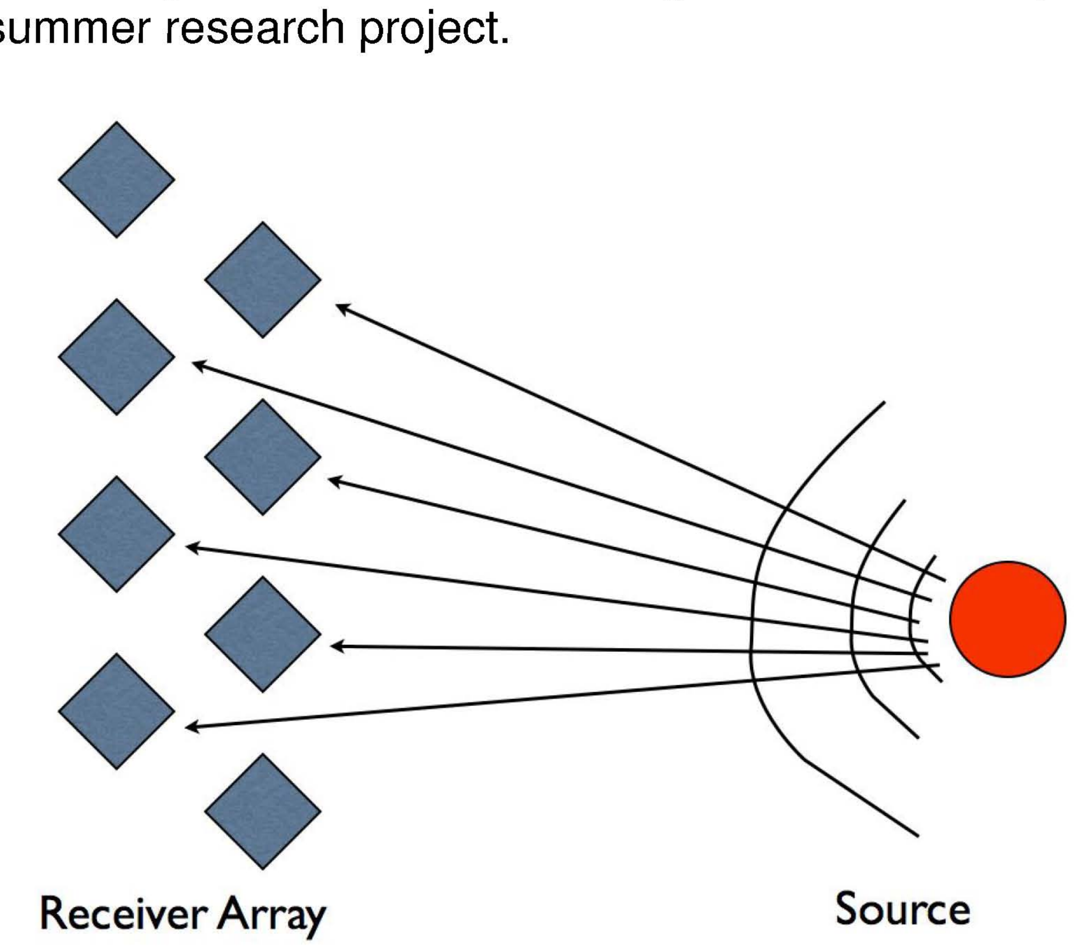
A microseismic simulation for algorithm testing

Michael P. Lamoureux*, Rituraj Shukla, Gary F. Margrave
Dept. Math & Stat, University of Calgary, mikel@ucalgary.ca

Overview

The goal was to build a small array of microphones, interfaced to a DSP unit, to perform data collection and analysis in a simulated microseismic experiment. Testing of the hardware and algorithms was completed in this short summer research project.

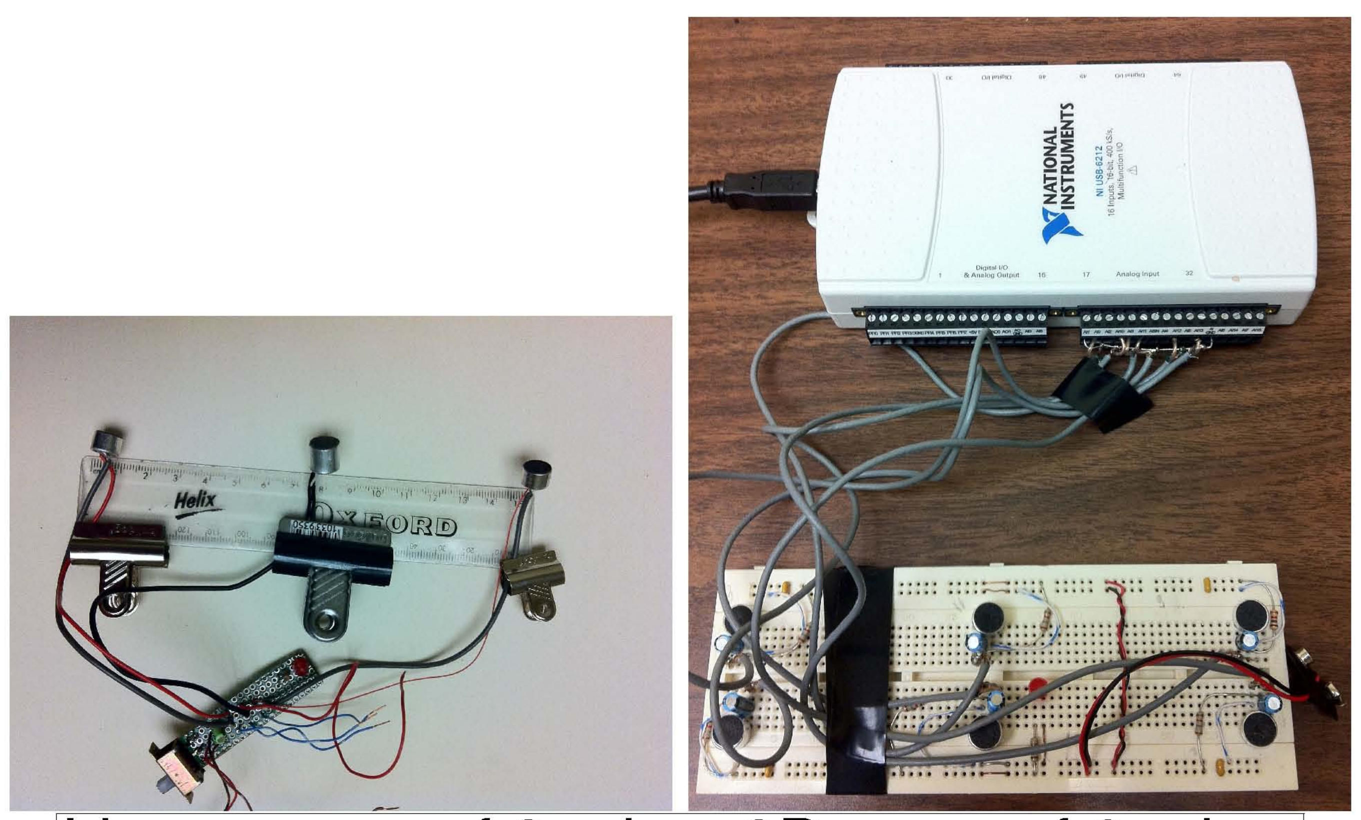


Receiver array and source.

A sound source (right) propagates a signal to an array of microphones, as a model for a microseismic source monitored by an array of geophones. The relative time delay between receivers is used to determine the source location.

Hardware

A National Instruments multichannel A to D converer (16 channels, up to 400 ksamples/sec) was connected to various arrays of small condenser microphones. 1D and 2D geometries were used, in a flexible arrangement.

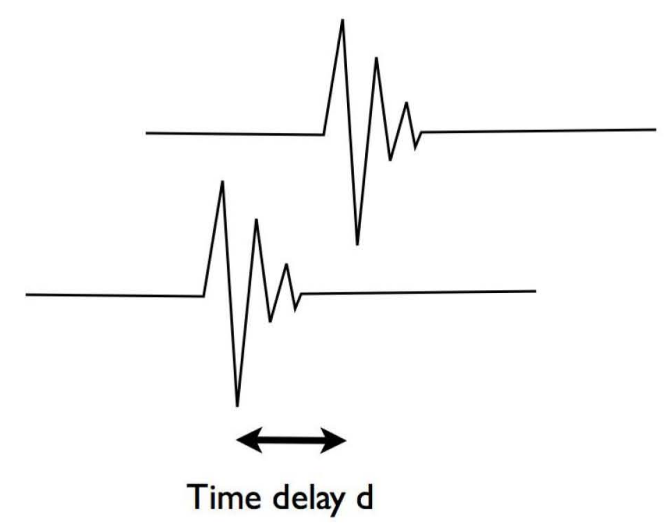


Linear array of 3 mics, 2D array of 6 mics.

The DSP unit was connected to a Windows PC through a USB cable, where data was be recorded and analysed in Matlab. It was also connected directly to the LabView virtual instrument environment provided by National Instruments.
Initial tests recording sound clicks in air (fingernail snaps, pen clip snaps) indicated a clean signal could be recorded in the desktop setting.

Algorithms

The first algorithms used cross-correlation of received signals from pairs of microphones to establish a relative time delay d between any two microphones.



Two received waveforms, time shifted by d.

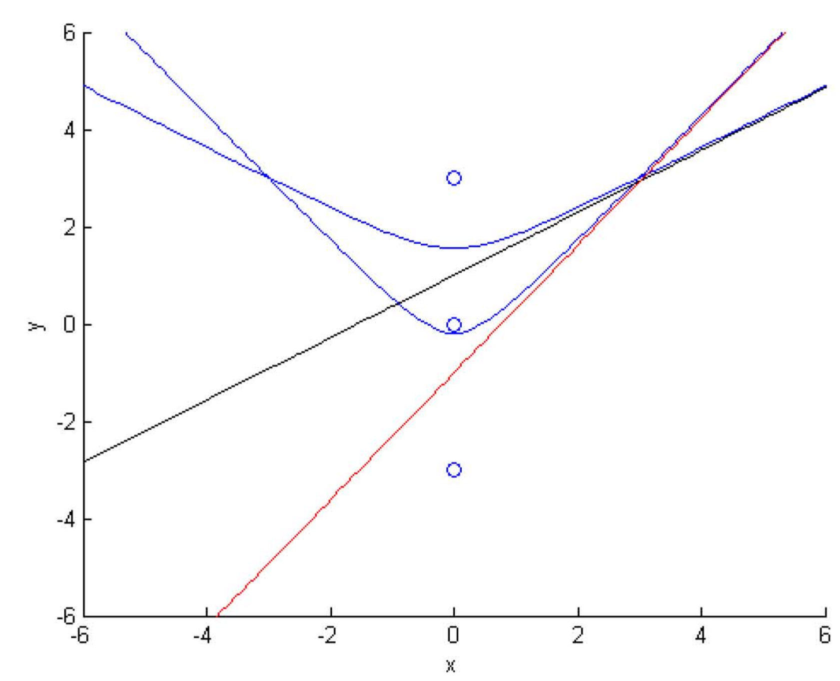
With microphones at points $(0,0)$ and $(0,s)$, a formula for the relative time delay for a source at (x,y) is give by

$$\sqrt{x^2+(y-s)^2}-\sqrt{x^2+y^2}=vd,$$

where v is the velocity of sound. This simplifies to a hyperbola:

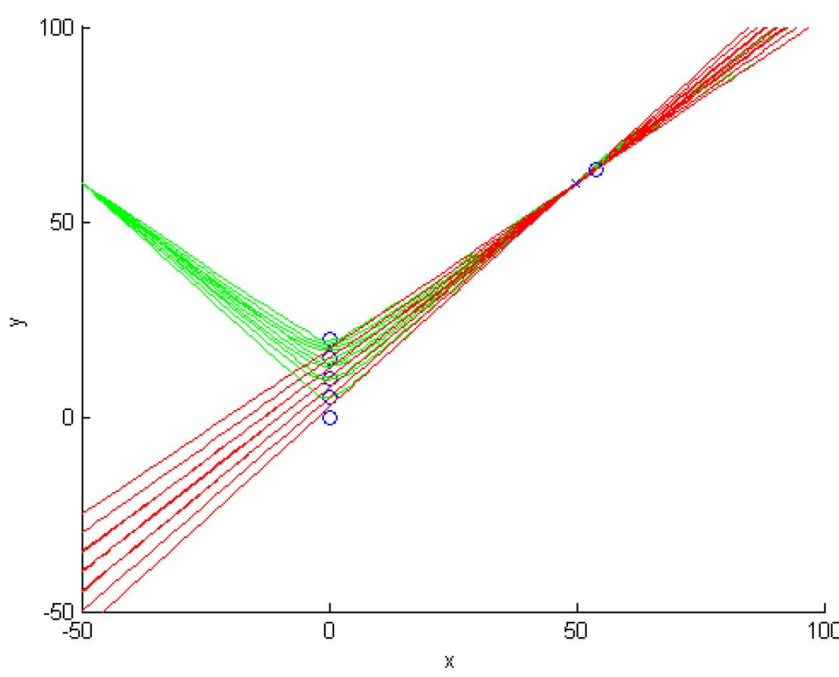
$$\frac{(y-s/2)^2}{(vd)^2}-\frac{x^2}{s^2-(vd)^2}=\frac{1}{4}.$$

Two receiver pairs gives two intersecting hyperbolas; linear asymptotes give an approximate solution for the source position (x,y):



Hyperbolas and linear asymptotes.

With many pairs, many hyperbolas and asymptotes are obtained. Solving the normal equations for the over-determined linear system of intersecting asymptotes gives an easy estimate for the location of the source.



Many mic pairs, many asymptotes.

Least squares

Source point (x,y) can be estimated by solving a least squares problem, with residual

$$r(x,y)=\sum_{i,j}\left(\sqrt{(x-x_i)^2+(y-y_i)^2}-\sqrt{(x-x_j)^2+(y-y_j)^2}-vd(i,j)\right)^2$$

where (x_i,y_i) are the locations of microphones, and $d(i,j)$ is the time delay between microphone pairs i,j.
A Gauss-Newton method quickly solves the minimization problem, and greatly increases the accuracy over the linear asymptote approximation. This method easily extends to 3D models and can be adapted for heterogeneous media.

PHAT method

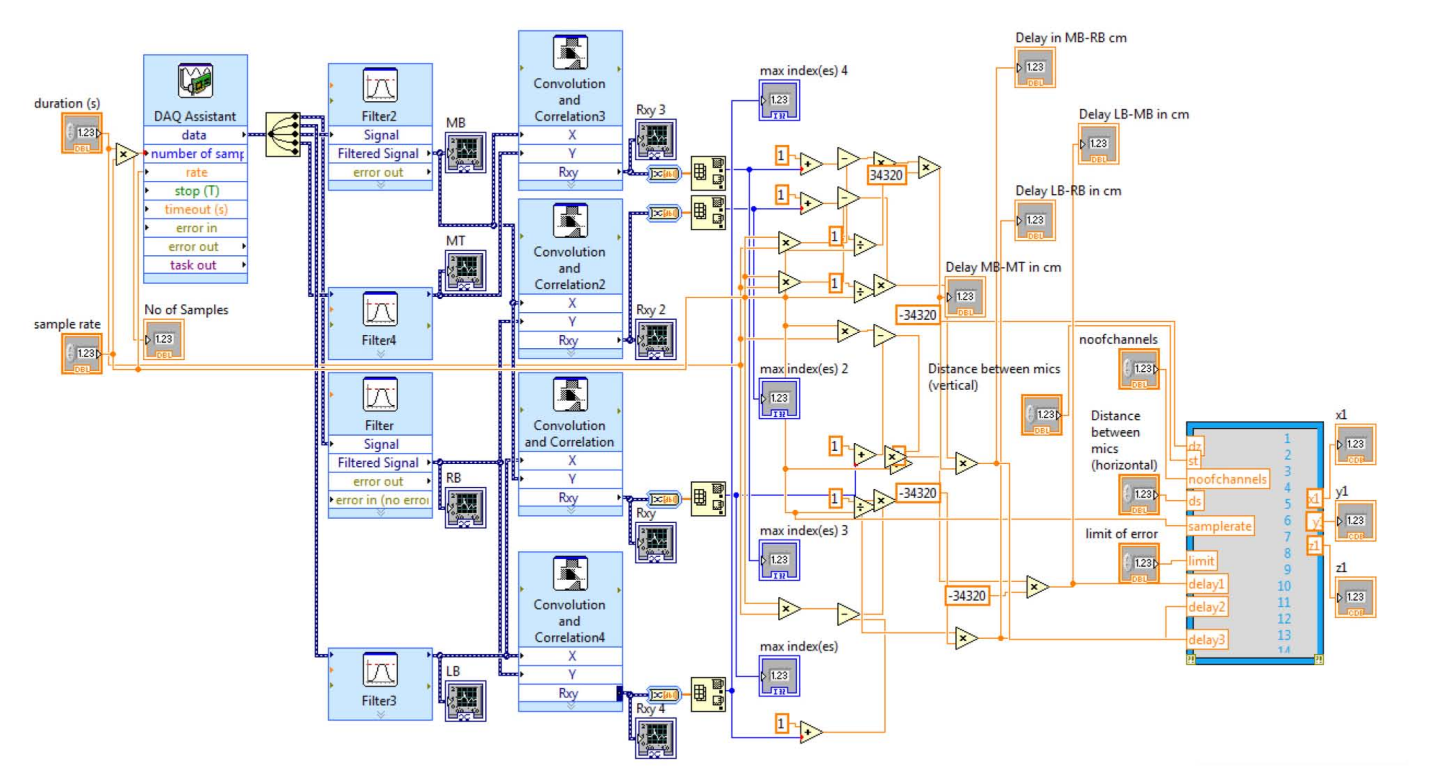
A literature review suggests the phase transform method for computing signal delay improves the accuracy of overall method. This method introduces weights to sharpen the peak of the cross correlation. The weighing function divides the cross spectrum with its own magnitude preserving only the phase information.

$$\begin{aligned}G_{s_1,s_2}&=FFT(crosscorrelation)\\X&=\frac{G_{s_1,s_2}}{|G_{s_1,s_2}|}\\C(d)&=FFT^{-1}(X)\\\text{delay } d&=\arg\max[C(d)]\end{aligned}$$

Our initial experiments indicate the PHAT methods performs significantly better than cross correlation in noisy environments.

LabView implementation

Nationsl Instruments provides an iconic programming language that samples and analyses data from the DSP unit. Our working algorithms were implement this way, as shown below.



Flow diagram of algorithm in LabView.

Bibliography

- Nocedal, J. & Wright, S. (1999) Numerical Optimization: Springer.
- Zhang, Y. & Abdulla, W. H. (2005) A comparative study of time-delay estimation techniques using mic arrays, University of Auckland.