Georgia Tech Earth and Atmospheric Sciences College of Sciences

Geophysics @ Georgia Institute of Technology School of Earth and Atmospheric Sciences

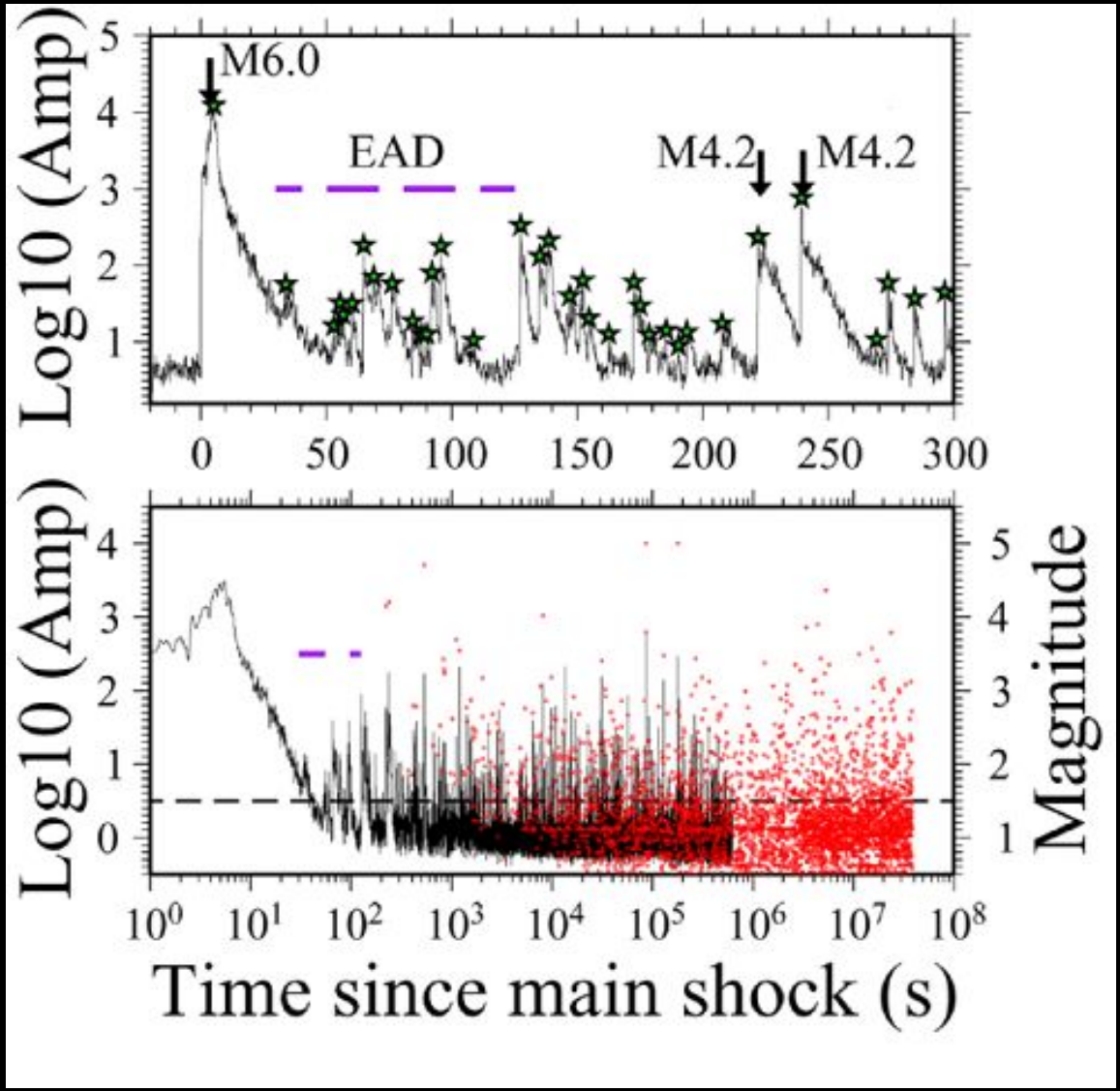
A brief tutorial of the Waveform Matched Filter Technique

Dongdong Yao

02/21/2016

Introduction

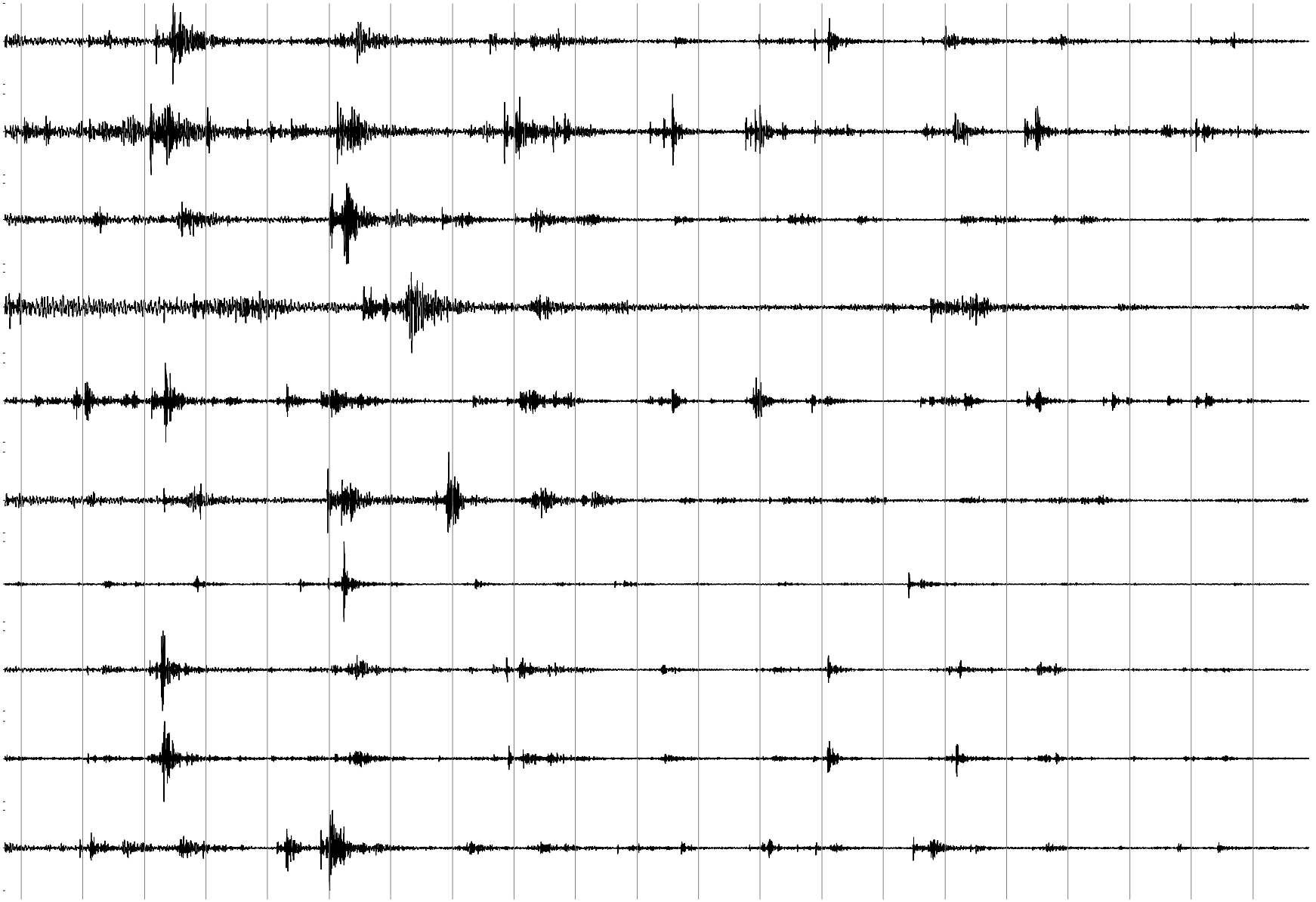
Why do we operate earthquake detection?



Peng et al., 2006

- Hardly clear phase arrivals: Low frequency earthquake
- Buried by larger amplitude coda wave: early aftershocks
- Earthquakes pop out so frequently, overlapping on each other: early aftershocks, earthquake swarms
- Limited by the signal to noise ratio (SNR) of smaller events, hardly picking by either manully or STA/LTA: microearthquakes

An example waveform



Introduction

How does matched filter technique work?

$$u(t)=s(t)\otimes p(t)\otimes i(t)$$

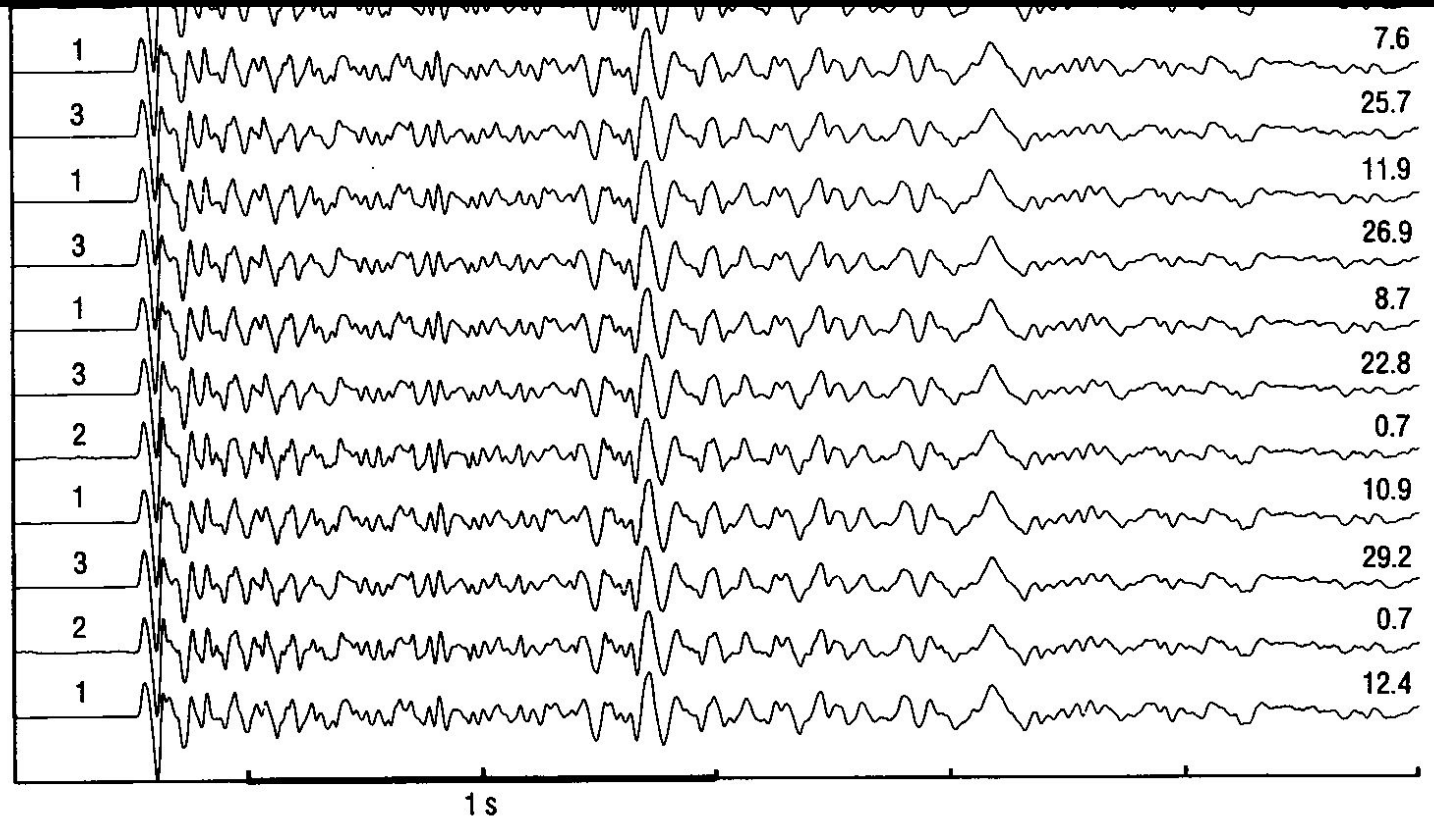
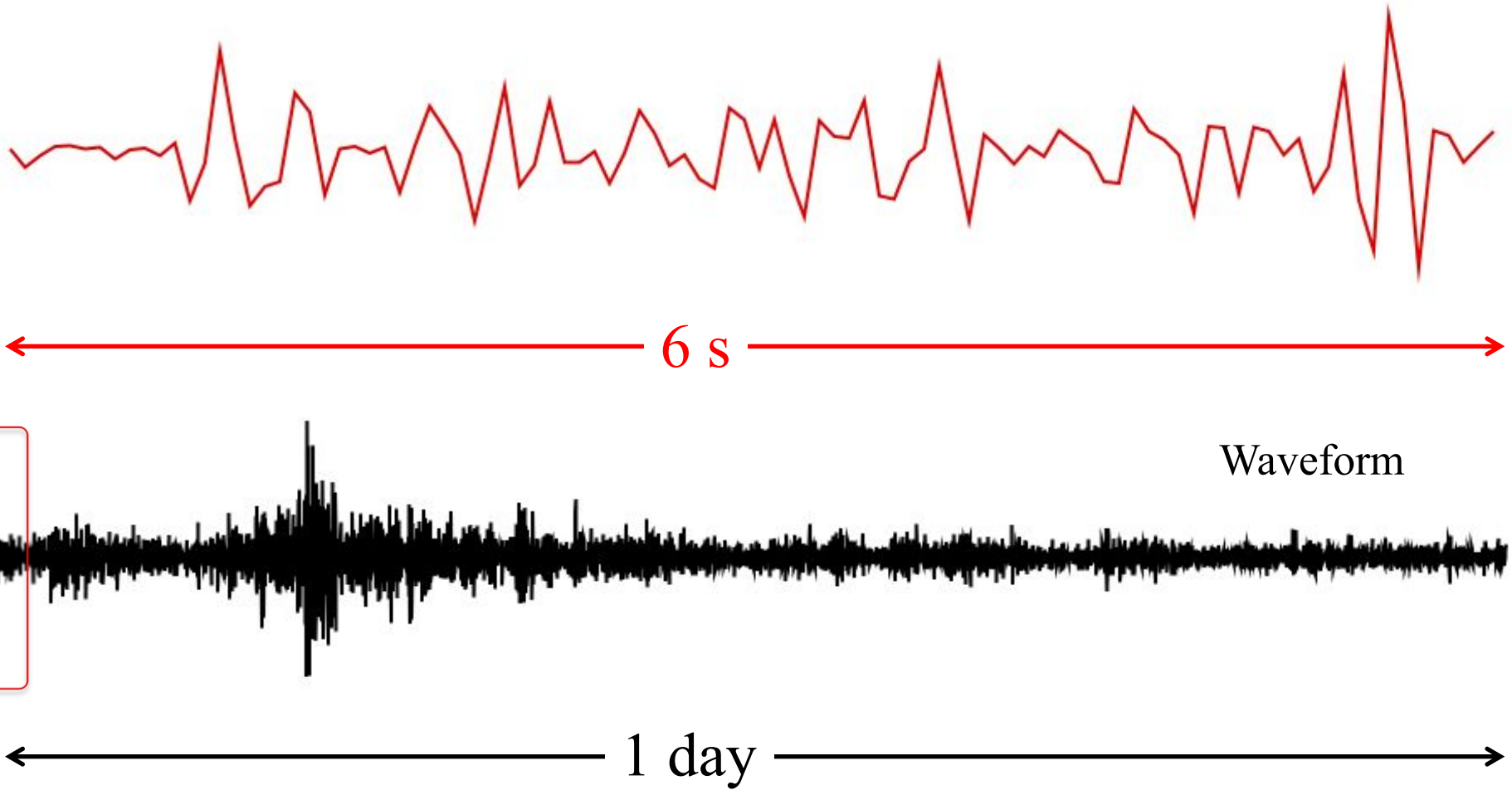
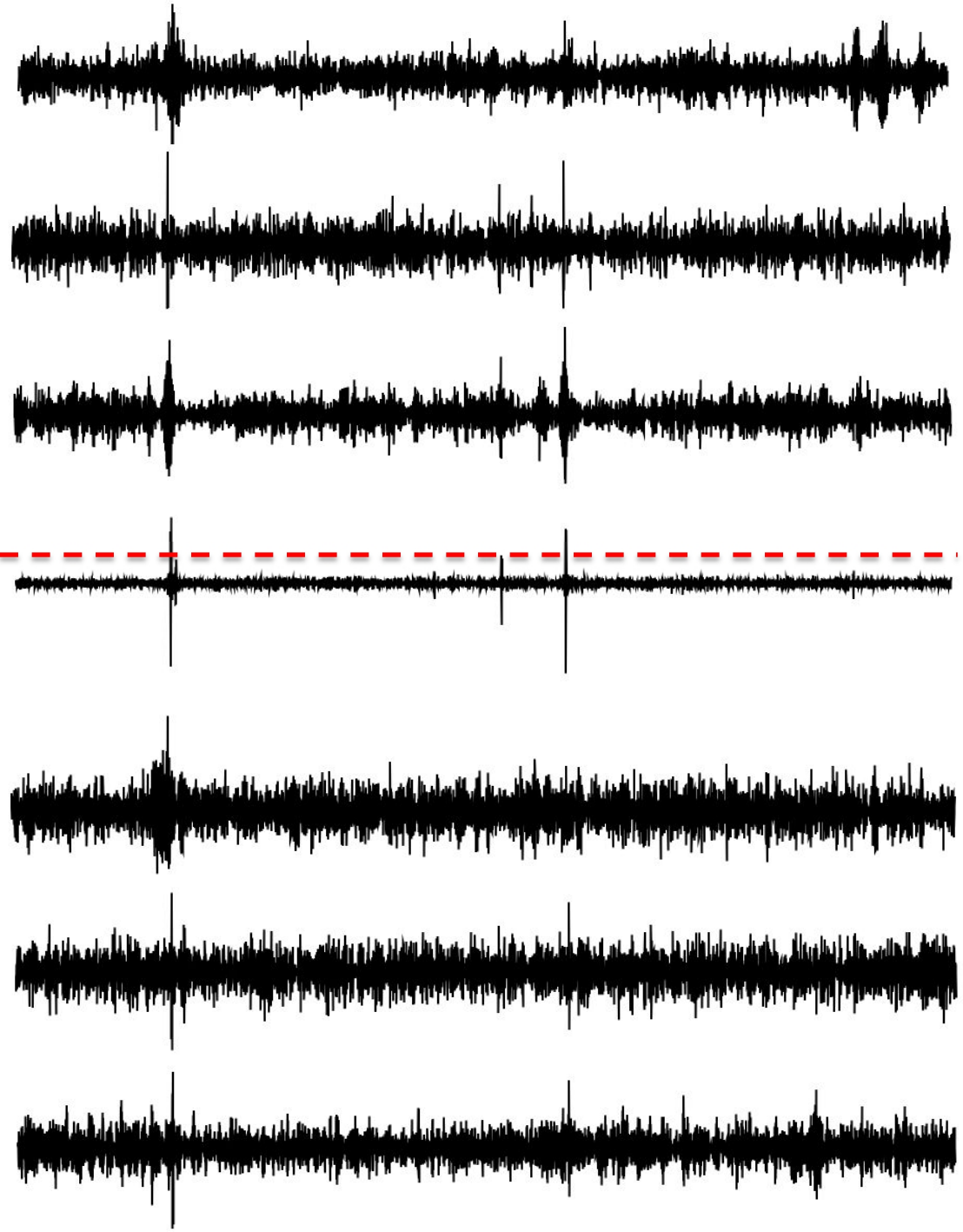


Fig. 1. Vertical-component waveforms recorded at station VCA (see Fig. 3) from cluster CL14, displayed chronologically, most recent on top. Recorded amplitude (counts ÷ 1000) are shown on the right. The cluster subdivides into three types of events on the basis of subtle differences in waveform, as indicated by the numbers on the left.

Nadeau et al., 1995



Sliding window cross-correlation (courtesy X. Meng)



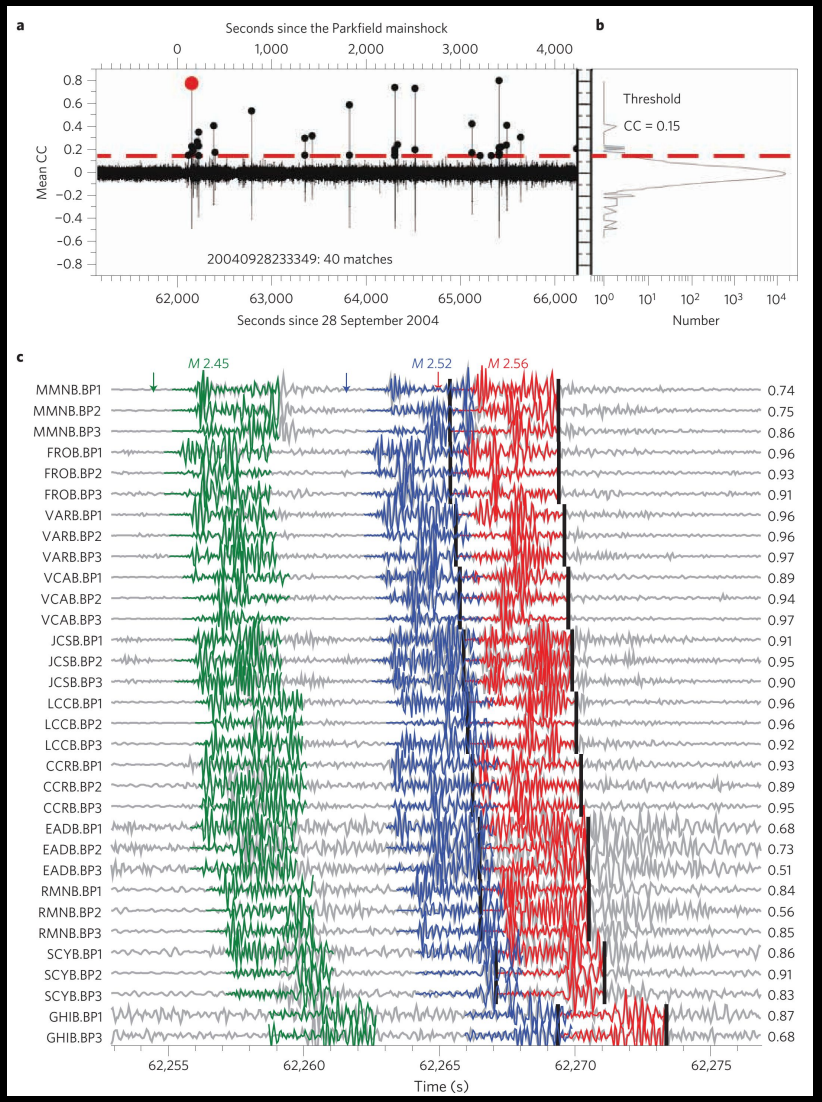
Stacked CC trace

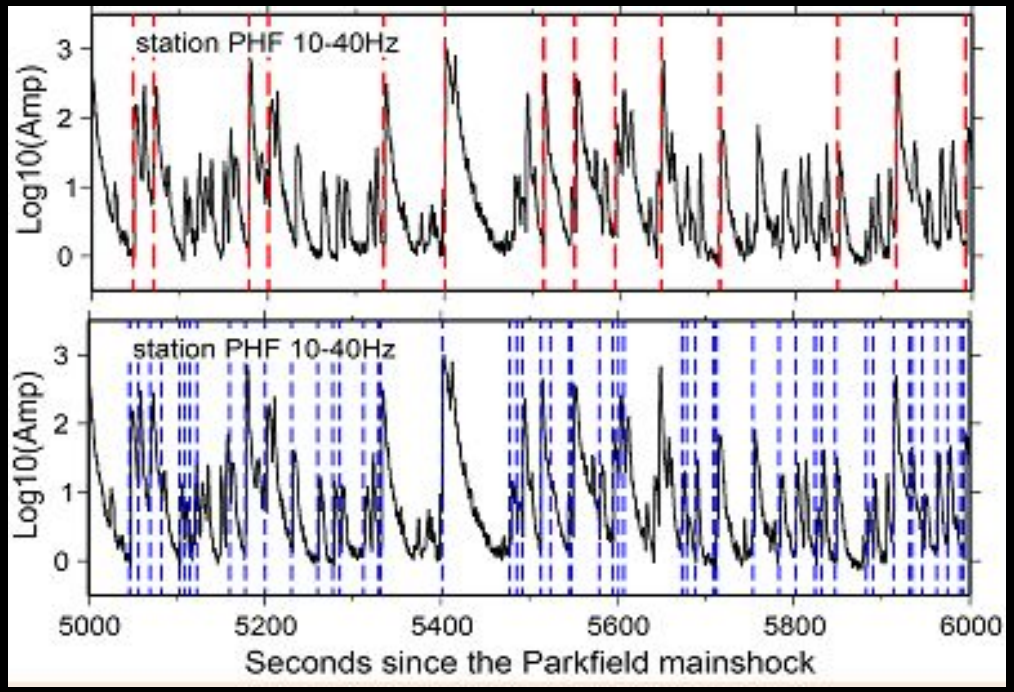
Stacking (courtesy X. Meng)

Introduction

Why we apply matched filter detection?

- Manully Picking: time consuming via visual inspection; limited by the SNR
- STA/LTA: may introduce lots of artificial events, and reply on the SNR

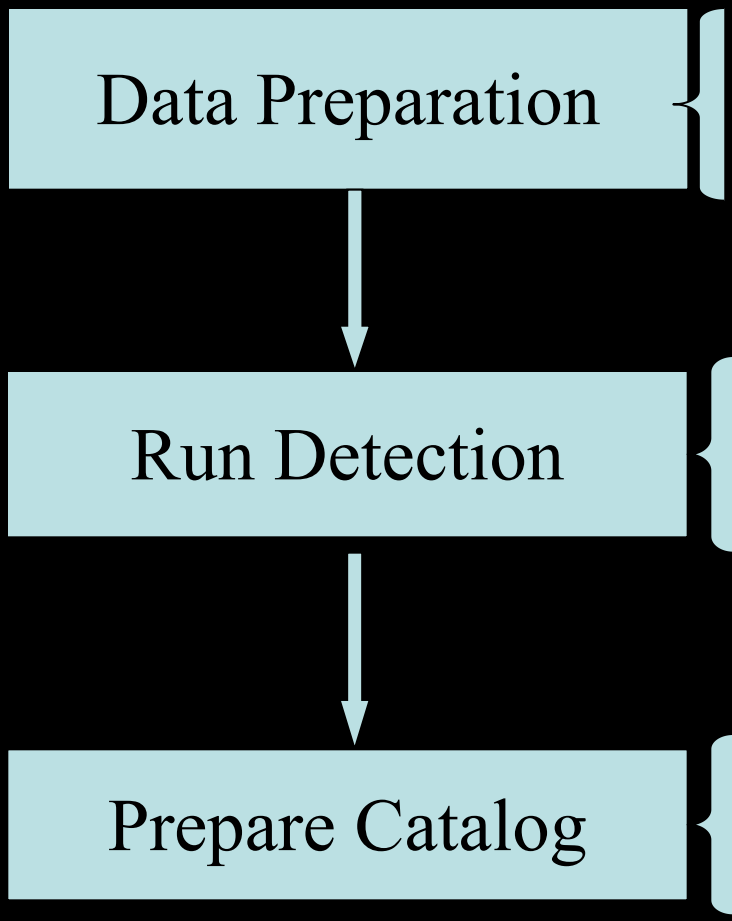




Courtesy Xiaofeng Meng

Peng and Zhao, 2009

Work Flow



Download data: continuous/event waveform
Pre-processing analysis: short/bad traces
Processing data: rename, filtering, arrivals, SNR

Naming Rule: **net.sta.chan.SAC**

Test: run a test before larger job submitted
Sliding-window cross-correlation: sliding_wfcc_fix_v5
Stacking: XmengStackShift

Cut-off Threshold: cc > median CC+MAD*TBD
Duplicate Detection
Compute Magnitude: amplitude ratio

Setup: prepare the directories and necessary commands

Creat a directory CPU_WFCC/ (under the available disk)

- cp /home/dyao/CPU_WFCC . -r

Containing following sub-directories:

ContWaveform/ TempWaveform/ WorkDir/ Catalog/

Required commands in src/

**Utilize your own dataset

[modify the bash before running]: change parameters, naming rules

Data Preparation: ContWaveform/

- cd ContWaveform

  - Common **naming rules** for continuous directories: a file lists all directories, cont.list. Example: beginning time of the data, 20130101000000/
    - ls all_cont_dir -d > cont.list

  - **Headers**: common kztime for all channels and change otime to 0

  Note: All waveforms should have the similar btime and etime, simply removing much shorter traces. *At the same time, all traces should have the same sampling rate delta*

  - check_cont_waveforms.bash cont.list

  if one day's continuous waveform has problem, it would be puto into the temporary file

  - **Band-pass filter**: filter out signals you are interested and compress "noise"
  - bp.bash 20130101
  - make bp (if you want to run the loop for all continuous waveform)

Checklist: cont.list

Data Preparation: TempWaveform/

➢ cd TempWaveform

- Template id named as the origin time of the event, e.g., 20130524054449/

A file lists all template directories: temp.list; template catalog: template.catalog

➢ ls all _temp_dir -d > temp.list

- **Template length**: half mintue before origin time, and contains the S wave
- **Headers**: set the kztime as the origin time of the template event and otime 0

➢ make shift

P and S arrival time t1 and t2: catalog phases, manual picking

- **Band-pass filter**: utilize the same filter as continuous waveform

➢ make bp

- **SNR**: compute the signal to noise ratio for each channel

➢ make SNR

Checklist: temp.list, template.catalog, */wf_SNR.dat, dir_event_SNR.dat

Run Detection: WorkDir/

- cd WorkDir/
  - ***test.bash*** (choose time window and run detection with different tasks)
  - perform sliding window cross-correlation, and stack all channels
  - perform a self-detection test
  - test.bash temp_id (it would loop over all contwfs, as listed in cont.list)

Notes for modifying test.bash

- slide_win (same as sampling rate) [line 59]
- cont_wf_dir [line 66]
- template_base_dir [line 74]
- minimum_wf_cutoff [line 84]
- median_SNR_file [line 91]
- channel_flag [line 145]

Note: replace all the same string in one file: open vi, and type :%s/oldstr/newstr/g

Prepare Catalog: Catalog/

➢ cd Catalog/

- **Combine detections** from all templates:

➢ **see_detection.bash** (two input files: temp.list, cont.list)

- **Remove duplicate detections**: close templates detect the same event with subtle differences in origin time. Only keep the one with highest CC every half-window length.
- **Compute the magnitude** of detected events
- **Generate final catalog**

➢ prepare_catalog.bash cont_id (prepare catalog for every day)

Notes for modifying bash files:

✓ Paths for directories in **amp_ratio_gain_BP.bash**

✓ Need to modify the **gen_best_detection_loop.bash**

See Result: Figure/

- cd Figure

•After getting the final catalog, detected events can be checked by compared the detected waveforms with templates

- waveforms_comparison.bash catalog_file N_row

(two inputs: different catalog file catalog_file, and number of the row N_row which stands for different events)

Note for modifying the bash file:

- **cont_dir**: Line 23
- **template_base_dir**: Line 24
- **chan_flag**: Line 80