

GEODYN II

- NASA/Goddard Space Flight Centerが開発。
- 旧GEODYNをデザインし直して1985年以來運用中。
- 衛星軌道予測、衛星軌道決定、測地パラメータ（重力場係数、観測局位置等）推定に多くの実績アリ。
- 「かぐや」の月重力場モデルもGEODYN IIを使って開発された。
- 今回は、衛星軌道予測および衛星軌道決定を体験する。

GEODYN II

- GEODYN IISとGEODYN IIEの二つのプログラムから成る。
- IISは主にファイルの入出力に関する処理(I/O intensive)、IIEは主に計算処理(CPU intensive)を行う。
- IISは、観測値、地球回転パラメータ、天体暦、重力場モデル、solar flux、入力カード等を読み、データをIIEが読める形に整理して中間ファイルに書き出す。
- カードによってコントロールパラメータを指定。カードの集まりのことをinput deckと呼ぶ。GEODYN IIの高い汎用性を反映して、カードの内容は多岐にわたる。

GEODYN II

- カードによって指定できるものの例：
 - 衛星の質量、有効断面積、初期軌道要素。
 - 観測局座標。
 - 重力場モデル、大気抵抗モデルの定義、あるいはその見積もりの要求
 - 極運動、固体潮汐、海洋潮汐、プレート運動等、観測方程式に関連するパラメータの指定あるいはその見積もりの要求。
 - 観測の補正值に関する情報。観測データの取捨選択。
 - 積分時間ステップ、初期軌道要素の初期共分散。
 - シミュレーションデータ作成の要求。
- IIEは、IISが出力した中間ファイルを読み、軌道の時間積分、衛星に働く力の計算、座標変換、観測のモデリング等を行う。

ridasへの接続

- GEODYN IIは国立天文台とNASAとの間の共同研究の枠組みで利用している。持ち出し不可。
- 天文台の計算機(ridas)にログインして使用する。ただし、ridasはファイヤーウォールの内側にあるので、webサーバをゲートウェイとして使う。
- Linux、Macの場合。

ターミナルを一つ開いて

```
% ssh -L 9000:ridas.miz.nao.ac.jp:22 web.miz.nao.ac.jp -l school5
```

このターミナルはログインしたままにしておく。

もう一つ別のターミナルを開いて

```
% ssh -p 9000 -x school5@localhost
```

(パスワードは当日お知らせします)

ridasへの接続

○ Windowsの場合。

- PortForwarderをダウンロード。

<http://toh.fuji-climb.org/pf/download/PortForwarder-2.9.0.zip>

- 解凍する。
- 以下の内容を記述した config.txt というファイルを作成。

Host ridas

HostName web.miz.nao.ac.jp

User school5

LocalForward 9000 ridas.miz.nao.ac.jp:5901

- PortForwarder.exe を起動し、config.txt を Config file として読み込む。
- Connect をクリック。パスワードを入力し、web.miz.nao.ac.jpへ接続。

- 以下のサイトから UltraVNC をダウンロードしてインストール。インストールするのは viewer のみで構わない。
<http://www.uvnc.com/downloads/ultravnc/95-downlaod-10961.html>
- vncviewer.exe を起動する。VNC Server のところに localhost:9000 と入力し、Connect をクリック。school5 のユーザー名でログイン。

(パスワードは当日お知らせします)

ログインしたら

- `/home/school5` がホームディレクトリです。
- ホームディレクトリの下に自分の名前の作業ディレクトリを作ります。
- 例: 小池さんの場合

```
% mkdir koike
```
- 今後の計算は、自分の作業ディレクトリ
(例: `/home/school5/koike`)
の下で行います。
- 他の人の作業ディレクトリに行って間違っ
てファイルを削除したりしないように注意。

例0: 軌道生成

- 月重力場の球状成分のみを考慮。太陽や他の惑星の重力を無視。太陽輻射圧もゼロとする。
- 高度100km、軌道傾斜角 10° のほぼ円軌道。

- ORBITG (orbit generationの意) というディレクトリを作成。

```
% mkdir ORBITG
```

- スクリプトをコピーする。

```
% cp /home/school15/matsumoto/ORBITG/ORBITG_* .
```

- スクリプトを眺めてみる

```
% less ORBITG_0
```

抜粋

```
set IIS      = /home/school5/bin/giis0812p2.exe      GEODYN IISの実行ファイル
set IIE      = /home/school5/bin/giie0812p2.exe      GEODYN IIEの実行ファイル
set EPHEM    = /home/school5/etc/ephem421.data       JPL暦ファイルDE421
set BIH      = /home/school5/etc/gdntable.data       原子時とUT1との差、flux、極運動(GEODYN Table)
set GRAV     = /home/school5/etc/zero.GCOEFCS       重力場係数のファイル(これはゼロ埋めファイル)
set G2T      = /home/school5/bin/g2tinertial_sander  binary軌道ファイルをdumpするプログラム
set OUTCUT   = /home/school5/bin/output             軌道要素の時系列ファイルを整形するプログラム
set PARAM    = 01.ftn05
```

```
#
# -----
#
```

```
if (-e $MainDIR) then
  ¥rm -rf $MainDIR
  echo $MainDIR removed and recreated
endif
mkdir $MainDIR
```

```
cd $MainDIR result_ORBITG_0 というディレクトリを作ってそこへ移動。
```

```
#
# -----
# Create setup file GEODYN II
```

```
cat << PARAMEND1 >! $PARAM GEODYN IIカードの集まり(input deckやsetupと呼ばれる)の書き出し
```


-----< GEODYN II orbit generation run >-----

Gravity Model = SGM100i

#23456789*123456789*123456789*123456789*123456789*123456789*123456789*123456789*

I32G2E 0

EPHEM 200

VLIGHT 299792458.0 299792458.0

VECOPT2 20 0.70000000D+08

90000.0

PLANET 0100 0.00000000000000D+00 2.4390000D+06

水星

PLANET 0200 0.00000000000000D+00 6.0520000D+06

金星

PLANET 0300 0.00000000000000D+00 6.3781370D+06

地球

PLANET02 0301 100100 10.49028011057653D+13 0.1738000D+07

月(301)

PLANET 0400 0.00000000000000D+00 3.3970000D+06

火星

PLANET 0500 0.00000000000000D+00 7.1398000D+07

木星

PLANET 0600 0.00000000000000D+00 6.0411280D+07

土星

PLANET 0700 0.00000000000000D+00 2.5400000D+07

天皇星

PLANET 0800 0.00000000000000D+00 2.4300000D+07

海王星

PLANET 0900 0.00000000000000D+00

冥王星

PLANET 9999 0.00000000000000D+00 6.9600000D+08

太陽

TIDES 20 0.0 0.000D+00

TIDES 21 0.0 0.000D+00

TIDES 30 0.0 0.000D+00

月の潮汐はゼロ

H2LOVE 0.6090

L2LOVE 0.0852

地球の潮汐(観測局の変位に関連)

REL300

PUNCH 1

REFRAC 00011

OBSVU 3

PRNTVU 2 1 1

TERMVU 22

TOLS 100

ENDGLB 11

重力場係数
degree, order
それぞれ100
まで考慮。

のGMと半径。
月以外はGM=0とした。

グローバルセクション:アークに依存しないグローバルパラメータの設定

```

*****
#Lunar satellite arc
*****
REFSYS 1110          080101000000.0 座標系を定義する日時:2008年1月1日00:00:00
SATPAR 7100001 satellite ID 10.400 質量 2765.0 断面積
EPOCH 080101000000.0 0801010000 0.0 080201000000.0 アークのstart & stop time
ELEMS1 20 301 1838000.0 0.000001 10.0 a, e, i (eを0にするとうまく走らない)
ELEMS2 0.0 0.0 0.0 Ω, ω, M
VARCOV 2 010 0 07100001 0.00000000D+00 0.000D+00 0.0D+00 0.0D+00 } 初期軌道要素の
VARCOV 2 011 0 07100001 0.00000000D+00 0.000D+00 } a priori sigmaはゼロ
SOLRAD 0 7100001 0.00000000D+00 0.0D+00 }
DRAG 0 7100001 0.00000000D+00 0.0D+00 } 太陽輻射圧
EDIT 3.5 1.D+50 と大気摩擦はゼロ
STEP 7100001 10.00000000 10秒刻みで時間積分
ORBFIL 30 7100001 080101000000.0 080201000000.0 60.0 軌道ファイル(binary)を1分間隔でUnit30に出力
ORBTUVU0021 080101000000.0 080201000000.0 60.0 軌道ファイル(ascii)を1分間隔で出力
ENDARC
ENDALL

```

1: カーテシアン要素
 2: ケプラー要素

中心天体は月

カーテシアン要素(fort.8)と
 ケプラー要素(fort.10)の両方を出力

iteration回数: 最大値1、最小値1

アークセクション: アークパラメータの設定

それぞれのカードは、数字を書く位置(カラム)が決まっていることに注意。
 例えば、satellite IDは18-24カラムに、軌道要素は21-40、41-60、61-80カラムにそれぞれ書かねばならない。

```
#
# -----
# Run GEODYN II
```

```
ln -s $EPHEM ftn01
ln -s $BIH ftn02
ln -s $GRAV ftn12
cp $PARAM ftn05
```

暦は Unit 1 に
GEODYN Table は Unit 2 に
重力場ファイルは Unit 12 に
Input Deck は Unit 5 に

決まり事

```
( $IIS >! iisout ) >&! iiserr
```

IISを走らせる。

```
mv ftn41 giie.ft12
mv ftn11 giie.ft11
rm -f ftn*
mv giie.ft11 ftn11
mv giie.ft12 ftn12
```

中間ファイルをrename

```
( $IIE >! iieout ) >&! iieerr
```

IIEを走らせる。

```
#
# -----
# Create trajectory file
```

```
$G2T fort.30 trj.xyz
$OUTCUT fort.8 sat.car
$OUTCUT fort.10 sat.kep
```

binary軌道情報をasciiに。
軌道要素の時系列ファイルを整形。

例0: 軌道生成

- スクリプトを走らせる。
% ORBITG_0
- 結果は result_ORBITG_0 にある。
- iiserr、iieerrのファイルサイズが0であれば問題なく計算は終了している。
- sat.carはカーテシアン要素 (& 緯度、経度、高度)
sat.kepはケプラー要素
- gnuplotで時系列を表示して遊んでみる。

```
gnuplot> set xdata time
gnuplot> set timefmt "%y%m%d %H:%M:%S"
gnuplot> set xrange ["080101 00:00:00":"080101 04:00:00"]
gnuplot> plot 'sat.car' us 1:3 w l
gnuplot> plot 'sat.car' us 1:4 w l
```

例1: 軌道生成

- 惑星のGM、月の潮汐、太陽輻射圧を考慮。

グローバルセクション

PLANET	0100	2.20420804937469D+13	2.4390000D+06
PLANET	0200	3.24858707526359D+14	6.0520000D+06
PLANET	0300	3.98600434494882D+14	6.3781370D+06
PLANET02	0301100100	10.49028011057653D+13	0.1738000D+07
PLANET	0400	4.28283579647735D+13	3.3970000D+06
PLANET	0500	1.26712767903558E+17	7.1398000D+07
PLANET	0600	3.79406260748394E+16	6.0411280D+07
PLANET	0700	5.79455912949934E+15	2.5400000D+07
PLANET	0800	6.83653406634824E+15	2.4300000D+07
PLANET	0900	9.83055111718998E+11	
PLANET	9999	1.32712440065915E+20	6.9600000D+08
TIDES	20	0.0255	0.000D+00

アークセクション

SOLRAD	0	7100001	0.12000000D+01	0.0D+00
--------	---	---------	----------------	---------

例1: 軌道生成

- ORBITG_1を走らせる。
- 計算結果は result_ORBITG_1 にある。
- 例0と例1の軌道と比較。

```
% differ_check result_ORBITG_0/trj.xyz result_ORBITG_1/trj.xyz
```

```
differ.out      :ケプラー要素の差
```

```
differ_rac.out  :radial, along-track, cross-track, totalの差。
```

- 1日でどのくらいの位置の違いが出るか？

例2: 軌道生成

- 現実的な月重力場非球状成分を考慮。

```
set GRAV      = /home/school5/etc/SGM100i.GCOEFCS
```

SGM100i:「かぐや」重力場モデルの一つ。

参考文献 Goossens et al, Lunar gravity field determination using SELENE same-beam differential VLBI tracking data, J. Geod., 85, 205-228, doi:10.1007/s00190-010-0430-2

- ORBITG_2 を走らせる。
- 計算結果は result_ORBITG_2 にある。
- iieerrのファイルサイズはゼロか？
- 何が起きたか、iieoutやsat.carを見て確かめよう。
- 軌道の一部をGMTでプロットしてみよう。

/home/school5/matsumoto/gmt/trj.gmt を使ってもよい。

例3: 軌道生成

- 月重力場モデルの10次までの係数のみを考慮。

```
PLANET02  0301100100  10.49028011057653D+13  0.1738000D+07
           ↓ ↓
PLANET02  0301 10 10  10.49028011057653D+13  0.1738000D+07
```

- ORBITG_3 を走らせる。
- 計算結果は result_ORBITG_3 にある。
- 100次までの係数で計算した結果(例2)との違いを見てみよう。
sat.carの比較。

```
% differ_check result_ORBITG_2/trj.xyz result_ORBITG_3/trj.xyz
```

- 重力場係数の低次項の影響について考察しよう。

例4,5: 軌道生成 (Lunar Prospector)

- グローバルセクションはORBITG_2と同じ。アークセクションが異なる。

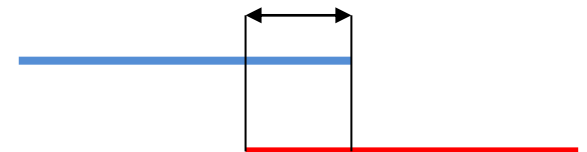
```
REFSYS 1110          980727160000.0000000
SATPAR      9601101          2.00000000          187.000
EPOCH      980727160000.000000 980727160000.000000 980730000000.0000000
ELEMS110 301      0.17223253459113D+070.44445387423103D+06-.49152226904951D+06
ELEMS2      -.41515163813234D+03-.13425265676379D+03-.15671716937558D+04
```

カーテシアン要素を指定。ELEMS1: X, Y, Z
ELEMS2: Vx, Vy, Vz

- 1998/07/27 16:00:00 ~ 1998/07/30 00:00:00 (ORBITG_4)と
1998/07/29 00:00:00 ~ 1998/07/31 00:00:00 (ORBITG_5)。
- fort.10かsat.kepを見て、どのような軌道であるか確かめよう。
(ORBITG_[1-3]で計算したものとは異なる軌道)
- オーバーラップ期間(7/29 00:00:00 ~ 7/30 00:00:00)の軌道の差を見てみる。

```
% differ_check result_ORBITG_4/trj.xyz result_ORBITG_5/trj.xyz
```

オーバーラップ



例6,7: 軌道決定 (Lunar Prospector)

- 実際の観測データを用いて軌道決定を試みる。
- 作業ディレクトリに戻ってORBTD (orbit determinationの意) というディレクトリを作成。

```
% mkdir ORBTD
```

- スクリプト と input deck をコピーする。

```
% cd ORBTD
```

```
% cp /home/school5/matamoto/ORBTD/ORBTD_* .
```

```
% cp /home/school5/matamoto/ORBTD/giis.in* .
```

- スクリプトとinput deckを眺めてみる。orbit generationと異なる部分がある。

スクリプト

set DATA = /home/school5/etc/FORT.40.LP.1998.JULY.ramp 観測データ

ln -s \$DATA ftn40 観測データはUnit40に。決まり事。

input deck

グローバルセクション

OLOAD 海洋潮汐荷重に関連するカード
POLTID 極潮汐に関連するカード
STAPOS - ENDSTA 衛星追跡局に関連するカード

アークセクション

REFSYS1821 9807271600 0.0000000 iteration回数:最大18、最少2
VARCOV 1 10 9601101 0.10000000D+32 0.100D+32 0.1D+320.10D+32
VARCOV 1 11 9601101 0.10000000D+32 0.100D+32
MBIAS 観測バイアスに関連するカード
RESIDU 残差を出力する(fort.19)
DATA 観測データを読む(このカードが無い場合がorbit generation mode)
SELECT 観測の選択
SIGMA 選択した観測の重み
DELETE 観測値の削除

初期軌道要素を拘束しない。GEODYN IIIは、a priori sigmaがゼロでないパラメータを推定しようとする。

例6,7: 軌道決定 (Lunar Prospector)

- オーバーラップ期間 (7/29 00:00:00 ~ 7/30 00:00:00) の軌道の差を見てみる。
- ```
% differ_check result_ORBITD_1/trj.xyz result_ORBITD_2/trj.xyz
```
- 軌道生成 (予測) の場合と比べて、オーバーラップ差は怎么样了か？