Massachusetts Institute of Technology

UNAVCO

COMET

GCRF
Global Challenges Research Fund

Basics of processing workflow for GAMIT/GLOBK

M. A. Floyd

Massachusetts Institute of Technology, Cambridge, MA, USA

GPS Data Processing and Analysis with GAMIT/GLOBK and `track`
Addis Ababa University, Ethiopia
24–25 & 27–29 November 2017

http://geoweb.mit.edu/~floyd/courses/gg/201711_AAU/

Material from R. W. King, T. A. Herring, M. A. Floyd (MIT) and S. C. McClusky (now at ANU)

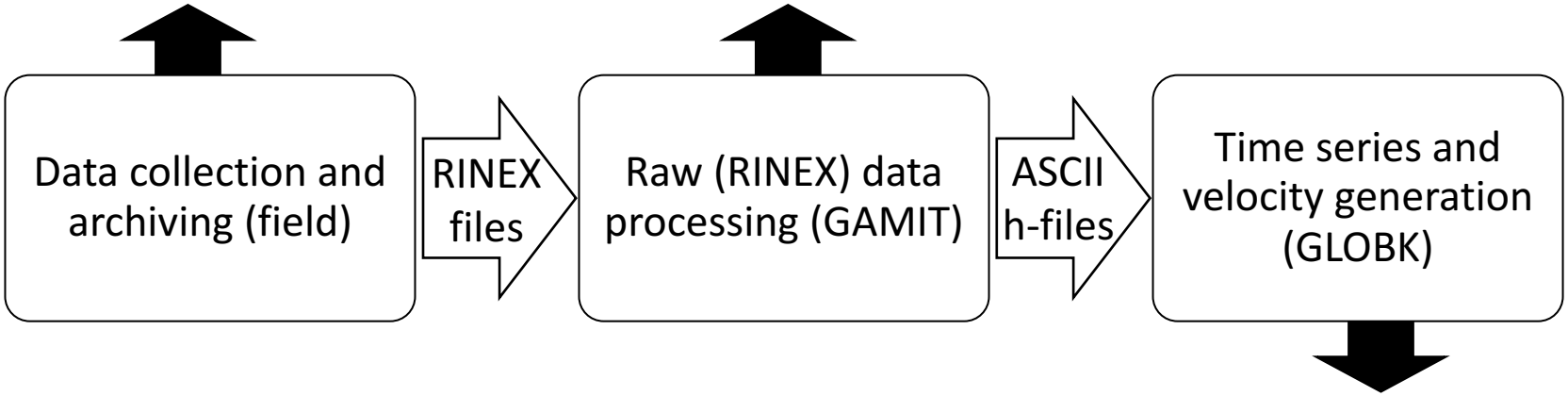
Structure

- The scripts that control GAMIT and GLOBK all have a built-in help page which can be evoked by entering command name only
  - ~/gg/com contains all of the scripts used
  - ~/gg/gamit/bin and ~/gg/kf/bin contain the program executables
  - (gg is a link in your home directory that points to the directory with the GAMIT/GLOBK software installed and should *not* be removed)
- Once the software is installed, user selects data to be processed over some interval of time and uses `sh_gamit` for the processing
- GLOBK is used after the daily processing to combine results and set the reference frame
- Everyone should have completed the installation of the software at this point
- Running the example case is a good idea to make sure installation was successful

Basic stages of GAMIT/GLOBK for geoscience

- `runpkr00`
- `teqc`
- etc.

- `model` (model observations)
- `autcln` (cleans data)
- `solve` (solve for parameters)



- `glred` (time series)
- `globk` (velocities)

Basic inputs and outputs

- RINEX data must be prepared for input to GAMIT
- Output from GAMIT are ASCII (text) “h”-files
  - Loosely constrained solutions with a priori parameter information, parameters adjustments and full covariance matrices
- Input to GLOBK are binary version of h-files
  - Convert from ASCII to binary h-files using `htoglb` (or “`–opt H`” option in `sh_glred`)
- Final output of GLOBK is “.org”-file, which contains all information for
  - Time series (“.pos”-files)

  and/or
  - Velocities (“.vel”-files)

GAMIT

1. Run `sh_setup`
   - Check all links, especially to grid files (otl.grid, atl.grid, map.grid, met.grid; see sestbl. for what is “switched on”)
2. Place RINEX data to be processed in rinex/ directory
   - Except any publicly-available RINEX files one has set to be FTP’d in sites.defaults
3. Prepare *and verify* station.info, e.g. `sh_upd_stnfo`
4. Prepare *and verify* apr-file, e.g. `sh_rx2apr`
5. Run `sh_gamit`

sh_gamit

- `sh_gamit` is the master script for running GAMIT
- The following files are important to verify and/or edit (e.g. after `sh_setup`)
  - autcln.cmd (probably unnecessary to edit)
  - process.defaults (not necessary to edit much, if anything)
  - sestbl. (controls experiment observations and models; defaults OK but may want to edit)
  - sites.defaults (list of sites to process in experiment)
  - sittbl. (controls a priori constraints on sites; probably unnecessary to edit)
  - station.info (*very* important file to get right)
  - .apr-file (*very* important file to get right)
- More detail in following lecture (last lecture this afternoon)

Phase data processing: GAMIT

- Preprocessing
  - Download (`sh_get_orbits`) and prepare (`sh_sp3fit`) orbits
  - Make clock files (`makej`)
  - Download publicly available sites (`sh_get_rinex`) and convert RINEX files to GAMIT internal format for phase-and-pseudorange observations (`makex`)
  - Write batch ("b") files
- Iterative solution (run b-files)
  - Calculate synthetic observations from a priori parameters and models (`model`)
  - Create observables (LC, L1+L2, etc.), clean data (`autcln`)
  - Fit calculated to observed by solving for parameter estimates (`solve`)
  - Update a priori information if large adjustments
- All the above command steps are run for the user by `sh_gamit`
  - Although preparation of orbits (`sh_get_orbits`/`sh_sp3fit`) and RINEX files (`sh_get_rinex`) often done manually, depending on resources

Post-processing: GLOBK

- Convert ASCII h-files to binary h-files (`htoglb` in glbf/)
- Generate and chronological list of binary h-files (`glist` in gsoln/)
- At this point, diverge in approach depending on solution sought
  - More details about `glred`, `globk` and `glorg` in lectures tomorrow
- Similarly to `sh_gamit`, the batch script `sh_glred` will run all of the above command steps (and more, introduced in next slides)
  - User may just need to edit `globk` and/or `glorg` command files to achieve most desired types of solution

GLOBK short-term combinations

- Combine days from a period over which velocities are negligible, e.g. a 10-day survey, bi-weekly or monthly combinations for continuous GNSS
  - Reduces short-term scatter
  - Reduces number of files to be carried forward to velocity solution
- Run `glred` to generate time series
- Plot time series (`sh_plot_pos`)
- Inspect time series to identify (and remove) outliers
- Run `globk` to form one solution file for survey (".org"-file) *without estimating velocities*, e.g. in `globk` command file:

```
apr_site all 10 10 10 0 0 0
```

or

```
apr_neu all 10 10 10 0 0 0
```

GLOBK long-term velocities

- Combine daily (continuous) or short-term combined h-files (e.g. surveys; see last slide)
- Plot long-term time series from short-term combination ".org"-file(s) (`sh_plot_pos`)
- Inspect time series to identify (and remove) outliers
- Run `globk` to form final solution file for all data (another ".org"-file) *with estimating velocities*, e.g. in `globk` command file

```
apr_site all 10 10 10 1 1 1
```

or

```
apr_neu all 10 10 10 1 1 1
```

- `sh_glred` capable of running all these individual commands to produce time series, short-term combinations and long-term velocity solutions