

Physical modelling

Henry C. Bland, Joe Wong, Eric V. Gallant and Kevin W. Hall

ABSTRACT

CREWES has been developing and improving its ultrasonic physical modelling facility. This modelling facility is capable of simulating seismic surveys in 3 dimensions over scaled down earth models composed of plastics and metals. Unlike prior physical modelling systems at the U of C, this one uses high precision linear motors with a 0.1 μm optical measuring system to execute and measure the survey with great precision. In 2005, components for a new modelling system were purchased, and the new system was designed and major components were test-assembled. Progress has been made on three fronts: First, much of the necessary fabrication and assembly has been completed. Several hundred electrical connections have been wired, and a new circuit board was designed and assembled to indicate the status of the eight drive motors. Second, a series of programs have been written to intelligently operate the modelling system's drive motors based on a high-level seismic survey design. Simulation has proved useful in verifying correct operation of the software. Third, a change of source signal generation strategies has been proposed. By changing from an impulse source to a swept-frequency source we hope to improve the signal to noise ratio of physical modelling data. A numeric analysis of the swept-frequency source illustrates the benefit of this change.

INTRODUCTION

The physical modelling system is designed to carry out simulated seismic surveys on scaled earth-models that measure approximately 1m by 1 m and are up to (but typically less than) 1 m deep. These models are usually built to a 1:10000 scale, and are constructed of plastics and metals of various velocities. The modelling system uses piezoelectric transducers as both sources and receivers. Transducers are placed on the surface of the model at a variety of source/receiver locations, and an ultrasonic signal is transmitted into the physical model. The output of the receiver is captured by a high-speed analog-to-digital converter and stored to disk in a format identical to real-world seismic surveys (SEG-Y). Unlike in real-world surveys, however, it is difficult to deploy more than one receiver, so instead of using a spread of multiple receivers, we use a single receiver instead. It is as though one were to shoot a seismic survey with a single geophone. Obviously, with such low fold per shot (a shot fold of one), many more shots need to take place to complete the survey than usual. Fortunately, the source is non-destructive and highly repeatable, so this does not pose a problem.

Transceivers are moved about the physical model using a complex robotic manipulator system. The system is comprised of eight motors which are able to move about the source and receiver transducers over the surface of the model. A complete description of the linear motor system is contained in the 2005 CREWES Research Report by Gallant et al.

The modelling system has progressed on three fronts: First, much of the necessary fabrication and assembly has been completed. Second, a series of programs have been written to intelligently operate the modelling system's drive motors based on a high-level

seismic survey design. Third, we have changed to a swept-frequency source as a way to improve the signal to noise ratio from our ultrasonic transducers. An analysis of the benefit of a swept-frequency source has been performed. We shall discuss each of these three areas of progress in the following report.

1. SYSTEM ASSEMBLY

The modelling system is a complex piece of robotics. Each of the eight servo motors have a set of sensors (magnetic and optical) used for motor commutation and precise position control. They are also fitted with limit-of-movement sensors, and two sets of home-position sensors (course and fine). Motors are ultimately controlled by a computer running control software. Each motor is programmed with an acceleration/deceleration, and peak velocity curve. A PC-mounted motion-control card executes all the motor move commands as directed by the software. The motion control card emits control signals which operate the eight drive control units. It is these drive control units which precisely adjust the drive currents in the motors, making them move in the prescribed fashion. With many levels of components, and a multiplicity of motors, sensors, control lines, and status lines, final wiring of the system has been a significant challenge. We have completed wiring the power circuits, drive control circuits, and feedback circuits and are now ready to activate the motor portion of the system

In order to assist in the system debugging and to simplify the wiring, a custom circuit board was built to help organize all the motor limit sensor outputs (Figure 2). This board helps organize the incoming sensor signals by re-grouping them by function (rather than motor). The board is also fitted with a series of LED indicators so that the status of these limit and home sensors can be checked visually. A further enhancement to the modelling system is a pair of digital control panels on each arm (Figure 3.). These control panels provide crucial operator feedback while performing interactive positioning operations with the system (such as manually driving motors to benchmark positions on a model).

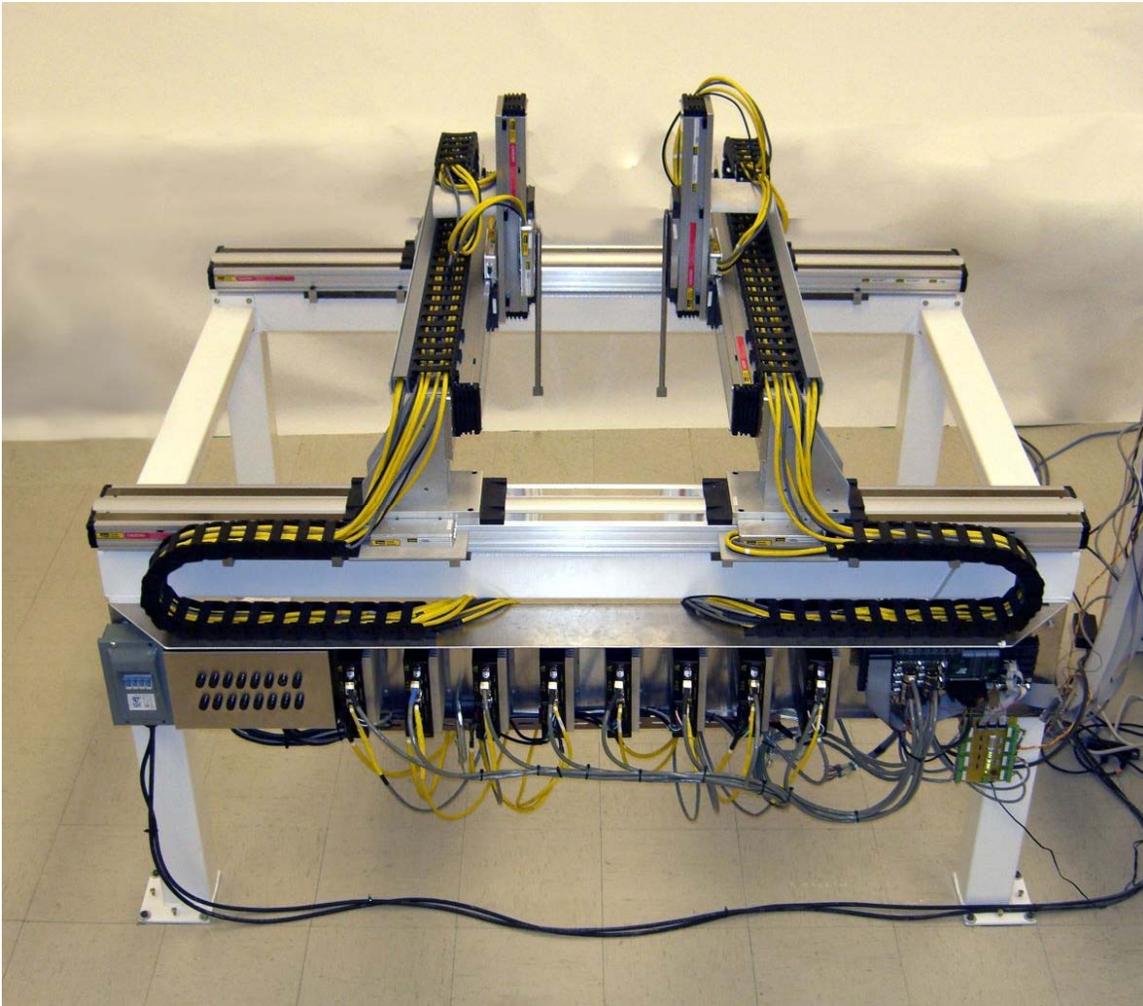


Figure 1. Modelling system

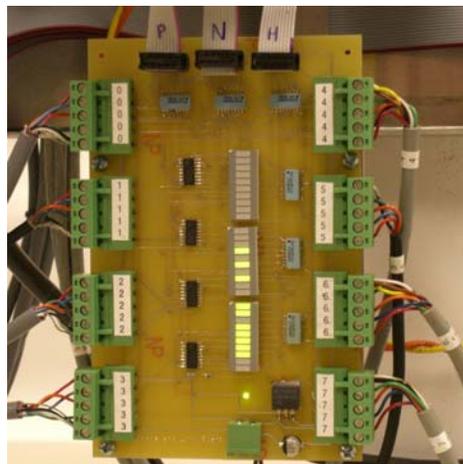


Figure 2. Limit and home routing and indicator board.

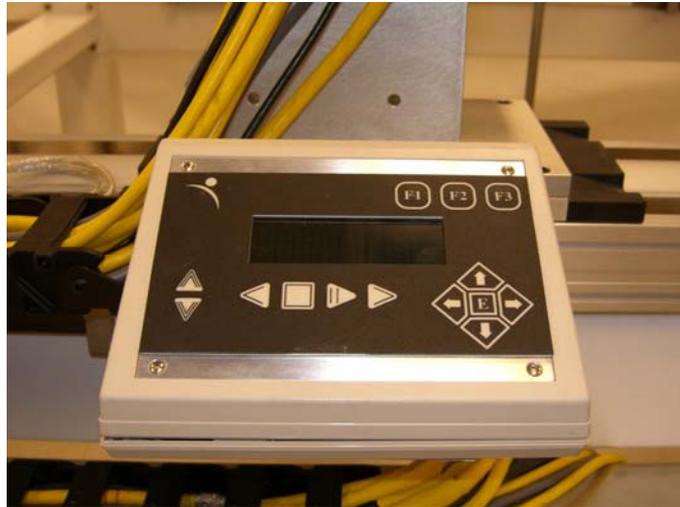


Figure 3. Operator control console

2. SOFTWARE

The modelling system uses sophisticated software to drive the motors that manipulate the transmitter/receiver transducers. The traces in a seismic survey can be considered as steps in a physical modelling survey: Each trace has a single source location and a single receiver location. The modelling software drives the source and receiver to the appropriate locations over the model, the transducers are lowered onto the model, and a signal is transmitted through the source while the received signal is recorded from the receiver. The signal is stored to disk and acquisition moves ahead to the next trace until all traces are acquired. Though this sounds relatively easy, it is in fact quite complicated: Since the source and receiver both move about the common area on the model, there is a very real danger of collision. The previous modelling system used a series of friction-driven cables to drive transducer carriages about the model. If collisions occurred, the friction drives would slip, and there was rarely serious damage. The new system uses direct-drive servo motors which are much more powerful. Motors on the new system are capable of causing serious damage if involved in a collision. To avoid any damage we wrote software which is aware of the true geometry of the motors and all attached appendages. Doing this, we can compute any projected motor moves, and hence foresee, and avoid, any possible collisions.

Control software for the modelling system has evolved over the years. Early versions of physical modelling software were written specifically for one kind of geometry. Any collisions had to be foreseen and collision avoidance was programmed manually for each case. The next iteration of software allowed for arbitrary survey geometries (Bland and MacDonald, 1999). It used a simplistic notion of collision avoidance by (1) assuming that all motors moved at exactly the same speed (2) modelling the transducers and their manipulator carriages as having a box-shaped footprint and (3) assuming that the model was much smaller than the modelling system (allowing plenty of room for collision bypass moves). This software occasionally ran into difficulties because of its non-rigorous collision avoidance strategies. It was also unable to shoot near-offset shots

where the box-shaped model of the transducer footprint was overly pessimistic about the space required to avoid a collision.

The new control software performs rigorous, time-variant collision detection. All motor moves are tested for collisions, both at the start/end points and while motors are in transit. The latest version of software is fully aware of the geometry of the components of the jig, the geometry of the model under test, and the range of motion of all motors. It can therefore intelligently plan how to shoot a model without having transducers collide with each other, with parts of the model, or parts of the modelling system. It has strategies to reach source/receiver locations that are tricky to reach.

Complex motion planning

One way of managing the different motors of the physical modelling system is to consider it a robot and use the well established mathematics for robot control. Robots are typically defined as a “kinematic tree” of interconnected joints (Figure. 4). Formulation of our modelling system as a kinematic tree allows us to mathematically describe its state using a series of matrix equations. For every source/receiver pair that needs shooting, a motion planning algorithm attempts to move the system from its initial state (one trace) to its desired state (the next trace). The motion planning algorithm must not only ensure that the final state doesn’t cause any collisions, but that any traverses that the motors make will also be collision free. The problem becomes a four dimensional one, as we must consider time as well as space to prevent collisions: Occasionally the traverse to a new source/receiver location places the two transducers on a collision course, even though the starting and ending points don’t collide. In most cases, careful ordering of motor moves prevents collisions. If, however, motors are moved in the wrong order, collisions are often the result.

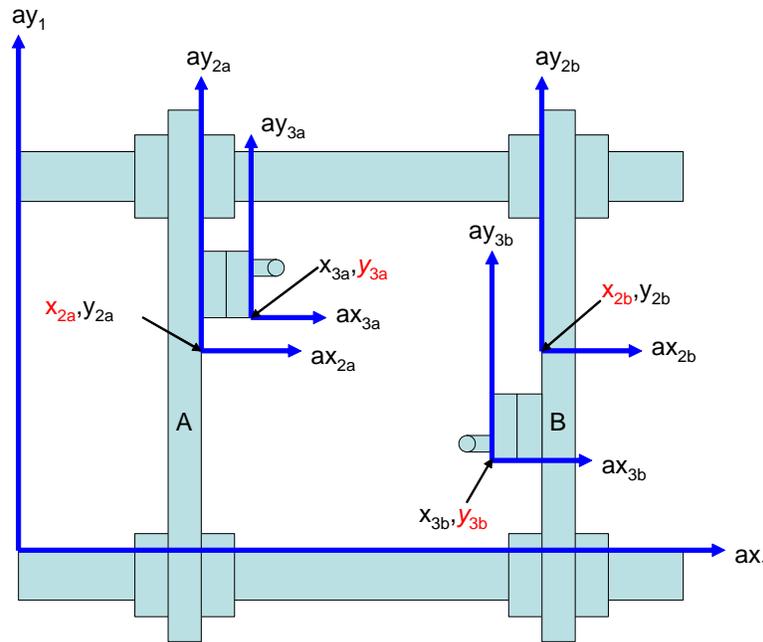


Figure 4. Top-down view of the modelling system in schematic form showing transformation axes (starting with “a”) and measurement reference points. Measurements in red vary with motor movement.

The processing flow

Our modelling system control software is broken down into a series of small programs which each perform a single task. Breaking down the software into small modules keeps the software more manageable and makes it easier to test. Executing a physical modelling survey thus becomes a processing flow, where the system input is a high-level seismic survey design, and the output is a SEG-Y format data file. Let us consider all the intermediate steps in the modelling processing flow:

- Generate survey design
- Transform the survey design into a field trace list
- Generate a model orientation file (MOF)
- Scale the survey from field coordinates to lab coordinates
- Partition the traces into three configuration groups
- Optimize the trace order
- Plan all the motor moves
- Execute the motor moves while acquiring SEG-Y data

1. Generate a survey

A survey design is built using a program like GEDCO’s OMNI or ARAM’s Aries field acquisition software. The survey design is stored in a Project File (PRF file). The project file contains the high-level definition of the survey, in field coordinates (several meters between geophone stations). The project file specifies the source/receiver coordinates and shot patterns (the set of live geophones for each shot). Figure 5 shows a typical project definition file for a 3-D seismic survey.

```

----- Patch Definitions -----
Receiver Flags
Receiver Line      From      To      By
PATDEF 1
PAT                2          2101    2110    1
PAT                4          4101    4110    1
PAT                6          6101    6110    1
PAT                8          8101    8110    1
PAT                10         10101   10110   1
----- Source Point Allocations -----
Source Flags
Source Line      From      To      By
USEPAT 1
SP               1          1101    1109    1
SP               3          3101    3109    1
SP               5          5101    5109    1
SP               7          7101    7109    1
SP               9          9101    9109    1
SP              11         11101   11109    1
----- Receiver Coordinates -----
Receiver Flags      From (Coordinates)      To
Receiver Line      From      To      By      Easting  Northing  Easting  Northing
RXY 2              2101    2110    1          450      400      0        400
RXY 4              4101    4110    1          450      300      0        300
RXY 6              6101    6110    1          450      200      0        200
RXY 8              8101    8110    1          450      100      0        100
RXY 10             10101   10110   1          450      0        0         0
----- Source Coordinates -----
Source      Source      Source      From      (Coordinates)      To
Source      Line      From      To      By      Easting  Northing  Easting  Northing
SXY 1              1101    1109    1          450      400      450      0
SXY 3              3101    3109    1          350      400      350      0
SXY 5              5101    5109    1          250      400      250      0
SXY 7              7101    7109    1          150      400      150      0
SXY 9              9101    9109    1           50      400      50       0
SXY 11             11101   11109   1          -50      400     -50      0

```

Figure 5. Project definition file generated by ARAM Aries software. This file provides a high-level definition of a 3-D seismic survey.

2. Expand the survey design into a list of explicit source/receiver traces

The survey design file is first transformed into a list of traces. Each trace has a source point and a receiver point specified. At this point, all coordinates are still *field* coordinates. An example of a field trace list (FTL) file is shown in Figure 6.

```

TRC      1 S (1,1101) R (2,2101) SXYZ (450,400,0) RXYZ (450,400,0)
TRC      2 S (1,1101) R (2,2102) SXYZ (450,400,0) RXYZ (400,400,0)
TRC      3 S (1,1101) R (2,2103) SXYZ (450,400,0) RXYZ (350,400,0)
TRC      4 S (1,1101) R (2,2104) SXYZ (450,400,0) RXYZ (300,400,0)
TRC      5 S (1,1101) R (2,2105) SXYZ (450,400,0) RXYZ (250,400,0)
TRC      6 S (1,1101) R (2,2106) SXYZ (450,400,0) RXYZ (200,400,0)
....
TRC     2695 S (11,11109) R (10,10105) SXYZ (-50,0,0) RXYZ (250,0,0)
TRC     2696 S (11,11109) R (10,10106) SXYZ (-50,0,0) RXYZ (200,0,0)
TRC     2697 S (11,11109) R (10,10107) SXYZ (-50,0,0) RXYZ (150,0,0)
TRC     2698 S (11,11109) R (10,10108) SXYZ (-50,0,0) RXYZ (100,0,0)
TRC     2699 S (11,11109) R (10,10109) SXYZ (-50,0,0) RXYZ (50,0,0)
TRC     2700 S (11,11109) R (10,10110) SXYZ (-50,0,0) RXYZ (0,0,0)

```

Figure 6. Field Trace List file

3. Generate a model orientation file (MOF file)

The model is next placed into the modelling system and the location of the model is registered relative to the origin of the modelling system. The registration is performed interactively by positioning a beam-mounted laser over two different benchmarks. The modelling system knows exactly where the beams point, so it can obtain the coordinates of two benchmarks very precisely. Benchmarks are typically two corners of a physical model. One benchmark is used to tie field coordinates to lab coordinates. The other benchmark is used as an azimuthal reference, so that any skew between the model and the jig can be measured and accounted for. The output of this registration operation is a model orientation file (MOF). It contains the coordinates of the two benchmarks (in “lab” coordinates) and the desired tie-point to the “field” coordinate system. The “lab” and “field” coordinate systems are usually related by an arbitrary translation, rotation, and a fixed scale factor of 1:10000.

Item	Lab X (m)	Lab Y (m)	Lab Z (m)	Field X (m)	Field Y (m)	Field Z (m)
BENCHMK1	0.21233	0.251312	0.12	0	0	0
BENCHMK2	0.21503	0.451311	0.12	0	2000	0

Figure 7. Model orientation file

4. Scale the survey from field coordinates to lab coordinates

We next convert the survey from field coordinates to lab coordinates using the field2lab program. This program reads the *Field Trace List* (FTL) file and the model orientation file (MOF) and produces a Lab Trace List (LTL) file. The lab trace list file applies the necessary rotation, translation, and scaling to the field source/receiver coordinates to generate a set coordinates in the lab coordinate space. All data present in the FTL file is carried into the LTL file so that the original field coordinates may be used to populate SEG-Y trace headers.

5. Partition the traces into three configuration groups

The modelling system has one source and one receiver transducer – each attached to its own beam. Since one beam is unable to reach beyond the other beam, the two transducers must be electrically or physically exchanged to get around this limitation. The traces, in “lab” coordinates are separated into two groups: the Source On Right (SOR) group, and the Source On the Left (SOL) group. In addition, certain source/receiver pairs (as created by the design program) are unshootable in any configuration due to overlapping source/receiver locations or out-of-bounds surface locations. The program reads the Lab Trace List (LTL) and Model Orientation Files (MOF) and outputs a new Lab Trace List where traces are tagged as SOR (source on right), SOL (source on left), or NS (for no shot).

6. Optimization

In order to acquire the survey in a reasonable amount of time, motor movement can be minimized by re-sequencing the acquisition order of traces. Rather than acquire traces in the order prescribed by the survey design program, we re-arrange the acquisition order to minimize the distance traveled by the transducers. Optimization of the two configurations

(SOE and SOW) is performed independently. The optimization program reads a lab trace list file (LTL) and outputs a new lab trace list file with traces (lines) re-sequenced in a more optimal shooting order.

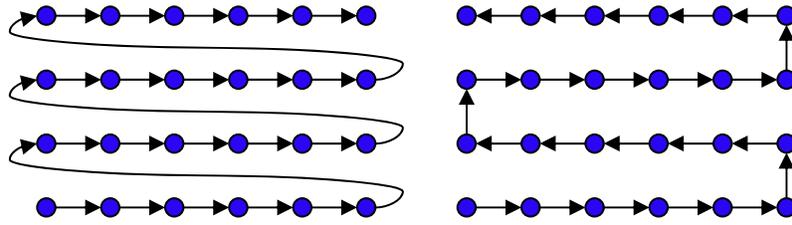


Figure 8. Optimization of shooting order. The distance traveled by traversing a set of receiver locations in their design order (left) is shortened by re-sequencing into a more optimal shooting order (right).

Optimization of the shooting order takes into account the distance traveled by both source and receiver transducers. This is a classic computer science problem known as the *travelling salesman problem (TSP)*. In the TSP, a salesman wants to visit all of his customers once, traveling the least distance possible. The distance between customers is the only information known. This problem is tricky to solve. For n customers, there are $n!$ possible paths from which to choose. In the case of the physical modelling system, each source/receiver pair (or seismic trace) can be considered one city. Even for a simple 3-D survey, n is large (10^3 to 10^9) and an ideal solution becomes computationally impossible.

There are many algorithms find an approximation to the ideal TSP solution. One of these is the nearest-neighbor algorithm. The algorithm starts at one city, and then advances to the next nearest city until all cities have been visited. The nearest neighbor algorithm is easy to implement and executes quickly, though it sometimes misses shorter routes which are easily noticed by human insight. Still, it turns out that this algorithm is “adequately optimal” for our purposes. Even with this optimization, dealing with values of n in the thousands is still very time-consuming. To reduce processing time, we partition the problem into several equally-sized sub-problems. The chain of sub-problem solutions, though not optimal, is still significantly faster than executing the survey with no optimization at all.

Motion Planning

Having obtained the sequence of traces to shoot, the motors must be instructed to drive to each source/receiver location without collisions. The motion planner reads a lab trace list file (LTL) and generates a motor move file (MMF). This file contains the sequence of motor moves necessary to carry out the acquisition. Like all other programs in the processing flow, it carries forward the book-keeping information about the traces (such as field coordinates, line and, station numbers) so that informative trace headers may be created as each shot is stored to disk.

Our current motion planning algorithm tries to run through all the source/receiver locations in the prescribed sequence. It computes the path (a volume of space) need for each transducer to be moved to its next location. Should the paths collide with each

other, or if one of the paths collides with the final position of the other transducer then a series of collision-evasion steps are taken. The first collision-evasion step is to see if moving transducers “one at a time”, or “one coordinate a time” will prevent the collision. If that does not work, then a series of intermediate “side steps” are introduced into the path. These side-steps often help avoid collisions. New paths with side-steps are tested one at a time to see if they are collision free. The side steps get increasingly more complex (varying in size and direction) until a collision-free path is found. If no collision free path is found, multiple side-steps are attempted by recursively calling the side-step algorithm. If, after a great deal of computation, no solution is found, the trace is dropped, and the planner moves on to the next trace.

Since correct operation of the motor control software is critical for avoiding damage to the system, testing has been taking place in simulation. Using computer animation we can see how the modelling system executes a variety of complex acquisition geometries. These simulations show that the collision-avoidance algorithm works well and that survey optimization saves a significant amount of acquisition time.

3. SWEEP FREQUENCY MODELLING

Models used in the physical modelling system are typically scaled by a factor of 1:10000; that is, a real world dimension of 10 meters is represented by 1 mm in the scale model. Wavelengths must be scaled in the same way. A wavelength of 40 meters in the real world must be represented by a wavelength of 4 mm. Frequencies must be scaled inversely as wavelengths if the velocities of real rocks and that of model materials are almost the same. For example, assume that the velocities of both rocks and model materials are both 4000 m /s. Then a real world wavelength of 40 meters means a real world frequency of 100 Hz. In the scale model, the wavelength must be 4 mm, and the corresponding frequency must be 1 MHz.

Such a large scaling factor leads to some gross approximations in using conventional small piezoelectric elements to represent real sources and detectors. If we use commonly-available piezoelectric cylinders of 1 cm diameter in the model, they would scale up to represent sources and detectors with an unrealistic surface footprint of 100 meters. Even if we reduce the piezoelectric diameters down to 1 mm, this still represents a footprint of 10 meter. When we reduce the diameters of the transmitting and detecting piezoelectric elements down to 1 mm, the areas of contact are reduced so much that we face large reductions in coupling the source energy from the transmitting element into the model and then out of the model into the sensing element. If we use a standard pulse technique with very small piezoelectric elements in our modelling system, we find that our desired signals (especially those from reflecting surfaces) are lost in system noise. The trade-off is between getting a good representation of seismic footprint and acquiring model data with adequate signal-to noise ratios.

We can improve the signal-to-noise ratios (SNR) of our model seismograms if we use frequency sweeps and crosscorrelation for acquisition. The vibrational energy generated by a piezoelectric cylinder driven by a single pulse with root-mean-square (RMS) voltage V_{RMS} for a duration of t seconds is proportional to $V_{\text{RMS}} * t$. The total energy produced by the same element when driven by a sinusoidal voltage with the same RMS voltage

over a much longer time T in a single pulse is proportional to $V_{\text{RMS}} * T$. If we then collapse (by some mathematical process) all that energy into a short time equivalent to t , we get an impulse-like signal with energy enhanced by a factor approximately proportional to T/t and an increase in SNR. This is essentially what happens in vibratory-source land-seismic acquisition, where the seismic source is driven for a long time with a frequency sweep of nominally constant power, and impulsive seismograms of high SNR are obtained by collapsing all the energy into a much shorter time through convolution. We can use exactly the same swept-frequency technique to increase SNR in our modelling data, and to improve our representation of seismic footprint. Alternatives to the vibratory-source frequency sweep exist (for example, the maximal length sequence PRBS discussed by Wong and Stewart, 2006, in this volume), but here we will limit discussion to the swept-frequency case.

Frequency Sweeps

A frequency sweep is a signal whose frequency changes monotonically with time. A linear sweep is defined by the equation

$$S_w = \sin (2\pi * [f_0 + (f_1 - f_0) / T * t] * t) , \quad (1)$$

where f_0 and f_1 are the frequency limits of the sweep, T is the time duration of the sweep, and t is the time. If f_1 is greater than f_0 , then we have an up-sweep (or chirp); if f_1 is less than f_0 we have a down-sweep (or whistle).

Figure 9A shows the first 200 microseconds of a frequency sweep with unity amplitude and a duration of 512 microseconds. The sweep starts at frequency .0625 MHz, and the frequency increases linearly with time until it reaches a final value of 1 MHz. Figure 9B is a delayed version of the sweep. Figure 9C is the crosscorrelation of Figure 9A and Figure 9B. Note that, other than the side lobes, the autocorrelation looks very much like a delta or impulse function. Figure 9D is a seismic wavelet, and Figure 9E is the convolution of the seismic wavelet with the delayed sweep in Figure 9A. Finally, Figure 9F is the crosscorrelation of Figure 9D with Figure 9E, recovering a filtered delayed version of the original wavelet. If we digitally deconvolve Figure 9E with Figure 9D, we would recover a higher fidelity representation of the original wavelet.

Figure 10 indicates the results of increasing the bandwidth of the sweep at low frequencies. In Figure 10A, we see the initial part of a frequency sweep with unity amplitude and a duration of 512 microseconds. The sweep starts at frequency 0.0 MHz, and the frequency increases linearly with time until it reaches a final value of 1 MHz. Figure 10B shows its autocorrelation, simulating an impulse function. Figure 10D is the same seismic wavelet shown in Figure 9D, and Figure 10E is the convolution of the seismic wavelet with the sweep of Figure 10A. Figure 10F is the crosscorrelation of Figure 10E with Figure 10A, and is higher-fidelity version of the original wavelet than shown in Figure 9E.

Figure 11 shows the noise rejection feature of crosscorrelation. Figure 11D shows random noise (incoherent with respect to the frequency sweep) added to the wavelet of Figure 9D giving a SNR of about 1. The wavelet is totally obscured by the noise. Figure 11E is the crosscorrelation of the noise-free signal 5D with the swept-frequency pilot

Figure 11B, plus the same noise that was added to give Figure 11D. The original wavelet appears reasonably well above the noise, whereas in Figure 11D, the wavelet is not discernible at all.

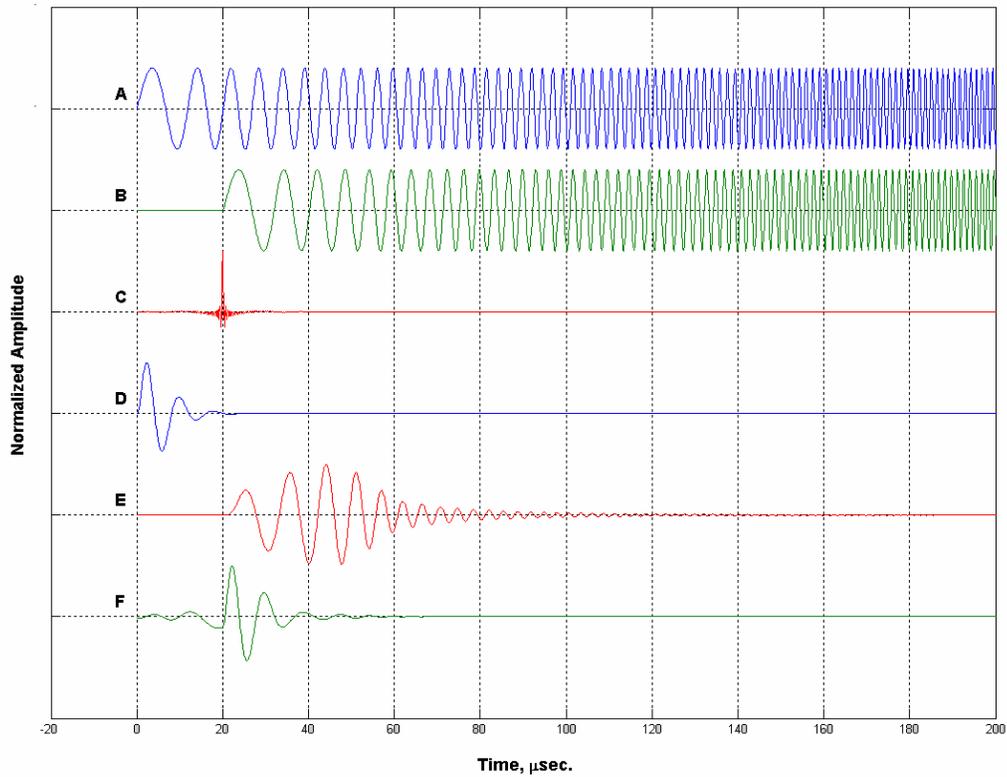


Figure 9. Example of frequency sweep acquisition. (A) Frequency sweep, with a start frequency of 0.0625 MHz, end frequency of 1.0 MHz, duration of 512 μ sec. (B) Sweep A delayed by 20 μ sec. (C) Crosscorrelation of sweep A and sweep B. (D) A seismic wavelet. (E) Convolution of seismic wavelet D with sweep B. (F) Cross-correlation of A and E, recovering a filtered delayed version of the wavelet.

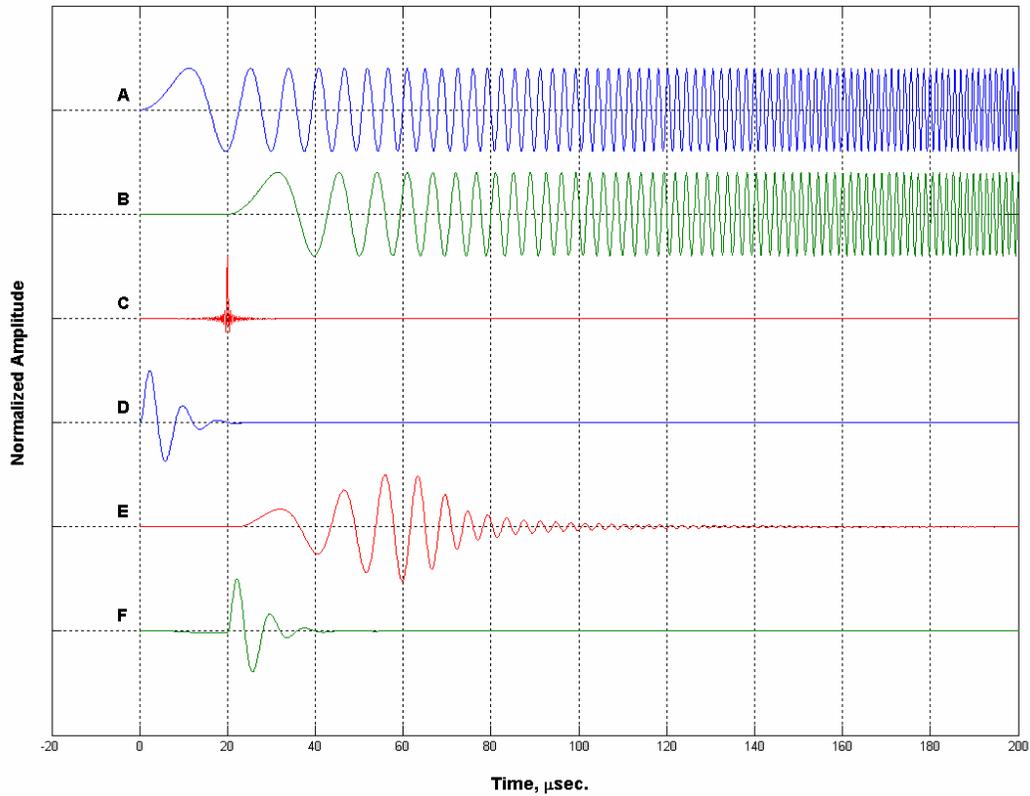


Figure 10. Example of frequency sweep acquisition. (A) Frequency sweep, with start frequency = 0.0 MHz, end frequency = 1.0 MHz, duration=512 μ sec. (B) Sweep A delayed by 20 μ sec. (C) Crosscorrelation of sweep A and sweep B. (D) A seismic wavelet. (E) Convolution of seismic wavelet D with sweep B. (F) Crosscorrelation of A and E. Compared to the result in Figure 9F, this is a better version of the wavelet. The improvement is due to the presence of lower frequency components in the sweep.

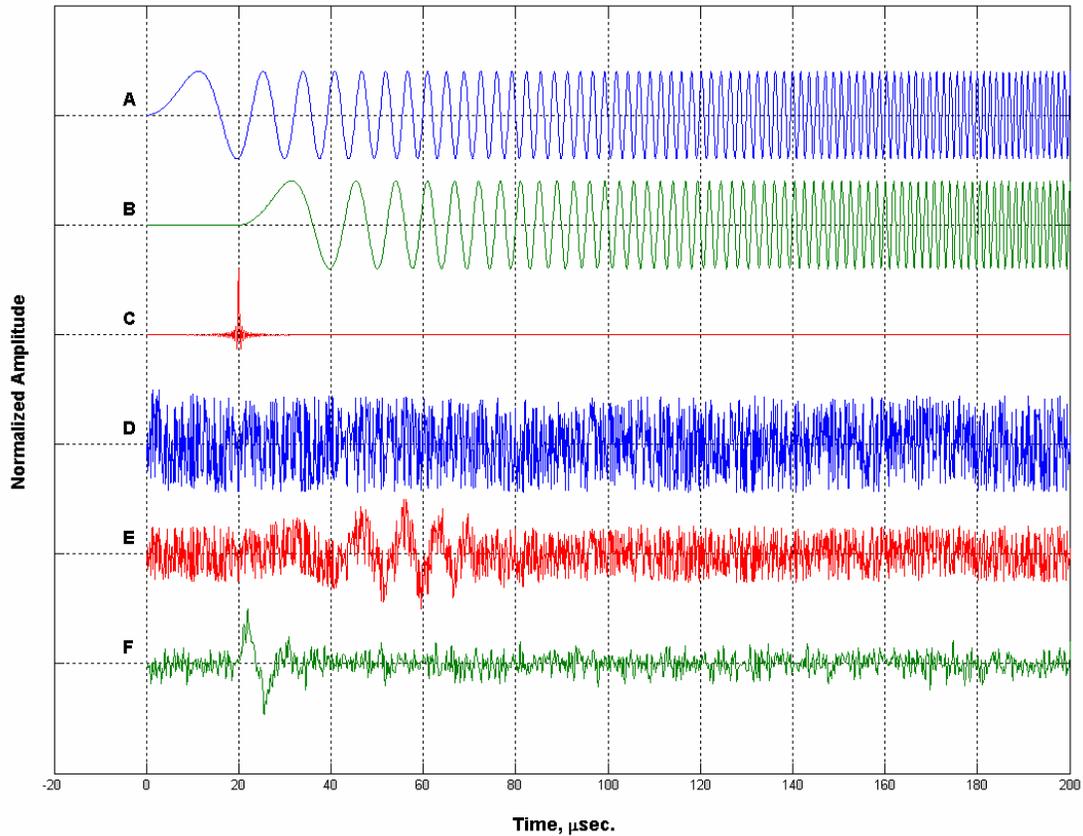


Figure 11. Noise rejection feature of crosscorrelation. (A) Frequency sweep, with start frequency=0.0 kHz, end frequency=1.0 MHz, duration=512 μsec . (B) Sweep A delayed by 20 μsec . (C) Crosscorrelation of sweep A and sweep B. (D) A seismic wavelet plus noise. The wavelet is lost in the noise. (E) Convolution of clean seismic wavelet in Figure 9D with sweep B, plus noise. (F) Crosscorrelation of A and E. The seismic arrival is discernible above the noise.

DISCUSSION

The report details much of the work which has gone into the new physical modelling system. We hope to have it functional in the near future and look forward to producing some interesting and informative modelling datasets.

ACKNOWLEDGEMENTS

We wish to thank Andy Read, Colin Branner, and the U of C Science Workshop technicians for their technical assistance.

We are grateful to GEDCO Corp. and Aram Systems Ltd. for supplying 3-D survey design software.

REFERENCES

- Bland, H.C., and MacDonald, P. R., 1999, Software for physical modelling survey design and acquisition: CREWES Research Report, **11**
- Gallant, E.F., Bland, H.C., Bertram, M.B., Lawton, D.C., The new CREWES/FRP seismic modelling system, an update: CREWES Research Report, Volume **17**
- LaValle, S.M., 2006, Planning algorithms: Cambridge University Press
- Wong, J., and Stewart, R.R., 2006. Development of a vibratory source for shallow reverse VSP applications: CREWES Research Report, this volume.