的 GPS 时，格式为 "YYYY-MM-DD hh:mm:ss"	<beg>
结束时刻	设置结束时刻的 GPS 时，格式为 "YYYY-MM-DD hh:mm:ss"	<end>
卫星系统	设置参与解算的卫星系统	<sys>
测站列表	设置参与解算的测站列表，4 位字符	<rec>
采样频率	设置观测值的采样频率	<int>
移除卫星	设置不参与解算的卫星 PRN	<sat_rm>
流动站名	流动站名和基站名，仅在 RTK/INS 模式下	<rover>
基站名	使用，在 PPP/INS 模式中不应设置	<base>
基站坐标（一级节点）		<receiver>
参考站坐标	在 RTK 与 RTK/INS 中设置的参考站的坐标(ECEF)	<rec>
输入文件设置（一级节点）		<inputs>
RINEX 观测值 文件	用于进行解算的 RINEX 观测值文件。支持 RINEX 2.10, 2.11, 2.12, 3.00, 3.01, 3.02, 3.03, 3.04 格式。	<rinexo>

项目	说明	XML 文件中的 标签或属性
	程序会根据基站名和流动站名自动识别， 文件名前四位应与设置的站名保持一致。	
RINEX 广播星 历文件	用于进行解算的 RINEX 广播星历文件。支持 RINEX 2.10, 2.11, 2.12, 3.00, 3.01, 3.02, 3.03, 3.04 格式。	<rinexn>
天线文件	用于天线相位中心改正的卫星天线信息文件	<atx>
海洋潮汐文件	用于潮汐改正的海洋潮汐文件	<blq>
行星星历文件	用于计算行星参数	<de>
EOP 参数文件	用于计算旋转矩阵的地球自转参数文件	<eop>
IMU 文件	用于解算的 IMU 文件，格式见后续节点 <ins>-<DataFormat>	<imu>
<b>输出文件设置（一级节点）</b>		<b>&lt;outputs&gt;</b>
日志文件	用于记录输出的日志信息	<log>
RTK 结果	基于图优化的 GNSS RTK 结果文件	<flt>
RTK/INS 结果文件	基于图优化的 RTK/INS 结果文件	<ins>
轨迹文件	可在谷歌地球上查看可视化结果的轨迹文件	<kml>
<b>解算设置（一级节点）</b>		<b>&lt;process&gt;</b>
相位观测值	是否使用载波相位观测值： --true: 使用 --false: 不使用	<phase>
对流层参数	是否估计对流层参数： --false: 不估计	<tropo>

项目	说明	XML 文件中的 标签或属性
电离层参数	是否估计电离层参数： --false: 不估计	<iono>
对流层模型	使用的对流层模型	<tropo_model>
待估参数的先验	测站坐标	<sig_init_crd>
sigma	测站速度	<sig_init_vel>
	对流层	<sig_init_ztd>
	模糊度	<sig_init_amb>
	Galileo 的系统间偏差和频率间偏差	<sig_init_gal>
	GLONASS 的系统间偏差和频率间偏差	<sig_init_glo>
	BDS 的系统间偏差和频率间偏差	<sig_init_bds>
	电离层	<sig_init_vion>
截止高度角	可用卫星观测值的最小高度角	<minimum_elev >
观测值组合	用于解算的观测值组合方式 --IONO_FREE: 无电离层组合, 支持双频 --RAW_ALL: 非差非组合, 支持双频和多 频	<obs_combinatio n>
最大验后残差	用于残差编辑的验后残差阈值	<max_res_norm >
动态模式	是否为动态模式： --true: 是 --false: 不是	<pos_kin>
最小卫星个数	参与解算卫星个数的最小阈值	<min_sat>
观测值加权	确定观测值权重的方法	<obs_weight>
基站坐标获取方	采用何种方式获取基站坐标	<basepos>

项目	说明	XML 文件中的 标签或属性
式	--CFILE: 从 XML 文件中读取 --spp: spp 模式解算基站坐标	
北斗卫星码偏差	是否改正北斗卫星的码偏差: --true: 是 --false: 不是	<bds_code_bias_corr>
周跳探测	周跳探测模型 --default 默认	<slip_model>
观测值频率	用于解算的观测值频率个数	<frequency>
优化设置（一级节点）		<fgo>
优化策略设置	RTK 窗口大小	gnss_window_size
	RTK/INS 窗口大小	gins_window_size
	模糊度传递 --1: default	amb_propagation
	是否使用 gnss 数据 --0: 否 --1: 是	gnss_enable
	是否使用 imu 数据 --0: 否 --1: 是	imu_enable
	最大解算时间	max_solver_time
	最大迭代次数	max_num_iterations
待估参数的先验	加速度计测量白噪声	acc_n

项目	说明	XML 文件中的 标签或属性
噪声	加速度计偏置随机噪声	acc_w
	陀螺仪测量噪声	gyr_n
	陀螺仪偏置随机噪声	gyr_w
	重力大小	gravity_norm
<b>参与优化的传感器设置（一级节点）</b>		<b>fgomsf_sensors</b>
多源传感器类型	多传感器融合类型 目前支持： GINS_TC：卫惯紧组合	msf_type
<b>模糊度固定设置（一级节点）</b>		<b>&lt;ambiguity&gt;</b>
固定模式	是否进行模糊度固定： --NO：不进行模糊度固定 --SEARCH：进行模糊度固定	<fix_mode>
部分模糊度固定	是否进行部分模糊度固定： --NO：不进行 --YES：进行	<part_fix>
部分模糊度固定的个数	部分模糊度固定模式下最小模糊度固定个数	<part_fix_num>
Ratio 值	LAMBDA 方法进行模糊度固定中 ratio 检验的值	<ratio>
参考卫星	是否设置参考卫星 --NO：不设置 --YES：设置	<set_refsat>
最小共视时间	进行模糊度固定时对同一颗卫星的最小共视时间	<min_common_time>
超宽巷观测值	对不同观测值组合进行模糊度固定的相关	<extra_widelane

项目	说明	XML 文件中的 标签或属性
	设置:	<code>_decision&gt;</code>
宽巷观测值	-- alpha&maxdev: 置信区间参数	<code>&lt;widelane_decis</code>
	-- maxsig: 最大 sigma 值	<code>ion&gt;</code>
窄巷观测值		<code>&lt;narrowlane_dec ision&gt;</code>
卫星设置（一级节点）		<code>&lt;gps&gt;/&lt;bds&gt; /&lt;gal&gt;/&lt;glo&gt;</code>
观测值的先验	伪距	<code>sigma_C</code>
sigema	载波相位	<code>sigma_L</code>
频率	卫星频率，对应频段，可选值 1/2/3/4/5	<code>&lt;freq&gt;</code>
卫星	卫星 PRN 号	<code>&lt;sat&gt;</code>
频段	设置不同卫星系统对应频段的观测值 --GPS: 1->L1, 2->L2, 5->L5 --GAL: 1->E1, 5->E5a, 7->E5b, 8->E5, 6->E6 --BDS: 2->B1I, 7->B2I, 6->B3I, 1->B1 C, 5->B2a, 9->B2b, 8->B2a+b --GLO: 1->G1, 2->G2	<code>&lt;band&gt;</code>
惯导设置（一级节点）		<code>&lt;ins&gt;</code>
处理时段	--Start: 启动时间，周内秒 --End: 结束时间，周内秒	<code>&lt;ProcTime&gt;</code>
数据格式	--AxisOrder: IMU 数据文件的格式顺序 (garfu/gaflu/gaf rd), 其中 g 和 a 分别表示陀 螺和加表，后三个字母表示方向(r: right, f:forward, u: up )	<code>&lt;DataFormat&gt;</code>



项目	说明	XML 文件中的 标签或属性
	--GyroUnit: 陀螺单位 (DPS /RPS /RAD /DPH /RPH ) --AcceUnit: 加速度单位 (MPS/MPS2) --Frequency: 采样频率 (单位: Hz)	
对准方式	--OFF: 不设置 --STATIC: 静态对准 --POS: GNSS 位置矢量辅助对准 --VEL: GNSS 速度矢量辅助对准	<Alignment>
初始状态	--Position: 位置, 支持 OFF /Cartesian /Geodetic , Cartesian 格式下单位为米 --Velocity: 速度, 支持 OFF/Cartesian 格式, Cartesian 格式下单位为米每秒 --Attitude: 姿态, 支持 OFF/ON 格式, 单位为度 --GyroBias: 陀螺零偏, 单位为 deg/h --AcceBias: 加速度计零偏, 单位为 mg	<InitialStates>
组合模式设置 (一级节点)		<integration>
GNSS 设置	--Type: 组合类型, 支持设置 TCI --AntennaLever: 天线杆臂, 属性 Type 表示天线在 IMU 坐标系下定义为 RFU, 即右前上, 在 X,Y,Z 中设置杆臂三轴长度 --DelayTime: 在组合量测的历元, GNSS 时间戳与 IMU 时间戳所容许的最大延迟时间	<GNSS>
估计器设置	--IMUErrorModel: IMU 误差模型, 支持 ADIS16470、StarNeto GI7660 两种 IMU,	<Estimator>

项目	说明	XML 文件中的 标签或属性
	也可通过 Customize 自定义误差模型，当使用预设 IMU 型号时，下面的属性不生效	
	--Attitude: 支持设置初始方差 InitialSTD 和过程噪声 ProcNoiseSD，单位为 deg	
	--Velocity: 支持设置初始方差 InitialSTD 和过程噪声 ProcNoiseSD，单位为 m/s	
	--Position: 支持设置初始方差 InitialSTD 和过程噪声 ProcNoiseSD，单位为 m	
	--GyroBias: 支持设置初始方差 InitialSTD 和过程噪声 ProcNoiseSD，单位为 deg/h	
	--AcceBias: 支持设置初始方差 InitialSTD 和过程噪声 ProcNoiseSD，单位为 mg	

下面给出一级节点<fgo>和<fgomsf\_sensors>的示例，其余示例可查看《GREAT-MSF\_1.1》或提供的 XML 文件：

```
<fgo>
  <gnss_enable> 1 </gnss_enable>
  <amb_propagation> 1 </amb_propagation>
  <gnss_window_size> 2 </gnss_window_size>
</fgo>
```

图 6.1 XML 配置文件<fgo>节点示例(RTK)

```
<fgo>
  <acc_n> 0.05 </acc_n>
  <acc_w> 1e-5 </acc_w>
  <gyr_n> 0.005 </gyr_n>
  <gyr_w> 1e-6 </gyr_w>
  <gravity_norm> 9.81 </gravity_norm>
  <imu_enable> 1 </imu_enable>
  <gins_window_size> 2 </gins_window_size>
  <max_solver_time> 8.0000000000000002e-02 </max_solver_time>
  <max_num_iterations> 10 </max_num_iterations>
</fgo>
```

图 6.2 XML 配置文件&lt;fgo&gt;节点示例(RTK/INS)

```
<fgomsf_sensors>
  <msf_type> GINS_TC </msf_type>
</fgomsf_sensors>
```

图 6.3 XML 配置文件&lt;fgomsf\_sensors&gt;节点示例(RTK/INS)

## 6.2 结果文件 (\*\*.fgo/\*\*/\*.ins) 说明

fgo 文件记录了 RTK 定位解算结果及其精度指标，具体如下：

表 6.2 GREAT-FGO fgo 结果文件说明

列号	列名(列 1-19)	列说明	单位	格式
1	Seconds of Week	GPS 周内秒	s	F18.4
2	X-ECEF	ECEF (地心地固系) 下 X 轴坐标	m	F15.4
3	Y-ECEF	ECEF (地心地固系) 下 Y 轴坐标	m	F15.4
4	Z-ECEF	ECEF (地心地固系) 下 Z 轴坐标	m	F15.4
5	Vx-ECEF	ECEF (地心地固系) 下 X 轴速度	m/s	F10.4
6	Vy-ECEF	ECEF (地心地固系) 下 Y 轴速度	m/s	F10.4
7	Vz-ECEF	ECEF (地心地固系) 下 Z 轴速度	m/s	F10.4
8	X-RMS	ECEF (地心地固系) 下 X 坐标 RMS	m	F9.4
9	Y-RMS	ECEF (地心地固系) 下 Y 坐标 RMS	m	F9.4
10	Z-RMS	ECEF (地心地固系) 下 Z 坐标 RMS	m	F9.4
11	Vx-RMS	ECEF (地心地固系) 下 X 轴速度的 RMS	m/s	F9.4
12	Vy-RMS	ECEF (地心地固系) 下 Y 轴速度的 RMS	m/s	F9.4
13	Vz-RMS	ECEF (地心地固系) 下 Z 轴速度的 RMS	m/s	F9.4
14	Nsat	可用卫星数量	#	I5
15	PDOP	卫星几何构型强度	-	F5.2
16	Sigma0	单位权中误差	m	F8.2
17	AmbStatus	模糊度固定情况(e.g., Fixed, Float)	#	A8
18	Ratio	Ratio 值	-	F10.2
19	BL	基线长度	m	F10.3
20	Quality	数据质量指标	#	I1

ins 文件记录了 RTK/INS 定位解算结果及其精度指标，具体如下：

表 6.3 GREAT-FGO ins 结果文件说明

列号	列名(列 1-19)	列说明	单位	格式
1	Seconds of Week	GPS 周内秒	s	F18.6
2	X-ECEF	ECEF (地心地固系) 下 X 轴坐标	m	F18.3
3	Y-ECEF	ECEF (地心地固系) 下 Y 轴坐标	m	F18.3
4	Z-ECEF	ECEF (地心地固系) 下 Z 轴坐标	m	F18.3
5	Vx-ECEF	ECEF (地心地固系) 下 X 轴速度	m/s	F10.3
6	Vy-ECEF	ECEF (地心地固系) 下 Y 轴速度	m/s	F10.3
7	Vz-ECEF	ECEF (地心地固系) 下 Z 轴速度	m/s	F10.3
8	Pitch	俯仰角 (下负上正)	deg	F10.4
9	Roll	横滚角 (左负右正)	deg	F10.4
10	Yaw	偏航角 (北偏西为正)	deg	F10.4
11	GyroBiasX	陀螺仪 X 轴零偏估值	deg/h	F12.4
12	GyroBiasY	陀螺仪 Y 轴零偏估值	deg/h	F12.4
13	GyroBiasZ	陀螺仪 Z 轴零偏估值	deg/h	F12.4
14	AcceBiasX	加速度计 X 轴零偏估值	mg	F12.4
15	AcceBiasY	加速度计 Y 轴零偏估值	mg	F12.4
16	AcceBiasZ	加速度计 Z 轴零偏估值	mg	F12.4
17	MeasType	当前时刻的测量类型 (e.g., GNSS、INS)	-	A10
19	Nsat	可用卫星数量	#	I5
20	PDOP	卫星几何构型强度	-	F7.2
21	AmbStatus	模糊度固定情况(e.g., Fixed, Float)	#	A8
22	Ratio	Ratio 值	-	F8.2

示例：

[illegible]

### 6.3 轨迹文件 (\*\*.kml) 说明

组合解算的轨迹文件，可通过谷歌地球等地图软件打开，可以查看估计精度和各个坐标点的解算信息，如下

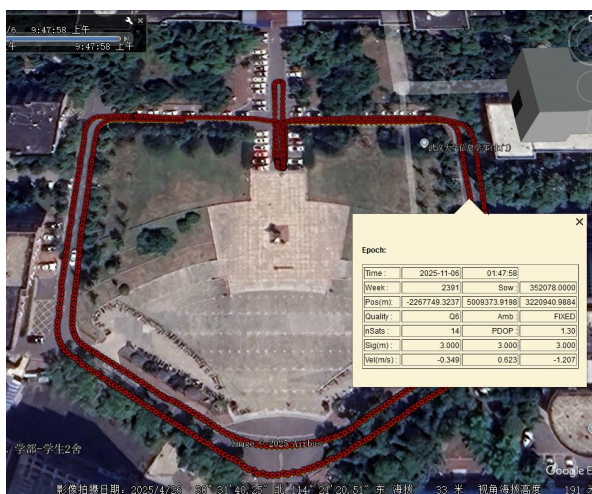


图 6.5 GREAT-FGO 轨迹输出文件打开示例