InsightTomo: In-Situ Seismic Tomographic Imaging System

Lei Shi, Wen-Zhan Song
Georgia State University
lshi1@student.gsu.edu,
wsong@gsu.edu

Yao Xie, Zhigang Peng
Georgia Institute of Technology
yao.xie@isye.gatech.edu,
zpeng@gatech.edu

Jonathan M. Lees
University of North Carolina at Chapel Hill
jonathan.lees@unc.edu

Abstract

In both industry and academic, the seismic exploration does not yet have the capability of illuminating the physical dynamics with high resolution and in real-time, as it involves collecting the raw seismic data from sensors to data loggers then manually retrieving data for post processing which may take months to complete. This study presents InsightTomo, a In-Situ Seismic Tomographic Imaging System based on sensor network technology and travel-time tomography principles. It is a large-scale sensor network system design that senses and processes seismic signals, and compute 3D tomography in-situ in real-time. The proposed InsightTomo system consists a distributed approach for tomography computation, collaborative signal processing and the associated sensor network design. We implemented InsightTomo and demonstrated that it can illuminate the 3D subsurface image of Parkfield in-situ in real-time. The design of InsightTomo in this paper is general, and can be implemented as a new field network paradigm for real-time imaging of highly dynamic and complex environments.

Categories and Subject Descriptors

C.2.4 [**COMPUTER-COMMUNICATION NETWORKS**]: Distributed Systems

General Terms

Algorithms, Design, Measurement, Performance

Keywords

Distributed Computing, In-network Processing, Seismic Tomography, Sensor Networks

1 Introduction

Volcanic eruption is one of the most dangerous threats to life on the Earth. In recent years, more volcano activities have drawn the attention of public and scientists. Most existing volcano monitoring systems employ expensive broadband seismometer as instrumentation. Also at present raw seismic data are typically collected at central observatories for post processing. Seismic sampling rates for volcano monitoring are usually in the range of 16-24 bit at 50-200Hz. With such high-fidelity sampling, it is virtually impossible to collect raw, real-time data from a large-scale dense sensor network, due to severe limitations of energy and bandwidth at current, battery-powered sensor nodes. As a result, at some most threatening, active volcanoes, fewer than 20 nodes [24] are thus maintained. With such a small network and post processing mechanism, existing system do not yet have the capability to recover physical dynamics with sufficient resolution in real-time. This limits our ability to understand volcano dynamics and physical processes inside volcano conduit systems. Substantial scientific discoveries on the geology and physics of active volcanism would be imminent if the seismic tomography inversion could be in real-time and the resolution could be increased by an order of magnitude or more. This requires a large-scale network with automatic in-network processing and computation capability.

To date, the sensor network technology has matured to the point where it is possible to deploy and maintain a large-scale network for volcano monitoring and utilize the computing power of each node for signal processing and distributed tomography inversion in real-time. The methods commonly used today in the procedure of seismic tomography computation cannot be directly employed under field circumstances proposed here because they rely on centralized algorithms and require massive amounts of raw seismic data collected on a central processing unit. Thus, real-time seismic tomography of high resolution requires a new mechanism with respect to system design, information processing and tomography inversion computation. To clearly address the challenges in this paper, we give a short description on the background knowledge of the first-arrival traveltime tomography.

The first-arrival traveltime tomography uses P-wave first arrival times at sensor nodes to derive the internal velocity structure of the subsurface. The basic workflow of traveltime tomography illustrated in Figure 1 involves four steps.

(a) P-wave Arrival Time Picking. Once an earthquake event happens, the sensor nodes that detect seismic disturbances record the signals. The P-wave arrival times need to be extracted from the raw seismic data.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SenSys'14, November 3–6, 2014, Memphis, TN.
Copyright © 2014 ACM 978-1-4503-1169-4 ...$10.00

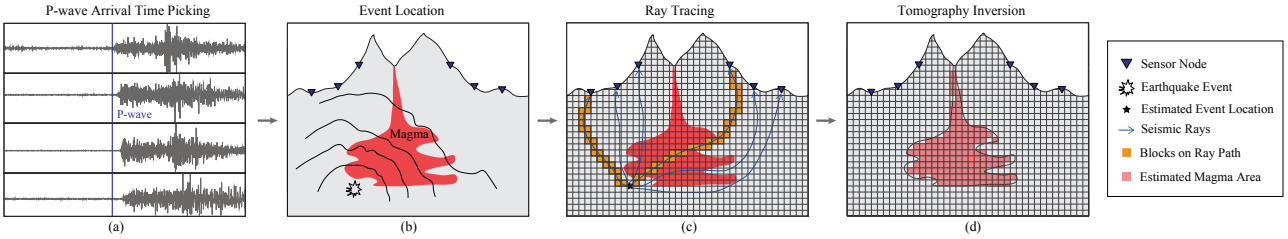


Figure 1: Workflow of first-arrival traveltime tomography.

(b) Event Location. The P-wave arrival times and locations of sensor nodes are used to estimate the event hypocenter and origin time in the volcanic edifice.

(c) Ray Tracing. Following each event, seismic rays propagate to nodes and pass through anomalous media. These rays are perturbed and thus register anomalous residuals. Given the source locations of the seismic events and current velocity model, ray tracing is to find the ray paths from the event hypocenters to the nodes.

(d) Tomography Inversion. The traced ray paths, in turn, are used to image a 3D tomography model of the velocity structure. As shown in Figure 1 the volcano is partitioned into small blocks and the seismic tomography problem can be formulated as a large, sparse matrix inversion problem.

In traditional seismology, the raw seismic data is collected for manual analysis including P-wave arrival time picking on the seismograms. Then centralized methods will process the data and compute seismic tomography. Keep in mind that our goal is to design a system which can deliver 3D tomography in real-time over a large-scale senor network by utilizing the limited communication ability in the network and the computation power on the sensor node. To reach this goal, first no raw seismic data should be transmitted over the network, which requires a light weighted algorithm that can accurately pick the P-wave arrival time on the sensor nodes locally inside the network. Second, an efficient distributed tomography computation method is needed for processing data and inverting volcano tomography in the network while avoiding both costly data collections and centralized computations.

This paper presents InsightTomo - a in-situ seismic tomographic imaging system framework in sensor network. The design of InsightTomo consists of a series of algorithms and network design to automatically process the seismic data, pick the P-wave arrival time, identify seismic events and compute the seismic tomography in-situ in real-time. The system design is evaluated with real data from the San Andreas Fault (SAF) on Parkfield and the results demonstrates the feasibility to implement and deploy the system on the future volcano monitoring sensor networks.

The rest of the paper is organized as follows. Section 2 shows an overview of the system design. Section 3 discusses the details of the system design and implementation. Section 4 gives extensive evaluations on the system with real data set. Section 5 reviews the related work and section 6 concludes the paper.

2 System Overview

2.1 System Model

The mesh network architecture is employed in the design of InsightTomo. Each sensor node in the network is equipped with a seismic sensor (e.g., single or three component geophone) that continuously samples and records the signal in an external storage (e.g., SD memory card). Also, the sensor node has a low power MCU (e.g., MSP-430 or Imote series) that has limited computation resources but keeps a very low power profile. Besides, all the sensor nodes have GPS modules on board [24] such that the clocks on sensor nodes are time-synchronized so are the time pickings. A powerful computation unit (e.g., BeagleBone Black board, cellphone or tablet) is also installed on each node; the unit can complete the computation-intensive tasks including the event location and tomography inversion.

2.2 System Architecture

In this section, we will give an overview of the system architecture and the data flow of InsightTomo respect to the design requirements mentioned above. InsightTomo consists of several algorithms running on sensor nodes and coordinator nodes, and a bundle layer protocol is proposed to improve the performance of communication. Figure 2 illustrates the architecture and dataflow of InsightTomo system. There are three steps in the architecture corresponding to the workflow of seismic tomography.

1. Sensor node receives the signal from the sensor and analyze the signal with P-wave arrival time picking algorithm. The algorithm will continuously monitor the samplings and alarm if an event is detected, then the picking algorithm will pick the P-wave arrival time. After the arrival time picking done, the sensor node only needs to send the arrival time to coordinator node. Refer to section 3.1 for more details.
2. The coordinator (could be any sensor node) receives the P-wave arrival times from many sensor nodes that detected events. The only information received by the coordinator node in this step is a series of arrival time pickings, based on this, the coordinator first needs to identify which arrival times are corresponding to the same event. The reason is that not all the sensor nodes can detect one specific event due to the event intensity, event position, sensor instability and so on. After the event is identified, the coordinator node can compute

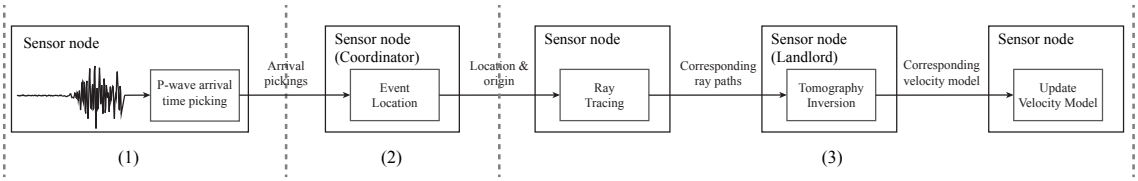


Figure 2: The architecture of InsightTomo system.

event location and origin time, then send the specific event location and origin back to the corresponding sensor nodes which detected it. Refer to section 3.2 for more details.

3. This step consists of several in-network computation and communication tasks. We put them together since they are all parts of the distributed tomography computation mechanism. Once the sensor node receives one event location from the coordinator node, it can trace the ray path from the event location to itself. Based on our distributed tomography algorithm, the sensor node sends ray paths to corresponding landlord node which will be in charge of the tomography computation. After the computation done, each landlord will broadcast the partial tomography model to the network so that each sensor node can update its model for future ray tracing and computation. Refer to section 3.3 for more details.

3 System Design

In this section, we study the design of InsightTomo step by step following the discussion above. Note that this design is based on careful analysis of the real data from previous Parkfield deployment.

3.1 P-wave arrival time picking

Primary waves (P-waves) are the seismic waves that travel faster than any other waves through the earth. P-waves arrive at the seismic sensors first and the arrival time of P-waves are essential to the first-arrival traveltime tomography. Figure 3 shows the seismograms from four seismometers deployed in Parkfield when an event happens. The vertical lines represent manual pickings of the P-wave arrival times. Due to the different wave propagation delays, the P-wave arrival times on sensors are different. In local seismic tomography, the scale of the field is up to tens of kilometers and the maximum difference of the P-wave arrival times among sensors is about several seconds, so that the accuracy of the picking is significant. Besides, manual analysis of seismograms and picking of arrival times require post processing of the data and are very time consuming, especially in a large sensor network. To avoid raw seismic data transmission and meet the real-time requirements, InsightTomo demands an on-line automatic event detection and P-wave arrival time picking method that runs on each sensor node.

From the seismograms in Figure 3, one can see that there is a big difference on the amplitude of the signal before and after the arrival of P-waves, the P-wave arrival time is a change point of the variance of the signal amplitude. Based on this observation, a method is proposed in this paper by utilizing the maximum-likelihood (ML) estimation to estimate

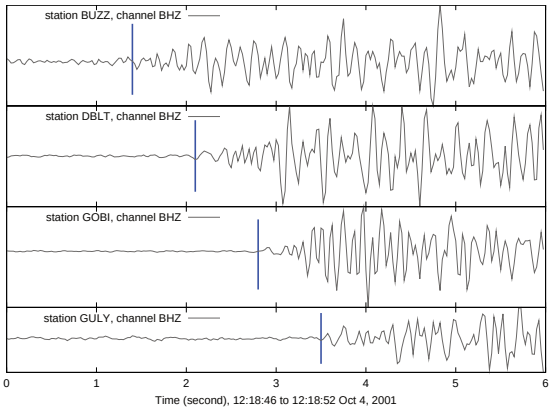


Figure 3: The seismogram from BHZ channel of four seismometers in Parkfield when an event happens. The vertical lines indicate the manual pickings of P-wave arrival times.

the variance of the signal amplitude following a statistical model. In the following discussion, we use pre- and post-change to describe the signals before and after the P-wave arrival.

Without loss of generality, we assume that both the pre- and post-change signals follow a normal distribution but with different variances. Let $\{x_i\}_{i=1}^{t}$ be the continuous sequence of samples from 1 to t, then the pre- and post-change sample has a normal distribution with zero mean respectively,

$$\text{pre-change sample,} \quad x_i \sim \mathcal{N}(0, \sigma_1^2) \tag{1}$$

$$\text{post-change sample,} \quad x_i \sim \mathcal{N}(0, \sigma_2^2) \tag{2}$$

The logarithm of the likelihood function at time k is,

$$\mathcal{L} = \sum_{i=k+1}^{t} \ln \frac{f_2(x_i)}{f_1(x_i)} \tag{3}$$

where $f_1(x_i)$ and $f_2(x_i)$ are the probability density functions (pdf) of the pre- and post-change signals, the likelihood function then can be rewritten as,

$$\mathcal{L} = \sum_{i=k+1}^{t} \ln \frac{\frac{1}{\sqrt{2\pi\sigma_2^2}} e^{-\frac{x_i^2}{2\sigma_2^2}}}{\frac{1}{\sqrt{2\pi\sigma_1^2}} e^{-\frac{x_i^2}{2\sigma_1^2}}} \tag{4}$$

$$= \sum_{i=k+1}^{t} \left[\frac{1}{2} \ln \frac{\sigma_1^2}{\sigma_2^2} - \frac{x_i^2}{2} \left(\frac{1}{\sigma_2^2} - \frac{1}{\sigma_1^2} \right) \right] \tag{5}$$

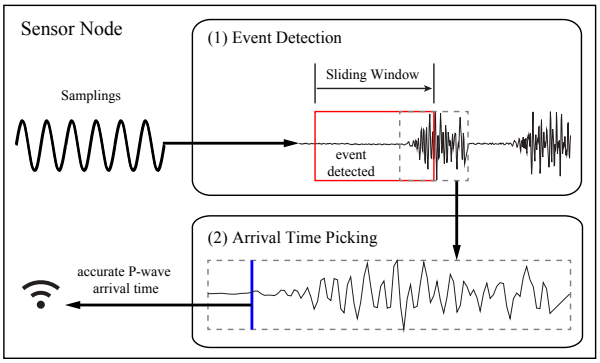


Figure 4: Two step P-wave arrival time picking.

Figure 4 illustrates how the proposed method works. The method consists of two steps, (1) Event Detection which continuously scanning the samplings from the sensor with a sliding window, claims if there is an event (change point) happens and extract a segment of signals around the change point; (2) Arrival Time Picking which takes the segment of signals from step (1), picks the exact change point (arrival time) from it and send the P-wave arrival time to coordinator node.

3.1.1 *Event Detection*

The goal of event detection is to continuously check weather there is a change point in the signal that is probably an earthquake event. Here we use a sliding window to scan the signal and compute the maximum value of the likelihood function out of the samples inside current window, we call this maximum value *statistics for ML change point detection*. If the statistics for ML change point detection of current window exceeds some threshold b, a change point is detected. Let w be the sliding window size, based on the definition of the likelihood function in equation (3) the *statistics for ML change point detection* can be defined as,

$$\mathcal{S} = \max_{t-w<k<t} \mathcal{L} \tag{6}$$

From the definition of statistics for ML change point detection, we can see that σ_1 and σ_2 are required to calculate the value of $\mathcal{S}$. Recall that σ_1^2 is the variance of the pre-change samples in the signal, which can be considered as the noise level of the signal. This noise level can be different for each sensor due to manufacturing, temperature and so on. So every senor needs to calculate its own σ_1^2 using some samples generated from it while no event happens. Let z_i be such a sample at time i, σ_1^2 can be calculated as,

$$\sigma_1^2 = \frac{1}{n-1}\sum_{i=1}^{n}(z_i - \bar{z})^2 \tag{7}$$

where n is the number of samples and $\bar{z}$ is the mean value of these n samples. Since the noise level of one sensor usually does not change much, the proposed method in this paper uses the σ_1^2 for a fixed period of time, e.g., one day and then update once.

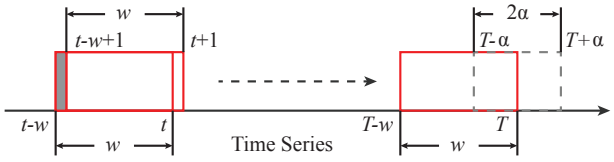


Figure 5: Illustration of sliding window in event detection.

Since σ_2^2 represents the variance of the post-change samples, the value of σ_2^2 depends highly on the property of event. This imposes that for each k, a σ_2^2 needs to be derived to maximize $\mathcal{L}$. This follows that $\partial\mathcal{L}/\partial\sigma_2^2$,

$$\frac{\partial\sum_{i=k+1}^{t}\left[\frac{1}{2}\ln\sigma_1^2 - \frac{1}{2}\ln\sigma_2^2 - \frac{x_i^2}{2}\left(\frac{1}{\sigma_2^2}-\frac{1}{\sigma_1^2}\right)\right]}{\partial\sigma_2^2} = 0 \tag{8}$$

and the σ_2^2 can be expressed as,

$$\sigma_2^2 = \frac{\sum_{i=k+1}^{t} x_i^2}{t-k} \tag{9}$$

In the discussion above, we showed how to calculate the statistics for ML change point detection. The event detection problem can be formulated as to find a time T such that,

$$T = \min\{t \geq 1 : \mathcal{S} > b\} \tag{10}$$

which means that to find the first time point where the statistics for ML change point detection is over the threshold.

Algorithm 1 Event detection algorithm

```
Calculate σ_1^2 as the noise level.
detected ← false.
for t ← w, ... do
    l* ← 0, s ← t − window.
    for k = s, ..., t − 1 do
        Calculate σ_2^2 and L, l = L.
        if l > l* then
            l* = l.
        end if
    end for
    if l* > b and detected is false then
        T ← t, detected ← true.
        Algorithm 2 picks arrival in [T − α, T + α] with σ_1^2.
    else
        if detected is true then
            detected ← false.
        end if
    end if
end for
```

Algorithm 1 gives the description of the event detection method. A sliding window keeps moving sample by sample and calculating the value of $\mathcal{S}$ with current samples in w, if the threshold is reached, a change point is detected at T, the signal in the window of $[T-\alpha, T+\alpha]$ is extracted and the arrival time picking algorithm will pick the accurate

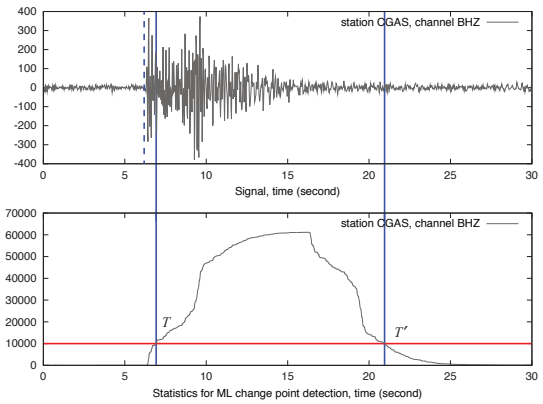


Figure 6: Event detection example on an earthquake event during 17:39:20 to 17:39:50 Feb 7, 2002.

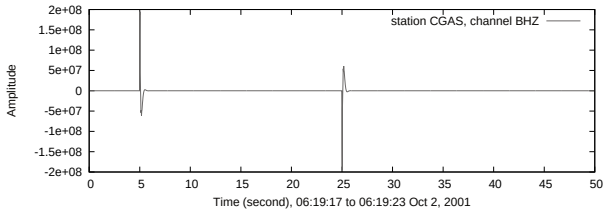


Figure 7: Spikes that causes change point detection.

arrival from it, see Figure 5. Then the sliding window continue moving from T and calculating the value of $\mathcal{S}$, since the event usually lasts for a period the value of $\mathcal{S}$ will be over the threshold for a while until time T'. We call the time window $T'-T$ the detection length, after time T' the algorithm will continue to find a *new* change point. A boolean variable *detected* is used to control this. Figure 12 gives an example of event detection result and the detection length of an earthquake event.

Note that the T claimed here is only a change point but not necessarily an earthquake event detection. Sometimes there might be a voltage spike from the sensor that can cause a change point detection, see Figure 7. Since the spikes have much larger amplitudes than events, to avoid the false alarm on spikes an upper threshold is set for $\mathcal{S}$. If the value of $\mathcal{S}$ is over the threshold the event detection will skip this change point.

Algorithm 2 Arrival timing picking algorithm

1: $l^* \leftarrow 0, k^* \leftarrow 0$.
2: **for** $k = T-\alpha+1,\ldots,T+\alpha-1$ **do**
3: Calculate σ_2^2 and $\mathcal{L}$, $l = \mathcal{L}$.
4: **if** $l > l^*$ **then**
5: $l^* \leftarrow l, k^* \leftarrow k$
6: **end if**
7: **end for**
8: **return** k^*

3.1.2 Arrival Time Picking

After the event detection done, the samples in the window $[T-\alpha, T+\alpha]$ is extracted with noise level σ_1^2. The arrival time picking is to find the change point $k*$ in $[T-\alpha, T+\alpha]$ which maximize the function value of $\mathcal{L}$. The problem can be simply formulated as,

$$k^* = \arg\max_k \mathcal{L} \tag{11}$$

The corresponding algorithm is described in algorithm 2. The vertical dashed line in Figure 12 indicate the arrival time picking of the algorithm for that detected event.

3.2 Event Location

Based on the architecture of InsightTomo, if a sensor node detects an event and picks the P-wave arrival time, it will send the time picking to the sensor node which acts as the coordinator for event location. The coordinator node only receives the time pickings from the sensor nodes and has the knowledge that which picking is from which node. In all of these pickings, there might be false alarms or some small and remote event is only detected by few stations. As we known, to estimate an event location, at least three pickings from different sensor nodes are required, the event detected only by one or two nodes is impossible to be located. Also, more pickings from different sensor nodes for one event usually lead to a better estimation. Thus there are two steps in event location, (1) Event Identification where the coordinator node identifies how many events existing in a series of arrival time pickings received and which pickings belong to the same event; (2) Location Estimation which uses Geiger's method to estimate the event location from the pickings of that event.

3.2.1 Event Identification

Since the InsightTomo system focuses on local seismic tomography (contrast to global tomography which focuses on spherical earth), the maximum arrival time difference among the sensor nodes is about several seconds. The event identification is based on two rules, (1) The maximum difference of the time pickings from the *same* event should be less than a threshold β; (2) The number of time pickings from the *same* event should be over a threshold θ.

Suppose that the coordinator node receives a series of time pickings and puts them in a list C. Each item in C is a pair $\langle$picking,node$\rangle$ which represent the arrival picking and the node picked it. The coordinator sorts list C increasingly according to the value of pickings and get C'. Then the coordinator will find the continuous subsequences in C' where the maximum difference among the items in the same subsequence is less than β, if the length of some subsequence is greater than θ the items in this subsequence are from the same event. Event identification will feed all the satisfied subsequences to step (2) that will estimate the event locations. Figure 8 gives an example of the event identification based on the 629 pickings of the P-wave arrival time picking method in section 3.1. In this example, the x-axis is the arrival time picking index for sorted list C' and the y-axis is the time difference between two adjacent pickings. Four events are identified (circles in the figure, $\beta = 4.0$ and $\theta = 14$) from

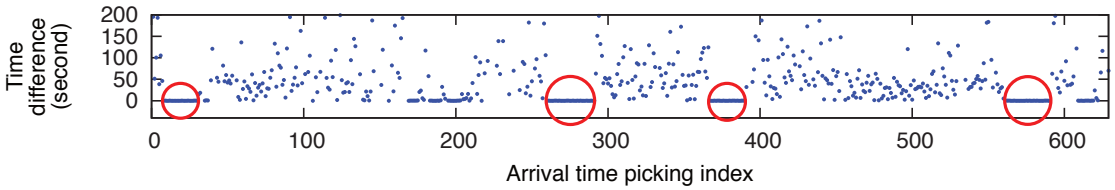


Figure 8: Sorted arrival time pickings from 30 stations with the entire day samplings on Feb 7, 2002 in Parkfield.

this data while three of them are identified in the manual pickings.

3.2.2 *Location Estimation*

The arrival time pickings in seismology are also called observed arrival times. To estimate the event location, an initial reference model of the subsurface P-wave velocities is required which can be used to calculate arrival time based on the guess of event location, this arrival time is referred to predicted arrival time. In this paper, we adopt 1D velocity model which treats the subsurface as a layered model and is commonly used in routine earthquake location (see 4 for details).

For sensor node v, we designate the observed arrival times by τ_v and the predicted arrival time by t_v which is a function of the estimated location x, y, z and the origin time q for the event. The residual, or the difference between the observed arrivals and the predicted arrivals is $r_v = \tau_v - \chi(q,x,y,z)$. Then an approximate location is estimated by seeking a small perturbation in the location that makes the residual smaller.

The guess, or estimation, is designated as, $\mathbf{x}_0 = \{q_0, x_0, y_0, z_0\}$ and let $\Delta\mathbf{x} = \{\delta q, \delta x, \delta y, \delta z\}$ be the small adjustments to the guess that bring X_0 closer to the correct location. The new location is represented as $\mathbf{x} = \mathbf{x}_0 + \Delta\mathbf{x}$ and the travel time associated with this location is $\chi = \chi_0 + \Delta\chi$. We can use the total derivative of χ to get the effect of perturbations,

$$\Delta\chi = \Delta q + \frac{\partial\chi}{\partial x}\Delta x + \frac{\partial\chi}{\partial y}\Delta y + \frac{\partial\chi}{\partial z}\Delta z \tag{12}$$

The residual calculated from the initial guess is $r_{0v} = \tau_v - \chi(q_0, x_0, y_0, z_0)$. The v-th residual is thus $r_v = \tau_v - (\chi_{0v} + \Delta\chi) = r_{0v} - \Delta\chi_v$. To estimate the location, the goal is to minimize the residual, i.e. find the perturbation that gives the least squared residual.

$$r_v = r_{0v} - \left(\Delta q + \frac{\partial\chi_v}{\partial x}\Delta x + \frac{\partial\chi_v}{\partial y}\Delta y + \frac{\partial\chi_v}{\partial z}\Delta z\right) \tag{13}$$

where the unknowns are $\Delta q, \Delta x, \Delta y, \Delta z$. Then the goal is to minimize the sum of the squares of the residuals with respect to these variables, i.e., the objective is,

$$\min \sum_v r_v^2 \tag{14}$$

Suppose there are n pickings in one event, the matrix of residuals can be written as,

$$\begin{bmatrix} 1 & \frac{\partial\chi_1}{\partial x} & \frac{\partial\chi_1}{\partial y} & \frac{\partial\chi_1}{\partial z} \\ 1 & \frac{\partial\chi_2}{\partial x} & \frac{\partial\chi_2}{\partial y} & \frac{\partial\chi_2}{\partial z} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \frac{\partial\chi_n}{\partial x} & \frac{\partial\chi_n}{\partial y} & \frac{\partial\chi_n}{\partial z} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} r_{01} \\ r_{02} \\ \vdots \\ r_{0n} \end{bmatrix}$$

this is the standard inversion of matrix equations $\mathbf{M}\Delta\mathbf{x} = \mathbf{r}$. We use Bayesian ART method (refer to section 3.3) to solve this equation system and get the perturbation solution $\Delta\mathbf{x}$. The event location is obtained by adding $\Delta\mathbf{x}$ to initial guess. This solution may not be close to the real location, then we can use this as a new guess and solve the system again until the solution is good enough.

After the event location done, the coordinator node needs to send the event location to corresponding sensor node that has a arrival time picking on this event. InsightTomo then can proceed to the ray tracing and tomography computation.

3.3 Tomography Inversion

On the reception of event location information, the sensor nodes will trace the ray paths and send them to the nodes which perform landlord for distributed tomography computation. Since the ray tracing method used in this paper is standard bending based method based on 1D velocity model and the distributed tomography computation method used here need to partition the ray paths after ray tracing done, we will discuss the ray tracing with tomography inversion in this section and will not give the details on ray tracing algorithms due to the space limitation.

Suppose that there are N nodes and J earthquakes, we consider a perturbation approach in this paper. Let $\mathbf{s}^*$ be the slowness (reciprocal of velocity) model with resolution M (blocks). $\mathbf{s}^*$ can be assumed to be a reference model, $\mathbf{s}^0$, plus a small perturbation $\Delta\mathbf{s}$, i.e., $\mathbf{s}^* = \mathbf{s}^0 + \Delta\mathbf{s}$. For simplicity, we use $\mathbf{s}$ to denote $\Delta\mathbf{s}$ in the following discussion.

The ray travel times can be estimated by the arrival time pickings and estimated event origin times from event location. Let $\mathbf{t}_i^* = [t_{i1}^*, t_{i2}^*, \ldots, t_{iJ}^*]^T$, where t_{ij}^* is the travel time experienced by node i in the j-th event. Based on the ray paths from ray tracing, the travel time of a ray is the sum of the slowness in each block times the length of the ray within that block, i.e., $t_{ij}^* = \mathbf{A}_i[j,m] \cdot \mathbf{s}^*[m]$ where $\mathbf{A}_i[j,m]$ is the length of the ray from the j-th event to node i in the m-th block and $\mathbf{s}^*[m]$ is the slowness of the m-th block. Let $\mathbf{t}_i^0 = [t_{i1}^0, t_{i2}^0, \ldots, t_{iJ}^0]^T$ be the unperturbed travel times where $t_{ij}^0 = \mathbf{A}_i[j,m] \cdot \mathbf{s}^0[m]$. t_{ij}^* and t_{ij}^0 represent observed travel time and predicted travel time respectively. In matrix notation we

have the following equations on sensor node i,

$$\mathbf{A}_i\mathbf{s}^* - \mathbf{A}_i\mathbf{s}^0 = \mathbf{A}_i\mathbf{s} \tag{15}$$

where $\mathbf{A}_i[j,m]$ represents the element at the j-th row and m-th column of matrix $\mathbf{A}_i \in \mathbb{R}^{J\times M}$. Let $\mathbf{t}_i = [t_{i1}, t_{i2}, \ldots, t_{iJ}]^T$ be the travel time residual such that $\mathbf{t}_i = \mathbf{t}_i^* - \mathbf{t}_i^0$, equation (15) can be rewritten as,

$$\mathbf{A}_i\mathbf{s} = \mathbf{t}_i \tag{16}$$

We now have a linear relationship between the travel time residual observations, $\mathbf{t}_i$, and the slowness perturbations, $\mathbf{s}$. Since each ray path intersects with the model only at a small number of blocks compared with M, the design matrix, $\mathbf{A}_i$, is sparse. The seismic tomography inversion problem is to solve the system,

$$\mathbf{A}\mathbf{s} = \mathbf{t} \tag{17}$$

where $\mathbf{A} = [\mathbf{A}_1^T, \mathbf{A}_2^T, \ldots, \mathbf{A}_N^T]^T$ and $\mathbf{t} = [\mathbf{t}_1^T, \mathbf{t}_2^T, \ldots, \mathbf{t}_N^T]^T$. This system is usually overdetermined and the inversion aims to find the least-squares solution $\mathbf{s}$ such that,

$$\mathbf{s} = \arg\min_{\mathbf{s}} \| \mathbf{t} - \mathbf{A}\mathbf{s} \|^2 \tag{18}$$

To solve the equation system in tomography inversion problem, there are many methods can be used as discussed in section 5. In InsightTomo, we employ Bayesian ART method which has been proved to be a good smoother for tomography inversion.

Algebraic Reconstruction Technique (ART) is a row action method to solve equation system. As an iterative method, ART produce a sequence of estimated vectors which converge to the required solution. Consider the system in tomography inversion $\mathbf{A}\mathbf{s} = \mathbf{t}$ where $\mathbf{A} \in \mathbb{R}^{L\times M}$ and $\mathbf{s}, \mathbf{t} \in \mathbb{R}^{M\times 1}$. The basic ART method can compute the approximation of the solution of the system with the following iterative formula,

$$\mathbf{s}^{(k+1)} = \mathbf{s}^{(k)} + \rho^{(k)} \frac{t_l - a_l^T \cdot \mathbf{s}^{(k)}}{\|a_l\|^2} a_l \tag{19}$$

where a_l is the $l-$th row (e.g., the $l-$th ray path traced) of $\mathbf{A}$, a_l^T is the transpose of a_l, t_l is the $l-$th component of vector $\mathbf{t}$ and $\rho^{(k)}$ is a relaxation parameter. $\|a_l\|^2 = a_l^T \cdot a_l$ and (k) denotes the iteration number, the procedure can repeatedly operate on the equations with $l = (k) \mod (L+1)$. The formula above provides a simple iteration routine, if the system is consistent, basic ART is proved to converge to the minimum-norm solution.

For the inconsistent system, a Bayesian version of the basic ART is proposed by G.T. Herman [4] for the image reconstruction in medical tomography. Suppose the system $\mathbf{A}\mathbf{s} = \mathbf{t}$ is inconsistent, then we consider the system $\mathbf{A}\mathbf{s} + \mathbf{u} = \mathbf{t}$ where $\mathbf{u}$ is chosen from given any $\mathbf{s}$. Then the system is transformed to a well-proposed problem, $\mathbf{s}$ and $\mathbf{u}$ can be solved simultaneously. Bayesian ART method has the following iterative

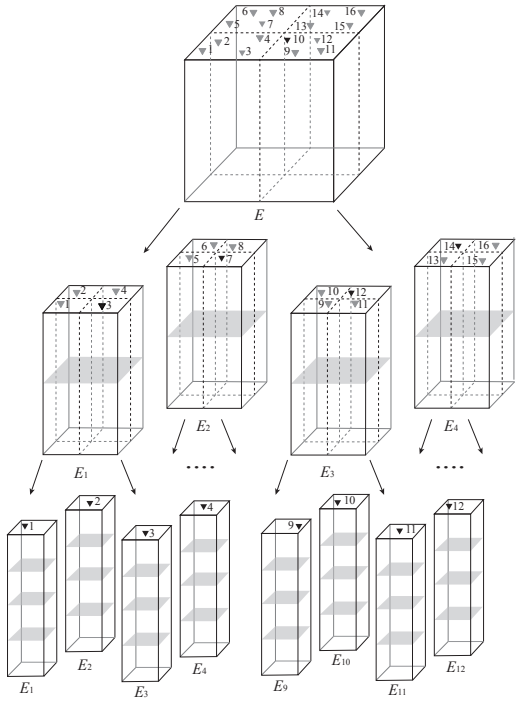


Figure 9: Multi-resolution evolving tomography.

formulas,

$$d^{(k)} = \rho^{(k)} \frac{\lambda t_i - (u_i^{(k)} + \lambda a_i^T \cdot \mathbf{s}^{(k)})}{1 + \lambda^2 \|a_i\|^2} \tag{20}$$

$$\mathbf{s}^{(k+1)} = \mathbf{s}^{(k)} + \lambda d^{(k)} a_i \tag{21}$$

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + d^{(k)} \mathbf{e}_i \tag{22}$$

where $\mathbf{e}_i$ is a unit vector with the $i-$th component equal to one, λ is the regularization parameter. Bayesian ART method find the solution $\mathbf{s}$ such that,

$$\mathbf{s} = \arg\min_{\mathbf{s}} \| \mathbf{t} - \mathbf{A}\mathbf{s} \|^2 + \lambda^2 \| \mathbf{s} \|^2 \tag{23}$$

Note that in Bayesian ART method, we need an additional vector $\mathbf{u}$ of length L, but in $(k)-$th step only one component of $u^{(k)}$ needs to update locally without communications.

To compute the tomography inversion in the network, a tomography partition and computation distribution algorithm with a multi-resolution evolving scheme [22] is employed to distribute the computation load, reduce the communication cost and approximate the least-squares solution of the seismic tomography inversion problem in the network. To distribute the computation load, this method first partition the volcano structure geometrically and the system $\mathbf{A}\mathbf{s} = \mathbf{t}$ correspondingly. Then some nodes are selected as *landlord*s to compute part of the tomography model. The computation on each landlord is entirely local so that the communication cost is bounded. Since the computation on each landlord only uses part of the system $\mathbf{A}\mathbf{s} = \mathbf{t}$, the result is not equivalent to the solution of the original system. To approximate the optimal solution, a multi-resolution evolving scheme is in-

troduced: the network initially computes a coarse resolution tomography without partition when small amount of earthquake events arrive; as more and more earthquake events arrive, the network will compute finer and finer resolution tomography with more partitions. The intuition behind this is that the network first computes an outline of the volcano structure in an low resolution then fills up with finer details inside. With the multi-resolution evolving scheme, people do not need to wait for all computation done and can retrieve the intermediate results under low resolutions in a real-time manner.

Algorithm 3 Distributed Multi-resolution Evolving Tomography

Upon the reception of an event location
  Trace the ray to get ray path a_l.
  Calculate $\vec{a}_{l,p}$ and $t_{l,p}$ for $1 \le p \le P_r$.
  Send $\vec{a}_{l,p}$ and $t_{l,p}$ to landlord h_p.
Upon the reception of $\vec{a}_{l,p}$ and $t_{l,p}$ (node is landlord)
  Add $\vec{a}_{l,p}$ and $t_{l,p}$ to $\mathbf{A}_p\mathbf{s}_p = \mathbf{t}_p$.
  if there is enough rays information in $\mathbf{A}_p\mathbf{s}_p = \mathbf{t}_p$.
    Solve $\mathbf{A}_p\mathbf{s}_p = \mathbf{t}_p$ with Bayesian ART.
    Broadcast $\mathbf{s}_p$.
  endif
Upon the reception of $\mathbf{s}_p$ from landlord h_p
  Update the corresponding part of $\mathbf{s}^0$ with $\mathbf{s}_p$
  if all $\mathbf{s}_p (1 \le p \le P_r)$ have been received
    if $r+1$ is equal to q
      TERMINATE
    else
      $r \leftarrow r+1$, $Q \leftarrow 2^r d \times 2^r d \times 2^r d$, $P \leftarrow 4^r$
      Increase the resolution of $\mathbf{s}^0$ to Q
      Partition the tomography model into P_r parts
    endif
  endif

The partition method involves the ray path and travel time splitting after ray tracing on each sensor node, this can be done locally since the partition scheme can be predefined. Suppose that there are P partitions in the subsurface structure. For partition $p (1 \le p \le P)$, let $\vec{a}_{l,p}$ be the partial ray path of the l-th ray a_l, $t_{l,p}$ be the corresponding partial travel time residual of t_l and $\mathbf{s}_p$ be the partial slowness perturbation. For more details about how to estimate $\vec{a}_{l,p}$ and $t_{l,p}$, please refer to [22]. Then we use $\mathbf{A}_p$ to denote the column partition of $\mathbf{A}$ corresponding to tomography partition p, $\mathbf{t}_p$ is the partial time residuals for $\mathbf{A}_p$. In each partition, some node h_p is selected as the landlord and the landlord in partition p solves the subsystem $\mathbf{A}_p \cdot \mathbf{s}_p = \mathbf{t}_p$. The global tomography can be obtained by combining all $\mathbf{s}_p$.

Figure 9 illustrates how the multi-resolution evolving scheme works. Notice that the computation of the partial travel time residuals highly depends on the slowness reference model. In the multi-resolution evolving scheme, the tomography model is not partitioned initially so that a good initial guess of the slowness model can be derived. This initial guess is used to estimate the partial travel time residuals later for approximating the optimal solution.

Suppose that the resolution of the tomography model is $d \times d \times d$ in the beginning, a single landlord (node 10 in the example) will compute the first perturbation for the reference model. Then the resolution increases to $2d \times 2d \times 2d$, and the tomography model is partitioned into 4 parts and distributed to 4 landlords (node 3, 7, 12 and 14) for computation as illustrated in Figure 9. Notice that, here the resolution of each partition is $d \times d \times 2d$. The aforementioned partition procedure will be recursively applied in each partition when sufficient more new earthquake events arrive, until the required resolution is achieved. Thus, at the $(r+1)$-th $(r = 0, 1, 2, \ldots)$ resolution, the tomography model has resolution $2^r d \times 2^r d \times 2^r d$ and is partitioned into 4^r parts and evenly distributed to $\max(N, 4^r)$ landlords. Algorithm 3 gives the description of Distributed Multi-resolution Evolving Tomography (DMET).

4 Evaluation

In this section, we evaluate InsightTomo performance with extensive experiments by utilizing (1) the CORE[1] and EMANE[2] network emulators [1] to emulate the sensor network with sensor nodes and (2) BeagleBone Black boards to perform as coordinator node for event location computation and landlord node for tomography inversion computation respectively. The advantage of emulation is that the code developed over the emulator can be transplanted to a Linux-based device, e.g., BeagleBone Black board, virtually without any modifications.

Real data set from the deployment on SAF at Parkfield is used in the evaluations of P-wave arrival time piking and event location. Both synthetic and real data are used to evaluate the DMET algorithm. The system implementation and experiments validate the correctness and accuracy for the proposed algorithms and the feasibility of InsightTomo system design.

4.1 System Implementation

InsightTomo is designed to compute the tomography in a wireless mesh network and requires both unicast and broadcast communication according to the system architecture and the algorithm requirements. On most remote deployment sites it is hard to rely on the pre-existing infrastructures (e.g. cellular infrastructure). Therefore, we need to utilize the wireless mesh networking which create its own infrastructure by multi hop relays. However, such systems may experience erratic link qualities and intermittent disconnections among nodes. These characteristics, combined with unpredictable environmental conditions, make it difficult to maintain efficient and reliable end-to-end connectivity that spans many hops. For example, the traditional end-to-end protocol like TCP is not suitable for a wireless mesh network in challenging environment because the packet lost ratio is higher than a wired network. On one hand, in a multi hop transmission the source node need to retransmit the packet through all hops once the packet lost on the path. On the other hand, the data rate can be very low after several hops due to packet loss and congestion control.

To address the challenges in wireless mesh networking, we adopt Disruption-Toleration Networks (DTN) techniques

[1] http://cs.itd.nrl.navy.mil/work/core/
[2] http://cs.itd.nrl.navy.mil/work/emane/

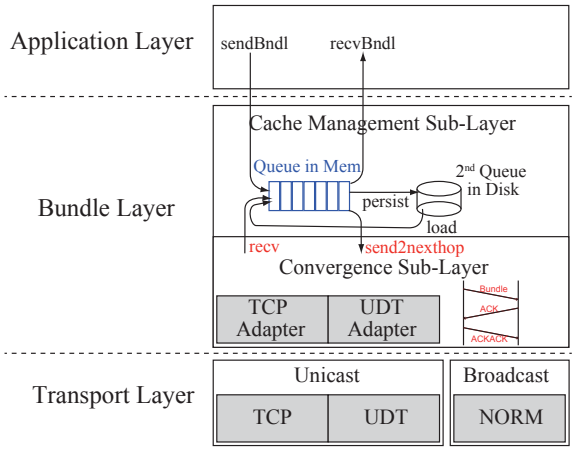
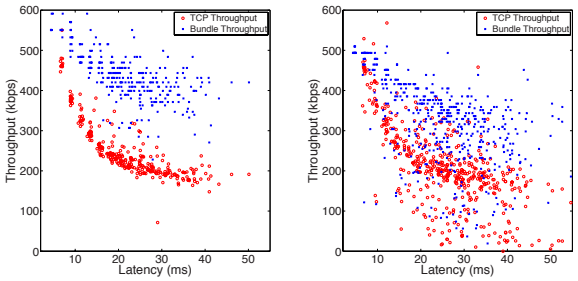


Figure 10: Bundle layer architecture.

to maintain efficient and reliable end-to-end connectivity that spans many hops for data delivery. In our design, the data is buffered in a bundle and then transferred hop by hop in a store-and-forward manner until it arrives at the destination. Our implementation of DTN technique does not make any changes to underlying network services, it uses TCP for one-hop reliable bundle transfer, and uses routing table to indicate the next hop. Figure 10 shows the application interfaces on each node for the integration of DTN and routing protocol. Figure 11 shows that the Bundle Layer outperforms TCP with routing protocol. The test is done using CORE and EMANE for 100 nodes multi-hop network settings.



(a) B.A.T.M.A.N. routing (b) OLSR routing

Figure 11: Performance of Bundle layer vs TCP.

Besides unicast, we implement a delay-tolerant broadcasting service based on the NACK-Oriented Reliable Multicast (NORM) protocol[3]. Using NORM interface, one node can push a bundle reliably to its one-hop neighbors. Our cache component can receive and store this broadcast bundle, and rebroadcast it again with NORM, to the nodes that are two hops away, and so on so forth. A redundancy check module is developed in the cache component guarantees each node receives the same bundle at most once.

The implementation of all the algorithms in InsightTomo is in ANSI C. The event location and tomography inversion related code are cross-compiled to run on BeagleBone Black board. All other code can be directly ported to embedded system such as ARM-based CPU or MCU.

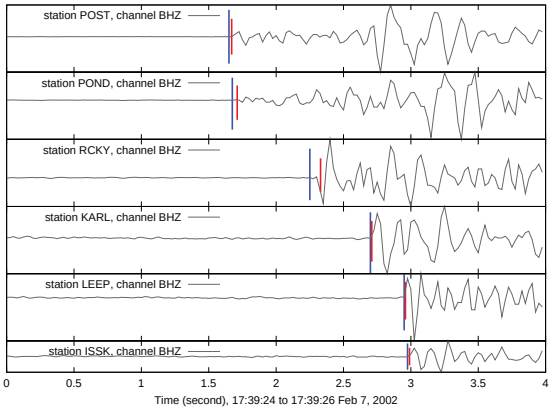


Figure 13: Manual pickings vs algorithm pickings on 6 stations in one event.

4.2 P-wave Arrival Time Picking Accuracy

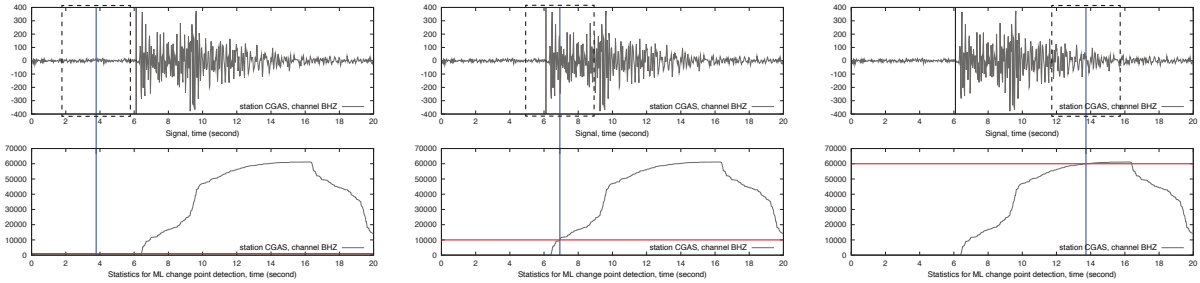
The raw seismic data set from the deployment on SAF at Parkfield is archived by IRIS[4]. The deployment is from Jan 1, 2000 to Dec 31, 2002 with 61 stations but the archived data we can download consists of 42 stations from Oct 2, 2001 to Oct 10, 2002. An extensive data set is obtained from Dr. Haijiang Zhang. This data set has more stations and longer period of data but not in seismic waves. The arrival timings in the extensive data set are from manual analysis, and the event locations and velocity model are from the previous computation with double-difference tomography method [33]. We use the extensive data set for comparison in this evaluation.

From the discussion in the system design, to get the accurate P-wave arrival time picking, the event detection algorithm needs to give the correct detection point that the arrival time point can be involved in the correct window to be picked. Most important parameter is the threshold in event detection. If the threshold is not set properly, the arrival time picking can be meaningless, see Figure 12. The dashed box indicate the window of signals to be processed by arrival time picking algorithm. If the threshold is too small, the event detection can trigger a false alarm while a delay detection could happen if the threshold is too big. In the implementation, after each detection, the algorithm will get some samples from the beginning of the current window, recalculate the statistics and get a new detection point. If this difference between these two points is bigger than the window size, the algorithm will reset the noise level for future detection. Figure 13 compares the pickings of P-wave arrival time picking algorithm with manual pickings, the long and short vertical lines indicate algorithm picking and manual picking respectively. We can see that two pickings are close and the picking is accurate.

Figure 14 shows the algorithm picking errors in comparison with manual picks from the extensive data set where

[3] http://cs.itd.nrl.navy.mil/work/norm/

[4] http://www.iris.edu/data/



(a) False alarm (threshold=20) (b) Change point detection (threshold=10000) (c) Delay detection (threshold=60000)

Figure 12: Earthquake event during 17:39:20 to 17:39:40 Feb 7, 2002 on BHZ channel of station CGAS in Parkfield.

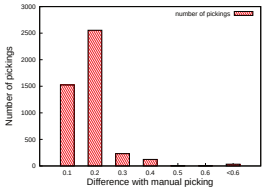


Figure 14: Picking errors.

x-axis is the picking difference between two methods and the y-axis is the number of pickings stand in that difference range. There are total 4478 pickings generated by the algorithm, which has a matching in the manual picking data for the Parkfield data. About 91% pickings from our algorithm are within 0.2 seconds of manual pickings. The mean value and the standard deviation of the difference between our pickings and manual pickings are 0.043 and 0.23.

4.3 Event Location Accuracy

As we discussed in the system design, the 1D reference velocity model is used for event location in InsightTomo. Figure 15 gives the reference model used in our evaluation. Note that this model is also used in the ray tracing and as the initial model for tomography inversion.

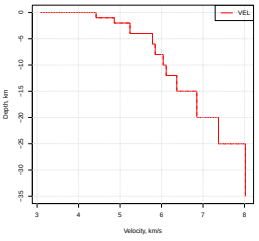
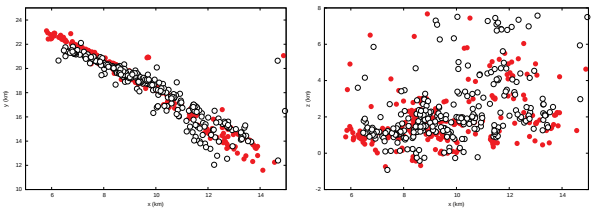


Figure 15: 1D P-wave velocity reference model.

The P-wave arrival time picking and event identification algorithm generate 433 events and 290 of them have corresponding event data in the extensive data set. The mean value and standard deviation of the difference between the positions of our algorithm and extensive data set are (0.33, 0.26) km respect to X, (0.46, 0.37) km respect to Y and (0.39, 0.40) km respect to Z.



(a) Event locations on XY plane (b) Event locations on XZ plane

Figure 16: Event location result comparison, the empty circles and solid disc indicate the event location from InsightTomo and the event location in extensive data set respectively.

4.4 Tomographic Imaging Result

Since the real structure of the Earth subsurface is unavailable, we can not directly compare the tomography image from the real data with ground truth. To verify the correctness, accuracy and the performance for DMET algorithm, here we use both a synthetic phantom and the real data set from Parkfield to construct the image of seismic tomography using InsightTomo and present the results.

4.4.1 Synthetic Phantom

The evaluation of algorithm is illustrated by simulating seismic data on a synthetic model of resolution 128^3 consisting of a magma chamber (low velocity area) in a 10 km^3 cube. 100 stations are randomly distributed on top of the cube.

We evaluate the algorithm starting with resolution 8^3 and one partition, evolving to resolution 16^3 with 4 partitions and complete at resolution 32^3 with 16 partitions. 50, 100 and 400 events are generated for resolution from low to high to make sure the system is overdetermined. To simulate the event location estimation and ray tracing errors, a White Gaussian Noise is added to the travel time to generate the sensor node observations (arrival times).

To validate the correctness and accuracy of the algorithm, we first visualize the tomography result in vertical slices in Figure 17. Each row of figures shows the same tomography slice on corresponding layer along with X or Y axes (the total layers of each figure is equal to the resolution dimension of

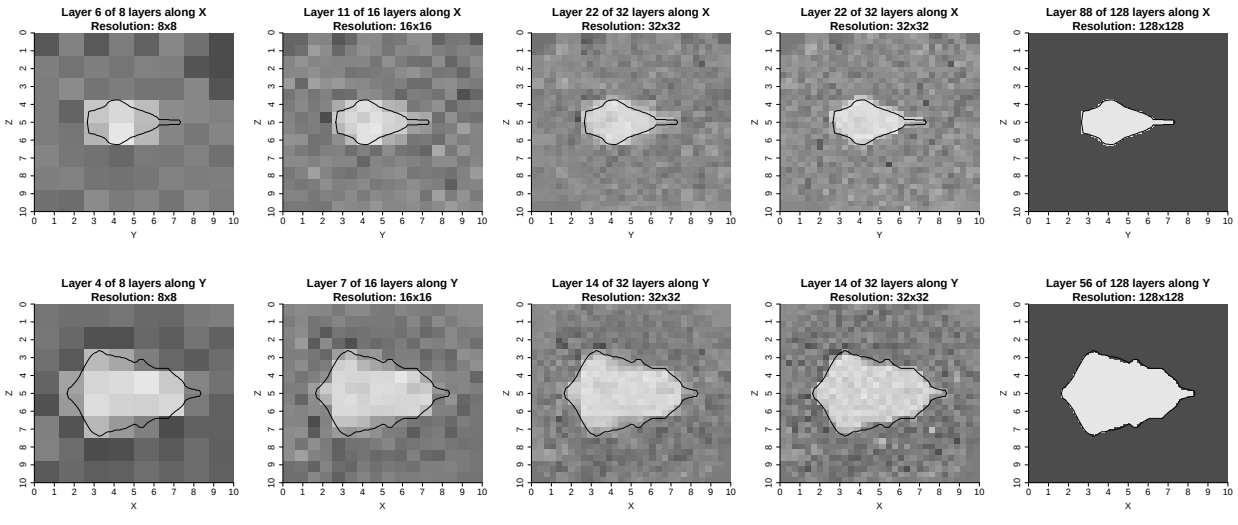


Figure 17: Vertical slices of tomography model. Column (a), (b) and (c) are the results from DMET algorithm with resolution dimension 8, 16, 32 respectively while column (d) is the result of centralized algorithm and column (e) is the ground truth.

the result). The black polygons give the cross section outline of the magma chamber surface. We can see that at the lowest resolution 8^3, the result can hardly indicate the outline of the magma chamber since the block size is big, especially for the small cross section in the first row. But it gives a good start point for the higher resolution to further refine the result. At resolution dimension 16, the result can closely show the outline of the magma chamber. The result in column (c) at resolution 32^3 is already very close to the centralized solution in column (d).

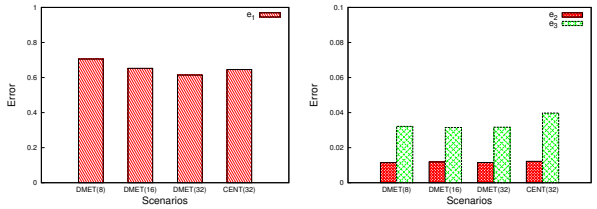


Figure 18: Measures of distance from synthetic phantom.

Using $\tilde{\mathbf{s}}$, $\mathbf{s}^*$ and $\bar{\mathbf{s}}$ to represent the synthetic model, the reconstructed model and the mean value of $\mathbf{s}^*$ respectively, we use the following quantitative measures of distance from the synthetic model provided in [10] to evaluate the estimation quality,

$$
\begin{aligned}
e_1 &= \left(\sum_{i=1}^{n}(\tilde{\mathbf{s}}_i - \mathbf{s}_i^*)^2 / \sum_{i=1}^{n}(\mathbf{s}_i^* - \bar{\mathbf{s}})^2\right)^{\frac{1}{2}} \\
e_2 &= \sum_{i=1}^{n}|\tilde{\mathbf{s}}_i - \mathbf{s}_i^*| / \sum_{i=1}^{n}|\mathbf{s}^*| \\
e_3 &= \max|\tilde{\mathbf{s}}_i - \mathbf{s}_i^*|
\end{aligned}
$$

These represent the normalized root mean squared distance, the average absolute value distance and the worst-case distance respectively. The result is shown in Figure 18, DMET(8) means that the distance analysis of DMET algorithm with resolution 8^3 and CENT(32) indicates the result of centralized algorithm. First we observe that in DMET algorithm, the distances are decreasing along with the increase of the resolution. The distances in DMET(32) are even smaller than CENT(32), this is because that we use the relative update as the stop criteria in Bayesian ART method and the centralized algorithm may stop before the distance is small enough. This analysis can imply that the multi-resolution evolving scheme can give a good approximation on each resolution level for estimating partial travel times and the computation can approximate the centralized solution while not diverging in the local computation in each partition.

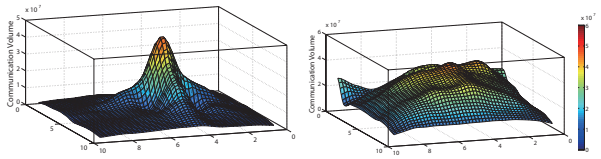
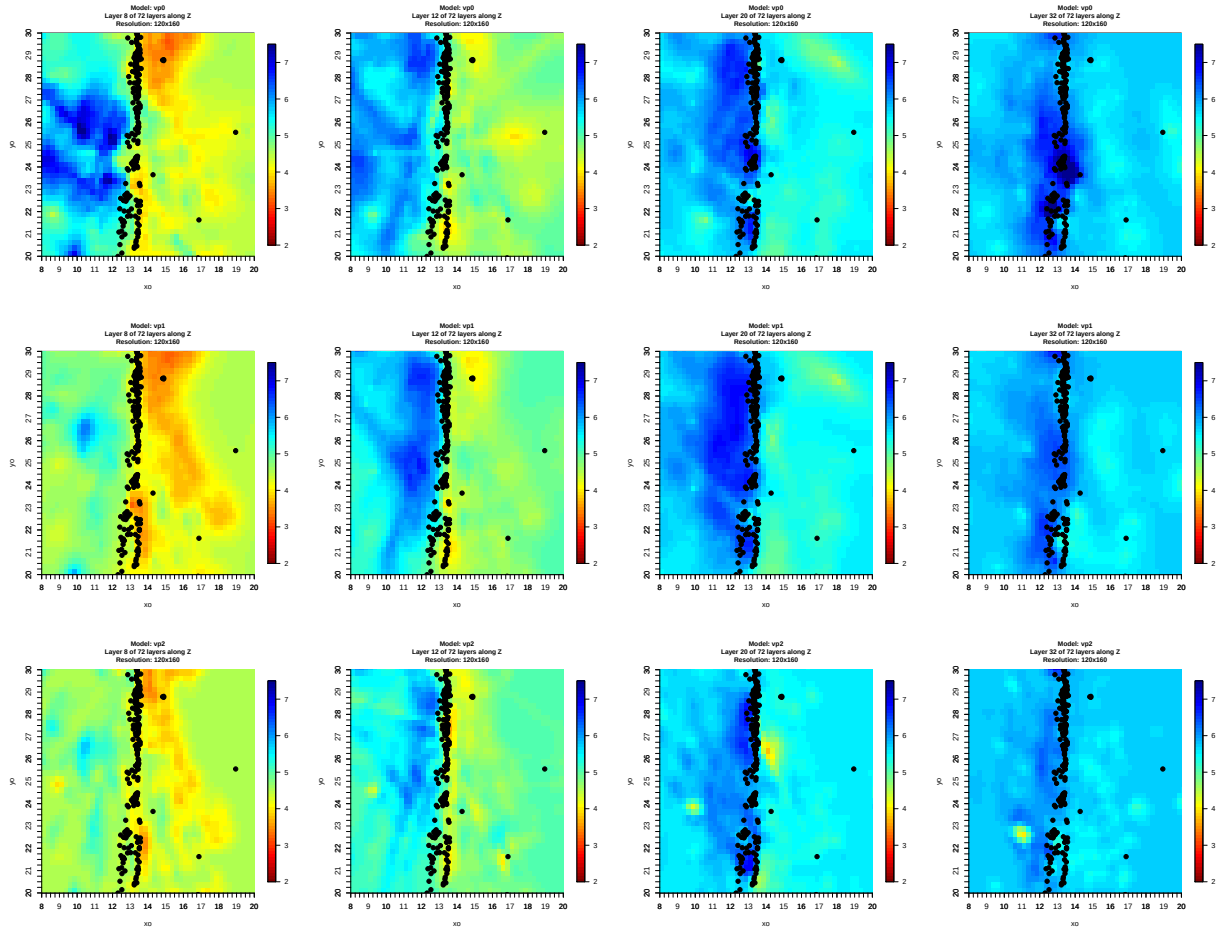


Figure 19: Communication and computation load balance.

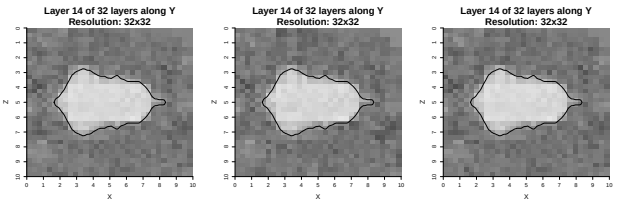
Figure 19 shows the communication cost on each node in the network for 2 different scenarios. From Figure 19(a), we can see that the communication cost in centralized scenario for the nodes near the base station in the center are much higher than other nodes since all the messages need to be delivered through them. The communication cost on each node



(a) Vp at depth = 1km (b) Vp at depth = 2km (c) Vp at depth = 4km (d) Vp at depth = 7km

Figure 20: Horizontal slices of the P-wave velocity at depths of 1, 2, 4, and 7 km. The fault is located around X=13.5km.

in our algorithm shown in Figure 19(c) is more balanced.



(a) 10% packet loss (b) 40% packet loss (c) No packet loss

Figure 21: Vertical slices of tomography model.

Next we did an evaluation on the robustness of DMET. The algorithm runs with the same data set and two different packet loss ratios of 10% and 40% set in the emulator. Figure 21 shows the rendering of one vertical slice along Y axes. We can see that with 10% or even 40% packet loss, compared with the result without packet loss, It is hard to tell the differences. This validate the robustness of DMET algorithm which can be tolerant to a severe packet loss.

4.4.2 Real Data: Parkfield

In this section, we present the tomography result of Parkfield delivered by InsightTomo System from the raw seismic data. Followed [33], the y-axis of the tomography rotated 40 degrees counterclockwise from north so that it is parallel to the local strike of the SAF.

Figure 20 gives the P-wave velocity model around SAF at Parkfield. The tomography in first row is centralized calculated from the extensive data with event locations (1538 events involved). It is close to the result in [33], it is easy to see that the velocity model is different on different side of SAF. The dots in the tomography indicate the event locations estimated in InsightTomo system. A scientific fact is that the events often happen around the fault which is verified by our result. The tomography in second row is centralized calculated from the extensive data, but not all the data is used. Only 438 events happened between Oct 2, 2001 and Oct 10, 2002 are used. This is because InsightTomo only used the raw seismic data inside that time period. It makes more sense to compare the InsightTomo result with the centralized result on similar data set. We can see that the fault feature is easy

to get from the second row of tomography but the velocity contrast at different sides of SAF is reduced.

The third row is the result from InsightTomo based on the the 433 events extracted from raw seismic data. A two-level DMET algorithm is applied where one landlord will start the tomography inversion with low resolution and four landlords compute partial model in high resolution. From the tomography image, except the Vp at depth 7km, the main feature of SAF can still be recognized and comparable with the centralized result. This verifies the feasibility of InsightTomo system.

5 Related Work

Static tomography inversion for 3D structure, applied to volcanoes and oil field explorations, has been explored since the late 1970's [5, 28, 8]. In volcano applications, tomography inversion used passive seismic data from networks consisting of tens of nodes, at most. The development and application to volcanoes include Mount St. Helens [7, 9, 29], Mt. Rainier [15], Kliuchevskoi, Kamchatka, Russia [11], and Unzen Volcano, Japan [17]. At the Coso geothermal field, California, researchers have made significant contributions to seismic imaging by coordinating tomography inversions of velocity [31], anisotropy [12], attenuation [30] and porosity [13].

P-wave arrival picking has been studied by the community. The widely used approach is the STA/LTA method [16] that has been using in real deployment [24] on volcano monitoring. The STA/LTA method continuously monitors the ratio of short-term average over long-term average on a signal. Since it is based based on RSAM (Realtime Seismic Amplitude Measurement), which is calculated on raw seismic data samples every second, the accuracy of STA/LTA is not enough for tomography computation.

In seismic tomography, the event location can also be formulated as a least-squares problem by Geiger's Method [2], and the estimation vector is of length 4 (event origin time and 3D coordinates). Since the dimension of the estimation vector is fixed and small, a centralized solution can be applied in the network for this problem. In step (2), each node traces ray path based on a reference model. This can be naturally distributed since the ray tracing computation is entirely local. The third step is the most computationally intensive and time-consuming aspect of high resolution seismic tomography. The sparse system can be solved by conjugate gradient method or row action method [10]. However, designed for high-performance computers, these centralized approaches need significant amount of computational/memory resources and require the knowledge of global information. As a result, they cannot be directly distributed in wireless sensor network. Thus, the key research challenge here is how to solve the least-squares problem in step (3) distributedly under the severe constraints of wireless sensor network. In this paper, we focus on distributed tomography inversion algorithm, while assuming that the event arrival timing at each node has been extracted from the raw seismic data by each node itself [23, 26], as well as that the event location and ray tracing have been done.

The methods to solve least-squares problem mainly fall into two categories, direct methods and iterative methods. Iterative methods for solving large sparse linear systems of equations are advantageous over the classical direct solvers, especially for huge systems [3]. Methods of parallelizing least-squares solutions on distributed memory architecture have been studied for both direct and iterative methods, but there are few studies on distributing the least-squares solutions from a wireless sensor network point of view.

Straková, Gansterer and Zemen investigate randomized algorithms based on gossiping for distributed QR factorization [25]. This algorithm demands asynchronous randomized information exchange among nodes. The problem is that gossiping based algorithms converge slow and a back substitution node by node is required to get the least-squares solution where the delay is introduced. Renaut proposed a parallel multisplitting solution of the least-squares problem [18] where the solutions to the local problems are recombined using weighting matrices. The problem to distribute this algorithm in the network is that it requires broadcast communications in each iteration. Yang and Brent describe a modified conjugate gradient least-squares method to reduce inner products global synchronization points and improve the parallel performance [32]. This can also be potentially distributed over the network but the broadcast communication is still required in each iteration.

In the literature of signal processing, there are a few studies on consensus-based Distributed Least Mean Square (D-LMS) algorithms [14, 20] in sensor network. These algorithms let each node maintain its own local estimation and, to reach the consensus, exchange the local estimation only within its neighbors. This can also be used for getting least-squares solutions statistically. The problem is that consensus-based methods can be slow [27] on convergence and the communication cost highly depends on the dimension of estimation. In large-scale network, consensus based algorithms introduce not only high communication overhead but also long delays involving frequent multi-hop communications. Sayed and Lopes developed a Distributed Recursive Least-Squares (D-RLS) strategy by appealing to collaboration techniques to achieve an exact recursive solution [19]. But it requires a cyclic path in the network to perform the computation node by node while exchanging a large dense matrix between nodes.

Kaczmarz's row projection method [6], also known as Algebraic Reconstruction Technique (ART), was one of the first iterative methods used for large nonsymmetric systems. Its main advantages are robustness and cyclic convergence on inconsistent systems. Dan and Rachel proposed scheme performs Kaczmarz row projections within the blocks and merges the results by certain component-averaging operations, called component-averaged row projections (CARP) [3]. CARP is a robust and memory efficient method to solve sparse systems. However, to distribute CARP in the network, the broadcast communication is required and the performance depends on the partition number of the equation system.

The methods mentioned above have been proved to be convergent, but there are several system design problems to distribute them in the wireless sensor network due to high

communication cost or long delay. In high-resolution seismic tomography, we intend to solve a large sparse system (n can be as large as tens of thousands even millions) in a large-scale network (hundreds of nodes), i.e., $N \ll m$ and $N \ll n$. Considering this, the multisplitting method and CARP have less communication cost if all algorithms converge in the same order of iteration number. But the iteration number highly depends on the matrix condition number (from the evaluation in [21], these methods need hundreds to thousands iterations to converge over a network with hundreds of nodes). Besides, some methods either requires broadcast communication per iteration or a path in the network to perform the computation node by node. This results to either high communication cost or long delay if the number of iterations is large.

6 Conclusion

In this paper we presented InsightTomo, an end-to-end system which performed in-network data processing and obtained tomography using distributed computation. Evaluation was carried out on real data and the obtained results where comparable to centralized method obtained by [33]. The communication cost was also flat meaning balanced unlike the centralized approach and was resilient to data loss. This project marks the collaboration between geophysicists and computer scientists that provided opportunities to introduce new technology for geophysical monitoring. The approach presented here has broader implication beyond tomography inversion and can be easily extended to oil and natural gas exploration.

7 References

[1] J. Ahrenholz, T. Goff, and B. Adamson. Integration of the CORE and EMANE Network Emulators. In *MILITARY COMMUNICATIONS CONFERENCE, 2011 - MILCOM 2011*, pages 1870–1875, 2011.

[2] L. Geiger. Probability method for the determination of earthquake epicenters from the arrival time only. *Bull.St.Louis.Univ*, 8:60–71, 1912.

[3] D. Gordon and R. Gordon. Component-Averaged Row Projections: A Robust, Block-Parallel Scheme for Sparse Linear Systems. *SIAM Journal on Scientific Computing*, 27(3):1092–1117, 2005.

[4] G. T. Herman, H. Hurwitz, A. Lent, and H.-P. Lung. On the Bayesian Approach to Image Reconstruction. *Information and Control*, 42:60–71, 1979.

[5] H. M. Iyer and P. B. Dawson. *Imaging volcanoes using teleseismic tomography*. Chapman and Hall, 1993.

[6] S. Kaczmarz. Angenäherte Auflösung von Systemen linearer Gleichungen. *Bulletin International de l'Académie Polonaise des Sciences et des Lettres*, 35:355–357, 1937.

[7] J. M. Lees. The magma system of Mount St. Helens: non-linear high-resolution P-wave tomography. *Journal of Volcanology and Geothermal Research*, 53:103–116, 1992.

[8] J. M. Lees. Seismic tomography of magmatic systems. *Journal of Volcanology and Geothermal Research*, 167(1-4):37–56, 2007.

[9] J. M. Lees and R. S. Crosson. Tomographic Inversion for Three-Dimensional Velocity Structure at Mount St. Helens Using Earthquake Data. *Journal of Geophysical Research*, 94(B5):5716–5728, 1989.

[10] J. M. Lees and R. S. Crosson. Bayesian Art versus Conjugate Gradient Methods in Tomographic Seismic Imaging: An Application at Mount St. Helens, Washington. *Institute of Mathematical Statistics*, 20:186–208, 1991.

[11] J. M. Lees, N. Symons, O. Chubarova, V. Gorelchik, and A. Ozerov. Tomographic images of kliuchevskoi volcano p-wave velocity. *AGU Monograph*, 172:293–302, 2007.

[12] J. M. Lees and H. Wu. P wave anisotropy, stress, and crack distribution at Coso geothermal field, California. *Journal of Geophysical Research*, 104(B8):17955–17973, 1999.

[13] J. M. Lees and H. Wu. Poisson's ratio and porosity at Coso geothermal area, California. *Journal of Volcanology and Geothermal Research*, 95(1-4):157–173, 2000.

[14] G. Mateos, I. D. Schizas, and G. B. Giannakis. Performance Analysis of the Consensus-Based Distributed LMS Algorithm. *EURASIP Journal on Advances in Signal Processing*, 2009, 2010.

[15] S. C. Moran, J. M. Lees, and S. D. Malone. P wave crustal velocity structure in the greater Mount Rainier area from local earthquake tomography. *Journal of Geophysical Research*, 104(B5):10775–10786, 1999.

[16] T. L. Murray and E. T. Endo. A real-time seismic-amplitude measurement system (rsam). volume 1966 of *USGS Bulletin*, pages 5–10. 1992.

[17] S. Ohmi and J. M. Lees. Three-dimensional P- and S-wave velocity structure below Unzen volcano. *Journal of Volcanology and Geothermal Research*, 65(1-2):1–26, 1995.

[18] R. A. Renaut. A Parallel Multisplitting Solution of the Least Squares Problem. *Numerical Linear Algebra with Applications*, 5(1):11–31, 1998.

[19] A. H. Sayed and C. G. Lopes. Distributed Recursive Least-Squares Strategies Over Adaptive Networks. In *Signals, Systems and Computers, 2006. ACSSC '06. Fortieth Asilomar Conference on*, pages 233–237, Nov. 2006.

[20] I. D. Schizas, G. Mateos, and G. B. Giannakis. Distributed LMS for consensus-based in-network adaptive processing. *IEEE Transactions on Signal Processing*, 57(6):2365–2382, 2009.

[21] L. Shi and W.-Z. Song. Distributed Least-Square Computing in Networks: A Survey and Comparison. `http://sensorweb.cs.gsu.edu/sites/default/files/publication/PDF/report/DistributedLS.pdf`.

[22] L. Shi, W.-Z. Song, M. Xu, Q. Xiao, J. M. Lee, and G. Xing. Imaging Seismic Tomography in Sensor Network. In *IEEE SECON*, 2013.

[23] R. Sleeman and T. 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*, 113(1-4):265–275, 1999.

[24] W.-Z. Song, R. Huang, M. Xu, A. Ma, B. Shirazi, and R. Lahusen. Air-dropped Sensor Network for Real-time High-fidelity Volcano Monitoring. In *The 7th Annual International Conference on Mobile Systems, Applications and Services (MobiSys)*, June 2009.

[25] H. Straková, W. N. Gansterer, and T. Zemen. Distributed QR factorization based on randomized algorithms. In *PPAM'11 Proceedings of the 9th international conference on Parallel Processing and Applied Mathematics - Volume Part I*, pages 235–244, 2012.

[26] R. Tan, G. Xing, J. Chen, W. Song, and R. Huang. Quality-driven Volcanic Earthquake Detection using Wireless Sensor Networks. In *The 31st IEEE Real-Time Systems Symposium (RTSS)*, San Diego, CA, USA, 2010.

[27] S.-Y. Tu and A. H. Sayed. Diffusion Strategies Outperform Consensus Strategies for Distributed Estimation over Adaptive Networks. *To appear in IEEE Transactions on Signal Processing*, 2012.

[28] A. L. Vesnaver, F. Accaino, G. Bohm, G. Madrussani, J. Pajchel, G. Rossi, and G. D. Moro. Time-lapse tomography. *Geophysics*, 68(3):815–823, 2003.

[29] G. P. Waite and S. C. Moranb. VP Structure of Mount St. Helens, Washington, USA, imaged with local earthquake tomography. *Journal of Volcanology and Geothermal Research*, 182(1-2):113–122, 2009.

[30] H. Wu and J. M. Lees. Attenuation structure of Coso geothermal area, California, from wave pulse widths. *Bulletin of the Seismological Society of America*, 86(5):1574–1590, Oct. 1996.

[31] H. Wu and J. M. Lees. Three-dimensional P and S wave velocity structures of the Coso Geothermal Area, California, from microseismic travel time data. *Journal of Geophysical Research*, 104(B6):13217–13233, 1999.

[32] L. T. Yang and R. P. Brent. Parallel MCGLS and ICGLS Methods for Least Squares Problems on Distributed Memory Architectures. *The Journal of Supercomputing*, 29(2):145–156, 2004.

[33] H. Zhang, C. Thurber, and P. Bedrosian. Joint inversion for Vp, Vs, and Vp/Vs at SAFOD, Parkfield, California. *Geochem. Geophys. Geosyst.*, 10(11):Q11002+, Nov. 2009.