SELF-TUNING SEISMIC SENSORS: REAL-TIME TRIGGER LEVEL ADJUSTMENTS

M. Peterson[⋆†], *H. Knox*[⋆], *E. Chael*[⋆], *B. Lawry*[⋆], *K. Phillips-Alonge*[⋆], *A. Faust*[‡], *C. Young*[⋆], *T. Draelos*[⋆]

[⋆] Sandia National Labs
[†] University of New Mexico, Computer Science
[‡] Google Inc.

ABSTRACT

Typical automated processing of time series data from seismic sensors produces many false signal detections, i.e. detections that are not associated with events of interest to the analyst. This is in part because the analyst does not want to miss any detections that are of interest, so they set the sensor detection parameters to be as sensitive as possible, accepting that this will lead to many false detections. This paper presents a model wherein the data processing parameters for each sensor are dynamically changed to achieve an optimal balance between missing signals from events of interest and detecting false signals. They key metric that guides the dynamic tuning is consistency of each sensor with its nearest neighbors: parameters are automatically adjusted on a per station basis to be more or less sensitive to produce consistent agreement of detections in its neighborhood.

***Index Terms*—** self-tuning, seismic, sensor network, adaptive parameters

1. INTRODUCTION

In the field of Seismology, the study of earthquakes and seismic waves, finding valid detections is a difficult task. A single seismic sensor can record thousands of automated detections in a single day where a large percentage of them could be false detections [1], even more so when the goal is to find microseismic events. In most cases an analyst has to verify these automated detections by hand, which can takes days to process.

One of the common ways of detecting events is by using the Signal to Noise Ratio (SNR). The SNR is calculated by taking the Short-Term-Average divided by the Long-Term-Average of the amplitude of the waveform, also known as the STA/LTA [2]. A signal detection occurs when the STA/LTA is greater than some threshold (i.e. trigger level) set by an analyst.

The trigger level is typically an analyst's best guess as to what they think will give them the least missed detections without causing too many false positives. Ordinarily, missing an event is very bad, so the trigger level is set low to error on the side of caution. The downside is that since the trigger level is lower there will be more false detections recorded and hence more detections for the analyst to verify.

In general, the trigger level is the same for all sensors placed in the network and never changes. Sensors can vary in their electronic characteristics, be located in different geophysical structure, and have different coupling to the Earth, supporting the proposition that custom parameters are important to optimal detection. Over a long period of time this can cause issues. Even on a day to day basis the background noise can vary. In a place like Antarctica, the ice shelfs tend to cause more ice quakes when the sun is up, and tend to be more dormant during the evenings. Even weather could potentially affect the noise levels.

All of these examples affects the level of the background noise which changes STA/LTA values. This begs for a non-static threshold approach. An approach where the trigger level can change based on an individual station's background noise.

The method explained in this paper reduces the amount of false detections while maintaining valid detections. This is done by allowing the seismic sensor network to share signal information across sensors. This approach allows the trigger level to change over time based on what the sensor sees compared to other sensors in the network. We are assuming that sensors which are near each other should be detecting the same events within a given time window, and therefore if a sensor is does not behave similarly to other nearby sensors, we assume its trigger level is not set properly and needs to be adjusted accordingly.

2. ALGORITHM

MajorityRules is a voting scheme where we assume the majority is correct, and hence we need to adjust the minority's parameters to better match what the majority said happened. The goal is to have consensus among the sensors across a neighborhood, as explained below. All sensors in a given neighborhood should be in agreement about the same events within a given time window; as well as being in agreement when there is no event.

2.1. Neighborhoods

Each sensor has its own individual list of neighbors which it uses to validate its own detection. These neighborhood lists can be defined however the user feels fit; one example would be distance based neighborhoods, where all neighbors in the list are a maximum distance from the sensor.

One perk of having a list of neighbors is that we can filter out detections that we deem as not important. An example of this is for locating event origins [3]. If we do not have three or more sensors detecting an event, then we cannot triangulate the origin of the event.

2.2. Majority vs Minority

In a given time window, each sensor will report whether they detected a signal or not. Each sensor consults its list of neighbors and determines if it agrees with its neighbors or not. Based on the consensus of its neighbors the parameters of the sensor are adjusted accordingly:

In Minority & Detected	make sensor **more** sensitive
In Minority & No Detection	make sensor **less** sensitive
In Majority	make sensor **slightly more** sensitive

In the cases where the sensor is in disagreement with the majority of its neighboring sensors, making the sensor more sensitive will help ensure that a detection wont be missed next time and making it less sensitive will help reduce the chance of detecting a false event.

3. RESULTS

The results in this paper are from adjusting a single parameter of the seismic sensor; the *Trigger Level*. This is what we deemed as the most direct way to control a sensor's sensitivity.

3.1. False vs Missed

Figure 1 is a graph of the percent of false events versus the percent of missed events from different starting trigger levels. The experiments are ran over a twelve hour period where all stations start out at the same trigger level. In the static version the trigger levels stay the same throughout the twelve hour period. In the Dynamic version the trigger levels are allowed to change over time using the MajorityRules algorithm described earlier.

The main point to be taken from this graph is that the MajorityRules results converge to a small cluster of points closer to the origin than the static method. This means that there is less pressure on the analyst to guess the best trigger level for all the stations since the algorithm will figure it out for them.

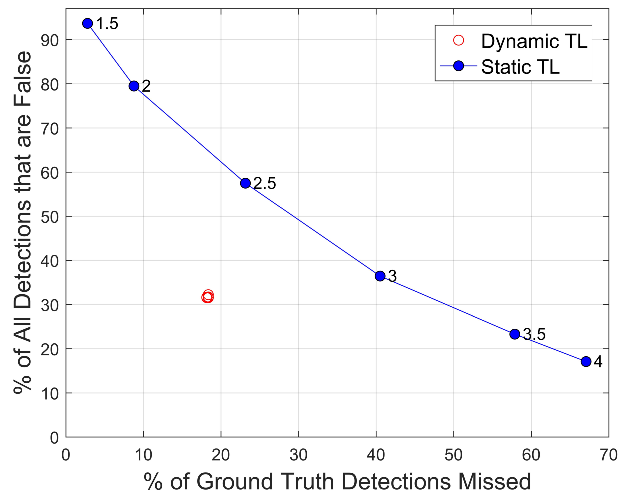

Fig. 1. The figure above shows the percent of false events versus the percent of missed events a twelve hour period from the Mt. Erebus data set [4]. The numbers on the points of the graphs are the starting Trigger Levels for the experiments.

4. FUTURE WORK

There are a few things we would like to incorporate into our algorithm. 1) Add the ability to adjust multiple parameters at a time. 2) Generate neighborhoods list dynamically. 3) Add uncertainty quantification to detection to better determine majority.

5. REFERENCES

[1] Reinoud Sleeman and Torild van Eck, "Robust automatic p-phase picking: an on-line implementation in the analysis of broadband seismogram recordings," *Physics of the Earth and Planetary Interiors*, vol. 113, no. 14, pp. 265 – 275, 1999.

[2] Ismael Vera Rodriguez, "Automatic time-picking of microseismic data combining sta/lta and the stationary discrete wavelet transform," in *CSPG CSEG CWLS Convention, convention abstracts*, 2011.

[3] Johannes Schweitzer, "Hyposatan enhanced routine to locate seismic events," in *Monitoring the Comprehensive Nuclear-Test-Ban Treaty: Sourse Location*, pp. 277–289. Springer, 2001.

[4] Hunter A Knox, *Eruptive characteristics and glacial earthquake investigation on Erebus volcano, Antarctica*, New Mexico Institute of Mining and Technology, 2012.