Application of dynamic programming to the correlation of paleoclimate records

— Source link ↗

Lorraine E. Lisiecki, Philip A. Lisiecki

Institutions: Brown University

Related papers:

- A Pliocene-Pleistocene stack of 57 globally distributed benthic δ18O records

- A long-term numerical solution for the insolation quantities of the Earth

- A Depth-Derived Pleistocene Age-Model: Uncertainty Estimates, Sedimentation Variability, and Nonlinear Climate Change

- Macintosh Program performs time-series analysis

- Pleistocene evolution: Northern hemisphere ice sheets and North Atlantic Ocean

Application of dynamic programming to the correlation of paleoclimate records

Lorraine E. Lisiecki

Department of Geological Sciences, Brown University, Providence, Rhode Island, USA

Philip A. Lisiecki

Quincy, Massachusetts, USA

[1] Signal matching is a powerful tool frequently used in paleoclimate research, but it is enormously time-consuming when performed by hand. Previously proposed automatic correlation techniques require very good initial fits to find the correct alignment of two records. A new technique presented in this paper utilizes dynamic programming to find the globally optimal alignment of two records. Geological realism is instilled in the solution through the definition of penalty functions for undesirable behavior such as unlikely changes in accumulation rate. Examples with synthetic and real data demonstrate that the dynamic programming technique produces accurate, high-resolution results with much less effort than hand tuning or preexisting automated correlation techniques.   *INDEX TERMS:* 4267 Oceanography: General: Paleoceanography; 4294 Oceanography: General: Instruments and techniques; 4299 Oceanography: General: General or miscellaneous; *KEYWORDS:* signal correlation, dynamic programming, wiggle matching, composite depth

Citation: Lisiecki, L. E., and P. A. Lisiecki, Application of dynamic programming to the correlation of paleoclimate records, *Paleoceanography*, *17*(4), 1049, doi:10.1029/2001PA000733, 2002.

1. Introduction

[2] Paleoclimate reconstructions are based largely on the interpretation of diverse climate proxy records. Chrono-stratigraphic correlation among records is often accomplished by matching signals between climate proxies and orbital parameters or between multiple climate proxies. Signal matching is typically performed by eye in a tedious and time-consuming process [*Shackleton et al.*, 1995; *Prell et al.*, 1986]. Previous attempts to automate signal correlation [*Martinson et al.*, 1982] and orbital tuning [*Brügge-man*, 1992; *Yu and Ding*, 1998] can only find locally optimized solutions and hence require good initial alignments. In this paper we present a new technique based on dynamic programming that overcomes this problem and others encountered by previous techniques. Our dynamic programming software and instructions for its use are available for download under the "Data & Software" link at http://pixie.geo.brown.edu/esh/paleo.html.

[3] A good automatic correlation technique should be easy to use, versatile, and at least as accurate as results produced by hand. The produced alignment should be physically realistic and globally optimal. In the context of signal matching, physical realism requires a monotonically increasing mapping function to preserve the sequence of events in each series and reasonable implied accumulation rates. Techniques that can only find locally optimal solutions might simply force each peak in one signal to fit to the nearest peak in the other signal, requiring the user to provide an accurate initial guess of the peaks' correlations. However, an algorithm that compares all possible alignments of two series to find the globally optimal one eliminates the need to produce a good initial guess and the potential error that accompanies such a guess. Another important gauge of a technique's performance is its ability to match signals with the complexities frequently encountered in paleoclimate proxies such as noise, gaps, mismatches, and changes in accumulation rate. We demonstrate that our dynamic programming technique can produce physically realistic, globally optimal solutions for the wide variety of signals encountered in paleoclimate research.

[4] Dynamic programming describes a class of algorithms, typically used in optimization problems, that divide one problem into many subproblems and store the solutions of these subproblems in a table to avoid duplicate computation. The technique has been used effectively in other areas of geology, e.g., in pore space analysis [*Eggleston and Peirce*, 1995] and in the correlation of geologic strata [*Waterman and Raymond*, 1987], but to our knowledge this is the first time it has been applied effectively to paleoclimate data. The dynamic programming algorithm that we developed divides each series into many small intervals and calculates an alignment score for each possible mapping of these intervals. The technique constrains the sequential ordering of the intervals, ensuring that the derivative of the matching function is not negative. The score of each mapping is primarily determined by the square of the difference between the two signals. Penalty functions added to the alignment scores allow the user to minimize accu-

mulation rate changes or to define other criteria for determining an optimal alignment. The evaluation of a wide range of mapping functions precludes the algorithm from becoming trapped in a local solution. No initial guess of a fit is required, and the starting and ending points of the two series may be different.

[5] In the following section we review other proposed automatic signal correlation techniques. The description of our procedure begins with a simple example to which dynamic programming may be applied and goes on to explain how this example is altered to best suit the problem of correlating paleoclimate records. We also describe the techniques that we find effective for the assignment of penalty functions and parameter values. The results section contains a comparison of our technique with hand-tuned data and an evaluation of its ability to align an artificially distorted signal. We demonstrate the algorithm's ability to match a variety of signals, including those with noise, gaps, and abrupt changes in accumulation rates. Finally, we explore some of the technique's potential applications and discuss its limitations.

2. Background

[6] One previous approach to inverse signal correlation [*Martinson et al.*, 1982] constructs a mapping function from a linear trend modified by a truncated Fourier series. The Fourier coefficients are adjusted according to the gradient of the curves' correlation. This technique requires a very good initial guess because it is susceptible to finding local maxima in correlation. *Martinson et al.* [1982] suggest that, for some data, fitting the low frequency component of the signal may produce a good initial guess for the fit of the entire signal, but this technique may not work well if significant sedimentation rate changes occur in the records. *Shure and Chave* [1984] observe that the truncated Fourier series produces "artificial, high-frequency fluctuations in the mapping functions." They suggest a parabolic spline parameterization, but *Martinson et al.* [1984] argue that splines make the algorithm more likely to find local solutions. The dynamic programming technique that we propose is superior to these techniques because it is not susceptible to finding local solutions and requires no initial guess.

[7] Most subsequent research has focused on finding age-depth (as opposed to depth-depth) relations through orbital tuning. In orbital tuning a data record is matched to an orbital forcing function that has a known age relation. Complications with orbital tuning include the selection of a forcing curve, unknown lag times, nonlinear climate responses, and noise. Generally, the climate signal and forcing function are fairly dissimilar. One common orbital tuning technique is to use bandpass filters on proxy records to compare their orbital frequency components with the same frequency components of the forcing function [e.g., *Imbrie et al.*, 1984; *Lourens et al.*, 1996]. However, uncertainty in the age-depth relation makes it difficult to define a filter that consistently preserves the orbital frequency without also including significant power outside of the desired frequency range [*Martinson et al.*, 1987]. The automated correlation technique presented in this paper does not perform orbital tuning in part because of these added complexities, but we present here a description of two automated orbital-tuning techniques that might also be applied to signal correlation.

[8] *Yu and Ding* [1998] propose an automated orbital tuning technique that maximizes the correlation of the 21 kyr and 41 kyr components of a signal with precession and obliquity curves respectively using a modified version of dynamic programming [*Råde and Westergren*, 1995]. The start and end ages of the signal are fixed, and one-at-a-time the age of each interior point is assigned the age between its two neighbors' ages that maximizes the curves' correlation coefficients while all of the other points in the series remain fixed. The process is iterated until the correlation coefficients are maximized. *Yu and Ding* [1998] allow only one point to move at a time because they must be able to calculate the filtered component of the signal at every step. Therefore, their algorithm no longer possesses all of the properties of dynamic programming; it requires a good initial guess and may not find the globally optimal solution of each problem, especially if accumulation rates change significantly at some point in the series. Additionally, the technique places no constraints on implied accumulation rates.

[9] *Brüggeman* [1992] proposes an automated tuning technique in which a parameterized frequency response function is tuned to fit the climate record to a forcing function. The algorithm uses a conjugate gradient method to adjust the time nodes of the data and the parameters of the frequency response function to optimize six penalty functions defined by (1) the difference between the data and the model, (2) the deviation from the initial guess, (3) implied accumulation rate changes, (4) estimated ages of dated points, (5) the behavior of the frequency response function, and (6) the sign of the derivative of the age-depth function. Our dynamic programming technique can incorporate penalties similar to all of these except point 5 because no frequency response model is used and point 6 because the derivative of the age-depth function is inherently positive. *Brüggeman*'s [1992] technique differs from our dynamic programming algorithm in that it does not discretize accumulation rates and the solution that it finds using the conjugate gradient method may be only a local minimum in the penalty functions.

[10] Previously developed dynamic programming techniques for the correlation of geologic strata [*Waterman and Raymond*, 1987] are similar to our algorithm, but they match defined geologic strata based on properties such as composition and thickness rather than signals that have not been divided into strata. The evenly spaced intervals into which our algorithm divides signals act in many ways like the strata in *Waterman and Raymond*'s [1987] technique; however, our algorithm compares a sequence of unevenly spaced data points within each interval rather than the bulk values typical of a single stratum. *Waterman and Raymond*'s [1987] technique is less sophisticated in its consideration of the relative sedimentation rates of the two stratigraphic sequences being matched but includes a more elegant treatment of stratigraphic gaps than our current algorithm. We discuss our

algorithm's treatment of gaps and plans for its improvement in the section on paleoclimate applications.

3.   Procedure
3.1.   Simple Example

[11]   First, let us consider an example problem, illustrated in Figure 1, in which we match all of the n data points in a series A to a subset of the m points in the series B so as to minimize the square of their differences. The only constraint we impose is that the sequence of points in A must be preserved. One way to solve this problem is to construct a table, as shown in Figure 1b, in which each column represents a point in series A, and each row represents a point in series B. Each table entry is a "score" equal to the square of the difference between the point in series A represented by that column and the point in series B given by that row. For example, the intersection of the third column with the second row has a score of 0.25, which is the square of the difference between the third point in A, 1.5, and the second point in B, 1.0. Because the score represents the difference between the two signals, smaller scores represent better fits. The best alignment of series A can be found by assigning one point in series B (row) to each point in series A (column) such that the sum of the scores is minimized and such that the sequence of points in A is preserved (i.e., no point in series A [column] is assigned to a point in series B [row] that precedes the point in B [row] assigned to the previous point in A [column]).

[12]   The structure of this problem makes it especially well suited to solution by dynamic programming. Matches of subsets of series A to subsets of series B serve as subproblems whose solutions are stored in a dynamic programming table. In the following description of our technique, we focus on the modifications to the standard technique of dynamic programming appropriate for application to stratigraphic alignment. A thorough discussion of the theory and techniques of dynamic programming is given by *Cormen et al.* [1990].

3.2.   Modifications for Stratigraphic Alignment

[13]   In the case of stratigraphic alignment, we want to make several modifications to the simple example above. We want to constrain an optimal solution's accumulation rate to realistic values, allow points in series A to fall between points in series B, and calculate a score by comparing points in both A and B with their corresponding, linearly interpolated values from the other series. We accomplish all of these goals by dividing each series into several hundred same-size depth intervals and using dynamic programming to search for the best alignment of the two series' intervals. These intervals make the alignment problem tractable by grouping data points, which may be irregularly spaced or more numerous than necessary, into a manageable number of evenly spaced depth or time intervals.

[14]   Groups of intervals are mapped to one another in ratios taken from a set of plausible relative accumulation rates or mapping function slopes. This set of relative rates, defined by the user, adds a third dimension to the table of values constructed by the dynamic programming algorithm. A table entry exists for the alignment of each interval in
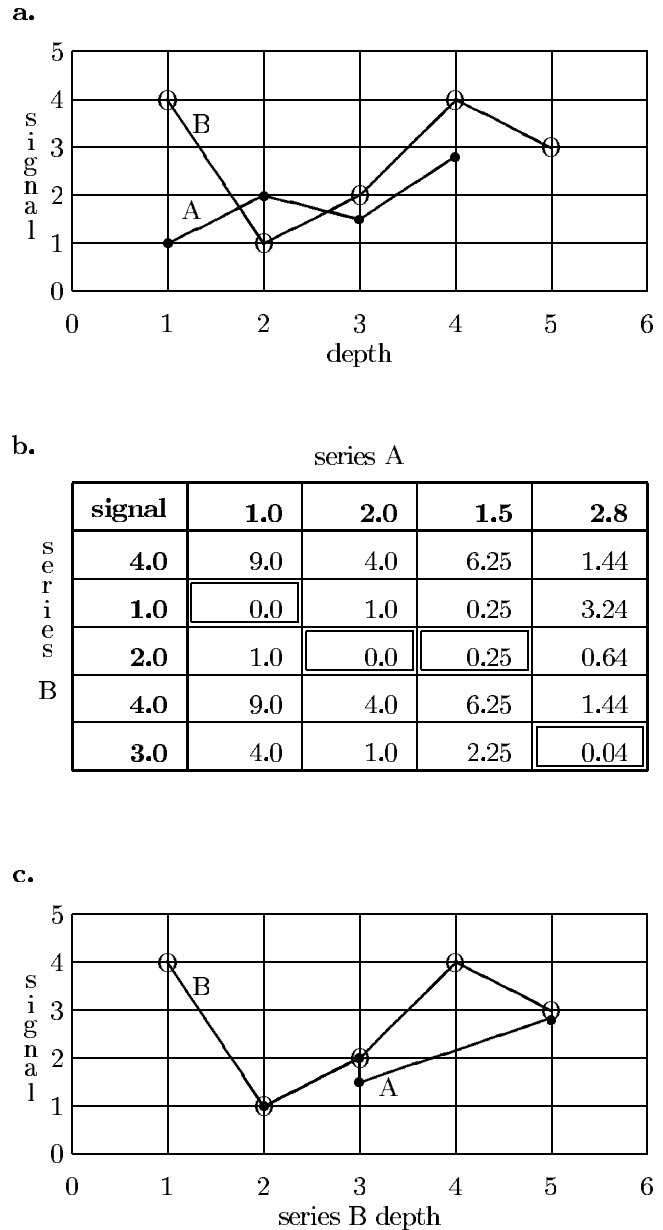
a.

b.

series A

signal	1.0	2.0	1.5	2.8
4.0	9.0	4.0	6.25	1.44
1.0	0.0	1.0	0.25	3.24
2.0	1.0	0.0	0.25	0.64
4.0	9.0	4.0	6.25	1.44
3.0	4.0	1.0	2.25	0.04

series B

c.

Figure 1.   A simple example. In this example we wish to match the points in series A to a subset of points in series B so as to minimize the sum of the square of their differences. (a) Series A with $n = 4$ points will be matched to a subset of the $m = 5$ points in series B. (b) We subtract each point in A by each point in B, square the differences, and place the results in a table in which each column represents a point in series A and each row represents a point in series B. The double boxes represent the alignment with the smallest sum. (c) Series A aligned to series B according to the solution shown in Figure 1b.

series A with each interval in series B at each possible relative rate. For each table entry we calculate a score, which is the sum of all applicable penalties, including any rate or rate change penalties, and the sum of the squared differences of all of the points within the matched intervals with the

linearly interpolated, corresponding values of the other series. The best alignment is the path from the beginning to end of the series that has the lowest cumulative score. This alignment is used to adjust the time or depth axis of the first series to fit the other, target series.

[15] By filling in the matrix starting with the beginning of one of the series, we can compute the cumulative penalty for a proposed match as we go along, add penalties for changes in accumulation rate, and store information to allow for the recovery of the optimal alignment. The cumulative penalty stored in the table is calculated using the technique illustrated in Figure 2. In this example, we calculate the smallest cumulative score for matching the intervals i and $i + 1$ in series A with the intervals n, $n + 1$, and $n + 2$ in series B. This is the case where the intervals i and n are matched at a relative rate of 2:3. Because the beginnings of intervals i and n are assumed to match, the ends of intervals $i - 1$ and $n - 1$ must also match, so we need to determine which relative accumulation rate for $i - 1$ and $n - 1$ produces the best cumulative fit. We define a penalty function for changes in relative accumulation rate and search through all of the matches of $i - 1$ and $n - 1$ to determine for which the sum of cumulative score and rate change penalty to 2:3 is smallest. This value is added to the sum of the squared differences of the points in the intervals i, $i + 1$, n, $n + 1$, and $n + 2$ and stored in the appropriate location in the table. In a separate table we record the selected relative rate for $i - 1$ and $n - 1$ so that we may recover the path of our alignment should it produce the lowest cumulative score.

3.3. Penalty Functions and Parameterizations

[16] The score for each table entry may include a variety of penalty functions. For example, because the first interval of series A is not constrained to match the first interval of series B, a "no match" penalty, μ, is added for each interval at the beginning or end of either series that is not paired with an interval from the other series. The magnitude of this penalty should be set according to how much overlap is expected between the two series. A penalty function can also be used to create tie points between the two series in order to add information from stratigraphic markers or correct mismatches produced by the algorithm. In this case, a penalty is added when an interval containing a tie point is matched. We use a penalty that is proportional to the square of the distance between the pair of tie points in the proposed match.

[17] The exact set of penalties used by the algorithm should be selected to best suit the type of data being matched. For the results presented in this paper, we use a rate change penalty of the form, γn_r^2, where γ is a rate change penalty coefficient, and n_r is the number of relative rate increments between two adjacent intervals. We choose a penalty proportional to n_r^2 to more strongly discourage fast changes in accumulation rate, but any desired function may be used. For example, if core data from two holes at the same site are matched, the relative accumulation rates of the cores may change many times due to compression and stretching of the cores during the drilling and extraction process, so a penalty for changes in relative rate may be undesirable. In this case a simple penalty for relative
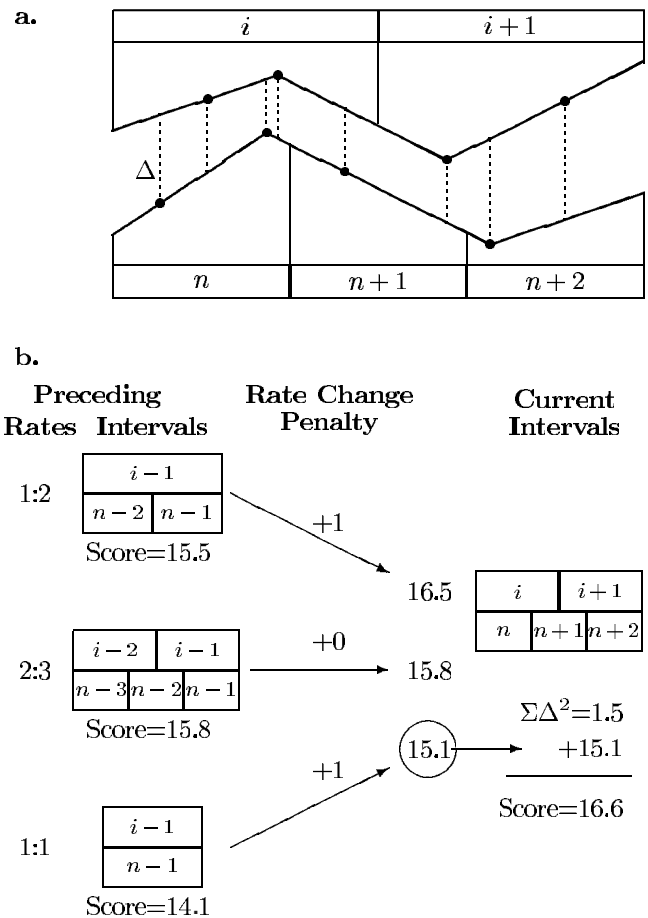

Figure 2. Matching two intervals in series A with three intervals in series B. (a) The intervals i and $i + 1$ in series A are stretched to fit across the intervals n, $n + 1$, and $n + 2$ in series B. Interpolation is used to calculate the difference Δ between the two signals at the location of each data point (solid circles). The sum of the squares of Δ, equal to 1.5 in this example, is the score for aligning these intervals. (b) The minimum cumulative score for this alignment will be stored in the table entry that corresponds to i, n, and the relative rate 2:3. The minimum score is found by determining all possible preceding interval matches, adding the necessary rate change penalty for switching from those previous relative rates to 2:3, selecting the minimum sum, and adding 1.5 to it. The previous relative rate selected is stored in a separate table to allow us to reconstruct our path later. This procedure is followed for every table entry, and when the table is full, the smallest of the entries matching the last interval of A with the last interval of B will contain the cumulative score of the optimal alignment of the two series.

rates beyond likely levels of distortion produces better results.

[18] The algorithm can handle data series with a wide variety of characteristics. However, the data series to be matched should be normalized if they do not have similar means and standard deviations. The optimal weighting of

Table 1. Algorithm Parameters

Parameters	Guidelines	Typical Values[a]
Interval size λ	smaller intervals increase alignment resolution runtime increases as the product of the number of intervals in each series more intervals may be desirable for data sets with more resolution or length more intervals should be used when using higher relative-rate resolution, e.g., 6:5, 7:6, 8:7	0.01–0.001 of signal length
"No match" penalty μ	penalty reflects degree of signal overlap if signals' endpoints are known to match, μ should be very high μ should be proportional to interval size and average difference between signals	0.1–10

Rate change penalty coefficient γ	noisy or dissimilar signals should have higher values of γ
	signals with very small intervals should have high values of γ to prevent
	    artificial fluctuations in relative rate
	penalties for extreme relative rates in addition to rate change may also be helpful
	    for difficult matches
		0.1–10

[a] For use with normalized data.

penalty functions may depend on the means and standard deviations of the data series because the penalties are added to the square of the two signals' difference. The beginning and ending points of the two series do not need to corre-spond, but the "no match" penalty, μ, should be adjusted according to the estimated degree of series overlap. We assume that linear interpolation between data points is valid and that the relative accumulation rate and position of the
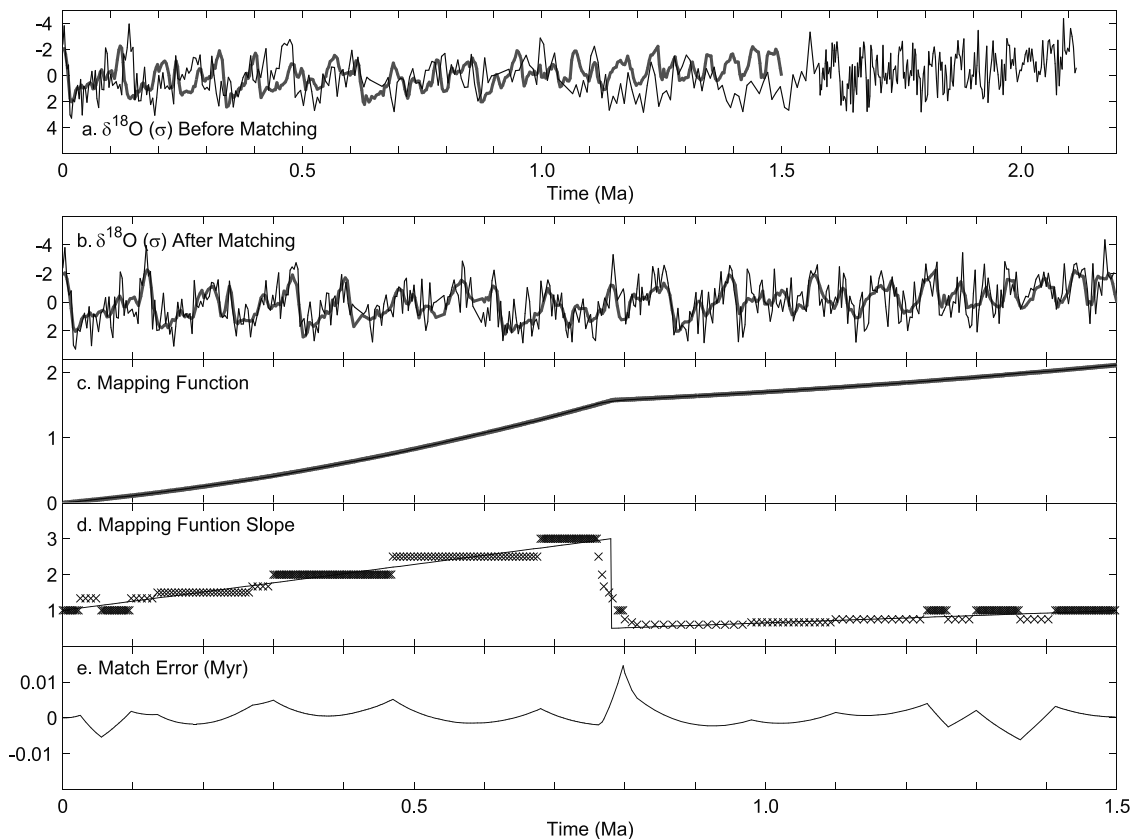

Figure 3. Matching an artificially distorted signal. (a) A normalized SPECMAP curve (thick shaded line) and an artificially distorted SPECMAP curve with a signal-to-noise ratio of 0.66 (thin line). (b) The artificial signal (thin line) after alignment with the normalized SPECMAP curve (thick shaded line) by the dynamic programming algorithm with a run time of 48 s using the parameter values: $\lambda = 2500$ years, $\mu = 5$, and $\gamma = 10$. (c) The mapping function used to produce the artificial signal (thick shaded line) and the mapping function reconstructed by the dynamic programming algorithm (thin line). (d) The slope of the real mapping function (solid line) and the relative accumulation rates (crosses) used by the algorithm to produce the alignment. (e) The age difference between the true age of each point and that determined by the automatic alignment.
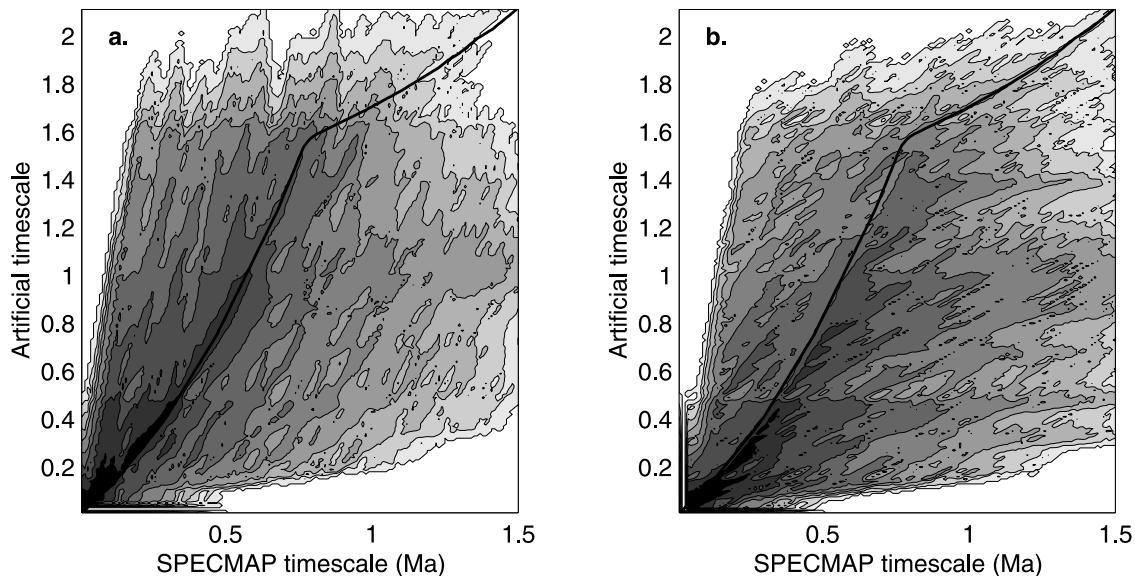
Figure 4. Cross sections of the table of cumulative scores. (a) A contour map of the cumulative scores recorded in the table produced by the dynamic programming algorithm for the alignment shown in Figure 3 at the relative rate of 3:1 overlain by the algorithm's reconstructed mapping function (thick line). Notice that from 0.5 to 0.7 Ma, where the slope of the mapping function is approximately 3, the mapping function lies along a valley defined by the contours. The contour lines mark score values of 200 (dark shading) to 1000 (white area) in increments of 100. (b) The table's scores for the relative rate of 2:3 displayed in the same format and with the same mapping function.

two series may be discretized without introducing significant error to the solution.

[19] The resolution of the produced alignment increases as the size of the intervals used for matching, λ, decreases, but the run time of the algorithm increases as the product of the number of intervals in the two series. In contrast, run time increases only linearly with the number of data points in the series. Because intervals are matched in groups and effectively discretize a continuous parameter such as depth or time, intervals that are smaller than the average spacing between data points may still be meaningful. The number and size of the intervals do not have to be the same for both series, but the set of relative rates defined by the user should reflect the approximate ratio at which they should be matched. For example, if a series matches only about half of a given target signal, the algorithm could either divide the series into half as many intervals as the target and use a set of relative rates centered about 1:1, or it could use the same number of intervals for both series and have the relative rates centered about 2:1. Any number of relative accumulation rates can be defined; we have found good results for many data sets using 13 relative rates between 3:1 and 1:3, with the greatest concentration of rates near 1:1. Over a distance of many intervals, the algorithm can effectively reproduce any relative accumulation rate within the range of defined rates by dithering between rates on either side of the true rate. The error that results from this approximation will depend on the resolution of the defined rates and the size of the series' intervals.

[20] We recommend that when beginning a correlation with our algorithm, the user select parameter values accord-

ing to the guidelines in Table 1. One should start with somewhat large intervals and tune the ''no match'' and rate change penalties until one obtains an approximately correct alignment. If tuning the penalty functions does not create a good alignment, the interval size is probably too large, but occasionally a couple of tie points will need to be defined by the user. After obtaining a reasonable alignment, progressively decrease the interval size, λ, until the degree of signal correlation stabilizes to prevent artificially high resolution in the optimal alignment. Typically, one should decrease the ''no match'' penalty in proportion to the change in interval size because it is applied to each interval. However, the rate change penalty coefficient, γ, may need to be increased as λ decreases because the existence of more locations at which the rate can change will allow the rate to change more quickly. With a little experience, an optimal tuning can usually be completed in less than 10 iterations, with each iteration taking no more than a couple of minutes. Little to no change in the parameters should be needed between tuning sets of similar data, and additional automation in the future may further reduce the number of tuning steps required.

4. Evaluation of Algorithm Results

4.1. Testing With an Artificially Distorted Signal

[21] To determine the dynamic programming algorithm's sensitivity to noise and changes in sedimentation rate, we distort a normalized SPECMAP curve [*Shackleton et al.*, 1990] with artificial noise and changes to its time axis and then map it back to the undistorted curve. Figure 3a shows the original SPECMAP curve and the distorted one, which

has a signal-to-noise ratio of 0.66 and a relative sedimentation rate that increases linearly from 1 to 3 over the first half of the data, instantaneously drops to 0.5, and then linearly increases back to 1. Both series have 490 data points, and the average time between points is ~3000 and ~4000 years for the original and artificial signals, respectively. We describe the time axis of the artificial signal in terms of years, but because of the distortion we have applied to it, one "year" of the artificial signal may represent as many as two real years or as little as one third of a year.

[22] The match shown in Figure 3b is obtained using an interval size, λ, of 2500 years for each series, a "no match" penalty, μ, of 5.0 per interval, and a rate change penalty coefficient, γ, of 10. The program runs in 48 s on a 750 MHz Pentium III with 128 MB of RAM. Most remarkably, this alignment is obtained with no tie points; the *Martinson et al.* [1982] technique requires at least five or six pairs of tie points to match these signals correctly even when the artificial signal has no added noise. The mapping function produced by our algorithm is virtually indistinguishable from the one used to construct the signal; the correlation coefficient between the two mapping functions is 0.999987. However, the plot of sedimentation rate in Figure 3d shows that the instantaneous rate change has been stretched over 20 kyr, producing a maximum error in estimated age of 12.9 kyr. The stair step pattern in Figure 3d is the result of the discrete nature of the defined relative rates. Increasing the number of defined rates results in a more accurate solution with a larger number of smaller steps but also increases run time.

[23] The dynamic programming algorithm constructs a three-dimensional table of cumulative scores to calculate efficiently the optimal alignment of the two signals. Figure 4a is a contour plot of the data stored in the slice of the table representing a relative matching rate of 3:1. The value at each position in the plot is the minimum cumulative score calculated for an alignment from the start of the series to a pair of intervals, with that pair matched at a rate of 3:1. The optimal mapping function follows the three-dimensional path through this table that produces the lowest cumulative score at the end of the series. This mapping function is projected onto two dimensions and displayed over the contour map in Figure 4a. Where the slope of the optimal mapping function is 3:1, it is passing through the part of the table that we have plotted and, therefore, it follows the contour plot's valley in this region. For comparison, in Figure 4b the mapping function is plotted with the table values for the relative rate of 2:3.

4.2. Parameter Sensitivity

[24] We examine the sensitivity of the algorithm to its parameters by varying each parameter individually while performing the same match as above except with a signal-to-noise ratio of 1.0 in the artificial signal. In Figure 5 we plot two measures of the amount of error in each alignment, the maximum age error of the produced fit and $10^6(1 - \rho)$, where ρ is the correlation coefficient of the true and produced mapping functions. These two indicators of fit behave similarly but not identically. One important feature of these curves is their flat, wide
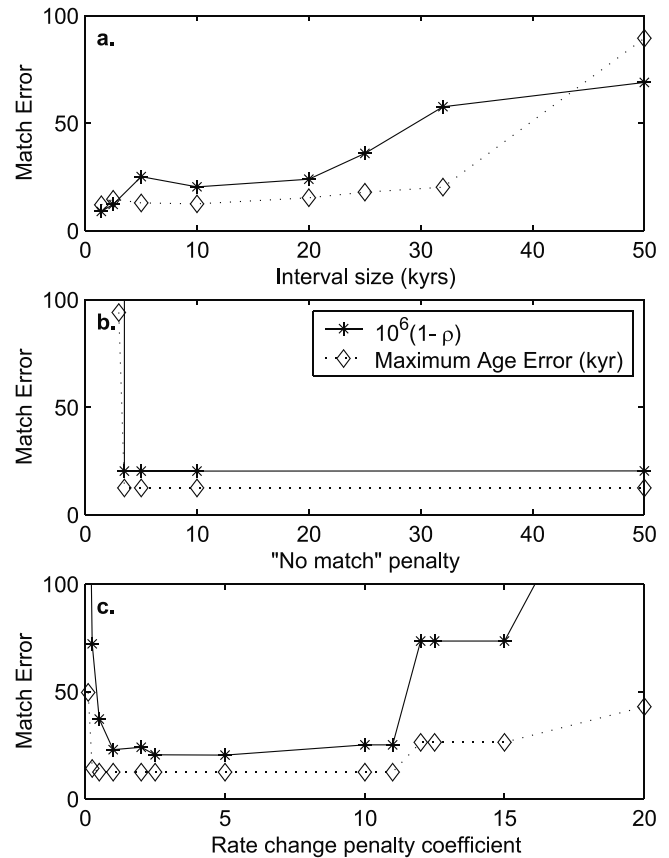

Figure 5. Parameter sensitivity. The same artificial signal as in Figure 3 but with an increased signal-to-noise ratio of 1.0 matched to the SPECMAP curve while varying one parameter at a time. Each star is $10^6(1 - \rho)$, where ρ is the correlation coefficient between the real and empirical mapping functions for a particular set of parameter values. Each open diamond is the maximum difference between the real and empirical ages of the data points for the mapping produced by that set of parameters. (a) Errors produced by different interval sizes, λ, in kiloyears with a constant "no match" penalty, μ, of 100 and a rate change penalty of $5n_r^2$. The run times for these experiments range from 1.1 second for λ = 50,000 years to 225 s for λ = 1428 years. (b) Errors produced by different "no match" penalties with a constant interval size of 10 kyr and a rate change penalty of $5n_r^2$. The run time for this interval size is about 5 s. (c) Errors produced by different coefficients to the rate change penalty with λ = 10 kyr and μ = 100.

minima; these demonstrate that small deviations from optimal parameter values will not greatly affect the accuracy of a fit. Below we describe the specific effects of variation in each parameter.

[25] Increasing the size of intervals reduces the number of locations at which the algorithm may adjust the relative rate of the two signals and thus increases error. However, intervals that are too small may unnecessarily increase run time and may make the mapping more sensitive to noise by allowing the relative rate of the series to change more rapidly. "No match" penalties that are too small cause the
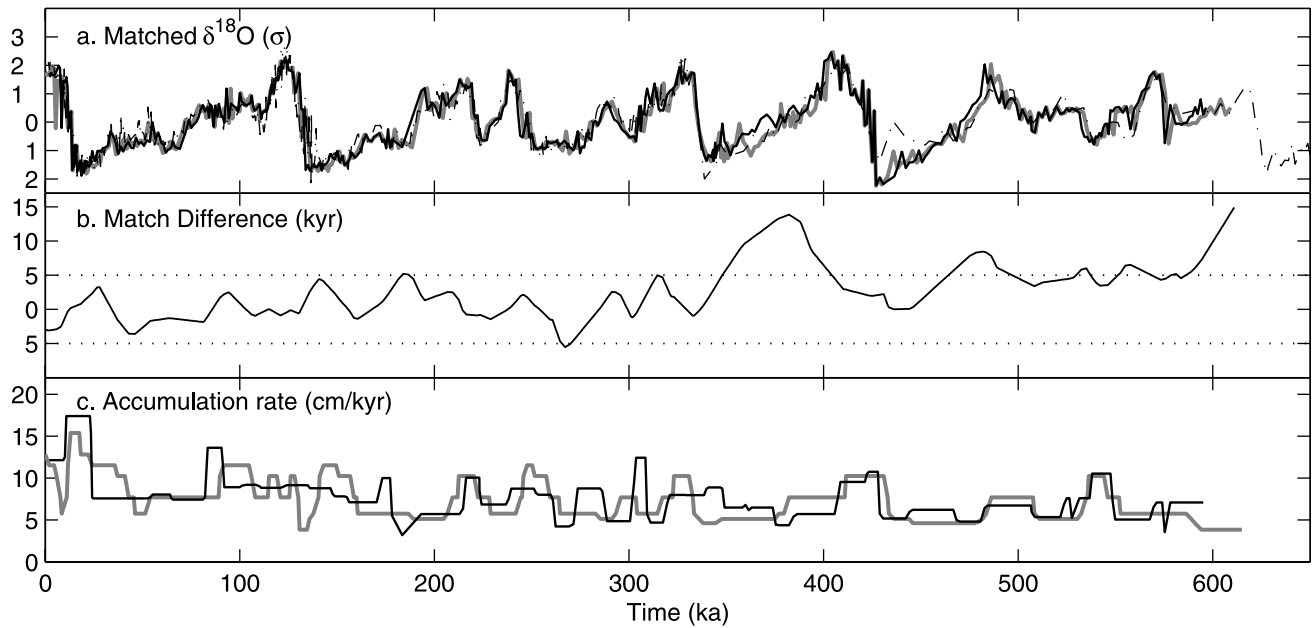
Figure 6. Automatic and hand tuning. (a) A normalized benthic oxygen isotope curve for ODP site 1012 independently tuned to a common target (dash-dotted line) by hand (thin line [*Herbert et al.*, 2001]) and by our dynamic programming algorithm (thick shaded line) with a run time of 23 s and parameter values of λ = 5 cm and 650 years, μ = 0.2, and γ = 0.3. (b) The difference in estimated age of the two tuned signals. (c) The accumulation rate versus time predicted by automatic (shaded line) and hand (solid line) tuning. Possible accumulation rates for the automatically tuned curve range from 3 to 17.5 cm/kyr.
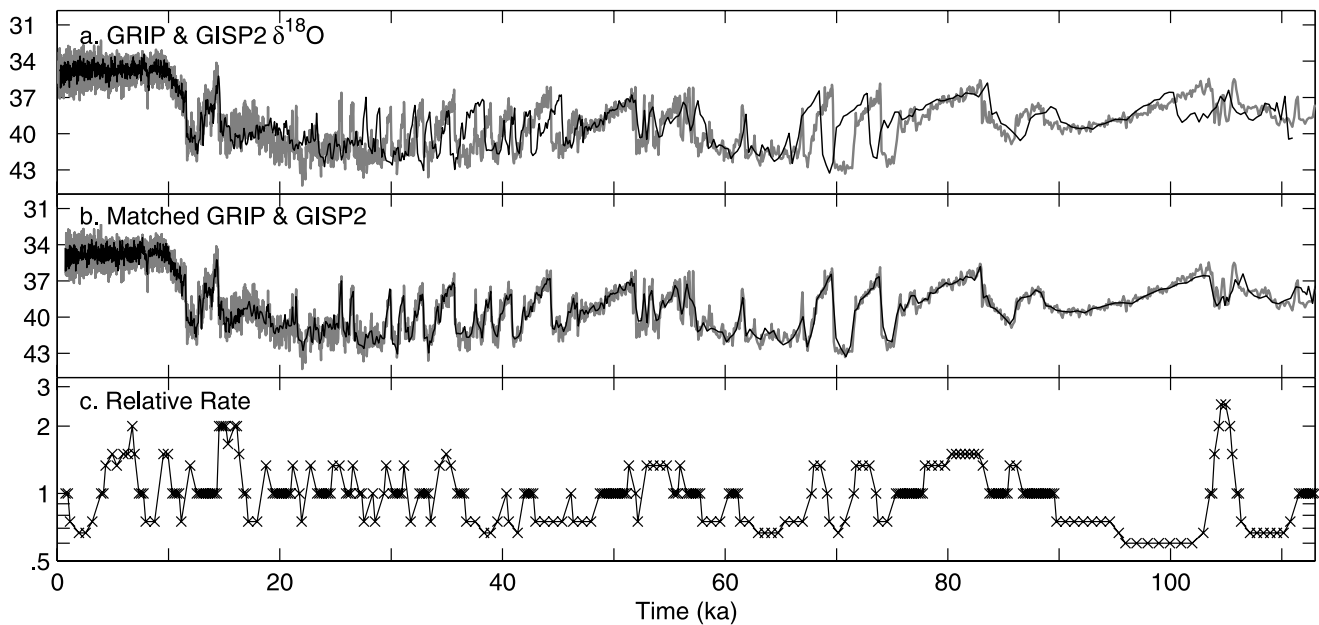

Figure 7. Matching GRIP and GISP2 curves. (a) The GRIP (shaded line) and GISP2 (solid line) ice oxygen isotope curves versus their respective published timescales [*Johnsen et al.*, 1997; *Grootes et al.*, 1993]. (b) The GISP2 curve aligned to the GRIP curve by the dynamic programming algorithm with a run time of 71 s, λ = 200 years, μ = 50, and γ = 1. (c) The duration of GISP2 events relative to GRIP events given by their respective timescales and the relative rates used by the algorithm to produce the presented alignment.
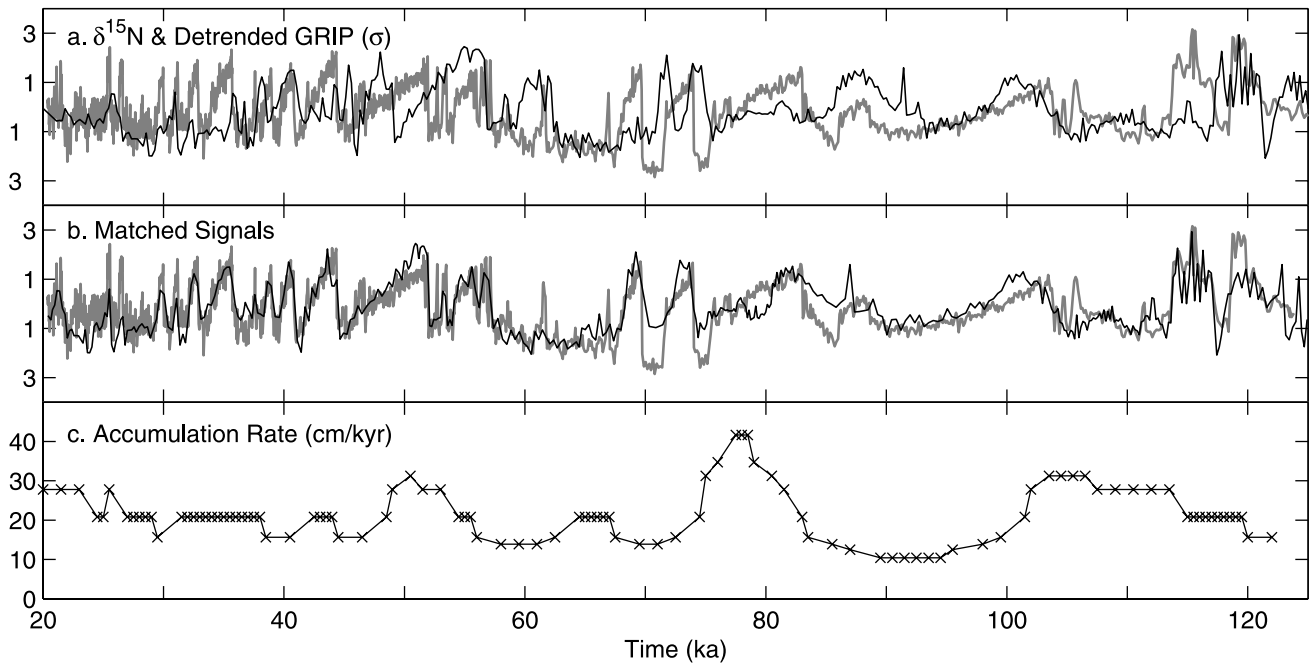
Figure 8. Matching two different proxies. (a) The GRIP oxygen isotope record from 20–124 ka, normalized and detrended (shaded line) and a normalized nitrogen isotope curve (solid line) from the Arabian Sea [*Altabet et al.*, 2002] from 4.0 to 26.7 m, plotted versus an arbitrary linear age scale. (b) Alignment produced by our dynamic programming algorithm in 17 s with the aid of tie points at 21.5 and 123.5 ka and with $\lambda = 10$ cm and 500 years, $\mu = 1.0$, and $\gamma = 4.0$. (c) Accumulation rate (cm/kyr) versus time (ka) in the Arabian Sea implied by the presented alignment.

algorithm to align only the very end of one series with the very beginning of the other, producing very large errors. In this example, the "no match" penalties above a certain threshold have no effect because the two series completely overlap. If the two series did not have the same end points, very large "no match" penalties would also produce errors because the two series' ends would be forced to match. High rate change penalties prevent the algorithm from changing rates when necessary to best match the signals. Lower rate change penalties allow more dithering between rates but cause distortion where the algorithm fits the signal too closely to the noise. Extremely low rate change penalties will allow physically unrealistic results, such as multiple peaks from one series squeezed into one peak from the other.

4.3. Paleoclimate Applications

[26] The dynamic programming technique has many potential applications in paleoclimate research. Above we demonstrated its ability to accurately align a very noisy artificial signal with a large, instantaneous rate change. We now examine its ability to align different types of real data to demonstrate its applicability to several important areas of paleoclimate research. For these applications, automated signal correlation can not only save time and effort but also provide a more consistent set of alignment criteria.

[27] Our algorithm is well suited to aid in orbital-scale tuning because with very little effort our automated algorithm produces results extremely similar to those produced by hand tuning, which is the most commonly employed correlation technique. Figure 6a compares the results of our automated tuning and an independent hand tuning [*Herbert et al.*, 2001] of a normalized benthic oxygen isotope curve for OPD site 1012 to a common target. The automated tuning uses parameter values of $\lambda = 5$ cm and 650 years, $\mu = 0.2$, and $\gamma = 0.3$. The algorithm does an excellent job of aligning the isotope curve to the target despite somewhat large differences between the two. The automated alignment agrees very well with the hand-tuned results on both the large and fine scale; the largest difference in estimated age of the two tuned signals is only 6200 years back to 350 ka and 17.3 kyr over the full 600 kyr range. Additionally, Figure 6c shows that the accumulation rates produced by the automated alignment are similar in character to rates derived from hand tuning.

[28] The match between GRIP [*Johnsen et al.*, 1997] and GISP2 [*Grootes et al.*, 1993] oxygen isotope records in Figure 7 demonstrates that our algorithm does an excellent job of matching fine-scale features. For this match, $\lambda = 200$ years, $\mu = 50$, and $\gamma = 1$. Because the algorithm matches signals using such small intervals, the high- and low-frequency components of the signals can be matched concurrently. Hence, our algorithm can be used to correlate millennial scale features or to produce stacked data without distorting sharp, high-resolution features. Another interesting feature that stands out in our results is the correlation between the isotopic measurement and the relative accumulation rate of the two signals. High isotopic values are accompanied by high relative rates because the peaks of

the GISP2 series are wider than the peaks in the GRIP series, which serves as the reference signal in this example. Hence, we may also think of the relative rate as the degree to which a signal must be compressed in order to match its reference signal.

[29] The algorithm can also be used to match two proxies representing different climate signals, but it may be necessary to detrend or otherwise slightly modify the signals to make their values sufficiently similar. In Figure 8, a normalized nitrogen isotope curve from the Arabian Sea [*Altabet et al.*, 2002] is matched to a linearly detrended and normalized GRIP oxygen isotope curve [*Grootes et al.*, 1993] . We add two tie points, one on either end of the series, to improve the alignment of the ends of the data. The parameter values used are λ = 10 cm and 500 years, μ = 1.0, and γ = 4.0. The algorithm performs extremely well and is not led astray by peaks that appear in only one of the two signals (e.g., peaks in the GRIP signal at 46, 53, 61, and 106 ka). In places where the two signals are quite different (e.g., 85 and 105 ka), the alignment may be adjusted with changes in the penalty weightings or new tie points, and the best alignment is a matter of interpretation just as it would be with any hand-tuned alignment. Of course, two signals are not necessarily synchronous, and one must take into account any error that may be caused by leads or lags between signals if attempting to use this technique to date events.

[30] The creation of composite depth scales for sediment cores is another area where our dynamic programming technique could be very useful. One useful feature of our algorithm is that it can match multiple core properties, such as reflectance and magnetic susceptibility, simultaneously. However, the main challenge for any automated algorithm in correlating data from cores is dealing with the gaps that occur between core sections. Currently, our algorithm assumes a linear interpolation across gaps and penalizes for the differences between two signals accordingly. This technique is robust enough to find an approximate mapping between two holes but produces some distortion around large gaps. One can use this approximate mapping to identify good tie points at the edges of these large gaps and then find an extremely good match. Figures 9 and 10 compare the results of this technique with the published composite depth mapping for a portion of ODP Hole 925 [*Curry et al.*, 1995]. Soon we hope to develop special instructions for dealing with core gaps and eliminate the need for the manual addition of tie points. This adaptation should also support the correlation of more than two holes and handle incomplete recovery realistically.

[31] Eventually, our dynamic programming technique may not only reduce the labor required to create composite depth scales but also produce more accurate results. The usual convention in the creation of composite depth scales is to define a constant offset from meters below seafloor (mbsf) depth for each 9-m core, but within-core depth-scale changes are often needed to align features on scales of <9 m due to stretching and squeezing within cores [*Hagelberg et al.*, 1992]. The mapping function produced by our dynamic programming algorithm with 10 tie points provides
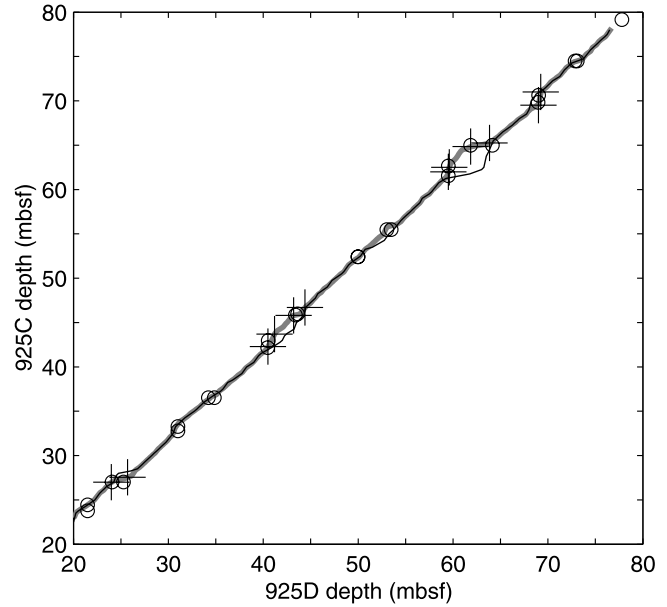

Figure 9. Correlation between holes from ODP Site 925. Mapping functions produced by the dynamic programming algorithm when aligning reflectivity data from Holes C and D at ODP Site 925. One mapping function is produced using no tie points (thin line), and one (thick shaded line) is produced by placing tie points (crosses) on both sides of 6 of the 11 gaps in the series and setting λ = 10 cm, μ = 100, and γ = 1.0. Both reflectivity series have nearly 1500 data points, and the program run time for this problem is 73 s. Circles mark the correlation of Holes C and D at specific depths according to the composite depth scale provided in the ODP initial report [*Curry et al.*, 1995].

a more consistent alignment, with a correlation coefficient of 0.8307 between the two reflectance signals in Figure 10, than the published composite depth scale [*Curry et al.*, 1995], which yields a correlation coefficient of 0.5843. These results suggest that the offset for each core is not constant, probably due to distortion during the drilling and extraction process. The traditional technique for the creation of composite depth scales also has a tendency to artificially inflate depths, as illustrated in Figure 10b, but dynamic programming may be able to overcome this distortion. We currently have plans to adapt our dynamic programming technique for the easy production of higher resolution composite depth scales with less distortion.

4.4. Limitations

[32] Unfortunately, dynamic programming is not well suited to the construction of age-depth scales by orbital tuning because, unlike signal difference, Fourier analysis and band-pass filtering cannot easily be broken into small enough subproblems for high-resolution alignment. However, one might use the dynamic programming technique to match climate signals to proposed insolation response functions. An automated technique would be very useful for considering a large number of possible response functions.
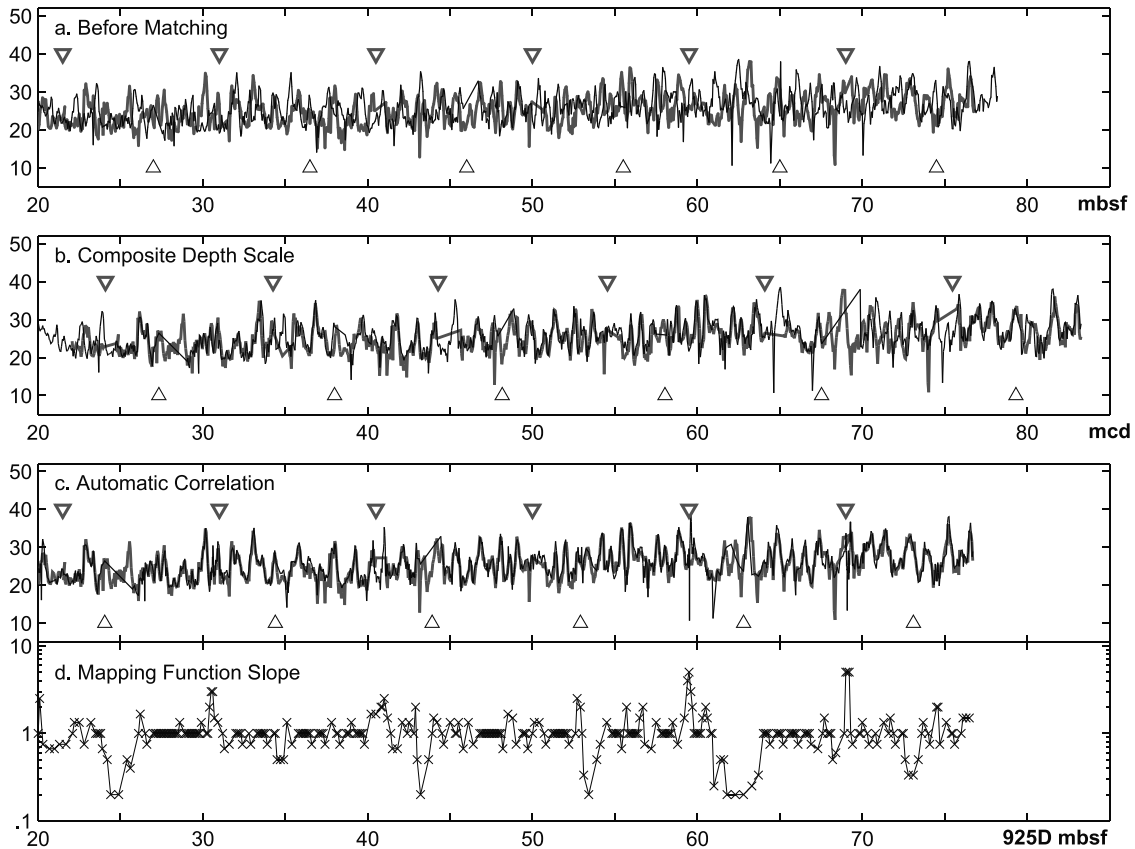
Figure 10. Producing a composite section. (a) Reflectivity plotted versus meters below seafloor (mbsf) for cores from ODP site 925, Holes C (thin line) and D (thick shaded line). Gaps between core sections are marked by upward pointing triangles for Hole C and downward pointing, shaded triangles for Hole D. (b) Alignment produced by the composite depth scale provided in the ODP initial report [*Curry et al.*, 1995] plotted versus meters composite depth (mcd). (c) Alignment of the Hole C reflectivity signal to the Hole D signal as determined by the automatically generated mapping function with tie points shown in Figure 9. (d) The accumulation rate of Hole C relative to Hole D. Values > 1 indicate that the algorithm has compressed Hole C relative to Hole D.

[33] Additionally, no automated technique can completely replace the knowledge and judgment of a trained professional. The results of this program should always be evaluated by eye. Hiatuses or other disturbances may produce erroneous results, and appropriate penalty weightings must be chosen for each pair of signals to be matched. Also, special attention should be paid to the beginning and end of a match because less surrounding data is available to constrain the match. For some data it may be desirable to increase the rate change penalty near the ends of the series to reduce the potential magnitude of these errors. Generally, incorrect matches are fairly obvious and the adjustment of penalty weightings or the addition of a couple of tie points can quickly correct the error.

5. Conclusions

[34] The dynamic programming algorithm proposed in this paper does an excellent job of aligning paleoclimate records with substantially less effort than hand tuning. Because the algorithm is not susceptible to local solutions, it also requires less effort to use than previously published automated correlation techniques, which require good initial guesses. The technique produces results that agree well with hand-tuned series and accurately fits signals with low signal-to-noise ratios, gaps, spectral power at a variety of frequencies, and large changes in sedimentation rate. In addition to making signal correlation faster and easier, our dynamic programming technique could provide a consistent set of criteria for the alignment of many climate proxies, improve the resolution of composite depth scales and stacked data, and automate comparisons of climate proxies with proposed climate response functions. However, any automatically produced correlation should be evaluated by eye because a record may contain unexpected complexities and because penalty weightings may need adjustment to produce the most desirable results. Additionally, the technique is not well suited to orbital tuning by frequency analysis. Automated signal

correlation by dynamic programming can be a powerful tool in the study paleoclimate signals because it offers speed, versatility, and the ability to find realistic global solutions.

Notation

μ "no match" penalty.
γ rate change penalty coefficient.
n_r number of relative rate steps between two intervals.
λ interval size.
ρ correlation coefficient between two mapping functions.

References

Altabet, M. A., M. J. Higginson, and D. M. Murray, The effect of millenial-scale changes in high frequency Arabian Sea denitrification on atmospheric CO_2, *Nature*, *415*, 159–162, 2002.

Brüggeman, W., A minimal cost function method for optimizing the age-depth relation of deep-sea sediment cores, *Paleoceanography*, *7*, 467–487, 1992.

Cormen, T. H., C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, 1028 pp., MIT Press, Cambridge, Mass., 1990.

Curry, W. B., N. J. Shackleton, and C. Richter, Composite depth subsection of the Site 925, *Proc. Ocean Drill. Program Initial Rep.*, *154*, 79–86, 1995.

Eggleston, J. R., and J. J. Peirce, Dynamic programming analysis of pore space, *Eur. J. Soil Sci.*, *46*, 581–590, 1995.

Grootes, P. M., M. Stuvier, J. W. C. White, S. J. Johnsen, and J. Jouzel, Comparison of oxygen isotope records from GISP2 and GRIP Greenland ice cores, *Nature*, *366*, 552–554, 1993.

Hagelberg, T., N. Shackleton, and N. Pisias, Development of composite depth sections for sites 844 through 854, *Proc. Ocean Drill. Program Initial Rep.*, *138*, 79–85, 1992.

Herbert, T. D., J. D. Schuffert, D. Andreasen, L. Heusser, M. Lyle, A. Mix, A. C. Ravelo, L. D. Stott, and J. C. Herguera, Collapse of the California current during glacial maxima linked to climate change on land, *Science*, *293*, 71–76, 2001.

Imbrie, J., J. D. Hays, D. G. Martinson, A. McIntyre, A. C. Mix, J. J. Morley, N. G. Pisias, W. L. Prell, and N. J. Shackleton, The orbital theory of Pleistocene climate: Support from a revised chronology of the marine $\delta^{18}O$ record, in *Milankovitch and Climate, Part 1*, edited by A. Berger, pp. 269–305, D. Reidel, Norwell, Mass., 1984.

Johnsen, S. J., et al., The $\delta^{18}O$ record along the Greenland Ice Core Project deep ice core and the problem of possible Eemian climatic instability, *J. Geophys. Res.*, *102*, 26,397–26,410, 1997.

Lourens, L. J., A. Antonarakou, F. J. Hilgen, A. A. M. Van Hoof, C. Vergnaud-Grazzini, and W. J. Zachariasse, Evaluation of the Plio-Pleistocene astronomical timescale, *Paleoceanography*, *11*, 391–413, 1996.

Martinson, D. G., W. Menke, and P. Stoffa, An inverse approach to signal correlation, *J. Geophys. Res.*, *87*, 4807–4818, 1982.

Martinson, D. G., W. Menke, and P. Stoffa, Reply, *J. Geophys. Res.*, *89*, 2501–2504, 1984.

Martinson, D. G., N. G. Pisias, J. D. Hays, J. Imbrie, T. C. Moore, and N. J. Shackleton, Age dating and the orbital theory of the ice ages: Development of a high-resolution 0 to 300,000-year chronostratigraphy, *Quat. Res.*, *27*, 1–30, 1987.

Prell, W. L., J. Imbrie, D. G. Martinson, J. J. Morley, N. G. Pisias, N. J. Shackleton, and H. F. Streeter, Graphic correlation of oxygen isotope stratigraphy application to the late Quaternary, *Paleoceanography*, *1*, 137–162, 1986.

Råde, L., and B. Westergren, *Mathematics Handbook for Science and Engineering*, 539 pp., Birkhäuser Boston, Cambridge, Mass., 1995.

Shackleton, N. J., A. Berger, and W. R. Peltier, An alternative astronomical calibration of the lower Pleistocene timescale based on ODP site 677, *Trans. R. Soc. Edinburgh*, *81*, 251–261, 1990.

Shackleton, N. J., T. K. Hagelberg, and S. J. Crowhurst, Evaluating the success of astronomical tuning: Pitfalls of using coherence as a criterion for assessing pre-Pleistocene timescales, *Paleoceanography*, *10*, 693–697, 1995.

Shure, L., and A. D. Chave, Comments on "An inverse approach to signal correlation" by D. G. Martinson, W. Menke, and P. Stoffa, *J. Geophys. Res.*, *89*, 2497–2499, 1984.

Waterman, W. S., and R. Raymond Jr., The match game: New stratigraphic correlation algorithms, *Math. Geol.*, *19*, 109–127, 1987.

Yu, Z. W., and Z. L. Ding, An automatic orbital tuning method for paleoclimate records, *Geophys. Res. Lett.*, *25*, 4525–4528, 1998.

―――――――――――――――

L. E. Lisiecki, Department of Geological Sciences, Brown University, Providence, RI 02912, USA. (lorraine_lisiecki@brown.edu)

P. A. Lisiecki, 150 Palmer Street, Quincy, MA 02169, USA. (lisiecki@alum.mit.edu)