**2D-InvGra USER MANUAL**

A C++ program package for

Starting and reference models in the inversion of gravity data: application to constrained and joint inversion

Version 2019

By

Shuang Liu(lius@cug.edu.cn),

Jamaledin Baniamerian (j.baniamerian@kgut.ac.ir),

Maurizio Fedi (fedi@unina.it),

Xiangyun Hu (xyhu@cug.edu.cn),

Mahmoud Ahmed Abbas (mahmoud.ahmedabbasahmed@unina.it),

and Mahak Singh Chauhan (mahaksiingh@gmail.com)

**1 Overview**

2D-InvGra is a C++ program package, so it must be compiled using a C++ compiler, such as Microsoft Visual Studio. 2D-InvGra has been tested in Microsoft Visual C++ 6.0. After compiling, an executable software with Graphical User Interface (GUI) will be produced. The readers easily use the GUI to implement the inversion of gravity data.

**2 Declarations for C++ Class / source code**

|  |  |  |
| --- | --- | --- |
| Class | Source file | Declarations |
| CMSIDoc | MSIDoc.h  MSIDoc.cpp | The main frame codes of the Graphical User Interface (GUI). |
| CMSIApp | MSI.h  MSI.cpp |
| CMainFrame | MainFrm.h  MainFrm.cpp |
| CMSIView | MSIView.h  MSIView.cpp |
| CSection | Section.h  Section.cpp | The profile discretization, read and write data, inversion algorithm, etc.  bool CSection::PCGM(const double Msmin,const double Msmax,int iter,double pre,int typeobs) //The function of implementing the inversion of gravity data. |
| CDike2D | Dike2D.h  Dike2D.cpp | The magnetic forward modelling of 2D rectangular model. |
| Matrix | Matrix.h  Matrix.cpp | All operations about matrix. |
| CMyPoint | MyPoint.h  MyPoint.cpp | Class of defining the point position. |

**3 Data file format**

Input data: 3 columns.

X(m) Z(m) gravity(mGal) (e.g., Gravity.dat)

0 -10 0.27778

20 -10 0.303438

40 -10 0.332473

60 -10 0.36536

80 -10 0.402589

100 -10 0.444629

...

**4 Test data**

The test data is the gravity data of synthetic example in the manuscript.