FALL 2019 – COS 397 COMPUTER SCIENCE CAPSTONE PROJECT PROPOSALS

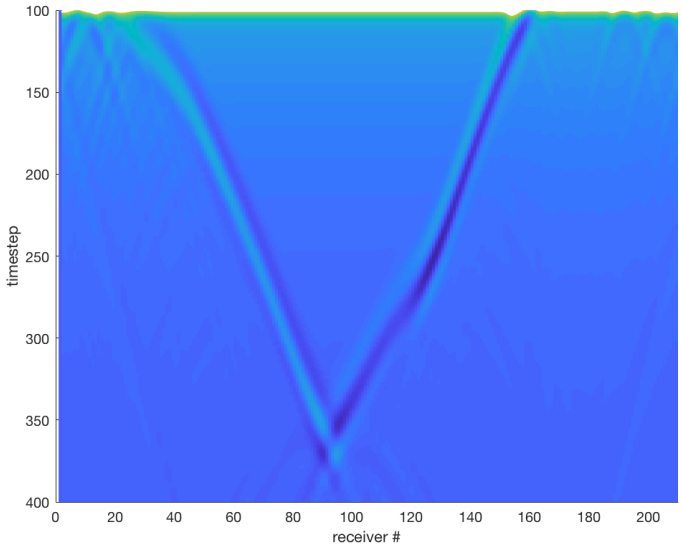
Project Title: Seismic-radar toolbox GUI development

Submitted By: Christopher Gerbi and Seth Campbell, School of Earth and Climate Sciences, christopher.gerbi@maine.edu

Brief Description (approximately 500 words)

Seismic and radar surveys are essential tools for geoscientists exploring a wide range of subsurface phenomena in the natural world. Seismic and radar instruments can, for example, gauge the depth of glacial ice, map out underground aquifers, determine Earth structure, find gas deposits, identify archaeological sites, and locate old graves. Our current work focuses primarily on glaciology and archaeology. For many sites, particularly in Antarctica, field time is expensive, so we can do more and better science by using field time more efficiently. Many geophysical surveys begin with a reconnaissance approach to get a sense of what the instruments can "see" at a particular location. If we could reduce that exploration portion of field time, we would be able to focus more on the scientific questions at hand. To reduce the exploration time, we must plan more effective deployments of sensors and survey patterns. That is where this proposed project comes in.

We have nearly finished building open-source software designed to predict the radar and seismic signals we would get for different field campaign designs. At present, the software is command-line driven, in Python. To make the software most useful and accessible to end users (mainly field-based geophysicists), we propose to build a graphical user interface. This GUI needs to accept and pass simple input parameters to the backend numerical model and provide tools to visualize the results. Visualizations will take the form of animations of wave propagation, signal profiles (see link and image below), and time series x-y plots. The resulting product will be widely distributed within the glaciological and other near-surface geophysics communities.



Example of one type of draft visualization product.

Goals for the project (approximately 50 words)

- Produce a graphical user interface that (1) accepts and passes model input to an already-developed backend computational scheme for radar and seismic wave propagation, and (2) provides tools to visualize model results. We are open to great creative solutions, including data/visualization products and approaches we have not considered.
- Produce documentation and a user guide for the GUI, which can become part of a publication.

Total Duration / Elapsed Time [in weeks]:

This project can scale to the time available. That said, we feel that the scope we propose is of the right size for the two-semester project.

External Schedules / Deadlines [if any]:

Parallel to the GUI development, we will be implementing some refinements in the back-end code, and a glaciology graduate student will be using the code to complete some scientific objectives. We would like to publicly launch the GUI by April, with as much functionality as it has at that point. Additional functionality and documentation can come online as it become available. The grant associated with this work terminates May 31, 2020, though we may be able to consider an extension.

Learning Objectives for student teams:

By undertaking this project, students will:

- Apply their COS skills to a natural science problem, benefitting the students through a deeper understanding of a new field, and benefitting society by providing a tool that improves the efficiency and effectiveness of near-surface geophysical research.
- Develop new skills in GUI design, including stakeholder engagement.
- Develop skills in software documentation.
- Follow the trajectory of and assist with publicly launching a software product.

Expected Project Experiences (select from the list):

Problem definition
Project scope definition
Data analysis (possibly)
Workflow analysis
Development of functional specifications
Examination of an unfamiliar technical area Identification of others' technical expertise Identification and evaluation of alternatives Development and presentation of

recommendations Responsibility and accountability for a
discrete product
Role definition in a task group and participation in group dynamics
Observation of management styles Observation
of organizational politics Preparation of a
manuscript for publication

Recommended experience (What operating system is required? What programming language? Other skills?):

The primary need is in Python, with the ability to deploy on multiple operating systems.

Expected Outputs/Products and likely requirements (specific programming language, operating system, integration with existing software, web-based requirements, *etc.*):

We desire the GUI to be available for multiple platforms (Windows, Mac, Linux), preferably a stand-alone application so that users do not need to install Python and its depenencies on their host machines. The back-end is written primarily in Python, with the heavy computation in Fortran. We do not anticipate the need for significant engagement with Fortran. In addition to a stand-alone application, we would like to explore the possibility of a web-served model.

The documentation should be in the form of a user-guide and tutorial. We are open to an interactive approach, in addition to a traditional item-by-item description.

Past experiences by the client (If software already exists, what is wrong? What has worked in previous versions, and what has not?):

This is new software, developed as part of a National Science Foundation grant. A Ph.D. student wrote the back end, including all the numerical computation. To make this software accessible to the widest array of users, we need a GUI front end as well. Because this is new, we are looking for any creative approach that will provide for the easiest on-ramps and best user experience possible.

Proposed Testing Plan (How will the team test their product? Do you have recommended/required testing strategies? What resources are available (test platform, stand-alone network, *etc.*)? Is test data available?):

The PIs and students involved in the project will be the first testers, after which we will share with colleagues in the field. We have built an international network of people interested in using the software and ready to test it when it becomes available. We anticipate that testing will be in the form of sharing a stand-alone executable file for the testers to run locally on their own computers.

Benefits to U Maine:

Gerbi has been part of two well-received software projects, and along with other School of

Eearth and Climate Science and Climate Change Institute colleagues, UMaine has an international reputation for strong physics-based Earth science numerical applications. This new software would build on that reputation and history in a new area of near-surface geophysics. We anticipate wide use in the glaciological and archaeological communities.

In addition to the direct scientific benefits, the engineering-Earth science-information partnership emphasizes UMaine ability to execute cutting-edge interdisciplinary work.

Project Sponsor(s):

The National Science Foundation funded the toolbox development

Other Resource People:
Ann Hill, graduate student
Steven Bernsen, graduate student

Software/server access required:
The software is currently hosted on GitHub, and easily runs on a laptop or desktop.