

## Nonlinear pairwise alignment of seismic traces

Christopher L. Liner and Robert G. Clapp<sup>1</sup>

### ABSTRACT

Alignment of seismic traces is a recurring need in seismic processing and interpretation. For global alignment via static shift there are robust tools available, including cross correlation. However, another kind of alignment problem arises in applications as diverse as associating synthetic seismograms to field data, harmonizing P-wave and mode converted data, final multilevel flattening of common image gathers, and so on. These cases require combinations of trace compression, extension, and shift - all of which are time variant. The difficulty is to find a mapping between the traces which is in some sense optimum. This problem is solved here using a modified form of the Needleman-Wunsch algorithm, which was originally developed for amino acid sequence alignment in proteins. Applied to seismic traces, this global optimization algorithm provides a nonlinear mapping of one seismic trace onto another. The method extends to alignment of any number of traces since that problem can be broken down into a cascade of pairwise alignments. The Needleman-Wunsch algorithm is discussed, extended to the seismic case, and applied to field data. The results show a promising new tool for nonlinear alignment or flattening of seismic data.

### INTRODUCTION

This paper is concerned with the process of aligning two seismic traces and, by repeated application, any number of seismic traces. At first appearance this is a trivial problem. One need only compute the cross correlation of the two traces and, from the peak of this function, the optimum alignment is known.

But this is merely the best alignment via static shift. In a complex data set, such as a CMP gather, the human eye can associate events that a global correlation alignment will not honor. What the eye is able to do is a time-variant, nonlinear association of events. To carry out this alignment process requires the determination and application of a nonlinear mapping between the trace samples – a combination of compression, stretching, and translation, all varying with time.

If it were possible to do this kind of alignment, what would be the use of it? In a sense, it is already done everyday in seismic processing. A collection of traces are analyzed for a set

---

<sup>1</sup>email: cll@utulsa.edu, Department of Geosciences, University of Tulsa;  
bob@sep.stanford.edu

of coefficients which drive a nonlinear stretch to make all events flat at all times. This process is, of course, normal moveout. But NMO is a model-driven process, with the model being the NMO equation. A general nonlinear trace alignment algorithm would make it possible to flatten all events in a CMP gather with no knowledge of the NMO equation. We are not advocating such a procedure, but making a point. There is value in using the NMO equation to flatten events, including the fact that it leaves multiples non-flat and therefore removable (at least partially). However, a the general alignment algorithm may be useful as a final flattening procedure for any type of gather (CMP, CIG, CAG, etc.). It could also have application in alignment of synthetic seismograms with field data, associating events on P-P and P-Sv data cubes, etc. In short it would be a useful, general utility.

While the literature on trace interpolation and estimation of missing data is vast, there is very little published work on nonlinear trace alignment. To our knowledge the only published work directly on point with the current study is Martinson et al. (1982) and a derivative paper Martinson and Hopper (1992). In the second work, an iterative, linear inverse approach is used to determine a set of coefficients describing a mapping function which relates features on one trace with those on another. The process is driven by maximizing the correlation or coherence between the modified traces, and used as a trace interpolation technique. This method is similar in spirit to our approach, but owing to the use of linear inverse theory it tends to be expensive, sensitive to the starting model, and does not guarantee a global solution.

Our solution to the pairwise trace alignment problem borrows a concept and algorithm from computational biology and modifies it to the seismic case. The concept is pairwise alignment of amino acid sequences, and the algorithm is due to Needleman and Wunsch (1970).

## METHODOLOGY

### **Needleman-Wunsch algorithm**

The Needleman-Wunsch (NW) algorithm (Needleman and Wunsch, 1970) is a nonlinear global optimization method that was developed for amino acid sequence alignment in proteins. This was the first of many important alignment techniques which now find application in the Human Genome Project.

Human DNA consists of some 30,000 genes which are in turn composed of 20 amino acids represented by letters of a reduced alphabet (ADCEFGHILKMNPQRSTVWY). The total genome is composed of about 3 billion chemical base pairs, or about 100,000 per gene. Finding where a particular string of amino acids fits is an optimization problem that aims to find the optimal alignment of the two strings with respect to a defined set of rules and parameter values for comparing different alignments.

The algorithm is an iterative method in which all possible pairs of amino acids (one from each string) are set up in a 2D matrix and alignments are represented as pathways through this array. The optimum alignment is the path (or paths) connecting maximum scoring values. This approach is an example of dynamic programming, which has also been applied to seismic

modeling (Darby and Neidell, 1966) and travel time computation (Schneider et al., 1992).

It is a global optimization process which yields a solution to the problem of pairwise alignment, meaning that we are interested in finding the best fit between only two strings. If alignment of more than two strings is of interest, the problem can be broken down into a cascade of pairwise alignments and thus solved.

In its simplest form, the Needleman-Wunsch algorithm can be summarized by Figure 1. A matrix is formed by placing the two strings, possibly of different length, along the left column and top row. In this step a one is allocated to a cell in the matrix if the letter in each list at this location is the same. Otherwise no entry is made (which is a defacto zero). It is at this stage that the letter-alignment problem becomes purely numerical. In fact, the original string could just as easily consist of integers as letters. The result of this process is the similarity matrix in Figure 1a.

From the similarity matrix a scoring matrix is formed beginning in the lower right corner. The procedure is to add the score value to the maximum value in a row-column pair whose upper left corner is down and to the right of the current working position. Thus in Figure 1b the similarity value 1 is added to the maximum value in the blackened cells (also 1) to give a score of 2. Figure 1c is a later stage of the computation, which continues up and to the left until every cell has been visited and the scoring matrix is complete, Figure 1d. In this simple form, a final score corresponds to how many character matches exist in the optimum alignment.

The final step (traceback) operates by starting at the highest score value (8 in this case) and determining the maximum score path by moving to the right, down, or diagonally down and to the right, Figure 1e. The fact that more than one 8 score alignment exists (Figure 1f) is an expression of non-uniqueness. An important aspect of the solution is that in the process of finding the best global alignment, we also find the best alignments of any sublength.

### Details of the algorithm

We now introduce a more flexible form to the Needleman-Wunsch algorithm (Karp, 2000). Let the two input strings,  $(x, y)$ , given by

$$x = (x_1, x_2, \dots, x_i, \dots, x_m) \quad (1)$$

$$y = (y_1, y_2, \dots, y_j, \dots, y_n) \quad (2)$$

where  $(m, n)$  need not be equal. The subscript  $i$  denotes the row direction, and  $j$  denotes columns. We write the scoring function  $V(i, j)$  as the equation set

$$V(i, j) = \max [G(i, j), F(i, j), E(i, j)] \quad (3)$$

$$G(i, j) = \sigma(x_i, y_j) + V(i + 1, j + 1) \quad (4)$$

$$F(i, j) = -(p + q) + \max [V(i + 1, j), F(i + 1, j) + p] \quad (5)$$

$$E(i, j) = -(p + q) + \max [V(i, j + 1), E(i, j + 1) + p] \quad (6)$$

$$V(i, n + 1) = -(p + (m - i + 1) q) \quad (7)$$

$$V(m + 1, j) = -(p + (n - j + 1) q) \quad , \quad (8)$$

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1						1	
C			1					1		1			
K													
C			1					1		1			
R						1						1	
B		1											
P													1

(a)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						1						1	
C			1					1		1			
K													
C			1					1		1			
R						1						2	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(b)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	1												
Y					1								
C			1					1		1			
Y					1								
N				1									
R						5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(c)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
Y	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
Y	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

(d)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
Y	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
Y	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

ABCNY-RQCLCR-PM  
 | | | | | | | | | | | | | |  
 AYC-YNR-CKCRBP-

(e)

	A	B	C	N	Y	R	Q	C	L	C	R	P	M
A	8	7	6	6	5	4	4	3	3	2	1	0	0
Y	7	7	6	6	6	4	4	3	3	2	1	0	0
C	6	6	7	6	5	4	4	4	3	3	1	0	0
Y	6	6	6	5	6	4	4	3	3	2	1	0	0
N	5	5	5	6	5	4	4	3	3	2	1	0	0
R	4	4	4	4	4	5	4	3	3	2	2	0	0
C	3	3	4	3	3	3	3	4	3	3	1	0	0
K	3	3	3	3	3	3	3	3	3	2	1	0	0
C	2	2	3	2	2	2	2	3	2	3	1	0	0
R	2	1	1	1	1	2	1	1	1	1	2	0	0
B	1	2	1	1	1	1	1	1	1	1	1	0	0
P	0	0	0	0	0	0	0	0	0	0	0	1	0

ABC-NYRQCLCR-PM  
 | | | | | | | | | | | | | |  
 AYCYN-R-CKCRBP-

(f)

Figure 1: The Needleman-Wunsch algorithm yields the globally optimum alignment between two strings, one along the left of the matrix and the other across the top. (a) Similarity matrix. (b) and (c) partially complete score matrix. (d) Complete score matrix. (e) Traceback route giving globally optimum alignment. (f) Alternate alignment illustrating non-uniqueness.

where the indices range  $i \leq m$  and  $j \leq n$ . In this form, the function  $\sigma(x_i, y_j)$  is the similarity matrix and it is calculated on the fly rather than precomputed. Further it can be customized to reflect different weights associated with matches and mismatches. In biological applications an element on each string either matches or does not, and this fact is represented in the choice of a similarity measure, for example

$$\sigma(x_i, y_j) = \begin{cases} \sigma(a, a) = +1 \\ \sigma(-, a) = 0 \\ \sigma(a, -) = 0 \end{cases} \quad (9)$$

in which a value of 1 is awarded for a match, and all other cases are awarded 0.

In the seismic case we do not expect or need an exact amplitude match, rather it is important to reward small amplitude differences and penalize large ones. We capture this idea in a similarity function as

$$\sigma(x_i, y_j) = c - abs[t_1(x_i) - t_2(y_j)], \quad (10)$$

where  $c$  is a constant chosen to keep  $\sigma(x_i, y_j) > 0$  and  $abs[]$  is the absolute value.

Computational complexity and cost of this algorithm applied to two strings of length  $n$  and  $m$  is  $O(n * m)$ .

### Modification to the seismic case

On first consideration, the NW algorithm seems ill-suited to the seismic case, primarily because seismic amplitude data is continuous, not discrete. To utilize the machinery of the NW algorithm we consider the histogram of data amplitudes on the trace pair and form a set of bins. That is to say, all of the floating point amplitudes are partitioned into a small number of intervals (20 in the examples given below), and the similarity matrix is formed by the equation 10 operating on the binned amplitudes. An important aspect of the algorithm is that a global optimum alignment function is found independent of the similarity measure that is used. To test feasibility, we used a one point similarity that captures amplitude differences. However we could easily have worked with a twopoint measure to emphasize slope similarity, or three points to match curvature. Extending this idea, one could work with short window cross correlations to fill the similarity matrix similar (Martinson and Hopper, 1992). Clearly, any of these more ambitious similarity measures would increase the cost of the algorithm. In any case, the NW algorithm guarantees a global optimum alignment solution using any similarity matrix as input.

As a final comment, we note that any number of traces that require alignment can be processed as a cascaded series of pairwise problems. Thus there is no loss of generality in discussing just the pairwise problem.

## EXAMPLES

To test the methodology we started with a simple synthetic Normal Moveout (NMO) gather with some random noise (Figure 2). We selected two traces some distance apart (the first and tenth trace) and applied the algorithm. The two traces can be seen in the left part of Figure 3. Note the time-variant alignment error.

Figure 2: A synthetic shot gather. The first and tenth trace were selected to test the algorithm.

`bob2-nmo.gather` [ER]

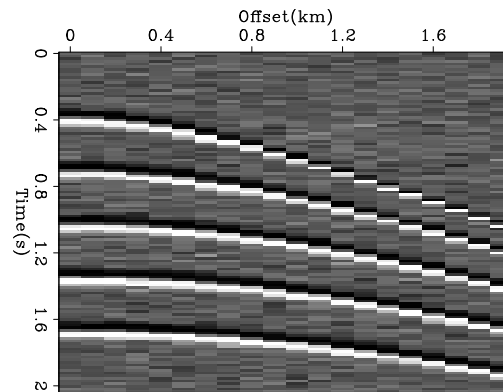
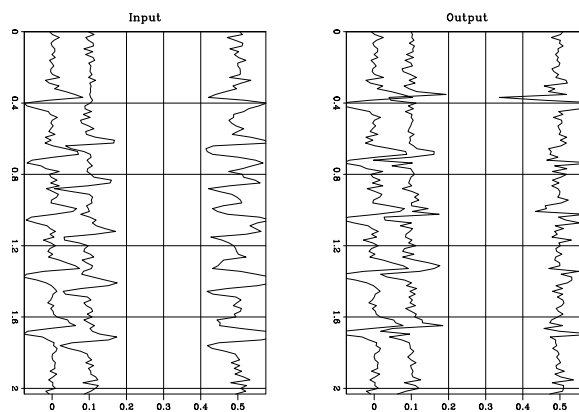


Figure 4 shows the similarity (left) and score (right) matrices. The black lines in the similarity matrix represent low scores and correspond to the events in the data, they only disappear (or match) when encountering another wavelet. The score matrix shows exactly what we expect to see. A slightly non-diagonal maximum (except for edge effects at low times corresponding to a lack of coherent events). Figure 3 shows the input (left) and output (right) along with their corresponding differences. The output is much better aligned and the overall differences reduced. The difference trace is a proxy for gauging the quality of alignment, but the goal is not to drive this difference to zero. The algorithm keys on strong events whose alignment may result in sizable differences at other levels. This is a significant departure from Martinson and Hopper (1992) who minimize a difference measure to determine alignment.

Figure 3: The left plot shows the two input traces and the right plot the traces after alignment. The third trace in each display shows the difference.

`bob2-nmo.in-out` [ER]



For a second test we chose a common reflection point (CRP) gather from a 2-D marine dataset (Figure 5). The gather is an angle gather (Prucha et al., 1999; Sava and Fomel, 2000) after phase-shift plus-interpolation (PSPI) migration (Gazdag and Sguazzero, 1985). Note that

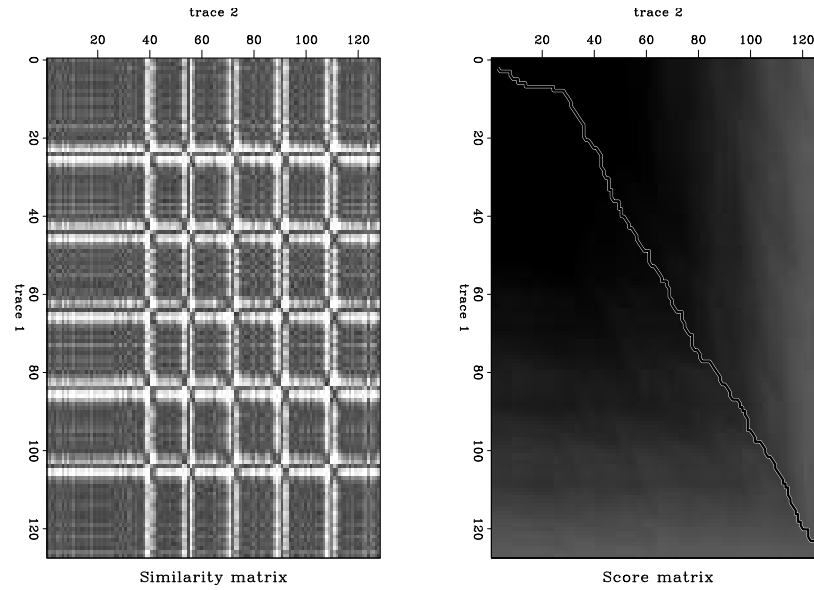


Figure 4: The right plot is the score matrix using first two traces from Figure 2 and the left panel is the similarity matrix. Axes labels refer to time sample numbers (not seconds). `bob2-nmo.score-sim` [ER]

we still see some residual moveout in the angle gather. The left panel of Figure 7 shows the input two traces (third and sixteenth).

After running the algorithm we obtained the score and similarity matrices seen in Figure 6. Note how the structure of the similarity matrix to the previous example (Figure 4). The score matrix and the corresponding maximum has the shape that we would anticipate. It is generally diagonal with some deviations. The output two traces appear to be better aligned (the right panel of Figure 7), but the difference isn't as reduced as we would hope. Our belief is this caused by a poor stretching algorithm.

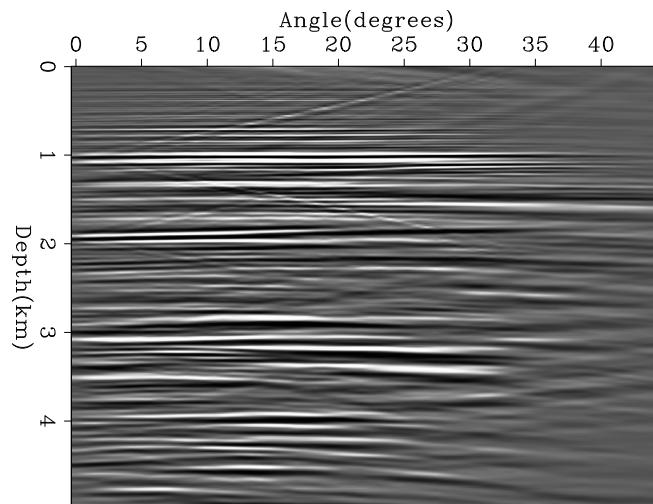


Figure 5: The CRP gather used for trace alignment. The third and 16th trace were used. `bob2-big.gather` [ER]

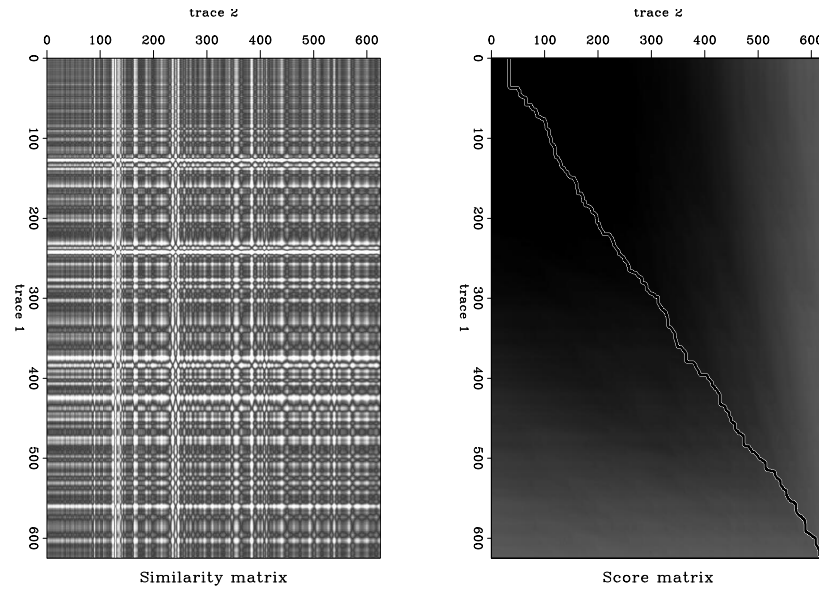
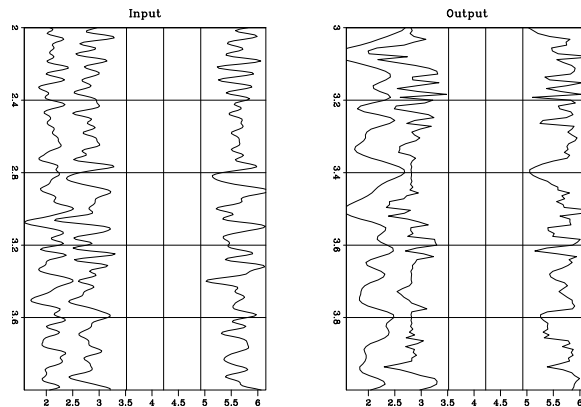


Figure 6: The right plot is the score matrix using two traces from a CRP gather and the left panel is the similarity matrix. Axes labels refer to time sample numbers (not seconds).

`bob2-big.score-sim` [ER]

Figure 7: The left plot shows the two input traces and the right plot the traces after alignment in the window from three to four seconds. The third trace in each display shows the difference. `bob2-big.in-out` [ER]





## CONCLUSIONS

We have shown that a robust and efficient algorithm originally developed for protein sequence alignment can be applied to the pairwise alignment of seismic traces. This has been demonstrated, in its simplest form, by application to synthetic and real seismic data.

With further work, this approach may supply a general tool for nonlinear alignment of seismic traces for use in processing and interpretation.

## ACKNOWLEDGMENTS

The first author would like to thank J. Claerbout and B. Biondo for the opportunity to visit SEP during May and June of 2002.

## REFERENCES

- Darby, E. K., and Neidell, N. S., 1966, Application of dynamic programming to the problem of plane-wave propagation in a layered medium: *Geophysics*, **31**, 869–883.
- Gazdag, J., and Sguazzero, P., 1985, Migration of seismic data by phase shift plus interpolation, *in* Gardner, G. H. F., Ed., *Migration of seismic data: Society Of Exploration Geophysicists*, 323–330.
- Karp, R. M. Cs 294-7 lecture notes: <http://www.icsi.berkeley.edu/epxing/lecture7.pdf>, 2000.
- Martinson, D. G., and Hopper, J. R., 1992, Nonlinear seismic trace interpolation: *Geophysics*, **57**, 136–145.
- Martinson, D. G., Menke, W., and Hopper, J. R., 1982, An inverse approach to signal correlation: *Journal of Geophysics Research*, **87**, 4807–4818.
- Needleman, S., and Wunsch, C., 1970, A general method applicable to the search for similarities in the amino acid sequence of two proteins: *J. Mol. Biol.*, **48**, 443–453.
- Prucha, M. L., Biondi, B. L., and Symes, W. W., 1999, Angle-domain common image gathers by wave-equation migration: *SEP-100*, 101–112.
- Sava, P., and Fomel, S., 2000, Angle-gathers by Fourier Transform: *SEP-103*, 119–130.
- Schneider W. A., J., Ranzinger, K., and Balch, A., 1992, A dynamic programming approach to first arrival traveltimes computation in media with arbitrarily distributed velocities: *Geophysics*, **57**, 39–50.

