UNIT 5. Testing coordinate repeatabilities using *globk/glred*

Ref. GLOBK manual

==========

1. GLOBK OVERVIEW

Globk is a Kalman filter whose purpose is to combine solutions from the processing of geodetic data. The primary input of *globk* is the loosely constrained solution (h-file) from GAMIT processing of GPS data. There are three applications in which *globk* is used:

1. To generate a time series of station coordinates (testing coordinate repeatabilities) to identify and remove any survey or stations which are outliers.
2. To obtain an estimate of station coordinates averaged over a multi-day experiment by combination of individual sessions (e.g., days) of observations.
3. To estimate station velocities from combination of averaged station coordinates obtained from several years of observations

Note that

1. *Globk* assumes a linear model.
2. *Globk* cannot correct any deficiency of the initial loosely constraint solution (h-file)
3. *Globk* can combine your local h-files with SOPAC global h-files. The SOPAC h-files are loosely constrained solution of stations from the IGS network.
4. *Globk* combinations are usually produced with loose constraints, but you can run *glorg* to define a reference frame by applying constraints (stabilization) on the coordinates of a reliable set of IGS stations (stabilization list).

The *globk* analysis of GPS data has the following structure:

1. Convert the experiment ASCII h-files into binary h-files (readable by *globk*) using the program *htoglb*. Run *glred/glorg* for all the (binary) h-files from a survey (or period of continuous observations) to obtain a time series of station coordinates, which can then be plotted and examined for outliers.
2. Remove outliers from the corresponding h-files by renaming the outlier station in an *earthquake file*.
3. Run *globk* to combine daily h-files into a single h-file that represents the averaged station coordinates for the chosen time range (e.g., monthly averages).
4. Run *glred/glorg* and*globk/glorg* again (using the combined h-files) to obtain a time series from *glred/glorg*, and estimates of station velocities from *globk/glorg* for the entire period spanned by your data

2. TESTING COORDINATE REPEATABILITIES

==========

FIRST STEP - Create the *globk* directory structure by adding the following directories to your experiment directory.

expt					
expt/doy	expt/templates	expt/tables	expt/gsoln	expt/glbf	expt/gplot
h-files	command files, the itrf00.apr file, stabilization lists	A priori station coordinate (itrf00.apr)	For running solutions (command files, lists of binary h-files, output files)	Binary h-files	For running plot of coordinate time series

Note that expt/doy and expt/templates are the same directories used in the previous GAMIT processing

==========

==========

SECOND STEP - Copy the SOPAC global h-files (igs1 & igs2 & igs3) in */glbf*

sh_get_hfiles

==========

```
 Script to get hfiles by day number from the scripps archive

 Usage: sh_get_hfiles -yr <yr> -doy <doy> -ndays <num> -net <networks> -soln -
ftp_prog <ftp/ncftp>

        <yr> 4 char year of hfile requested  [Required]
        <doy> 3 char day of year of hfile requested [Required]
        <num> Number of consecutive days of hfiles to retreive [Default 1]
        <networks> List of networks to be retreived from the ftp archive [Default
ALL]
        Network choices: igs1 igs2 bard cors noam dgga eura pgga net1 net2 net3 net4
        -soln Check the solution archives rather than the h-file archives
        <ftp/ncftp> choose the ftp program to be used [default is ftp]

 Examples: sh_get_hfiles -yr 1999 -doy 246
           sh_get_hfiles -yr 1999 -doy 246 -ndays 1 -net igs1 igs2 eura
```

==========

You can also download the global h files directly from the SOPAC web site OR anonymous FTP server (update information available at http://sopac.ucsd.edu/dataArchive/)

==========

==========

THIRD STEP – Prepare the binary h-files in */glbf* using *htoglb*

htoglb converts ASCII h-files into bynary h-files readable by *globk*.
htoglb will generate 4 different binary files:
gcr and gcx (biases free and biases fixed tight constrained solutions)
glr and glx (biases free and biases fixed loose constrained solutions)

The binary file wanted by *globk* is the glx solution.

```
Runstring:

Usage: htoglb [glbf_dir] [ephemeris_file] <GAMIT h-file>
               output       output           input
where
[dir] is the directory for the output files.
[empheris file] Name of the file for output of the ephemeris for the satellites. Can
                then be used as input to GLOBK.
<input files  ... > is a list of input files

Example:
      htoglb    . ../tables/svs_myexp.svs ../008/hnbaya.01008
      htoglb    . ../tables/svs_myexp.svs higs?a.01008
================================================================================
```

==

FOURTH STEP - Copy the *globk* (globk_rep.cmd) and *glorg* (glorg_rep.cmd) command files for repeatabilities from */templates* to */gsoln*

==

==

FIFTH STEP - Edit globk_rep.cmd (directory */gsoln*) -= see GLOBK manual for commands explanation

```
*
* Globk file for repeatabilities using glred with glorg stabilization
*
eq_file ../templates/scec_eq.v2.1
  make_svs ../templates/sat1.apr
  com_file globk_rep.com
  srt_file glb_rep.srt
  sol_file globk_rep.sol

# apr site file(s)
  apr_file ../templates/itrf00.apr

# (1) Max chi**2, (2) Max prefit diff, (3) Max rotation; defaults are 100 10000 10000
  max_chi 50 10 2000.0

* Apply the pole tide whenever not applied in GAMIT
  app_ptid ALL

# Allow the network to be loose since using glorg for stabilization
  apr_neu  all  10 10 10  0 0 0

# Satellites are loose if combining with global SOPAC H-files
#                X       Y      Z    XDOT  YDOT  ZDOT  DRAD YRAD ZRAD BRAD XRAD DCOS
DSIN YCOS YSIN BCOS BSIN
  apr_svs all   100     100    100   10     10    10    1     1  .02  .02  .02 . 02
.02 . 02 . 02  .02 . 02
#  tight if not combining with global data
x  apr_svs all .05 .05 .05 .005 .005 .005 .01 .01 F F F F F F F F F F F F F F

# Keep EOP loose
  apr_wob  100.  100.  10. 10.0  0.0  0.0  0.0  0.0
  apr_ut1  100.  10.  0.0  0.0  0.0  0.0
#  unless not using global data
x  apr_wob  .25 .25  .001 .001 0 0 0 0
x  apr_ut1  .25 .25 .001 .001 0 0
```

```
# Set minimal globk print options since using GLORG output
   prt_opt cmds gdlf

# Invoke glorg for stabilization
  org_cmd glorg_rep.cmd
  org_opt cmds psum gldf
  org_out globk_rep.org
```

==========

==========

SIXTH STEP - Edit glorg_rep.cmd (directory */gsoln*) -= see GLOBK manual for commands explanation

```
*
* Glorg file for repeatabilities
*
# apr site file(s)
# ITRF96 for global stabilization
#    NNR frame
  apr_file ../templates/itrf00.apr
#    Eurasian frame
x  apr_file /data5/sites/itrf96_gps_eura.apr
# Regional stabilization for filtering spatially-correlated errors
x  apr_file ../tables/globk_vel_990306.apr

#  Define the stabilization frame
 source ../templates/stab_site.global
x source ../templates/stab_site.serf

# Set parameters to estimate in stabilization
  pos_org  xrot yrot zrot xtran ytran ztran
x no rotation if regional stabilzation
x  pos_org  xtran ytran ztran

# Set height ratios
x  cnd_hgtv 1000 1000 2.0 2.0
x loosen height tolerance
  cnd_hgtv 1000 1000 2.0 10.0

 # Iterations and editing
  stab_ite 4 0.8 4.
```

==========

==========

SEVENTH STEP - Generate the input global file list (*.gdl) in the directory */gsoln*, see GLOBK 10.0 manual p. 19 for more details

```
A quite straigthforward approach is to use the ls command
Example: ls -1 ../glbf/*.glx > nbay01_glx.gdl

Edit the file eurf_glx.gdl to coupling (with weight 1) the igs global h-files and the
experiment h-files. E.g.:

../glbf/h9702031200_igs1.glx 1.0 +
../glbf/h9702031200_igs2.glx 1.0 +
../glbf/h9702031200_igs3.glx 1.0 +
../glbf/h9702031200_eurf.glx
```

==========

```
================================================================================
EIGTH STEP - Run GLRED

 Usage: glred  crt prt log input_list markov_file
   crt = 6
   prt = globk_rep.prt
   log = globk_rep.log
   input_list = gdl file (e.g., nbay01_glx.gdl)
   markov_file = globk_rep.cmd

 Example: glred 6 globk_rep.prt globk_rep.log eurf99_glx.gdl globk_rep.cmd
================================================================================
```

3. INTERPRETING THE GLRED/GLORG OUTPUT

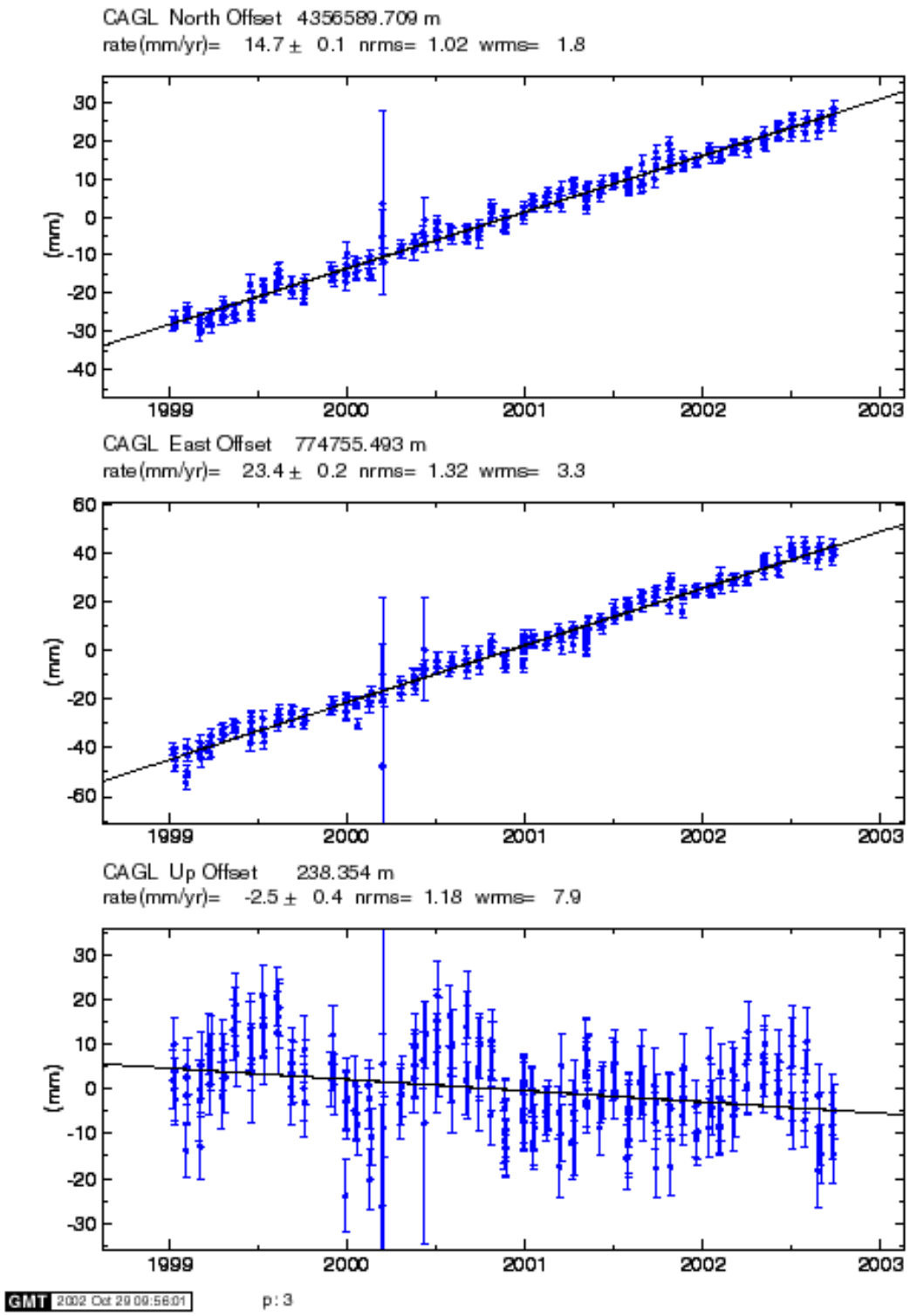
There are three types of output produced in running *glred/glorg*
The "log" file that contains a log of the run as each new h-file is added
The "prt"*globk* solution file. Since the *globk* output is loosely constrained, only th eheight and baseline length components have small enough uncertainties to be useful for careful evaluation.
The "org" *glorg* solution file. This is the tight constrain solution in the reference frame defined by the *glorg* run.

```
================================================================================
Run GLIST (in /gplot) list of stations included in all h-files in a .gdl list

GLIST, Input list, <output_file>, <sort_direction> [eq_file] [out gdl] [apr file]

where Input list is the name of the file containing the list of global
          files to be included in the solution.
      <Output_file> is the optional name of an output file (Default is user's
          terminal.
      <sort_direction> optional value which determines in which order the
          data will be time sorted.  The default is +1 meaning sort in
          ascending time order.  -1 may be specified to have data sorted in
          decending time order.
      [eq_file] GLOBK earthquake file to be used in generating the site names
      [out gdl] Output GDL file sorted in time order according to sort_direction
           (Feature added 991122)
      [apr file] A globk apriori file to used in checking the apriori coordinates
           in the hfiles (Feature addedd 000901)

glist ../gsoln/*****.gdl nbay.list ../templates/**** ../templates/itrf00.apr

================================================================================

================================================================================
Plot coordinate repeatabilities from the GLRED solution (*.org) - directory /gplot

sh_plotcrd
 Use GMT to make a histogram and/or time series plots from glred output.

USAGE: sh_plotcrd -f <files> -s <span> -k <sites>

 OPTIONS:
-f[ile]  : Input glred format .org file/s [Required]
-s[pan]  : (long or short) Long- or short time span plots. [Default: short]
```

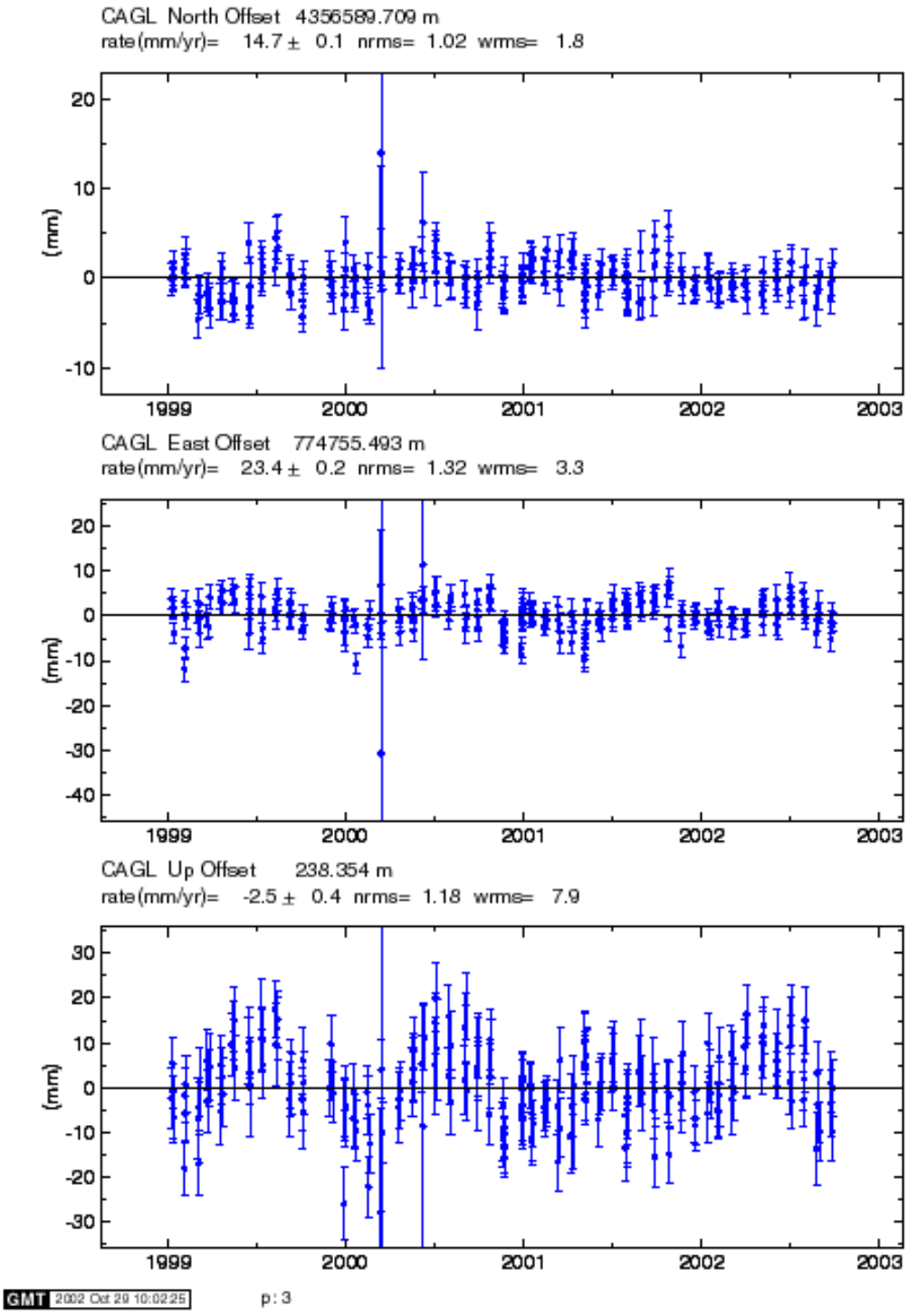
```
-e[xpt]  : Experiment name; optional, used for naming files and labeling plots
-k[ill]  : sites to remove from plotting.
-u[scale] : Scale uncertainties by this value
-r[es]   : Plot timeseries as residuals
-c[ols]  : Number of timeseries columns to plot (1 or 2)[Default: 2]
-o[rder] : (0 or 1) Order of polynonial to fit to data [Default: 0]
-b[ase]  : Name of file containing sites to be used (blank in col 1) [Default: all].
-v[ert]  : Do not estimate vertical rate. [Default: estimate]
-ed[it]  : Flag any N - sigma outliers in the time series. [Default: none]
-maxsigne : Maximum north and east sigma to plot in time series. [Default: all]
-maxsigu  : Maximum up sigma to plot in time series. [Default: all]

EXAMPLES:   sh_plotcrd -f *.org -expt emed00
================================================================================


================================================================================
```

Buy a crystal ball to interpret the results

```
================================================================================
```



sh_plotcrd -f globk_adri_all.org -s long -e adri -k ../templates/ex_sites -c 1 -o 1 -b ../templates/in_sites



sh_plotcrd -f globk_adri_all.org -s long -e adri -k ../templates/ex_sites -r -c 1 -o 1 -b ../templates/in_sites