

pyGrav

Basile Hector

Оглавление

1	Обзор	3
2	Быстрый старт	4
3	Функции pyGrav	5
3.1	Загрузка данных	5
3.1.1	Загрузка необработанных данных	6
3.1.2	Загрузка обработанных данных	6
3.2	Поправки данных	6
3.2.1	Земные приливы	6
3.2.2	Океаническая нагрузка	7
3.2.3	Атмосфера	7
3.3	Выбор данных	7
3.4	Уравнивание дрейфа	8
3.5	Вычисление двойных разностей	8
4	Пример для проверки	9
4.1	Использование графического интерфейса pyGrav	9
4.2	Использование сценариев python и объектов pyGrav	10
5	Структура кода	11
5.1	Объектно-ориентированное программирование и данные микрогравиметрии	11
5.2	Графический интерфейс и PyQt	11
5.3	Код pyGrav	11
5.3.1	Объектный файл данных	11
5.3.2	Файл графического интерфейса	12
6	Протокол получения данных	14
6.1	Геометрия петли и номенклатура	14
6.2	Транспортировка	14
6.3	Настройка прибора	14
6.4	Сбор данных	14
7	Среднеквадратическая инверсия и распределение ошибок	16
8	Установка сторонних программ	18
8.1	Установка ETERNA	18
8.2	Установка MCGRAVI	18

1 Обзор

pyGrav [Hector and Hinderer, 2016] разработан для обработки данных микрогравиметрии, и в основном посвящен интервальным гравиметрическим исследованиям. Пользователь может выбрать между графическим пользовательским интерфейсом (GUI) или классическими сценариями Python путём запуска функций **pyGrav** для обработки данных: поправки за приливы и атмосферные явления, выбор данных и корректировку дрейфа. Открытый исходный код написан на Python 2.7 и соответствует объектно-ориентированной схеме, которая позволяет быстро внедрять новые функции/опции. В настоящее время для чтения доступен только формат файла Scintrex CG5 ASCII, но любой другой формат может быть легко добавлен в процедуры чтения **pyGrav**. Ниже кратко излагаются ключевые моменты:

- Вместо управления различными конкретными программами с соответствующими форматами файлов ввода/вывода, дан единый интерфейс для различных этапов обработки (поправок, выбора данных, уравнивание дрейфа, двойные разности ...),
- Обеспечен уникальный и простой в использовании интерфейс для выбора данных, как с графическим, так и с табличным отображением, а также автоматические критерии выбора для ускорения обработки.
- Пакет написан на языке Python 2.7 с открытым исходным кодом, и также может использоваться для переноса других программ (таких как MCGRAVI для компенсации сети, или ETERNA PREDICT для вычисления синтетических приливов).
- Пакет написан в объектно-ориентированном стиле, подходящем для микрогравиметрических данных, для которых четко определены объекты с определенными свойствами/функциями и соблюдена интуитивно понятная иерархия (гравиметрическая **кампания** с несколькими **съёмками**, каждый из которых состоит из разных **петель**, состоящих из нескольких **пунктов**).
- Структура кода (GUI также закодирован в объектно-ориентированном стиле с использованием PyQt) позволяет легко реализовать дополнительные функции (такие как формат ввода-вывода или взаимодействие с другими программами).

Данное руководство пользователя представлено следующим образом:

- В разделе 2 предлагается «**быстрый запуск**» для запуска **pyGrav**.
- **Функции pyGrav** подробно описаны в разделе 3.
- В разделе 4 перечислены шаги для запуска предоставленных **тестовых данных**. Это лучший способ перейти к **pyGrav**.
- **Структура кода** кратко представлена в разделе 5 для тех, кто заинтересован в модификации сценариев на языке python
- В разделе 6 предложен **протокол по сбору данных**, основанный на опыте автора и позволяющий лучше использовать текущие возможности **pyGrav**.
- Раздел 7 напоминает формулы для **уравнивания по методу наименьших квадратов**, и распределение ошибок
- В разделе 8 приведена процедура установки внешних программ, которые могут использоваться в **pyGrav** (ETERNA и MCGRAVI).

2 Быстрый старт

pyGrav – это набор из четырех скриптов на Python с расширением `.py`: `pyGrav_main.py`, `data_objects.py`, `model_Classes_tree_and_table.py`, и `synthetic_tides.py`. Основные функции для обработки микрогравиметрических данных определены в `data_objects.py`, а графический пользовательский интерфейс (GUI) определен в `pyGrav_main.py`. Третий сценарий, `model_Classes_tree_and_table.py`, содержит только функции для отображения и взаимодействия с данными на этапе выбора данных **pyGrav**, в то время как четвертый скрипт, `synthetic_tides.py` это всего лишь модуль, в котором размещены функции для расчета синтетических приливов. **Чтобы запустить пользовательский интерфейс pyGrav, запустите сценарий pyGrav_main.py.** Хороший способ запустить и/или отредактировать скрипт на Python – использовать Spyder, визуальный интерфейс, аналогичный Matlab, который предоставляет редактор сценариев и структуру кода, консоль ... Оказавшись в Spyder, откройте скрипт `pyGrav_main.py` и запустите его с помощью клавиши F5.

- Для пользователей Windows или Mac хорошим вариантом является загрузка и установка Spyder вместе с обширным списком модулей Python. Хорошие варианты – Анаконда (<https://store.continuum.io/cshop/anaconda/>) или Python(x,y) (<https://code.google.com/p/pythonxy/>).
- Для пользователей Linux Spyder можно получить в менеджере пакетов. Также, если установлен Python и все необходимые модули, возможно напрямую запустить скрипт на Python с помощью команды `python pyGrav_main.py` из каталога исходных файлов.

Для получения краткого обзора функционирования **pyGrav** следуйте руководству тестовым данным из раздела 4 данного руководства.

3 Функции pyGrav

Этот раздел является фактическим руководством по использованию программы, описывающим функции и технические характеристики. Программа запускается с единственной доступной опцией «Start project» в меню «File». Пользователю предлагается указать папку ввода, в которой должны храниться все входные данные, и папку вывода, которая будет использоваться программой для записи выходных файлов.

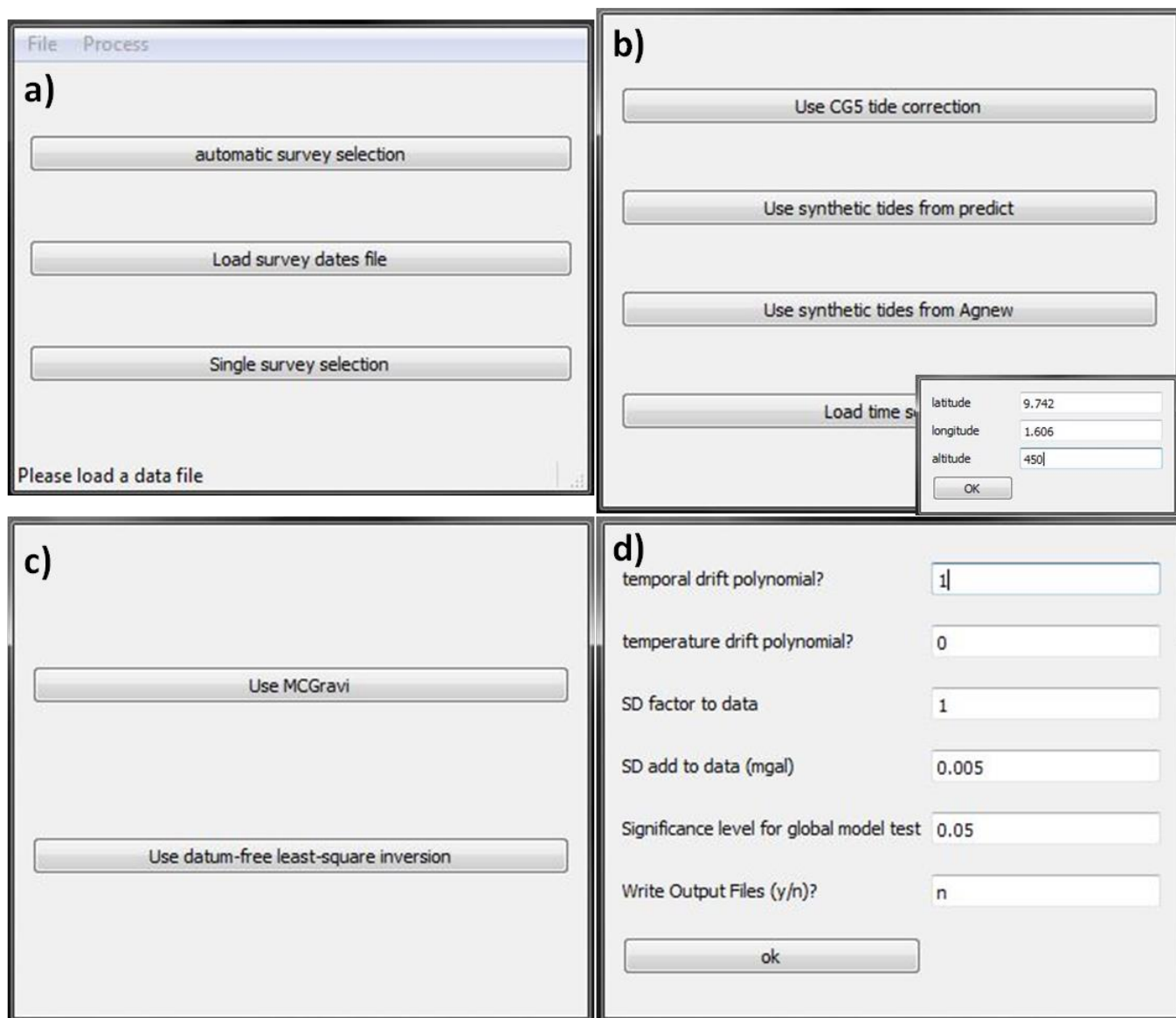


Рис. 3.1: Пример снимков **pyGrav**: а) Экран загрузки данных, б) Экран поправок за прилив, в) Экран компенсации сети и d) параметры компенсации сети.

3.1 Загрузка данных

Доступны два варианта. Для продолжения/изменения обработки данных можно загрузить «raw data» или «processed data». При выборе «processed data», данные уже отсортированы в соответствии с иерархией съёмка/-петля/пункты, в то время как для «raw data» должны быть отсортированы. Таким образом, этот шаг не только загружает данные, но и сохраняет их в соответствии с их иерархией (съёмки – петли – пункты).

3.1.1 Загрузка необработанных данных

Для выбора необработанных данных доступно три варианта:

- Автоматический выбор съемки: для простой геометрии съемки, когда базовая станция всегда одна и та же, эта опция позволяет автоматически определять различные съемки на основе простого временного порога, запрошенного программой: выделяются разные съемки, если время между двумя последовательными сменами станций превышает пороговое значение. Если базовая станция и станция петли – одни и те же, но имеют разные названия, следует выбрать параметр 0.
- Загрузка из файла даты начала/окончания съемки: эта опция позволяет считывать только съемки, определенные между датами начала и окончания, указанными во входном файле. Формат такого входного файла следующий:

```
2012/07/11 05:17:00 2012/07/11 13:00:00
2012/07/13 05:00:00 2012/07/13 22:00:00
...
```

- Ручной ввод дату начала/окончания одной съемки (выбор единичной съёмки)

Затем, петли в съёмках идентифицируются следующим образом: каждый раз, когда обнаруживается базовая станция, запускается новый цикл, а предыдущий завершается. Однако в более поздней процедуре корректировки дрейфа обрабатываются вложенные петли, поскольку каждый цикл является частью одной и той же системы уравнений, которая инвертируется с использованием метода наименьших квадратов.

3.1.2 Загрузка обработанных данных

Загрузка обработанных данных эквивалентна загрузке «project». Она позволяет повторно загрузить уже обработанные данные, которые должны быть предварительно сохранены с помощью «Save processed data» (что эквивалентно сохранению «project»), или упорядочены требуемым образом. Пользователю предлагается загрузить файл, описывающий иерархию данных. Формат такого файла следующий:

```
Directory C:/Users/.../test_case/output_data/
Survey: 2013-09-19 nloops: 4 directory: 2013-09-19
Loop: 1 filename: fn111c13.262.txt
Loop: 2 filename: fn211c13.262.txt
Survey: 2013-09-21 nloops: 4 directory: 2013-09-21
Loop: 1 filename: fn111c13.264.txt
Loop: 2 filename: fn211c13.264.txt
Loop: 3 filename: fn311c13.264.txt
Survey: ...
Loop: ...
```

Строки «Survey» описывают доступные съёмки, названия съёмов (дата первого измерения), количество петель в съёмках, и названия папок съёмов в корневом каталоге. Строки «Loop» описывают для каждой строки в каждой съёмке название петли и файл данных петли. Файлы данных похожи на файлы CGxTool 'c', но без заголовка и дополнительного столбца, содержащего статус данных (1 или 0, независимо от того, сохранена строка данных для корректировки дрейфа или нет).

3.2 Поправки данных

3.2.1 Земные приливы

Доступны четыре варианта поправок за приливы:

- Использование приливной поправки CG5 (ничего делать не нужно): до начала съемки в прибор должны быть введены географические координаты.
- Использование синтетических приливов из прогноза: для пользователей Window для создания синтетического прилива на основе приливных параметров, используемых для поправок, к программе подключается функция PREDICT из программного пакета ETERNA [Wenzel, 1996] (см. раздел 8 об установке). В **pyGrav** запрограммирован приливной потенциал HW95 [Hartmann and Wenzel, 1995], но при необходимости его легко изменить. Приливные параметры – это либо стандартные приливные параметры, либо вводимые пользователем в виде отдельного файла. Необходимо ввести географические координаты съемки.

- Экземпляр прогнозной программы (.exe) должен быть доступен в папке с исходным кодом и копируется **pyGrav** в выходную папку, где производится вычисление.
- Стандартные приливные параметры считываются из файла (200D.INI) в папке основного кода
- Формат файла приливных параметров, вводимых пользователем, должен быть следующим:

0.023812	0.044652	1.13344	0.5445 MM
0.060132	0.080797	1.12607	-0.1195 MF
0.096423	0.249951	1.14548	0.6833 MTM
...			

(частота начала полосы пропускания (cpd) – частота конца полосы пропускания (cpd) – амплитуда – фаза – название прилива)

- Использование синтетических приливов из Agnew: это негармонический метод, предусмотренный кодами Fortran [Agnew, 2007, Agnew, 2012], и позже переведен в MATLAB™ [Cattin et al., 2015] в программе **GravProcess**. В данном случае поправка представляет собой прямое вычисление приливного потенциала по [Munk and Cartwright, 1966]. Он основан на внутренних эфемеридах (для определения положения Луны и Солнца). Необходимо указать местоположение съемки.
- Загрузка временных рядов: если в качестве временных рядов доступен другой синтетический прилив, его можно загрузить и использовать для вычисления поправок. Принятый формат файлов – файлы **ETERNA** или **Tsoft** (.TSF). Если расширение файла не .tsf, он будет рассматриваться как внешний файл. По умолчанию данные должны храниться в первом канале (столбце).

3.2.2 Океаническая нагрузка

Загрузку океана можно скорректировать, используя два различных подхода:

- Если анализ приливов может быть выполнен на съёмочном пункте (например, благодаря близкому расположению сверхпроводящего гравиметра), можно перейти к поправке земных приливов, используя **PREDICT** из пакета **ETERNA** (см. выше) и предоставляя приливные параметры из анализа приливов. Этот эмпирический подход учитывает как земные приливы, так и поправку океанической нагрузки (но может также учитывать другие воздействия окружающей среды, такие как давление воздуха, которые происходят с аналогичной частотой).
- Коррекция океанической нагрузки в программе **pyGrav** такая же, как и в коде процесса **Grave** [Cattin et al., 2015]. Он основан на [Agnew, 2012] и на транскрипции MATLAB™ от [Cattin et al., 2015]. Коэффициенты океанической нагрузки должны быть загружены из отформатированного файла, предоставленного свободным океаническим поставщиком Scherneck (<http://holt.oso.chalmers.se/loading/>). Параметры требуются для полусуточных (M2, S2, N2, K2), суточных (O1, P1, Q1, K1) и долгопериодических (MF, Mm, Ssa) приливных гармоник.

3.2.3 Атмосфера

Загрузите один временной ряд. Принятый формат файлов – файлы **ETERNAL** или **Tsoft** (.TSF). Если расширение файла не .tsf, он будет рассматриваться как внешний файл. По умолчанию данные должны храниться в первом канале (столбце).

3.3 Выбор данных

Это оригинальная функция программы. Иерархия данных видна в виде дерева, таблицы и графического представления. Съёмку, циклы, пункты и единичные измерения можно проверить или снять флажок, следует ли их сохранять для процесса корректировки дрейфа (и окончательных расчетов единичных разности). Это можно сделать либо вручную, либо следуя автоматическим процедурам отбора, основанным на простых пороговых критериях. В настоящее время к ним относятся

- пороговое значение для наклонов: не отмечаются абсолютные значения наклонов, превышающие входное значение
- пороговое значение стандартного отклонения силы тяжести (SD): не отмечаются значения SD, превышающие входные значения
- пороговое значение, основанное на значениях силы тяжести: не отмечаются значения абсолютной силы тяжести, превышающие среднее значение трех последних значений + входное пороговое значение

- критерий длительности: если длительность данных отличается от входного значения, они не отмечаются. Это часто происходит, когда пользователь сохраняет текущие данные в поле при остановке сбора CG5.

Когда выбраны отдельные пороговые значения, они применяются только к текущей таблице. При нажатии кнопки **«apply to all data»** используются все входные пороговые значения, причем для всего набора данных.

Другой вариант быстрой проверки/снятия флажков с данных – выделить несколько строк с помощью мыши и нажать кнопки **«check selected»** или **«uncheck selected»**.

Каждая съёмка и петли идентифицируются по датам их первых измерений. Указаны номера станций, а число в скобках – это номер повторения, поскольку некоторые станции повторяются (например, базовая станция). Данные организованы в хронологическом порядке.

Ручной выбор может быть выполнен на основе табличных значений или графических отображений, в зависимости от предпочтений пользователя. На графическом дисплее непроверенные данные отображаются черным цветом, в то время как отмеченные данные – синим. На экране значения силы тяжести, среднее значение выбранных данных отображается синей горизонтальной линией.

После выбора данных следует нажать кнопку ОК. Это действие мало что дает, но может быть важным: оно используется для проверки того, что на некоторых пунктах данные не выбраны, и в этом случае они удаляются. Это может произойти с помощью автоматического выбора.

3.4 Уравнивание дрейфа

Как только поправки будут применены и данные выбраны, можно приступить к корректировке дрейфа. Доступны два варианта. Можно запустить либо **MCGravi** [Beilin, 2006] (если он установлен, см. раздел 8), либо использовать простую схему инверсии наименьших квадратов без данных [Hwang et al., 2002]. Единственный интерес в использовании **MCGravi** заключается в том, что выполняется компенсация сети и необходимо использовать несколько фиксированных абсолютных (априорных) значений (используя опцию **«weighted constraint»**). Для этого требуется чтобы была доступна `mcgravi.exe`. Программа записывает входные файлы **MCGravi** в выходной каталог и считывает выходные файлы **MCGravi** (файл `*.gra` в папках `mix...`).

Пользователь может выбрать параметры для применения к данным перед инверсией:

- Коэффициент SD: мультипликативный коэффициент на наблюдаемое стандартное отклонение (SD)
- SD_add: константа, добавляемая к каждому наблюдаемому SD

Также следует указать время и температурный дрейф в градусах.

Единичные разности отображаются в консоли и могут быть сохранены в виде файла `SimpleDifferences.dat` в каждой папке опроса выходного каталога, выбрав **File** → **Save simple differences**.

3.5 Вычисление двойных разностей

Как только вычислены единичные разности, двойные разности могут быть вычислены с помощью **Process** → **Compute double differences**. Затем двойные разности можно сохранить в выходной папке, выбрав **File** → **Save double difference**. Это создаст файлы двойных разностей силы тяжести и SD в выходном каталоге. Доступны два формата (даты – пункты или пункты – даты).

4 Пример для проверки

В этом разделе подробно описаны два способа обработки тестового примера. Первый использует графический интерфейс **pyGrav** для быстрой и удобной обработки данных микрогравиметрии. Второй использует сценарий Python, который вызывает функции **pyGrav**. Это больше всего подходит для тех, кто заинтересован в разработке **pyGrav** и добавлении новых функциональных возможностей.

4.1 Использование графического интерфейса pyGrav

Тестовый пример доступен в каталоге `test_case`. Это набор из 4 съемок, проведенных в ходе Западноафриканской кампании в небольшом водосборном бассейне на севере Бенина. Каждая съемка обычно состоит из четырех петель, охватывающих в общей сложности 13 пунктов и базовую станцию (станция № 1).

Чтобы обработать этот набор данных из файла необработанных данных, выполните следующие действия:

- Запустите **pyGrav**: запустите файл Python `pyGrav_main.py` используя `python(x,y)`, например, в Windows, или выполнив `python pyGrav_main.py` в оболочке Linux.
- Запустите проект: выберите подкаталоги `input_data/raw` и `output_data` из каталога `test_case`.
- Выберите **Load raw data** в меню **File**: откройте `Atest_Raw_data.txt`
- Выберите **Load Survey dates file** и загрузите `Atest_start_end_dates.txt`, или в качестве альтернативы, выберите одну съёмку путём введения начальных и конечных дат (см. файл `Atest_start_end_dates.txt`). Номер базовой станции равен 1.
- Обработка приливов: Выберите **Tide corrections** в меню **Process** и выберите **Use CG5 tide corrections** или в качестве альтернативы **Use Synthetic tides from predict** (если установлено, см. раздел 8): Широта = 9.742; Долгота = 1.606; Высота над уровнем моря = 450. В этом случае выберите **Load Tidal parameters** и загрузите `tide_param.txt` из папки `input_data`. Этот список параметров был получен на основе приливного анализа сверхпроводящего гравиметра, доступного на месте исследования, и следовательно, эмпирически включает океаническую нагрузку. В качестве последнего варианта, выберите **Use Synthetic tides from Agnew**: Широта = 9.742; Долгота = 1.606; Высота над уровнем моря = 450.
- Обработка океанической нагрузки: выберите **Ocean Loading correction** в меню **Process** и выберите файл `oceantidal.txt` из папки `input_data`. Этот файл был получен из <http://holt.oso.chalmers.se/loading/> с широтой = 9.742; долготой = 1.606; высотой над уровнем моря = 450.
- Выбор данных: выберите **Data selection** в меню **Process**.
 - Перейдите в дерево на левой панели, чтобы проверить временные ряды для некоторых пунктов, щелкнув на пункт в петле. Сначала все измерения проверяются и отображаются синим цветом в окнах графика (правая панель).
 - Попробуйте снять галочки с некоторых данных в табличном представлении. Графики обновляются при повторном нажатии на пункт в дереве (левая панель) или при нажатии кнопки **update plots** на верхней панели. Черные точки появляются там, где данные не отмечены. Эти измерения не будут сохранены для компенсации сети. Горизонтальная синяя линия на графике силы тяжести – это среднее значение силы тяжести для отмеченных значений.
 - Автоматический выбор. Заполните следующие критерии автоматического выбора и выберите **apply to all data**:
 - * **auto uncheck tilts** >: 5
 - * **auto uncheck g** >: 4
 - * **auto uncheck SD** >: 20
 - * **auto uncheck dur** <>: 60
 - Просмотрите подборку, чтобы завершить фактический выбор (при необходимости).

- Нажмите кнопку ОК. Это действие мало что дает, но может быть важным: оно используется для проверки того, что на некоторых станциях данные не выбраны, и в этом случае они удаляются. Это может произойти с помощью автоматического выбора.
- Теперь обработанные данные можно сохранить: в меню **File** выберите **Save processed data**. В папке **output_data** будут созданы подкаталоги, имена которых являются именами съёмок и которые содержат файлы петель, аналогичные файлам CGxTool 'c', с дополнительным столбцом 0 или 1, указывающим, следует ли сохранить станцию или нет. Вся иерархия данных записана в файле **gravity_data_hierarchy.txt** в папке **output_data**. Этот файл можно открыть в другом проекте для загрузки обработанного набора данных (выбрав пункт загрузить обработанные данные в меню Файл).
- Коррекция дрейфа: выберите **Drift Adjustment** в меню **Process** и **Use datum-free least-square inversion** без привязки к данным с параметрами:
 - **Temporal drift polynomial?: 1**
 - **Temperature drift polynomial?: 0**
 - **SD factor to data: 1**
 - **SD add to data (mgal): 0.005**
 - **Significance level for global model test: 0.05** (т. е. 5%)
 - **Write Output Files (y/n)?: y**
- Теперь дрейф скорректирован и можно проверить выходные файлы, чтобы увидеть, удовлетворяет ли вывод дрейфа или нет. В каждой подпапке съёмки (которая была создана, если она ранее не была создана функцией **Save processed data**) из папки **output-data** два файла, начинающихся с **LSresults**, отображают результаты единичных разностей для первого и полную процедуру LS (ввод, выходные данные и полная проверка) для второго.
- Сохранение единичных разностей: выберите **Save simple difference** в меню **File**. Как и в случае с функцией **Save processed data**, это создаст файлы единичных разностей во вложенных папках папки **output_data** вместе с файлом иерархии, начинающимся с **simple_diff_data_hierarchy...txt**
- Вычисление двойной разности: выберите **Compute double difference** в меню **Process**. Выберите опорную съёмку, например, первую. Затем выберите **Classic double difference**.
- Теперь возможно сохранить двойные разности: выберите **Save double difference** в меню **File**. Это создаст файлы **gravity** и **SD double difference** в каталоге **output_data**. Доступны два формата (даты – пункты или пункты – даты).
- Альтернативой здесь является загрузка уже обработанных данных (в данном случае, уже выбранных данных), и для сравнения результатов уравниваний дрейфа, единичных и двойных разностей с выбранными вами результатами. В этом случае начните с изменения пути к данным в первой строке файла **gravity_data_hierarchy.txt** в каталоге **input_data/preprocessed**. Затем выберите пункт **Load processed data** в меню **File** и выберите файл **gravity_data_hierarchy.txt** в каталоге **input_data/preprocessed**. Этот файл является файлом иерархии, описывающим, где найти файлы петли в каждом каталоге съёмки (как указано в функции **Save processed data**).

4.2 Использование сценариев python и объектов pyGrav

Другой способ использования функций **pyGrav** (в **data_objects.py** файл) заключается в написании сценария на Python и последовательном вызове этих функций. Это особенно подходит, если кто-то хочет добавить функцию в основной код и протестировать ее, не проходя через все программирование с графическим интерфейсом. Как только функция настроена должным образом, дополнительные кнопки/действия графического интерфейса могут быть легко добавлены в **pyGrav_main.py** путем адаптации существующего кода.

Пример сценария (**example_script.py**) можно найти в каталоге **main_code**, и он должен выполняться как скрипт на Python. Цель скрипта – загрузить необработанные данные, извлечь отдельные съёмки на основе дат начала и окончания, выбрать данные для сохранения на основе простых пороговых значений, уравнивать дрейф для каждой съёмки, и в конечном итоге вычислить как единичные, так и двойные разности. Комментируя/раскомментируя некоторые части кода, также можно загрузить уже выбранные и упорядоченные данные.

5 Структура кода

5.1 Объектно-ориентированное программирование и данные микрогравиметрии

Этот мощный способ программирования, для которого особенно подходит Python, основан на наборе определенных объектов (классов), которые содержат оба атрибута (или свойства), такие как временной ряд, имя, словарь других объектов. . . и методы (или функции/определения), которые описывают, что может делать объект. Таким образом, как только программист привыкает к таким концепциям, этот метод программирования не является линейным, но способствует удобочитаемости кода. Метод, например, для записи новых выходных форматов, может быть легко добавлен в соответствующий объект и вызван из основной программы, добавив всего несколько строк и сохранив общий фрейм исходного кода.

Это особенно подходит для данных микрогравиметрии, главным образом потому, что данные могут быть организованы в виде структур (объектов) в соответствии с иерархическим определением (кампания, которая включает в себя несколько съёмок, которые включают в себя несколько петель, которые включают в себя несколько пунктов, которые включают временные ряды полученных данных). Объекты могут быть физически идентифицированы (кампания, съёмка, петля, пункт), а также связанные с ними методы (чтение `ascii`-файла `CG5` должно быть определено в самом широком объекте – кампании; запись входного файла `mcgravi` – файла `'c'` – для корректировки дрейфа должна быть определена в объекте петли, в то время как корректировка дрейфа должна вызываться или основываться на объекте съёмки; выбор данных будет в основном касаться пунктов). Эта логическая иерархия позволяет хранить данные в виде структур.

5.2 Графический интерфейс и PyQt

PyQt использует мощный язык программирования Python и его пригодность для объектно-ориентированного программирования вместе с библиотеками **Qt GUI** (графический пользовательский интерфейс). **PyQt** состоит из нескольких модулей, таких как фундаментальный **QtCore** (для функций, отличных от GUI, таких как использование файлов, потоков, процессов или времени) и **QtGui** (для графических компонентов), среди которых определены от десятков до сотен классов, содержащих многочисленные функции и свойства.

5.3 Код `pyGrav`

Код основан на объектно-ориентированном программировании (ООП). Это можно рассматривать как два параллельных набора определений:

- Функции хранения данных и манипулирования ими: ядро программы, где определены классы и функции для операций с данными. Это тот самый `data_object.py` исходный файл.
- Функции GUI: функции графического интерфейса пользователя, которые связывают требования пользователя с операциями с данными, определенными в предыдущем наборе определений. Это тот самый `pyGrav_main.py` исходный файл.

В целом, код обильно прокомментирован, и описан каждый класс, подкласс, функции и свойства.

5.3.1 Объектный файл данных

Этот модуль содержит основные классы программы:

- Базовый класс – это объект типа `ChannelList`, который в основном содержит списки каналов, подобные тем, что содержатся в выходных файлах `CG5 ascii` (сила тяжести, наклоны, температура, стандартное отклонение, время . . .).

Производные классы следуют логической иерархии, где каждый «подкласс» создается как элемент словаря из родительского класса:

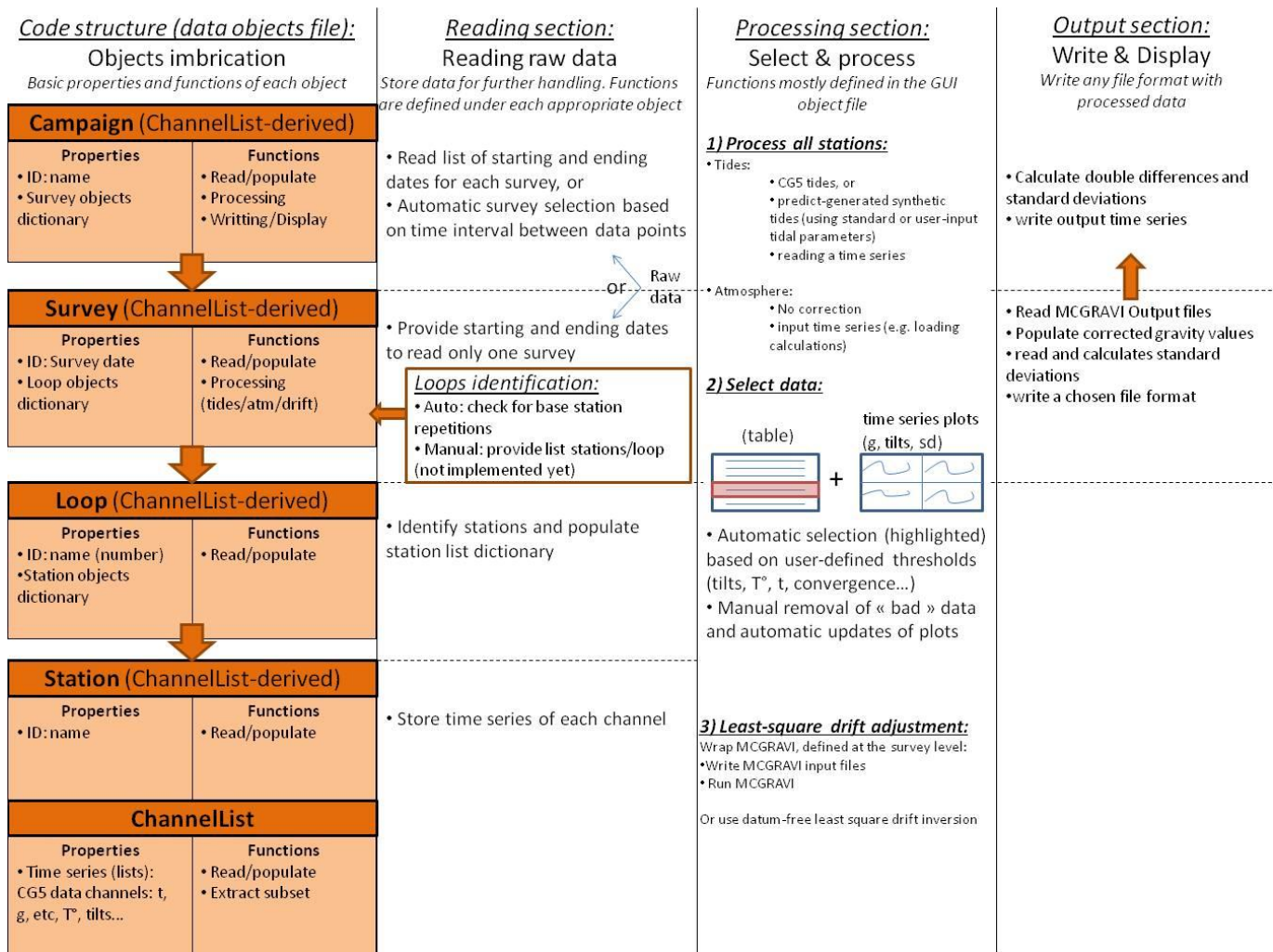


Рис. 5.1: Диаграмма **pyGrav** и обозначения структур (объектов): в объекте **Campaign** есть словарь из нескольких **Survey**. В объекте **Survey** есть словарь из нескольких **Loop**. В объекте **Loop** есть словарь из нескольких **Station**. Каждый из этих объектов является производным от объекта **ChannelList** (т.е. они содержат несколько временных рядов).

- класс **Campaign** содержит словарь объектов типа **Survey**
 - * класс **Survey** содержит словарь объектов типа **Loop**
 - класс **Loop** содержит словарь объектов типа **Station**
 - класс **Station** о содержит временные ряды для каждого пункта

Каждый из этих объектов является производным от объекта списка каналов. Экземпляр базового класса **Campaign** содержит весь набор данных. У каждого класса также есть определенные свойства и функции заполнения, записи, манипуляции и обработки, вызываемые из основной программы (рис. 3.1).

5.3.2 Файл графического интерфейса

Это файл, который должен быть выполнен для запуска **pyGrav**. Он содержит единственный класс под названием **mainProg**, который является объектом **QMainWindow**, производным от **Qt**. Наиболее важными свойствами класса **mainProg** являются объект **Campaign**, который содержит весь набор данных, а также каталоги данных и выходных данных. Большинство функций класса **mainProg** связывают пользовательский интерфейс (определенный в функциях) с кодом обработки, записанным в файле объекта данных, для изменения состояний объекта **Campaign** (набора данных).

Ссылки

Несколько ссылок для тех, кто заинтересован в изменении кода

- Несколько руководств по Python:
<https://docs.python.org/2/tutorial/> Руководство для Python версии 2.7

<http://zetcode.com/lang/python/>

<http://marvin.cs.uidaho.edu/Teaching/CS515/pythonTutorial.pdf> (Руководство G. Van Rossum)

- Руководство PyQt:

<http://zetcode.com/gui/pyqt4/>

- Ссылки на класс PyQt:

<http://pyqt.sourceforge.net/Docs/PyQt4/classes.html>

- Демо графиков PyQt:

<http://eli.thegreenplace.net/2009/05/23/more-pyqt-plotting-demos/>

- Руководства программирования вида модели:

<http://www.yasinuludag.com/blog/?p=98>

- Руководство Matplotlib:

http://web.archive.org/web/20100830233506/http://matplotlib.sourceforge.net/leftwich_tut.txt

6 Протокол получения данных

Для правильной работы программы требуются некоторые основные шаги, касающиеся процедуры сбора данных. Некоторые процедуры сбора данных также следует применять в более общем смысле, чтобы получать более точные результаты и проводить эффективные обследования. Подробный анализ и описание гравитационных сетей см., например ([Lambert and Beaumont, 1977, Torge, 1980]). Требования и рекомендации по высококачественным проектам обследования можно найти в [Seigel et al., 1995].

6.1 Геометрия петли и номенклатура

- Отдельные пункты всегда должны иметь один и тот же идентификационный номер (название), независимо от их статуса (опорный/рядовой пункт). Если он изменен, его можно модифицировать вручную, используя процедуру поиска/замены в файле необработанных данных в текстовом редакторе.
- Различные опорные пункты в рамках одной схемы обработки до сих пор не обрабатываются, но программа может запускаться последовательно для каждого подмножества данных.
- Предпочтительно, но не обязательно, чтобы опорный пункт имел другое название, чем пункт цикла (если он соответствует тому же местоположению).

6.2 Транспортировка

- CG5 очень чувствителен к транспортировке: на короткие расстояния его могут переносить два оператора, которые держат рейку, на которую за ручки подвешена сумка с гравиметром. На больших расстояниях он должен быть в лучшем случае изолирован от вибрации автомобиля.
- Транспортировка обычно подразумевает так называемый кратковременный «транспортный дрейф», и стабилизация пружины после приезда на пункт может занять некоторое время. Этап выбора данных в **pyGrav** позволяет выбрать данные после стабилизации прибора.

6.3 Настройка прибора

- Поскольку вертикальные перемещения сильно влияют на изменение силы тяжести ($\approx 0,3$ мкГал/мм при номинальном градиенте в свободном воздухе), пункты должны быть оборудованы бетонными опорами, чтобы ограничить последствия мягкого грунта или других неустойчивых элементов. В качестве альтернативы можно было бы использовать передаточные стержни для ограничения роли усадки/набухания грунта. Для каждой съемки (в режиме интервальной съемки) прибор должен располагаться точно в одной и той же точке и удерживаться под одним и тем же азимутом, например, благодаря винтам, зафиксированным в бетонной опоре. Высоту относительно опоры следует поддерживать постоянной с помощью латунного кольца, прикрепленного к одной ножке штатива прибора, как предложил [Montgomery, 1971].
- CG5 и его штатив должны быть защищены от ветра и прямых солнечных лучей, например, мусорным баком, покрытым изоляционным материалом, или соответствующим зонтиком.

6.4 Сбор данных

- В настоящее время нет возможности применить различные поправки к приливам в рамках одной и той же съемки. Следовательно, если съемка состоит из пунктов, удаленных друг от друга на большое расстояние (несколько десятков км, не рекомендуется для высокоточных интервальных гравиметрических съемок), пользователь должен позаботиться о том, чтобы ввести правильные местоположения пунктов в CG5 на местах, чтобы можно было применить поправку CG5 за земные приливы.

- Длительность: CG5 производит выборку с частотой 6 Гц и усреднение по заданной пользователем длительности для получения одного единственного измерения. [Merlet et al., 2008] показали в своем исследовании, что отклонение по Аллану их CG5 достигло 1 мкГал через 40 с, до минимума в 0,8 мкГал через 85 с, но еще больше увеличилось из-за влияния приливов. [Gettings et al., 2008] также показывают пример из необработанных данных за 1 секунду (их рисунок 2), где среднее значение сходится примерно через 40 секунд. Кроме того, по истечении 60 секунд стандартная ошибка на экране дисплея CG5, работающего в полевых условиях, всегда, по-видимому, сходится. Сбор данных за 60 секунд часто позволяет сделать динамичный и быстрый вывод о стабильности прибора (см., например, [Hector et al., 2015]).
- Идентификация стабилизации: Существует столько вариантов, сколько требуется пользователям и научным целям для определения того, является ли полученный временной ряд достаточно длинным (стабилизированным) или нет. Строгая процедура идентификации сигналов малой амплитуды при интервальных исследованиях может быть следующей: на станции проводится первая серия из 5 измерений, пока операторы находятся на расстоянии нескольких метров от прибора, прежде чем в первый раз проверяется горизонтирование. Затем проводится еще один набор измерений, и операторы приходят проверять стабильность работы прибора примерно каждые 5 минут. При любой проверке, если наклоны выходят за пределы диапазона $0 \pm 5''$ (или любого другого выбранного порогового значения), прибор снова горизонтируется. Это следует из [Merlet et al., 2008], которые обнаружили, что внутренняя поправка наклона CG5 является точной на уровне 1 мкГал при $\pm 20'$. Кроме того, им удалось сохранить наклоны в пределах $0 \pm 3'$ в условиях их внутреннего и стабильного постаментов. Измерения можно считать выполненными (и гравиметр стабильным), если выполнены все следующие критерии:
 - выполнено минимум 10 соответствующих измерений;
 - изменения силы тяжести составляют 3 мкГал или менее в течение 5 последовательных измерений;
 - в 5 последних измерениях нет видимого дрейфа (дрейф < 1 мкГал/мин).

Все это означает, что долгосрочный дрейф CG5 внутренне корректируется прибором, скажем, при остаточном дрейфе < 100 мкГал/сут (около 1 мкГал/15 мин).

7 Среднеквадратическая инверсия и распределение ошибок

Гравиметрические наблюдения на пункте представляют собой временные ряды из нескольких измерений силы тяжести, каждое измерение является средним значением нескольких выборок. Например, типичные результаты измерения CG-5 получены в результате сбора фрагментов в течение более минуты (наилучшим считается 85 секунд [Merlet et al., 2008]) при частоте 6 Гц. Таким образом, стандартная погрешность (SE) при каждом измерении составляет:

$$SE = \frac{\sigma}{\sqrt{n}} \quad (7.1)$$

где σ – стандартное отклонение измерения, а n – количество выборок.

После отбора данных из оставшихся временных рядов выводятся одно наблюдение l_i и стандартное отклонение σ_i для каждого пункта i с использованием средних, взвешенных по дисперсии:

$$l_i = \frac{\sum_{j=1}^n \frac{1}{SE_j^2}}{\sum_{j=1}^n \frac{1}{SE_j^2}} \quad (7.2)$$

$$\sigma_i = \frac{1}{\sum_{j=1}^n \frac{1}{SE_j^2}} \quad (7.3)$$

Стандартное отклонение σ_{ij} для наблюдения относительной силы тяжести Δl_{ij} между пунктами i и j определяется как

$$\sigma_{ij} = \sqrt{\sigma_i^2 + \sigma_j^2}, \quad (7.4)$$

где Δl_{ij} определяется как

$$\Delta l_{ij} + v_{ij} = g_j + g_i + \sum_{T=1}^m a_T (t_j - t_i)^T + \sum_{k=1}^m b_k (t_j - t_i)^k \quad (7.5)$$

Здесь v_{ij} – невязки, g_i – значение силы тяжести на пункте i , m – степень полинома коэффициентов a_T для дрейфа гравиметра и n – степень полинома коэффициентов b_k для температурного дрейфа. Таким образом, система уравнений является

$$\mathbf{L}^b + \mathbf{V} = \mathbf{A}\mathbf{X}, \quad (7.6)$$

где \mathbf{L}^b вектор, содержащий n наблюдений относительной силы тяжести, \mathbf{V} – вектор с n невязками, \mathbf{A} – дизайн-матрица и \mathbf{X} – вектор u неизвестных (значений силы тяжести и параметров дрейфа). Чтобы получить решение \mathbf{X} , необходимо удерживать фиксированным по крайней мере одно значение силы тяжести во время уравнивания (так называемое исходное значение силы тяжести). Здесь это делается путем добавления наблюдений абсолютного значения силы тяжести (обычно равной нулю) на опорном пункте:

$$\mathbf{L}_g^b + \mathbf{V}_g = \mathbf{A}_g \mathbf{X} = [\mathbf{I}0] \begin{bmatrix} \mathbf{X}_g \\ \mathbf{X}_I \end{bmatrix} \quad (7.7)$$

Введя вектор S с k (числом значений силы тяжести), первые значения которых равны единице, а последние значения $u - k$ равны нулю, что удовлетворяет:

$$\mathbf{S}^T \mathbf{X} = 0 \quad (7.8)$$

можно найти решение по методу наименьших квадратов для X :

$$\mathbf{X} = (\mathbf{A}^T \mathbf{P} \mathbf{A} + \mathbf{S} \mathbf{S}^T)^{-1} \mathbf{A}^T \mathbf{P} \mathbf{L}^b \quad (7.9)$$

Матрица весов \mathbf{P} состоит из членов, обратных дисперсии для наблюдений.

Апостериорная дисперсия единичного веса (σ_0^2) задается формулой

$$\sigma_0^2 = \frac{\mathbf{V}^T \mathbf{P} \mathbf{V}}{n + 1 - u} \quad (7.10)$$

и апостериорная ковариационная матрица \mathbf{S}_X задается путем распространения ковариации:

$$\mathbf{S}_X = \sigma_0^2 (\mathbf{A}^T \mathbf{P} \mathbf{A} + \mathbf{S} \mathbf{S}^T)^{-1} \mathbf{A}^T \mathbf{P} \mathbf{A} (\mathbf{A}^T \mathbf{P} \mathbf{A} + \mathbf{S} \mathbf{S}^T)^{-1} \quad (7.11)$$

Код также переходит к проверке по глобальной модели, чтобы оценить, адекватна ли математическая модель или в данных есть отклонения. Если выполняется следующее условие, то модель уравнивания можно считать правильной и заверенной до уровня значимости α :

$$\chi^2 = \frac{\mathbf{V}^T \mathbf{P} \mathbf{V}}{\sigma_{0i}^2} < \chi_c^2(1 - \alpha; m) \quad (7.12)$$

где σ_{0i}^2 – априорная дисперсия единичного веса, а $\chi_c^2(1 - \alpha; m)$ – критическое значение распределения χ^2 , когда доверительный уровень равен α , а степень свободы уравнивания равна m .

$$\chi_c^2(1 - \alpha; m) = m \left[\chi_{1-\alpha} \left(\frac{2}{9m} \right)^{\frac{1}{2}} + 1 - \left(\frac{2}{9m} \right) \right]^3 \quad (7.13)$$

и для $0 < \alpha < 0.5$

$$\chi_{1-\alpha} = t - \frac{2.515517 + 0.802853t + 0.010328t^2}{1 + 1.432788t + 0.189269t^2 + 0.001308t^3} \quad (7.14)$$

и

$$t = \sqrt{2 \ln \left(\frac{1}{\alpha} \right)} \quad (7.15)$$

8 Установка сторонних программ

pyGrav предоставляет конструкцию для обработки данных микрогравиметрии. Таким образом, вызовы внешних программ в **pyGrav** возможно легко включить. В настоящее время могут быть вызваны две программы: функция **PREDICT** из пакета **ETERNA** [Wenzel, 1996] для расчета синтетических приливов и **MCGRAVI** [Beilin, 2006] для уравнивания сети. Чтобы добавить любой другой вызов к внешним программам, лучший способ – просмотреть скрипты **pyGrav** и вдохновиться тем, как эти две программы взаимодействуют. Здесь описана установка таких программ, если они будут использоваться в **pyGrav**.

8.1 Установка ETERNA

Для пользователей Windows можно использовать функцию **ETERNA** из пакета **ETERNA** [Wenzel, 1996] для поправок за прилив. Это особенно подходит, когда доступны параметры приливов для конкретного места (например, с помощью анализа приливов по данным сверхпроводящего гравиметра, поскольку это также включает эффект океанической нагрузки). Каталог **pyGrav** включает в себя облегченную версию **ETERNA** с минимумом, необходимым для правильной работы функции **PREDICT**. Просто скопируйте папку `/eterna33` из папки `main_code/external_files/` в корневой каталог (`C:\`). Когда **pyGrav** попросит запустить `predict`, он скопирует экземпляр файла `predict.exe`, присутствующий в папке `main_code/external_files/`, в выходной каталог съёмки и запустите его. Эта программа `predict.exe` затем вызовет данные о приливном потенциале из папки `C:/eterna33`. Полный пакет **ETERNA** также доступен бесплатно на сайте ICET: <http://www.upf.pf/ICET/soft/index.html>.

8.2 Установка MCGRAVI

MCGravi [Beilin, 2006] может использоваться для поправки за дрейф по методу наименьших квадратов и уравнивания сети в случае сложной сети с несколькими известными абсолютными точками (инверсия методом наименьших квадратов со взвешенными ограничениями, см. [Hwang et al., 2002]). В качестве альтернативы, алгоритм инверсии наименьших квадратов без данных [Hwang et al., 2002] закодирован в **pyGrav**.

- Скопируйте папку **MCGRAVI** в корневой каталог (`C:\`) (или в другое место).
- Возможно потребуется перекомпилировать **MCGRAVI**. В этом случае необходим g95. Его можно найти здесь: <http://www.g95.org/downloads.shtml>, или <http://math.hawaii.edu/~dale/190/fortran/fortran-windows-installation.html>, или <http://www.fortran.com/the-fortran-company-homepage/whats-new/g95-windows-download/>
- с Intel Fortran:
 - Запустите **Visual Studio** и откройте проект `Mc_gravi.vfproj`. Никаких конкретных параметров не требуется
- с g95:
 - `cmd.exe` (или запустите `cmd` в начальном меню), чтобы открыть консоль `dos`. Перейдите в директорию `mcgravi`

```
cd /mcgravi
```

Если необходимо

```
make clean
```

или

```
del *.o  
del *.mod
```

`make all`

– чтобы запустить **mcgravi**, скопируйте `mcgravi.exe` в рабочую директорию и `conf.conf` в окне `dos`

- Добавьте путь к исполняемому файлу в переменную окружения **PATH**. Перейдите в раздел **Advanced system parameters** на странице **System** (панель конфигурации). Выберите переменную окружения. На нижней панели (системные переменные) выберите строку **Path** и **Modify**. Скопируйте/вставьте значение переменной в текстовый редактор (**Ctrl+A/Ctrl+C** → **Ctrl+V**). Добавьте путь, за которым следует `;`. Больше ничего не меняйте.
- Установите **Perl**
- Также для вывода карт из **mcgravi** требуется **GMT**, но все работает нормально, если он не установлен.

Литература

- [Agnew, 2007] Agnew, D. C. (2007). *Treatise on geophysics*, chapter 3.06 - Earth Tides, pages 163–195. Elsevier, 1 edition.
- [Agnew, 2012] Agnew, D. C. (2012). SPOTL: Some programs for ocean-tide loading. Technical report, UC San Diego: Scripps Institution of Oceanography.
- [Beilin, 2006] Beilin, J. (2006). Apport de la gravimétrie absolue á la réalisation de la composante gravimétrique du réseau géodésique français. Master’s thesis, Inst. Géogr. Natl., Paris, France.
- [Cattin et al., 2015] Cattin, R., Mazzotti, S., and Baratin, L.-M. (2015). GravProcess: An easy-to-use MATLAB software to process campaign gravity data and evaluate the associated uncertainties. *Computers & Geosciences*, 81:20–27.
- [Gettings et al., 2008] Gettings, P., Chapman, D. S., and Allis, R. (2008). Techniques, analysis, and noise in a Salt Lake Valley 4D gravity experiment. *GEOPHYSICS*, 73(6):WA71–WA82.
- [Hartmann and Wenzel, 1995] Hartmann, T. and Wenzel, H.-G. (1995). The HW95 tidal potential catalogue. *Geophysical Research Letters*, 22(24):3553–3556.
- [Hector and Hinderer, 2016] Hector, B. and Hinderer, J. (2016). pygrav, a python-based program for handling and processing relative gravity data. *Computers & Geosciences*, 91:90–97.
- [Hector et al., 2015] Hector, B., Séguis, L., Hinderer, J., Cohard, J., Wubda, M., Descloitres, M., Benarrosh, N., and Boy, J. (2015). Water storage changes as a marker for base flow generation processes in a tropical humid basement catchment (B enin): Insights from hybrid gravimetry. *Water Resources Research*, 51(10):8331–8361.
- [Hwang et al., 2002] Hwang, C., Wang, C.-G., and Lee, L.-H. (2002). Adjustment of relative gravity measurements using weighted and datum-free constraints. *Computers & Geosciences*, 28(9):1005–1015.
- [Lambert and Beaumont, 1977] Lambert, A. and Beaumont, C. (1977). Nano variations in gravity due to seasonal groundwater movements: Implications for the gravitational detection of tectonic movements. *Journal of Geophysical Research*, 82(2):297–306.
- [Merlet et al., 2008] Merlet, S., Kopaev, A., Diament, M., Geneves, G., Landragin, A., and Pereira Dos Santos, F. (2008). Micro-gravity investigations for the LNE watt balance project. *Metrologia*, 45(3):265–274.
- [Montgomery, 1971] Montgomery, E. L. (1971). *Determination of coefficient of storage by use of gravity measurements*. PhD thesis, The University of Arizona.
- [Munk and Cartwright, 1966] Munk, W. H. and Cartwright, D. E. (1966). Tidal spectroscopy and prediction. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 259(1105):533–581.
- [Seigel et al., 1995] Seigel, H. O., Brcic, I., and Mistry, P. (1995). *A guide to high precision land gravimeter surveys*. Scintrex LTD.
- [Torge, 1980] Torge, W. (1980). *Geodesy*. Walter de Gruyter, Berlin, New York.
- [Wenzel, 1996] Wenzel, H. G. (1996). The nanogal software: Earth tide data processing package eterna 3.30. (124):9425–9439.