12.010 Computational Methods of Scientific Programming
Fall 2008

12.010 Computational Methods of Scientific Programming

Lecturers

Thomas A Herring

Chris Hill

Review of Last Lecture

- Analysis of the some of the functions needed for the GUI development in Matlab
- Look at functions which would be appropriate for the GPS data analysis GUI
- Looked at how to use graphics objects

Today's Lecture

- This is the last lecture on Matlab
- Analyze a complete GPS time series data analysis program
  - Basic idea is to display position estimates as function of time and to work with these data
  - Operations:
    - De-trending (rate of change and breaks)
    - Editing
- Implementation through a Graphical User Interface (GUI)

Aims and Layout of program

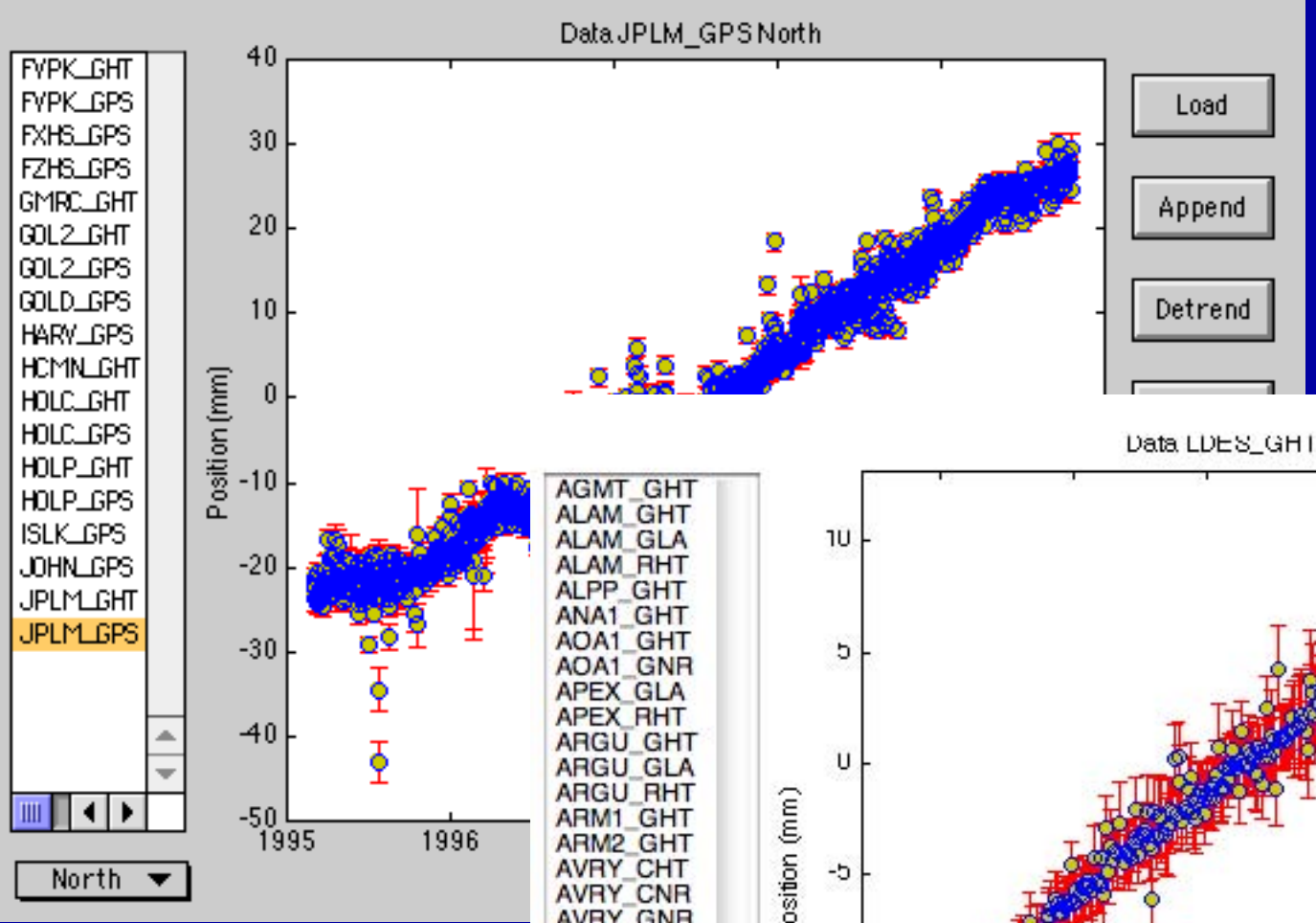
- The GUI layout was designed using the Matlab `guide` tool.
- Basic features of GUI:
  - Time series of GPS positions are stored in files which contain 3 header records and then
    - time (in decimal years),
    - position (north measured as distance from equator, east measured as distance from Greenwich meridian, Up as ellipsoidal height). Values in meters
    - Uncertainty given as a standard deviation
    - Since the positions are large numbers, the leading digits greater than 10 meters are removed from the values in the file.

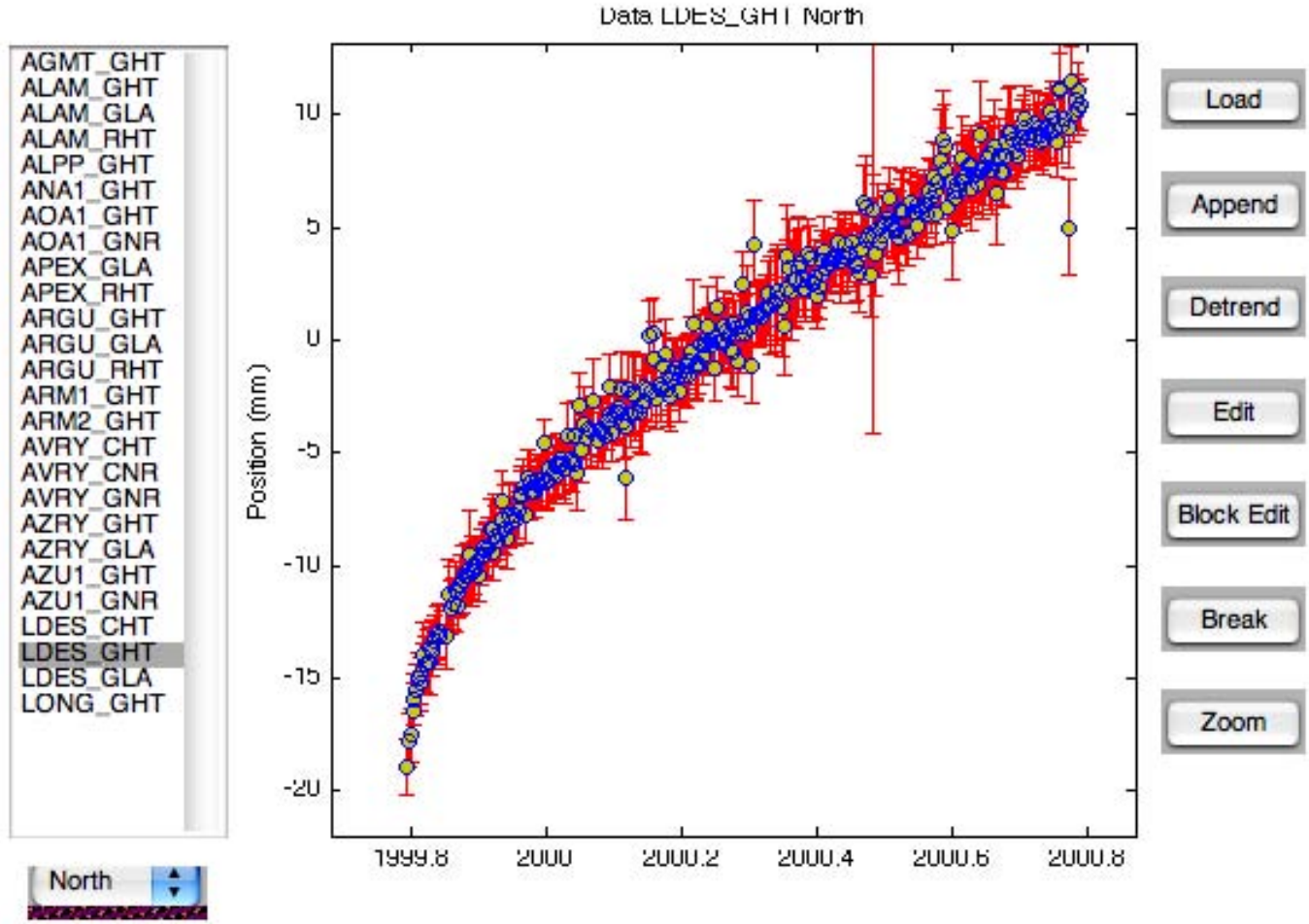
GUI Features

- GUI needs to be able to:
  - Load data file
  - Append another file to the loaded file (needed since the data file are separated into before and after a major earthquake in California: Hector mine Oct 17, 1999)
  - Remove the linear trend from the data and possible jumps in the time series
  - Edit the time series (both as individual points and a selected region of the data)
  - Allows breaks when there appears to be jumps in the time series to be added
  - Allow zoom of the plot so that break times can be set.

GUI Features 02

- Figures below shows the lay out the GUI from different versions of Matlab





GUI Features 03

- In MatLab code that implements the GUI is contained in the M-files, GPSanal.m and gpsgui.m. (These files are in the Matlab section of the web page). GPSanal.mat is also needed to initialize some variables (can be avoided)
- In the lecture, these M-files will be examined in detail.
- The features of the GUI are:
  - Rather than listing file names, the names of the sites are extracted from file names and user selects a site and a component (North, East, Up)
  - The file name is regenerated by re-concatenating the parts of the file name back together.

GUI Features 04

- Nearly all of the callbacks for the GUI use the gpsgui.m function M-file and a series of cases in a `switch` function. (This approach has the advantage on not requiring large number of M-files to implement the GUI.)
- Because functions "forget" there values between calls (unless `global` or `persistence` is used specifically save values), the gpsgui.m function saves the values needed in the `UserData` structure for each of the objects.
- Some of the handles are known when the GUI is called because the handles of the call back objects (`gcbo`) are available, but in other cases we need to find the object using the `findobj` function and a unique Tag added to each object.

Algorithms needed

- The main algorithms needed by this GUI are:
  - Least square fitting: We want to fit a function of the form
    $pos(t) = pos(t_o) + vel*(t-t_o) + Sum(dpos_i)$
    where pos(t) is position at time t,
    t_o is a reference time (we use the mean time)
    vel is velocity
    $dpos_i$ are jumps in the time series at time given by the break times.
  - We can write the above equation in matrix form by making pos(t) a vector and the quantities to be estimated another vector

Algorithms 02

- Defining another vector soln with entries
  soln = [pos(t_o) vel $dpos_1$ $dpos_2$... $dpos_n$]
  where n is the number of breaks
- Matrix form of equation is written as
  pos = A* soln
  where A is made up of rows that look like:
  A = [1 ($t-t_o$) 0 0 … 1]
  where the last group [0 0 … 1] depend on when the time of the measurement is with respect to the times of the breaks. The column value is 1 for all data times after the time of a break

Solution to Least squares problem

- Solution to the above problem is given by
  soln = inv(A'*A)*A'*pos
  (see Detrend case in gpsgui.m)
- We also compute the root-mean-square (RMS) of the fit to the model using
  rms = sqrt(res'*res/numdata)
  where res is the residual vector computed as the difference between the observed positions and the model estimate of the position, and numdata is the number of data.
- The rms of the residuals is used to estimate the standard deviation of the model parameter estimates.

Analysis of GUI

- Run through the gpsgui function, examining the way that the call backs and the objects in the GUI interact.
- The main features to examine are:
  - Use of UserData to save values
  - Use of tags to name and find objects
  - Use of button tool tips
- The sequence of using the GUI is to select a site and component (left hand sided options), Load the data, possibly select a continuation of the data and append it. Detrend and the edit, block edit, break, and zoom features allow detailed analysis of the data.

Data access in GUI's

- Basic approaches are:
  - Use persistent variables. Easiest but there are problems if multiple instances of the GUI are used
  - Store the data (often as a structure) in the User Data part of the handle for a particular object. (This is method used examples)
  - Use the getappdata, setappdata, rmappdata and isappdata to get data, set data (done first), remove data. Generally tag is used to find correct handle.
  - Use guidata to save data in figure handle and retrieve data using data = guidata(gcbo); [graphic call back object). With this approach switchyard programming is not needed.

Final GPS Analysis tool

- On the web page, under Matlab is also the final version of GPS analysis tool called tsview.m
- This tool has the GUI interface re-arranged and enhanced with new button and features.
- This version also explicitly accounts for screen size and has a ResizeFcn to correctly change the GUI when the window is re-sized.

Conclusions

- In general in using Matlab: Use the helpdesk and helpwin to find functions that could be useful
- Remember the basic features of the language in the way that arrays, structures and cells are defined
- Remember the range selection devices
- Remember the control structures
- Remember that functions forget their values (and handles can forget the UserData and Tags as well).
- Matlab is very powerful for quick and easily set up problems for analyses involving matrices.