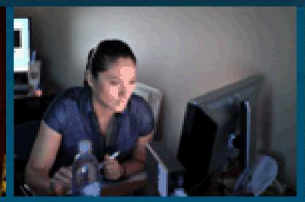
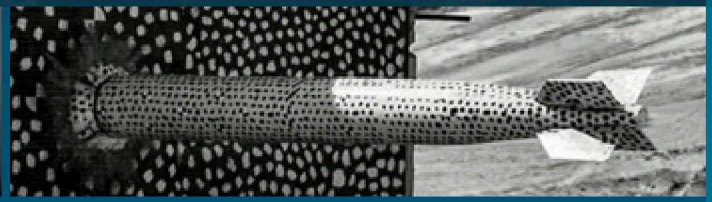
SAND2020-3351PE

Geophysical Monitoring System (GMS)
Generalized F-detector Prototyping





PRESENTED BY

Julia Sakamoto

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2020-????

GMS Generalized F-detector Requirements

Specifications (SSD)

- 3.2.4.1.7 The System shall use Generalized F-detectors to form signal detections for array stations.

Use Cases

- 2.4 System Detects Signals

User Interface Storyboards (UIS)

- N/A

Current USNDC Implementation

- Leidos is working on integrating Neil Selby's Generalized F-detector Fortran code into DFX
  - C callable Fortran module
  - Will be made available to the IDC
- Based on Selby's 2013 paper, which calculates minimum-power (MP) beam and F-statistic in time domain.
- Integration work is essentially complete, but testing and tracking down differences in processing results vs. non-integrated version of Selby code is ongoing.
- This implementation will not be used for GMS.
  - Not written in Java
  - Does not follow GMS architecture

Generalized F-detector Prototype Motivation

- Current implementation would not effectively guide GMS development team that will code Gen-F.
  - Code is well-organized but excessive and extremely difficult to follow (requires expertise both in Gen-F methodology and in Fortran).
  - Not fully consistent with theoretical framework of 2013 paper.
- We are re-coding in Python with improved structure and commenting, increased efficiency, more screen outputs during execution, and accompanying algorithm description documentation
- We will use our prototype to test and tune the algorithm.
  - Our prototype will come with test data sets and processing results to check against.
- Prototype will likely be used by USNDC for continued testing and tuning of Gen-F.
  - Should be significantly easier to work with than the current USNDC implementation (e.g., no need to work with DFX).

Generalized F-detector Prototyping Plan

- Based on Neil Selby's time-domain method (2013) but with critical reworkings of the algorithm to match theoretical framework.
  - Modifications will address issues currently experienced by Leidos (e.g., program crashing, bizarre results for certain arrays).
- Prototyping is being done from scratch in Python using object-oriented programming.
  - Includes ObsPy open-source library for few specialized subroutines (e.g., reading IMS data).
  - References: Selby's publications (2011, 2013) and Fortran code.
- Develop modularly and in same order of execution as the Fortran code.
  - Allows examination and matching of outputs after each subroutine or major data manipulation steps.
  - Make reasonable modifications to existing method (often involves conferring with customer and Leidos).
  - Run and test code on real datasets (e.g., CMAR array) with different tuning parameters.
  - After upcoming divergence from Fortran code, compare/contrast results (e.g., MP beam, F-stats, detections) between algorithms. Continue to confer with Leidos.

Generalized F-detector Prototyping Accomplishments

In-depth study of Neil Selby's papers (ongoing)

- Both frequency-domain (2008, 2011) and time-domain (2013) methodologies

Environment setup to run Fortran code (Aug. 2019)

- Linux-based but required specific, outdated Fortran version to be built from source, plus other system environments.

Python prototyping:

- Classes/methods for importing data and arbitrary input files: IMS (waveform) data, array geometry, beam recipe, tuning parameters, noise model (for initialization), instrument response (Nov. 2019)
- Classes/methods for data handling and quality control (Dec. 2019)
- Module for frequency-time analysis (Jan. 2020)
- Module for MP beamforming (Feb. 2020*)
  - * This is where inconsistencies between Fortran code and theoretical framework were discovered and will be reworked. Productive discussions with Leidos will continue.

Generalized F-detector Prototyping Upcoming Work

Create Python Prototype of Gen-F

- Ready to approach detection phase of algorithm
- Run Selby's test cases and match results with Selby's code to acceptable level of fidelity (Sept. 2020*)

Python Prototype Report

- Purpose of project, comparisons with and modifications made to Selby's code, etc. (Sept. 2020*)

Development of User Guide

- Description of how to install and use software, intended only for AFTAC SMEs (Oct. 2020*)

Delivery to AFTAC

- Code and documentation for version 1.0 to be delivered to AFTAC SMEs (Oct. 2020*)

* Subject to change, dependent on significant reworkings of Selby's code.

Generalized F-detector Prototyping Summary

- We are developing a Generalized F-detector prototype that will be used to guide operational (Java) code development later.
- Prototype will be based on Selby's time-domain methodology (2013), currently implemented in Fortran, but will be significantly easier to understand by developers.
- We are considerably reworking the Fortran code based on inconsistencies with the mathematical framework.
  - Leidos has experienced critical issues as a result of these discrepancies.
- A first version of the prototype is projected to be available in Sept. 2020, contingent on the extent of changes made to Selby's code.