Functional Requirement for a Deformation Modeling System

Introduction

A framework for a deformation modeling system includes these five basic parts:
(1) Source models
(2) Earth models
(3) Data objects
(4) Visualization and interaction platform
(5) Optimization algorithms

Source models should take as input: their own model parameters, coordinates, and an Earth model. At a minimum, they should return a displacement vector and its spatial derivatives for each coordinate. Ideally, they would also return derivatives with respect to the model parameters for use in optimization. Also, they should in general accept 3D coordinates, though this may not always be available. Source models should be validated to ensure that they satisfy compatibility, compare favorably to one another, etc.

Earth models contain elastic properties (and implicitly topography) as a function of spatial position. An Earth model can be as simple as a single scalar (Poisson's ratio). Earth models are inputs to source models. Mathematically, Earth and source models are not quite separable. However, maintaining these as separate objects will result in several usability enhancements.

Data objects contain observations (and covariance) measured at specific coordinates. Data objects should also contain methods (functions or operators) that map the output of Earth models onto themselves. For example, a data object with InSAR data would include a function that takes the output of a source model and dots it with the look vector. Data objects should also have methods for adjusting the scale of their covariance for varying their weight in comparison to other data objects.

The visualization and interaction platform along with the optimization algorithms will allow the user to manipulate various sets of the previous three objects (models). Questions about simultaneous inversion of multiple data sets, weighting, etc., now become far less about programming challenges than about mathematical analysis, validation, and justification. Moreover, new methodologies, source models, and data sets can plug-in to the existing system with very little (programming) work.

Specific requirements (initial)

- Support for the following data types: GPS, EDM, leveling, LIDAR, InSAR, tilt, strain, and gravity

- Ability to load data in standard formats where available as well as comma delimited ASCII and simple binary formats.
- The ability to load both time series data, and pre-calculate velocities, displacements, etc.
- Support for multiple geographic datums and coordinate systems (e.g., global Cartesian, ellipsoidal, etc.)
- Support for multiple projections (e.g., polyconic, polar, transverse Mercator, etc.) The GUI should support both LLH and Cartesian coordinates.
- For its initial release, the software should include the following deformation model types: shear and opening dislocations, spherical and ellipsoidal point sources, penny-shaped cracks, cylindrical sources, plunging ellipsoids, constant pressure dikes, and user-defined (e.g., FEM-derived) Green's functions.
- The software will be written in Java. The Java Native Interface (JNI) will allow the software to call libraries written in other languages such as C, Fortran, or assembly.
- The GUI will enable the user to interactively define and modify model geometries and locations. This will include the ability to build up complex shapes from simple (e.g., rectangular) constituents.
- Optimization algorithms should allow user defined norms, as well as standard norms such as L1, L2, chi-squared, etc.
- Support for importing and exporting in a wide variety of formats such as ASCII, KML, Arc, GMT, etc. Export capability should be available at every stage of analysis and apply to all calculated or user-defined quantities.
- The software should support a simple scripting language that allows non-interactive use of the software's modeling capabilities.
- The user should have the ability to visualize predicted deformation (vector field) on a user-defined coordinate grid.
- The software should support visualization of observation and prediction on a number of graphs, including map views, radial distance, distance along transects, etc.
- The software should have a basic time series analysis capability, allowing the user to calculate velocities, displacements, de-trended displacements, etc., for all time series data.
- For GPS data, the software should support the modeling of baselines as well as absolute coordinates.
- The software should contain an error analysis capability that allows the user to estimate uncertainties of model parameters and to visualize change in misfit as a function of individual parameters.
- The software will include the following optimization algorithms: non-linear least squares (quasi-Newton), simulated annealing, random cost, grid search, and genetic algorithms.
- The general engineering philosophy behind the software will emphasize extensibility and code growth. The software should be modular and easily added to.

- The software should have basic data editing capabilities so that the user can remove obvious outliers, window particular time intervals, etc.
- Limits on amounts of data, numbers of stations, etc., should be governed only by hardware limitation, i.e., no hard-coded limits.
- The software will be well documented, including a user's manual, a code maintenance and development manual, and scientific publications explaining and justifying the employed methodology.
- The software will not depend on external libraries for its normal operation, nor will it depend on any commercial software.

Specific Requirements (wish list)

- The ability to solve for spatially variable slip (and opening) patterns on dislocations (dikes).
- Temporal modeling capabilities, including Kalman filters, viscoelastic response, etc.

Other plans and goals

- A working group will be formed that will oversee development of proposed software
- A web page will be created that includes code documentation, user's manual, etc., a source repository, a wiki, a discussion forum (or mailing list), and other general information. The Alaska Volcano Observatory will host and maintain this web page, unless the working group prefers another host.
- As the software is being developed, the working group will issue short reports (perhaps quarterly) on progress

Funding opportunities

The USGS members of the working group will actively pursue funding from the Volcano Hazards Program management (and from other parts of the survey). Academic members will investigate other funding sources (NSF, NASA, UNAVCO, etc.)

Working Group Nominees

Falk Amelung, University of Miami
Maurizio Battaglia, USGS/University of Rome
Peter Cervelli, USGS, Alaska Volcano Observatory
Jeff Freymueller, UAF/GI, Alaska Volcano Observatory
Chuck Meertens, UNAVCO
Andrew Newman, Georgia Institute of Technology