

**AN EXPERT SYSTEM FOR WELL-TO-WELL
LOG CORRELATION**

by

David J. Lineman

B.S., MASSACHUSETTS INSTITUTE OF TECHNOLOGY
(1985)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF

MASTER OF SCIENCE IN GEOPHYSICS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

December 1986

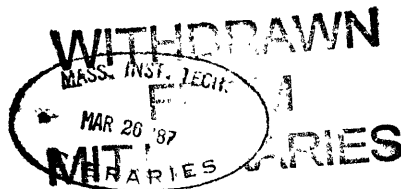
© David J. Lineman 1986

The author hereby grants to M.I.T. permission to reproduce and
to distribute copies of this thesis document in whole or in part.

Signature of Author.....
Department of Earth, Atmospheric, and Planetary Science
December 10, 1986

Certified by.....
Prof. M. Nafi Toksöz
Thesis Advisor

Accepted by.....
Prof. Theodore R. Madden
Chairman, Department Committee



Lindgren

An Expert System for Well-to-Well Log Correlation

by
David J. Lineman

Submitted to the Department of Earth, Atmospheric, and Planetary Sciences on December 10, 1986 in partial fulfillment of the requirements for the Degree of Master of Science in Geophysics.

Abstract

This thesis presents a robust method for correlation of geologic sequences known as 'Dynamic Depth Warping' (DDW). This method uses dynamic programming to find an optimal depth matching between two sets of log data. The advantage of dynamic programming over conventional spectral correlation methods is the ease at which local geologic knowledge can guide the matching process both before and during the correlation.

The Dynamic Depth Warping algorithm is implemented in a LISP program called COREX. This program uses a knowledge-based system to integrate all of the information usually available to geologists for log correlation. For this thesis the data includes digitized wireline logs, simple lithologic information, seismic lines, interpreted dipmeter logs, and local geologic knowledge. The system uses rules stored in the knowledge base to analyze these data and impose geologic constraints on the correlation algorithm.

We use the knowledge base of COREX for two purposes. First, the program performs an initial match of the wells based on the lithologies present, the scale of the correlation, and the depositional environment under consideration. Second, using the initial match to establish tie lines, the knowledge base analyzes the local geologic structure and log quality, and translates this information into constraints on the DDW algorithm. This guidance allows meaningful correlations in areas that were previously too complex because of geologic structure or large data volume.

We demonstrate the method with synthetic examples in which the program successfully correlates across geologic structures and pinch-outs. We also apply the program to field examples from West Africa and Turkey. In both cases, the automated correlation agrees very well with hand correlations provided by geologists.

Thesis Supervisor: M. Nafi Toksöz
Title: Professor of Geophysics

Acknowledgements

Thanks first go to my sooth-saying advisor M. Nafi Toksöz. He had faith that Artificial Intelligence would have a place in geophysics when Lisp was still a speech impediment. Nafi has a practical approach to science that has always impressed me. Even though he doesn't know it, the best thing he has taught me is that science is nothing without the people who do science. Greg Duckworth proofread the entire text, asked a lot of good questions early on, and gave helpful pointers during my all-to-many presentations. He can consider himself responsible for my future employment in the wonderful world of well logging at Schlumberger, and when they come after him, he'll wish he wasn't!

Jim Mendelson, co-director of the A.I. Group and great friend, tunneled me through my thesis from outline to copyright. His advice, support, and good humor through my some of my mental dilemmas were valuable beyond price. Jim taught me that one word – perspective – can do the work of a thousand words. May he always live by our motto, "Mahammana hammana hammana". Immeasurable thanks also go to the duke of late night computing, Gilles Garcia. It was his idea to try this dynamic programming stuff, and he gave me advice on the cans and can'ts of artificial intelligence at the beginning.

It's difficult to accomplish anything on a LISP machine here without the help of Steve Gildea. He answered a thousand questions, made LATEX do a mitthesis, and got the LISPM out of many stuck states. Roger Turpening convinced guys at DEC that my stuff was worthwhile – which makes him as good a magician as he is geophysicist. His laboring to get the BIG lisper in here was essential to my work – thanks Rog. Carol Blackway always provided encouragement whenever she stopped into the "hole" at room 400, which can mean a helluva lot when you feel like throwing the entire machine into a basket. As John Lennon once said, "Behind every idiot is a great woman", and Sarah Saltzer was the one. She listened to all of my complaining, and still forced me to get motivated when I really needed it.

And to the people who paint some color on the gray hallways of ERL. The great old-time zoners, Danny Fain, Mike Guenette, and Rich "Gerry Heals" Herrmann made two summers of

Fortran programming as fun as is humanly possible. When you look up "Genuine nice guy" in the dictionary, there is a picture of George Blumberg there— with a shit-eating grin on his face. Ed Reiter has such a charming townie way about him that you just wanna kiss him. I hope Bob Cicerone will someday convince Nafi that former oil company employees are only human— but I doubt it. Jane Maloof deserves a job and a man she is happy with— or at least one of the two! Dan Burns, João Rosa, Eric Shortt, and Tom Wissler finally made it out of here, showing me that there really is a faint flicker at the end of the tunnel.

I must also thank the ghost of Carl Godkin. Even though he left ERL, I think I still see his shadow meandering down the hallways, solving inverse problems and debugging my Fortran programs. I thank him for accomplishing unreasonable amounts, and causing Nafi to expect too much from the five-year-fifty-thousand-dollar planners. He is the only guy I know who spent more time doing other peoples work than his own, and still managed to finish. Now that's a rare breed.

Finally, I must thank those amazing folks Mr. and Mrs. Lineman. Not only did it start with them, but it couldn't have ended without them either. They busted butt for four years while I drank beer and tried to learn something about life in between swallows. Hey Dad, is this what you mean by "Go big, or go home?" I hope so.

Contents

Abstract	1
Acknowledgements	3
1 Introduction	8
2 Correlation with Dynamic Depth Warping	12
2.1 Dynamic Depth Warping	12
2.2 Discrete Curve Matching	13
2.3 Constraints on the Dynamic Depth Warping Algorithm	14
2.3.1 Endpoint Constraints	15
2.3.2 Local Continuity Constraints	15
2.3.3 Global Path Constraints	17
2.3.4 Axis Orientation	18
2.4 The Distance Metric	19

2.4.1	Distance Metric for Logs	19
2.4.2	Multiple log distance measure.	20
2.5	Weighting and Normalization	22
2.6	Smoothing the Distance Metric	23
2.7	Computational Savings in Dynamic Depth Warping	24
2.8	The COREX System	25
3	Using Geologic Knowledge in the Match: Examples	27
3.1	The Initial Match	27
3.1.1	The Environment Consideration	28
3.1.2	The Scale Consideration	29
3.1.3	The Distance Metric for the Initial Match	29
3.2	Geologic constraints on Dynamic Depth Warping	30
3.2.1	Geologic Expression in the Warping Path	31
3.2.2	Global Path Constraints	32
3.2.3	Local Path Constraints	33
3.2.4	Weighting the local distance metric	34
4	Results	36
4.1	Field Example from West Africa.	36

4.2	Field Example from Thrace Basin, Turkey.	37
5	Discussion and Conclusions	40
A	Dynamic Waveform Matching	42
A.1	An Example With String Matching	42
A.2	Dynamic Waveform Matching	46
B	Outline of the COREX system	48
C	Noise analysis by Walsh Transform	50
C.1	The Noise Measure	52
C.2	Noise Analysis by Dynamic Programming	54
D	Rules in the geologic Knowledge Base	56
D.1	Lithologic Rules	57
D.2	Scale Rules	58
D.3	Shoreline Rules	59
D.4	Structural Rules	60
D.5	Distance Metric Rules	61
E	Other geophysical applications of Dynamic Programming.	63

Chapter 1

Introduction

This thesis presents a new method for well-to-well log correlation known as Dynamic Depth Warping. This method combines expert-system programming with a dynamic waveform matching algorithm in order to overcome many of the problems that face present automated correlation algorithms. It attempts to imitate the thought process of the geologist by breaking down the problem into smaller parts, and then exploits the speed and dependability of the computer to analyze these parts on numerical scale that is impossible for human analysts.

The basic correlation problem is illustrated in figure 1. A formation pattern is present in one well, and we wish to discover the same pattern in another nearby well, thus tracing the geologic horizon through the area. Sometimes an entire suite of well logs is available, plus information from seismic, cores, and biostratigraphic analysis. Other times these data are uncertain or unavailable, and we may have only the well logs themselves with little outside information. Further, any number of geologic factors may change from one well to the next, including geologic structure and lateral changes in lithology, porosity or bed thicknesses, each of which may drastically alter the log's response. As exploration for hydrocarbons continues to encounter more complex geologic traps, both structural and stratigraphic, the classic correlation problem only escalates in difficulty. In the past, petroleum geologists have correlated well logs by hand. The geologist lines up two traces and matches events by eye, incorporating local

knowledge of geologic structures and lithologies into the matching. As the data from a logging run continues to grow in volume and complexity, automated correlation methods become more attractive to reduce the burden on the scientist and to speed up and improve the matching process.

Testerman (1962) first proposed automatic correlation based on comparing permeability means. Although it laid the foundation for subsequent work, his methods did not analyze geologic sequences, the way in which modern log correlation is performed. One of the first attempts to automatically correlate sequences of geologic data was made by Sackin, Sneath, and Merri-man (1965), using cross-association. As more powerful computers allowed quick computations of Fourier Transforms, a number of spectral correlation techniques were presented. Rudman and Lankston (1973) used the statistical cross-correlation coefficient as a measure of best correlation. They iteratively stretched and lagged one shorter log relative to a stationary, longer log and looked for the peak in the cross-correlation function. Cheng and Lu (1985) performed correlation of waveforms by representing them as 'trees' describing the various features of the peaks and troughs. These relational trees are a more sophisticated way to represent features of a time series, and although they showed good results when large distortions occurred between traces, the correlation algorithm is still too complex for the large amount of data necessary for well log processing.

All statistical cross-correlation methods share a number of difficulties. As Robinson (1978) points out, the peak in the cross-correlation function is highly dependent on window length. Thus, many restrictions apply to the amount of stretching and iteration allowed for specific window lengths. In addition, random white noise contains a mix of all frequencies, and thus noisy logs will correlate poorly with any section of any log. Perhaps the greatest pitfall of statistical techniques, however, is the inability to handle correlation across missing or discontinuous rock units. This is such a common situation in geological sequences that it must be handled by any automated technique that is to be applied in a variety of areas. The dynamic depth warping method developed in this thesis can handle this and other problems by incorporating geologic knowledge.

Dynamic programming provides many advantages over more classical solutions to the correlation problem. Although widely used in biological sciences to match strings of DNA (Delcoigne and Hansen, 1975) and in speech processing for speaker identification (Myers, 1980; Sakoe and Chiba, 1978), it has barely crept into the geophysical literature. Smith and Waterman (1980) were the first to use dynamic programming in the geosciences as a technique to allow for insertions or deletions of rock units. Other early methods using dynamic programming in well-to-well correlation were proposed by Gordon and Reymont (1979), Anderson and Gaby (1983), and Kerzner (1983).

Anderson and Gaby (1983) pointed out that automated correlation techniques, no matter how powerful, will have very limited applications unless they can be guided in some way by geologic knowledge. Geological layers in the earth, although represented as 'series' on geophysical logs, do not represent continuous time sequences. Thus, correlation techniques that exploit properties of continuous time signals are doomed to failure when applied to geologic records. Wu and Nyland (1986) and Startzman and Kuo (1986) used artificial intelligence techniques to input geologic knowledge into the correlation. They proposed automated correlation of entire zones by pattern recognition, and used a knowledge-based system to eliminate correlations that were geologically unreasonable. Their method takes an important step in solving the correlation problem— to find ways of imitating the thought process of the geologist, while at the same time exploiting the speed and flexibility of the computer.

In this study we develop a new automated well-to-well correlation technique which bridges the gap between statistical techniques and the geological inputs to the correlation process. This new approach uses dynamic programming to allow for complex correlations across geologic structures and discontinuous units, and employs 'expert systems' programming to allow the correlation to be guided by a geologic knowledge base. The geologic knowledge base contains models of depositional environments, and rules relating these models to different correlation strategies appropriate for that environment. This approach provides two improvements over existing techniques. First, it will allow large-scale matching of the lithologies in the wells based on the depositional environment and the geologic structure. Second, it allows use of a virtually unlimited number of logs in each correlation, and the ability to weight the significance of these

logs based on geologic information.

The Dynamic Depth Warping method is described in chapter 2, which outlines the basic theory behind dynamic programming, and shows how constraints can be applied to improve the performance of the method. Chapter 3 will show that geological constraints, which are commonly applied to correlations, have natural extensions into the dynamic programming algorithm. Some examples will illustrate how geologic information from the knowledge base is used to complete an initial match of the wells based on the depositional environment. Chapter 4 demonstrates how dynamic depth warping worked with field data from offshore basins in West Africa and in Turkey.

Chapter 2

Correlation with Dynamic Depth Warping

2.1 Dynamic Depth Warping

The main failure of previous automated correlation methods is their inability to correlate sections with discontinuous or rapidly varying rock units. This is because a match in such a case would result in a non-linear transformation of the depth axis of one well with respect to the other. In addition to being computationally tedious, simple linear log stretching by interpolation does not always represent the geologic process.

The same problems that arise in well log correlation also appear in isolated word recognition problems for speech processing. In order to properly compare two waveforms, temporal variations between the two patterns, due to differences in the way a speaker may say a word on different occasions, must be accounted for. These differences may include absolute differences in the length of the utterances, as well as local variations where one portion of the word is sped up relative to the same section in the reference word. These problems are exactly the type we wish to solve in well to well correlation. In speech processing, the problem is overcome by a pattern matching technique known as Time Warping.

Dynamic Time Warping (DTW) or Dynamic Waveform Matching (DWM) as described by Anderson and Gaby (1983) is a generalization of cross-correlation useful in pattern matching and feature extraction. While statistical cross-correlation schemes match two waveforms by continually shifting the time axis of one waveform relative to another, DTW deforms the test waveform by warping and shifting the time axis until it best matches the reference waveform (Figure 2). The resulting warping function, shown as $W(n)$ in the figure, can be considered a mapping function which ties corresponding points on the curves. The advantage of correlating with the warping function is that non-linear translations of the time axis are now possible. As the next chapter will show, this feature when applied to Depth Warping will allow correlations in complex geology where conventional cross-correlation would fail.

Since there are many possible warping functions, but only one that performs the correct match, we must use Dynamic Programming techniques as proposed by Bellman (1960) to explore all of the possible warping paths. The optimal warping function will be the one that completes the match while minimizing some distance function, known as the distance metric, which represents the “cost” of performing the match. In keeping with Bellmen, the “distance measure” and the “cost” are used interchangeably throughout this thesis. Thus, the best warping function will be the one that performs the match with the minimum total cost.

Since the problem of log correlation concerns “depth” series as opposed to time series, the term “Dynamic Depth Warping” is used. Later sections will reveal that this distinction is valid because warping functions for depth series produced through geologic time have different characteristics than the typical warping functions for time series.

2.2 Discrete Curve Matching

In the dynamic depth warping algorithm, the warping function $W(n)$ of figure 2 is never solved as a continuous function. Instead, the warping function can be represented as a path finding problem consisting of discrete moves through a matrix (Figure 3). In figure 3 we see two discrete curves, separated by a matrix of points representing the possible indices that a warping path

may touch. This matrix of values represents what is called the “search space” or the “global area”. In other words, all legal warping paths must be contained in this area. Each point in the global area has a value associated with it, this being the total cost or distance required to match the two curves up to that point. A quick look at the matrix reveals that of all the total cost values, there is a path through the matrix that continuously runs through the cheapest or minimum-distance values. This minimum distance path is the discrete version of the warping function of figure 2. Once this path is found, tie lines connecting appropriate features can be drawn from well to well. For those readers who are not familiar with dynamic programming methods, Appendix A contains a more detailed description.

2.3 Constraints on the Dynamic Depth Warping Algorithm

Before we can solve the dynamic curve matching as a pathfinding problem, there are several parameters which must be considered:

1. Endpoint Constraints – the way the path begins and ends in the matrix.
2. Local Continuity Constraints (or slope constraints) – restrictions on the motion from one point to another in the matrix.
3. Global Path Constraints – limitations on where the path can wander in in the global space.
4. Axis Orientation – the effects of interchanging the roles of the test and the reference patterns.
5. The local distance measure (or distance metric)
6. Weighting and Normalization

The following sections will discuss these properties as they apply to dynamic depth warping, and discuss examples of how geological constraints can be imposed on the matching process

through these parameters.

2.3.1 Endpoint Constraints

In speech problems with word recognition, breath noise at either end of the speech waveform can cause inaccurate determination of beginning and end points. In well log correlation with DDW we assume that the beginning and ending are determined such that the regions we are interested in correlating in one well are present in the other well. In other words, we count on DDW to correlate the appropriate features in each well, as long as we provide some buffer zone on either side of the regions of interest. Thus in this application, all depth warping algorithms should be restricted to have all paths start at the point (1,1) (the first depth point in each well), and end at the point (N,M) (the final depth values in each well) as shown in figure 3.

One possible alteration of these constraints is to allow some zone over which the path could begin and end (Figure 4). This is a relaxed endpoint constraint, and would be appropriate if we had some isolated smaller unit in one well, and wished to discover the corresponding unit in a series of units in a longer well. The warping procedure would then be equivalent to the “local minimum DTW” algorithm proposed by Myers (1980) for applications in connected word recognition. This algorithm requires a number of warps proportional to the number of relaxed points at the ends. Considering the amount of depth points used in the average correlation for the wells in this thesis, relaxed endpoint algorithms are considered too costly for their limited application and are not considered. As Anderson and Gaby (1983) point out, however, relaxed endpoint algorithms would be essential in certain geophysical applications such as seismic signal detection, where the waveforms end by gradually fading into the background noise.

2.3.2 Local Continuity Constraints

Local continuity constraints, or simply local constraints, are restrictions on how the warping path can move from point to point in the global matrix. There are a number of possible continuity constraints for dynamic depth warping, and they can be adjusted, even within the program,

for specific applications. All of the constraints, however, require that the warping function (or running cost) be monotonically nondecreasing. In other words:

$$W(n) \geq W(n - 1) \text{ and}$$

$$W(m) \geq W(m - 1)$$

When the warping path is moving from one point in the grid to another, a local constraint has three basic features:

1. Slope Restrictions – The previous points which the path is allowed to come from.
2. Weights – Different directions of the local motion may be weighted accordingly to bias the path in a specific direction.
3. A memory – The amount of points backward in the grid the path is allowed to come from. This is termed memory since it determines how many columns of data must be kept in memory for local distance calculations.

Appendix B shows the production rules associated with a word matching problem, where the sequences are strings of letters. The three possible edit operations, insertion, deletion, substitution, or no change, corresponded to three possible motions from point to point in the grid. In this case the slope of the motion is restricted to three directions– horizontal, vertical, or 45 degree diagonal. In discrete curve matching the same rules apply. Vertical motion indicates deletion of points in the first curve $T(n)$, and horizontal motion corresponds to deletion of points in the second curve $R(m)$.

Figure 5 shows some possible slope constraints and their associated weights and production rules. Chapter 3 will discuss the various ways in which local geology can be used to choose a local slope constraint and weighting appropriate for the algorithm.

2.3.3 Global Path Constraints

Global path constraints are restrictions on where the path can wander in the $M \times N$ grid. They are imposed to make the warping algorithm more efficient, and to eliminate paths that do not make sense based on the local geology. Global path constraints are achieved in three ways:

1. Internal Tie Points – where the search space is limited by forcing two points in the curves to match.
2. Local Slope Constraints – where local constraints are manifested throughout the range as global constraint.
3. Range Constraints– where there is some absolute restriction on how much two points can be shifted and still be allowed to correlate.

Global restrictions in the path can be forced if we specify beforehand that two points in the curves will match, this is equivalent to forcing the warping path to go through one particular point in the grid. Since the **dynamic programming principle** applies to the minimum distance path (see Appendix A), the resulting path from the tie point to the end points will only depend on the path chosen so far in the match. Thus the larger path problem is broken into two smaller pathfinding problems with an intermediate point, reducing excessive calculations over paths that are no longer available (Figure 6).

Local continuity constraints force certain points out of the search space to be excluded from legal paths because they would require excessive expansion or compression of the depth axes. For example, the type II local constraint of Myers (1980) (Figure 5) restricts the path locally to slopes of $1/2$, 1 , or 2 at any point in the grid. This forces the path to lie within a region bounded by lines that leave the points $(1,1)$ and (N,M) with slopes 2 and $1/2$, representing the parallelogram of figure 7.

Finally, global path constraints are forced from local continuity constraints by specifying the restriction that at any point in the match

$$|n - m| \leq R$$

In other words, no two points separated by a distance greater than R will be allowed to match. This restriction manifests itself in the global path as two parallel lines running from the points $(1, 1 + R)$ and $(1 + R, 1)$ through the grid, further reducing the allowable global paths (Figure 8).

2.3.4 Axis Orientation

The local path constraints discussed in section 2.3.2 can be asymmetric. If this is the case, complications arise in the warping path such as one waveform being warped easier than another, or longer paths being favored over shorter ones. In this case, whether or not the test or reference waveform is placed on the x-axis becomes a consideration. For example, consider the type I continuity constraint of figure 5 with non-symmetric weighting, as shown in figure 9. A smaller weight on horizontal motion makes it easier (less costly) to delete section from the x-axis well, if we expect from the geology that this may occur. Obviously, if we interchange the wells, deletion of the appropriate sections will become very costly, and a large bias may force an incorrect solution.

Axis orientation can also be a problem when we consider the global cost as a measure of the matching quality (Section 2.7). If the local continuity constraint is asymmetrically weighted, longer paths may be warped with a cheaper total cost than shorter ones, thus biasing the optimal total cost toward longer paths. Again, interchanging the two axes can have drastic effects on the final matching cost, and an incorrect switching of the axes can cause erroneous solutions.

For the examples studied in this thesis, only symmetric local constraints are considered. For convention, however, the longer log is chosen to be along the x-axis or horizontal axis. Thus, the optimal warping path will most often end along the bottom of the global matrix, making it easier for the program to search for the correct minimum path.

2.4 The Distance Metric

2.4.1 Distance Metric for Logs

Essential to the dynamic programming approach to correlation is the assumption that all relevant differences between each pair of objects being matched may be summarized by a single measure of pairwise dissimilarity. In other words, we must be able to measure how closely the two logs are related at any depth point in two different wells. In dynamic programming methods this measure is called the distance metric. The distance metric when applied to matching problems is a measure of how similar two objects are: the greater the similarity, the shorter the distance, the lesser the similarity, the greater the distance between them. The idea is simple enough, but the subtleties of this quantity and how it is measured is the backbone behind an efficient dynamic programming algorithm. This section describes the basic attributes of this distance measure, and how we can use this measure to incorporate geologic knowledge into the machinery of the algorithm.

In order for our distance metric to supply us with a good measure of how closely two objects are related, it must first satisfy the requirements of a general distance measure (Findler and Leeuwen, 1979). If we define $d(a,b)$ as the distance between two objects, these requirements are:

1. That $d(a, b) \geq 0$
2. That the distance should be irrespective of the ordering of the objects: $d(a, b) = d(b, a)$.
3. That $d(a, b) = 0$ if and only if $a=b$.
4. That the distance over any other path must be longer than or equal to the minimum distance path: $d(a, b) \leq d(a, c) + d(c, b)$ for any object c .

Every specific application of dynamic programming has its own distance measure. In speech pattern recognition, it is often a log-likelihood ratio between the cepstral coefficients of the two

words that are to be matched. (See Sakoe and Chiba, 1978 for a more detailed description of isolated word recognition.) In molecular biology, it is a match of amino acid substrings to similar strings in another protein (Sankoff and Kruskal, 1983). Gordon and Reymont (1979) proposed that an obvious quantity for well log correlation would be the absolute value of the difference between the readings of one kind of log, at one particular depth in the two different boreholes. If we consider a depth point n in well A and another depth point m in well B, then the local distance between them would be given by:

$$d_i(n, m) = |A_i(n) - B_i(m)| \quad (2.1)$$

where $A_i(n)$ represents the value of the i 'th log at the depth n in well A, and $B_i(m)$ represents the value of the same log at the point m in well B. Figure 10 illustrates how the local distance measure for logs is effective at identifying changes in lithology.

2.4.2 Multiple log distance measure.

One of the first decisions that one makes when doing a stratigraphic correlation is which logs to use in the process. The most important variable for stratigraphic correlation is lithology. Logs chosen for correlation must be diagnostic for lithologic properties in a given geological environment. For example, in most paleoenvironments of interest to petroleum geologists, the shale units cover the largest geographical extent. One would assume, therefore, that if our goal is to correlate wells separated by a substantial distance we would be interested in the shales as marker beds instead of the sands. For this reason, the ability of a correlation program to use logs such as interval transit time, bulk density, or gamma ray is essential.

A more fundamental reason for using multiple logs is the effect of "stacking", or enhancing the signal to noise ratio by combining signals over a given range. If a formation boundary is not evident in one log, it may show up as a deflection in another log. In this way, by correlating with all of the common logs, we can be assured of matching as many common characteristics

between wells as possible.

We can modify our local distance measure to consider multiple logs. If we are matching two wells A and B, each with $i=1, \dots, k$ logs, then

$$d(n, m) = \frac{\sqrt{\sum_{i=1}^k |A_i(n) - B_i(m)|^2 W(k)}}{k} \quad (2.2)$$

Where $W(k)$ is a family of weighting coefficients. This weighting can be used to:

1. Normalize the logs to some arbitrary scale.
2. Adjust the confidence level assigned to the log based on local geology or some measure of the noise in the log.

Figure 11 shows a diagram of the multiple log distance matrix as used in the COREX program. Notice from the figure that “cross-correlations”, or matches between logs of two different kinds, are possible with the multiple log approach. In this thesis, however, cross-correlations are only used if logs of the same type are so noisy as to require matching with another type of log.

The multiple-log distance measure allows us to compare a large number of logs simultaneously with only a slight increase in computer time. A correlation with six logs requires roughly six times as many computations as with one log, whereas the same comparison using spectral techniques would require 2^6 times as many calculations. This multiple log distance measure, with the cross-correlation feature, may prove useful in the future when spectroscopy tools become more widely used. In this case, correlation would require the simultaneous comparison of many vector quantities.

2.5 Weighting and Normalization

The previous section showed that we can adjust the influence of each log on the correlation. This is done through the family of weighting coefficients shown in equation 2.2. First, the weighting coefficients are used to normalize each log with respect to its own maximum and minimum to a scale from 1 to 100. The reason for this is twofold. Normalization of logs is required to account for amplitude differences between corresponding logs in different wells, and between two different types of logs in one well if we were interested in depth matching. Second, the hundred-point scale allows the correlation program to use integer arithmetic only, without a loss of resolution. In LISP programming, integer variables are computed faster and require less memory than floating point numbers.

We can also adjust the weighting coefficients to account for uncertainties from noisy logs. Once the logs are properly normalized, they are low-pass filtered with Walsh Transforms. The Walsh filtering reduces all the logs to a common minimum resolvable bed thickness, which is taken to be 1.2 m. Appendix C details how the filtered logs are then compared with the unfiltered versions to obtain a measure of the noise content of the log.

For adequate weighting, the program must know the maximum amount of noise that is allowed in a log before the noise begins to deteriorate the correlation results. Experiments with the depth warping algorithm show that this noise “threshold” is dependent on the the maximum signal deflection across a bed, and on the number of data points in the feature. In other words, if the logs are sampled every .5 meters, a 1 meter thick bed deflection is smeared by noise much easier than a 3 meter deflection. This is because the warping algorithm often checks neighboring points in local distance calculations, and a spike deflection may get missed when surrounded by noisy data. If we take some maximum deflection amplitude, like the deflection in the gamma ray response between sands and shales, and progressively add random noise, then the noise threshold will be a function of the feature thickness only. Examples with synthetic logs show that once a deflection reaches three data points in thickness (a 1.5 meter bed in .5 points/meter sampling), the noise threshold is no longer a function of bed thickness. The results of the noise analysis are summarized in Table 1. If we assume that all of the beds are

adequately sampled, then 70% noise is taken as the point where the log is considered too noisy to be used in correlation. In the synthetic examples, signals due to bed boundaries were no longer distinguishable from noise in logs with 70% added noise.

Num. of data points in bed.	% noise where boundaries are missed	Signal / Noise
1	35	2.83
2	50	2.125
3	70	1.41
4	70	1.41

Table 1. Effects of noise on Depth Warping resolution.

Once the noise measure is determined for each log, the weighting coefficients of equation 2.2 are used to adjust the influence that each log has on the correlation. At present, only a 0 or 1 weighting is used. Attempts to adjust local distances of individual logs only seem to make all points more similar, and hence the program suffers a loss of resolution. As chapter 3 will show, an “on” or “off” weighting is adequate in many applications.

2.6 Smoothing the Distance Metric

Just as we can filter the logs to minimum resolvable bed thicknesses by Walsh transforms, we can use a smoothing technique on the local distance measure to effectively filter out small deflections in amplitude. As previously discussed, in each trace there will be an amount of deflection that is due to noise, and not due to the “signal” of a changing lithology. We can determine this average deflection from the noise analysis either by Walsh or Dynamic Depth Warping analysis. Then a rule is imposed on the local distance metric which determines if the measured local distance is below this threshold of noise. If this is the case, then the matching distance is set to zero, forcing the points to match. If the local cost is above the noise threshold, it is considered to be signal and is weighed accordingly. If $d(n, m)$ is the local cost for matching

the points n and m , and T is the noise threshold, then the smoothing operator is:

$$d(n, m) = \begin{cases} 0 & \text{if } d(n, m) < T \\ d(n, m) & \text{otherwise} \end{cases}$$

Thus, we can use the smoothing operator to force the program to look only for signals above a certain amplitude, which can be equivalent to searching for bed boundaries. Figure 12 shows the effect of this smoothing operator on the correlation and warping path of a synthetic log with 50% noise added to the signal.

2.7 Computational Savings in Dynamic Depth Warping

In statistical cross-correlation, the degree of match is reflected in the amplitude of the peak in the correlation function. In dynamic depth warping, the analogous measure is the total matching cost. Dynamic programming will find a solution at any cost, but two identical curves will have a total matching cost of zero, whereas two very different curves will require a high total matching cost. In the same way, matching very noisy logs will result in a high total matching cost (Figure 13). Thus, if we are only interested in the quality of the match, as we would be if we were matching test waveforms to a set of references, then we would only record the total cost. This method would be appropriate for waveform clustering or waveform classification schemes. Dynamic programming can proceed in this case with a very limited amount of computer memory. However, as the next chapter will show, in well-to-well correlation and many other geophysical applications of dynamic programming the warping path has geological significance, and must be recorded. Dynamic programming in this case requires permanent storage of a matrix that has the same dimensions as the global area ($N \times M$).

Myers (1980) shows that for an average match the computation time required for local distance calculations is 300% more costly in terms of computer time than dynamic programming decisions (combinatorics). For example, a correlation of $N \times M$ points, requires $N \times M \times 3$ calculations because each decision requires three local distance calculations. Thus, if we are

concerned with an efficient depth warping algorithm, we should concentrate on limiting the number of local distance calculations, or in other words, limiting the global area. As the next chapter will show, global path constraints like those discussed above are necessary to make large correlation problems computationally feasible.

2.8 The COREX System

The computer program written to perform log correlation using dynamic programming and artificial intelligence is called COREX. COREX is written in LISP using "object-oriented" programming. Object oriented programming means that computer code is used to structure, store, and retrieve information about objects in much the same way that humans do in their mind. With this programming, the COREX system can store geologic information in a fashion that will allow an easy exchange of facts between the knowledge base and the correlation algorithm. The system is described by the diagram in figure 14, and has four main parts:

1. A database that contains the well logs, available lithology information, a seismic cross-section, and dipmeter information.
2. A dynamic programming algorithm which performs a minimum cost match between the wells based on the data.
3. A knowledge base containing models of depositional environments and their common lithologies, relations between structural information and dynamic programming constraints, and relations between specific lithologies and their local distances.
4. A set of rules that manipulate the geologic data and impose constraints on the dynamic programming algorithm.

The COREX program provides the interpreter with the ability to use all of the information usually available for a prospect, including digitized well logs, seismic sections, and lithologic interpretations. From the raw digitized logs, COREX can walsh-transform and filter the logs, and

then perform a noises analysis in each well. The lithologic inversion for the wells is preformed in a separate routine on a VAX/11-780, and is stored in the COREX data base for use in the initial lithologic match. When the initial correlation is complete, this match is then combined with the seismic tie lines to constrain the dynamic programming algorithm. For seismic data, the information in the seismic section is reduced into a series of tie points with depth in each well. Structural dip can either be inferred from these tie points or entered from a file containing interpreted dipmeter information. The dynamic programming then uses the raw digitized logs, which have already been filtered and normalized, to do the final correlation.

The knowledge base of COREX contains structured geologic knowledge about depositional models including common lithologies, average horizontal and vertical scales, and the continuity of each facies present in the environment. The knowledge base also contains rules which govern how the geologic information will interact with the dynamic depth warping algorithm in the form of constraints. For a more detailed description of the flow of the COREX system, see Appendix B.

Chapter 3

Using Geologic Knowledge in the Match: Examples

This chapter discusses the various ways in which the COREX system uses geologic knowledge to aid the correlation. First, we will see how the system uses geologic depositional models stored in the knowledge base to perform an initial match on the wells. Examples illustrate how the continuity of different lithologies depends on the depositional environment, and how this dependence can alter the correlation strategy. Second, we see how the various dynamic depth warping constraints outlined in chapter 2 are used to incorporate local geologic information into the correlation. Examples illustrate how constraints can be imposed to include both structural and stratigraphic information.

3.1 The Initial Match

The first step COREX takes in the correlation process is a coarse matching of the lithologies present in each well. For each depositional model, the system assigns some measure of continuity to a particular lithology, and from these it infers similarity or distance measures for the initial match. A question arises, however, of whether to perform lithologic correlation on a small scale,

where isolated strong deflections should be correlated, or on a scale where larger patterns are mapped from well to well. Thin shale units may be excellent marker beds in one area, but only isolated lenses in another area. This scale problem is entirely dependent on the lithologies present in the wells, and the geometry of these units in relation to common deposition models. Lithologic correlation for COREX requires two pieces of information from the user before the matching process can begin, the depositional environment under consideration and the scale of the correlation problem in relation to this environment.

3.1.1 The Environment Consideration

In most ancient sedimentary environments of interest in exploration, shale units represent conditions of slow deposition and calm water. For this reason they are more continuous in time (vertically) and geographical extent (horizontally). Sandstones, on the other hand, represent conditions of quick deposition that are spatially isolated to areas such as river-mouths or point bars. A reasonable correlation strategy in this case would be to correlate the shale units first and then the sands.

Situations arise, however, where this strategy should be reversed. In modern braided stream environments, for example, it is the sand units which are laterally continuous and the shales which form as isolated pods (Figure 15). A correlation strategy can also change as a result of more local geologic factors including the strike of the paleoshoreline. In submarine fan and delta lobes, sand bodies are fairly continuous parallel with the shoreline. If we were correlating in tide-dominated sand ridges, however, this situation would be reversed (Figure 16). Another example is a limestone barrier reef, or the wave-dominated beach sand deposits, which both have great continuity parallel to the ancient shoreline, but are only a few tens of meters wide in the perpendicular direction (Figure 17). Obviously, any lithologic correlation strategy is specific to the paleoenvironment. If we can specify the geometry of the correlation problem with respect to a depositional model for the environment, highly successful correlation will result.

3.1.2 The Scale Consideration

After the determination of the environment, the second decision the system makes is the scale of the correlation in relation to common geometries for the environment in question. What are “thick” units for one environment are not necessarily thick units in another environment. If, for example, the correlation is in a deltaic environment where a complete vertical sequence commonly spans 100 meters, and the correlation interval is hundreds of meters, then an obvious strategy would be to first correlate marker beds which cap isolated vertical sequences, and then let units within each individual sequence correlate as the numerical results dictate. If, on the other hand, our vertical correlation sequence spans only tens of meters, correlation should follow individual sand lobes, or other locally continuous units.

The same analysis proceeds in the lateral direction as well. One of the main considerations in any correlation strategy is the possibility of spatial aliasing. In other words, do we have well spacing dense enough so that we can reliably trace units from well to well. As figure 18 shows, if correlation is in a meandering stream environment where the wells run parallel to the rivers paleoflow, well spacing of a roughly 1-2 km is adequate to trace point-bar sands. In the same environment, however, well spacing on the order of 8-10 km is not enough to reliably map sandstone units from well to well. If we were correlating in beach sands, however, and running parallel to shoreline, 8 - 10 km well spacing may be adequate. Thus, the problem of spatial aliasing must be addressed with respect to depositional models, and the continuity of particular facies in these models. Obviously, tight well control on a scale of .5 to 1 km. in any environment would insure more meaningful results.

3.1.3 The Distance Metric for the Initial Match

As described above, shale units are generally more continuous than sands, and thicker units are generally more continuous than thinner ones. The knowledge-base of COREX has a generic hierarchy of continuous units structured by these rules. If no specific information about the depositional environment is available, the program use these general rules to perform an initial

match. It also has provisions for specific lithologies such as coals or volcanics, which if indicated by the user, are taken to be marker beds. Figure 19 shows how designation of a marker bed can alter a correlation. This information is stored as a table which contains all of the possible distances or local costs between units, as shown in figure 20. The most continuous units are assigned a matching cost of zero, whereas units that are not expected to match at all are given the default cost of 100. For each correlation problem, this matrix is calculated based on a depositional model stored in the knowledge base. If a shoreline is meaningful in this environment, then rules are run concerning the positions of the wells with respect to the strike of the ancient shoreline. In this way, the local distances can be modified to include lithology changes which may occur down the paleoslope, or changes in continuity of specific units due to the strike of the shoreline. Once the rules are processed, dynamic programming proceeds as outlined earlier, using the distance measures of figure 20 to calculate the costs. The results of this initial match are then used as constraints to guide the rest of the dynamic depth warping. Figure 21 shows an example of an initial match of lithologies including the matched units, the computed global cost grid, and the corresponding warping function that performed the match. In all of the initial matches, a simple type I local constraint is used without weighting.

3.2 Geologic constraints on Dynamic Depth Warping

Once the initial lithologic match has been performed on the wells, we can use the coarse correlation to constrain a more detailed correlation. The initial match will give a series of tie points that will force the final correlation to make sense based on the geologic environment. When we combine these tie points with structural information from the seismic data, we can further use the geology to guide the match, at the same time saving unnecessary computation. Geologic conditions can be imposed from the knowledge base to the dynamic programming through global path constraints, local path constraints, and weights in the local distance metric.

3.2.1 Geologic Expression in the Warping Path

Before discussing how we can force the matching to incorporate geologic knowledge, we should first see how simple geology can affect the matching. Figure 22 shows how geologic structures can define the warping path which matches two wells. Recall from section 2.2 that purely horizontal motion corresponds to deletion of section in well A relative to well B, vertical motion corresponds to deletion in well B, and diagonal motion corresponds to simple stretching of A relative to B. A 45 degree diagonal corresponds to a perfect match between the wells, and a curved path results from non-linear stretching between sections. In figure 22a, for example, correlation across a normal fault with a throw of 200 meters would result in a simple shift of well A relative to well B. To accommodate this shift, the correlation will effectively delete all of the section in well A (200 m.) that is not in well B. Thus, the warping path would proceed along the edge of the matrix perpendicular to the section being deleted, until it reached the point where the wells would begin to match perfectly. At this point the path would be a 45 degree diagonal, proceeding to the lower edge of the matrix. In figure 22b, when a growth fault separates the two wells, the amount of stretch and shift is no longer constant, but instead increases as a function of depth. In this case the warping path will be a curved line, concave toward the longer section.

Notice from figure 22 that a wide variety of structural situations can be accounted for by only a few different restrictions on the warping path— namely, horizontal or vertical motion, diagonal motion with some slope, and curved paths which will most likely be restricted to one half of the global space. Thus, it becomes simple to program very general structural rules into the dynamic depth warping algorithm. Figure 23 shows the correlations resulting from synthetic logs generated to represent the structures of figure 22. Correlations were performed on wells separated by a normal fault, and on wells where sandstone units are missing from each well. Also shown are the warping paths that resulted from the match. As the figure shows, the paths are very similar to those discussed in figure 22. Departures from the synthetic examples and the theoretical paths can be accounted for by noise present in the logs. By removing the noise from the logs, we effectively smooth the warping function to the straight lines shown in 1

figure 22.

3.2.2 Global Path Constraints

When a correlation is done by eye, one uses information that reduces the problem into smaller, more manageable parts. An obvious example is an easily recognized “marker” bed or a seismic well tie which is a known correlation point. This would be equivalent to forcing the warping path to go through one point in the grid, thus reducing the global area as seen in chapter 2. Another example would be limitations on the amount of shift between the wells based on the geology. This would show up in the warping path as a range constraint as mentioned in section 2.6 (Figure 8). If the maximum allowable shift is 200 m, then any points separated by more than 200 m distance automatically are assigned an extremely high distance value so that the minimum-distance path will never match those two points. These types of rules are easily input into the dynamic depth warping algorithm, and in doing so they can result in vast savings in computation time and computer memory. Figure 24 shows how regional tie lines and shift constraints greatly reduce the number of possible warping paths. A correlation with t well-spaced tie lines can reduce the effective number of calculations to roughly $1/(t + 1)$. Thus, large correlations with many logs over thousands of feet of section can be reduced to a manageable size with only a few simple constraints.

The following is a sample structural rule that would lead to a global path constraint on the correlation:

- **If** there is a normal fault between the two wells with a throw of 80 meters,
and the beds in the areas dip less than 5 degrees,
and the well separation is less than two kilometers,
- **Then** start the warping path 80 meters into the downthrown well,
and impose a maximum shift constraint of 40 meters.

Figure 25 demonstrate the three ways in which global path constraints are used in the correlation process. Figure 25a represents the simple correlation problem with two wells and no information to guide the match. In the last section we saw how an initial match is performed on the wells, which results in a series of tie points between the wells. Figure 25b shows how the problem is then broken down with these tie points into a series of smaller problems. The program then looks at the structure and lithology of each individual section and decides upon further constraints. In section 1, for example, the section to be matched in well A is much shorter than the corresponding section in well B. As seen in figure 22, a warping path in this case should be some kind of diagonal line through the matrix, but one which does not wander into the upper section of the global area. In this case, we can impose the global constraint that the path is not allowed to wander above the 45 degree diagonal, further limiting the possible warping paths. In another section a maximum shift rule may apply, imposing yet another constraint. Figure 25d shows the final problem for dynamic programming to solve. The initial problem required $2000 \times 2000 \times 3$ or 12,000,000 operations. After the geologic input it is reduced to roughly 1,500,000 calculations, and the chances of a geologically meaningful result have been increased tremendously.

3.2.3 Local Path Constraints

Another way in which geology is imposed on the dynamic depth warping algorithm is through local path constraints. As seen in section 2.3.2, different local continuity constraints, or local restrictions on how the path can move from point to point, can be applied to the algorithm. By weighting different local motions that the path can take, we can control how the path will wander in the global range. For example, figure 26 shows how different weighting of the simple type I local constraint of section 2.2 can affect the warping path. If we assign some extremely high cost to the horizontal motion, then as the dynamic programming searches for the locally cheapest move, it will never choose horizontal motion. If this weighting is carried out over the entire path, we will get a matrix of distance values with extremely high values above the 45 degree diagonal, and lower values below. The warping path will have no choice but to proceed somewhere in the lower section of the grid.

Another way to restrict the path is to choose an entirely different continuity constraint. If we use the type II constraint of section 2.3.2, where the local motions are restricted to slopes of 1/2, 1, or 2, this restricts the path to lie within a parallelogram as shown in figure 7. If we can determine from the structural information that no great expansions or contractions occur between the wells, then we can use a modified continuity constraint such as the type II to further increase the chances that our result be geologically meaningful. The following rule illustrates how the geologic information could lead to modifications of the local continuity constraints:

- If the two sections to correlate are the same thickness,
and the structural dip between the section is less than 5 degrees,
- Then choose a type II constraint,
and impose a maximum shift of 20 meters.

3.2.4 Weighting the local distance metric

The final way in which the system imposes knowledge on the dynamic depth warping algorithm is through the local distance metric. Equation 2.2 showed that a family of weighting coefficients is used to alter the influence that particular logs had on the correlation. For example, if one log is not diagnostic in a particular formation, or is determined to be too noisy, it will be weighted accordingly. If the program expects to correlate down a paleoslope where shaliness would increase away from the shoreline, then a sandy-shale can be made to correlate with a shaly-sand of the same dimension. Before the matching process begins, the COREX program runs through rules concerning the lithologies and the noise analysis of the logs to adjust the local distance measure between points. Figure 27 shows how the correlation can be improved by modifying the local distance measure because of noisy logs. The following is a sample rule that would alter the local distance measure between two points:

- **If** we are in a meandering stream environment,
and our lithology is a thin shale,
and the caliper log contains a spike at this point,

- **Then** we expect the log response to be altered by a washout,
and the local distance should be increased by 50.

A more complete description of the rule structure used in the COREX system is given in Appendix D.

Chapter 4

Results

4.1 Field Example from West Africa.

Figures 28–31 show an example of correlation on log data from two offshore wells in the Western margin of Africa. In this study, well 1 contains 250 depth points and spans 250 feet (150 m) of section, and well 2 contains 225 depth points and spans 225 feet (135 m) of section. Each well had five logs available for correlation— gamma ray, sonic, density, neutron porosity, and resistivity. Each of the logs was normalized from 0 to 100, and then walsh-filtered to a minimum resolvable bed thickness of four feet (1.2 m). The depositional environment is interpreted to be deep marine, along the continental shelf of the passive East Atlantic margin of West Africa. Figure 28 shows the results of an initial lithologic match between the two wells. As the figure shows, a section of sandy-shales in well 1 correlates with a section of shaly-sands in well 2. This is allowed by the program since the environment is a shelf and the strike of the well correlation line is roughly perpendicular to the shoreline. In this case, it is likely for a sand sequence to increase in shaliness away from shore, and hence the matching cost between the units is reduced.

Figures 29–31 show the resulting correlation, the final warping path, and the constrained global cost matrix for the West African wells. Also displayed are the basic assumptions the program made in the correlation process. Figure 29 shows that the program does very well

matching the particular sand and shale units across the wells, even when thickness changes are occurring. We can see from figure 29 the advantage of using multiple logs in the correlation. Looking at the gamma ray log, it appears that the program is making meaningless correlations over the depth range of 8300 – 8400 feet. Looking at the density log correlation, however, we see that changes in rock properties are occurring, even though they are not reflected in the gamma ray log. This is direct evidence of the need for multiple logs in the correlation.

In figure 30 we can see how the transformations from one well to the next are represented in the warping path. First, referring back to figure 22, we can see the superposition of different geologic factors on the warping path. Over the entire depth section, the beginning and ending regions match quite well in depth, which accounts for the general 45 degree diagonal trend in the warping path. Notice, however, that in a number of sections non-linear expansions and contractions occur between wells 1 and 2, and that this imposes a curved section on the warping path. Most notably, the strong radioactive zone near 8250 feet is expanded in well 2 relative to well 1, which shows up as vertical motion in the path. At 8500 feet, however, we have expansion of a sand body from well 2 relative to well 1, which shows up as horizontal motion in the path. Finally, we have an entire shale section at 8550–8600 in well 1 which is expanded to the section 8825–8925 in well 2, which shows as a significant curve in the warping path. A glance at figure 22 will show the similarity with the features displayed in parts (a) and (c) of that figure.

Figure 31 shows the global cost matrix that resulted from the constraints imposed by the initial match. As the figure shows, maximum shift constraints were imposed by the system for each section in the initial match. The original, unconstrained correlation required $250 \times 225 \times 3$ or 168,750 distance calculations, and took about 7 minutes. The final problem, after initial matching and shift constraints, required only 20,500 calculations, and took only 127 seconds.

4.2 Field Example from Thrace Basin, Turkey.

The dynamic depth warping algorithm is efficient because it can detect simultaneous changes in log character that are too small or too complex for the eye to detect. As a test of the small-

scale matching ability of the COREX system, we used a set of wells from the Thrace Basin, Turkey. These wells contain 500 and 456 depth points respectively, and span 250 meters and 225 meters of section. They were filtered to a minimum resolvable bed thickness of 2 meters, with an original sampling of 2 points per meter. These wells consist mainly of a shaly-sand sequence which is capped by a limestone, with little character displayed in the logs. Lithology changes show up as only small deflections on each trace and are very difficult to observe by eye. There are two noticeable features, however, one being the top of the formation coming in early in each well, and the other being a more subtle area of volcanic tuffs which are considered to be reliable time surfaces in each well. Both of these sections are shown in the lithology log on either side of the wells.

Here, no lithologic match was performed, but we imposed a maximum shift constraint of 80 meters. Because of the close well spacing in this section (1 km.), and the very small structural dip in the area, a maximum shift of 80 meters is a safe estimate. The results of the correlation are shown in figures 32–34, along with the warping path and the global cost matrix. As figure 32 shows, the top of the sandy-shale sequence is clearly reflected in the correlation. Close scrutiny of the correlation also reveals three individual volcanic beds there are traced from well to well. Another feature seen from the correlation is a general thickening of corresponding sections with depth from well 1 to well 2. This shows up in the warping path (figure 33) as a diagonal line through the matrix, with a non-linear trend toward the bottom sections. Looking at the warping path, we see two of the basic kinds of motions superimposed on the path. First, the length difference between the logs shows up as horizontal motion at the bottom portion of the path. In other words, an artificial stretching of well 1 relative to well 2, which shows up as a diagonal line through the matrix, as we saw in figure 22. The non-linear increase in bed thickness with depth that is reflected by the tie lines, however, curves the warping path from the upper left corner to the bottom of the match. Once again, the separate geological features show up as basic motions in the warping path. A comparison with figure 22 will show the similarity with parts (b) and (d) of that figure.

At this point we should mention the importance of imposing maximum shift constraints, whenever applicable. In addition to saving machine costs, they can also help the program locate

a minimum cost warping function. As figure 31 shows, there are oftentimes a number of paths that run parallel to the minimum path that are actually local minimums, not global. Even if the matching distances are calculated correctly, the program may get "lost" tracing its way down another path that is simply a local minimum Figure 35 shows an example of an incorrect correlation that results from such a case. In all of the synthetic and real data examples studied here, this factor accounts for almost all of the incorrect correlations that result from improper path tracing. The rest of the errors in back-tracing result from edge effects on the matrix. The global cost matrices are initialized with some cost proportional to the depth in the well. If the number of logs is large, or the distances abnormally high due to noise, then the initial cost penalty for distance may not be adequate, and the minimum distance path will simply run along the edge of the matrix. Whenever these path-tracing errors occur, they are very obvious in the final result, and some modifications to the trace-back procedure can be made easily.

Chapter 5

Discussion and Conclusions

The previous chapters presented a new approach to well-to-well correlation which combines dynamic programming and expert-systems techniques. Combining these two rather new geophysical techniques, we can overcome many of the fundamental problems which have hindered automated correlation in the past.

Dynamic Depth Warping alone has many advantages over conventional cross-correlation techniques, the largest among these being the ability to incorporate non-linear stretching between two sections to be matched. With a non-linear warping function, it becomes easy to handle missing or discontinuous units, and to correlate across common geologic structure. Chapter 3 showed how many different structural situations can be accounted for by only a few simple motions in the warping path— namely horizontal, vertical, and diagonal motion. Another great advantage of dynamic depth warping is the tremendous computational savings over conventional cross-correlation. Correlations involving as many as six logs in each well, with up to 500 depth points in each, can be completed in only a few minutes. When tie points or range constraints are imposed on the dynamic programming algorithm, even greater savings are accomplished, while at the same time increasing the likelihood of a meaningful result.

Perhaps the most promising aspect of the dynamic depth warping, however, is the ease at

which both large scale, and very local geologic knowledge can be input into the algorithm. This makes dynamic programming an excellent algorithm to be used in conjunction with artificial intelligence techniques. The global path constraints outlined in chapter 2 provide a natural avenue through which basin-wide geologic knowledge can be applied. On the other hand, very local geologic knowledge concerning lithologies or log quality from point to point can be applied through local continuity constraints or weights in the local distance metric.

The ability to apply knowledge from the geologic knowledge base to the matching algorithm provides another set of advantages over previous approaches to correlation. As chapter 3 showed, correlation strategies should change from basin to basin and even well to well, based on the depositional environment and the geometry of the wells in relation to this environment. Stored information in the geologic knowledge base allows the program to make decisions about the continuity of specific units, and how this continuity can change within a depositional model. From this the program performs a coarse matching of the lithologies in the wells, and uses this match to further constrain the dynamic programming. The knowledge base can also manipulate rules concerning geologic structures which force global range and slope constraints, and rules concerning log quality and formation sensitivity which will show up as weights in the local distance measure. Again, the kinship between local and global constraints, and local and basin-wide geologic knowledge make dynamic depth warping an ideal algorithm to use in conjunction with expert systems programming.

The field data examples from West Africa show that the COREX system can be used effectively on a large data set with many logs, and produce excellent results with a minimum of pre-processing and on-line computational time. The field example from the Thrace Basin, Turkey showed that the system does well at picking small deflections that are too small to be adequately picked by eye. Thus, the system should be a valuable tool for correlation on a very small scale, as well as a good assistant for large scale matching.

Appendix A

Dynamic Waveform Matching

A.1 An Example With String Matching

The following is a detailed description of dynamic programming for dynamic waveform matching. For further information the reader is referred to Bellmen (1960), Myers (1980), or Anderson and Gaby (1983). To illustrate the principles behind dynamic waveform matching, we will consider the simple problem of matching some reference word R with characters $R(n)$, $1 > n > N$ with a test word T with characters $T(m)$, $1 > m > M$. The word matching will be performed by editing some of the letters of the input test word $T(m)$ until it matches the reference word $R(n)$. For alphabetical characters, these edit operations are defined as:

operation	Description	Edit Cost
d	Delete the character	1
cX	Change the character to X	1
iX	Insert character to the right of this character.	1
-	No Change	0

Associated with each edit operation is a cost, listed on the right. We will assume for simplicity that each operation has a cost of 1 except for the “no change” option, which has no cost. For example, below are several possible edit sequences that transform the word HILLIER into the word MILLER, and the total edit cost of each operation:

M I L L E R : T(M)

H I L L I E R : R(N)

d cM cI - cL - - D=4

M I L L E R : T(M)

H I L L I E R : R(N)

cM d cI - cL - - D=4

M I L L E R : T(M)

H I L L I E R : R(N)

cM - - - d - - D=2

From the example we see that of the three possible edit sequences, one is less costly than the others. This match represents the “minimum distance” edit sequence, where here the “distance” that we are concerned with is the total cost of matching the words.

Since there are at most $N' = \max(N, M)$ edit operations required to transform T into

R and there are 4 possible edit operations, there are 4^N possible edit sequences. Dynamic Programming¹ is used to efficiently explore this space of possible solutions and determine the minimum distance (or least costly) match.

Let $D(n, m)$ be the minimum distance (or cost) required to match the first n characters of R to the first m characters of T. $D(n, m)$ will depend at any point in the match on the choices of edit operations used so far in the match as follows:

Edit	$d(n, m)$
-	$D(n - 1, m - 1)$
c	$D(n - 1, m - 1) + 1$
i	$D(n, m - 1) + 1$
d	$D(n - 1, m) + 1$

The creed of dynamic programming is to minimize the total cost of the match by always choosing the minimum distance operation (cheapest local cost) at each point in the matching process. In other words:

The total cost of matching two words up to the letters $T(m)$ and $R(n)$, $[D(n, m)]$, is equal to the total cost up to the previous move $[D(n - 1, m - 1)$ or $D(n - 1, m)$ or $D(n, m - 1)]$, plus the cost of the next move [Either 1 or 0].

Thus, as the matching process continues, our total matching cost increases every time we perform an operation besides a perfect match. Thus our cost function is a monotonically increasing function. For each costly operation (ie. an insertion, deletion, or no change), we will choose the least costly one. Therefore, our matching costs are continuously increasing, but increasing by a locally minimal amount.

¹Dynamic Programming was first introduced by Bellman in 1960 as a method of optimizing by linear programming.

We can now recursively define the running total cost of the match $D(n, m)$ in terms of the previous total cost as:

$$D(n, m) = \min \left\{ \begin{array}{l} D(n-1, m-1) + d(n, m), \\ D(n, m-1) + 1, \\ D(n-1, m) + 1 \end{array} \right\}$$

with the boundary condition $D(n, m) = 0$ whenever $n = 0$ or $m = 0$, and

$$d(n, m) = \min \left\{ \begin{array}{l} 0 \text{ if } R(n) = T(m), \\ 1 \text{ otherwise.} \end{array} \right.$$

where $d(n, m)$ is the local cost and is referred to as the “distance metric”²

Figure 36 shows a graphical representation of this matching process, with the distances computed for the MILLER and HILLIER example. Referring to figure 36a, the running total matching costs $D(n, m)$ are computed on an N by M grid, starting from the point $D(1, 1)$ and proceeding to the point $D(N, M)$ column by column. The minimum total edit cost at the end of the match [the point $D(N, M)$] is $D(6, 7) = 2$, as was shown above. The path drawn from $D(N, M)$ to $D(1, 1)$ for which the value of $D(n, m)$ is monotonically decreasing corresponds to the optimum edit sequence.

Figure 36 shows how the individual moves through the grid correspond to the edit operations described above. Horizontal motions in the grid correspond to deleting portions of the reference pattern, and always have a cost associated with them. Likewise, vertical motions correspond to deletion of the test pattern with the associated costs. Motions along a diagonal can either

²The distance metric is a measure of how similar two objects are. For letters the values are simply 0 (they are the same) or 1 (they are different). This quantity will appear later as an important feature in Dynamic Depth Warping.

result in no cost, if the corresponding features match ($R(n) = T(m)$), or a cost of 1, if they are different.

Since $D(n, m)$ (the running total cost) depends only on $d(n, m)$ (the cheapest next move) and the $D(n, m)$ values to its left (the total matching cost so far), the path is locally constrained to follow one of the three paths shown in figure 36. Since these motions correspond to the edit operations, the matching problem becomes equivalent to a pathfinding problem. If we can find the path through the grid that minimizes the total distance $D(N, M)$, we have found our optimal match. Notice that this path is exactly the warping function $W(n)$ described in Section 2.2.

Dynamic programming can efficiently determine the optimal path whenever the **dynamic programming principle** applies:

Whenever the path from a starting point S [$D(1, 1)$ here] to an intermediate point I does not influence the optimum choice of paths for traveling from I to a goal point G [$D(N, M)$ here], then the minimum distance from S to G is the sum of the minimum distance from S to I and the distance from I to G

Dynamic programming is efficient because it ignores all paths from S to I other than the minimum distance one. This feature will prove vital when we wish to reduce some of the grid calculations in matchings that involve hundreds or thousands of points.

A.2 Dynamic Waveform Matching

Now that we have outlined the properties for dynamic matching of patterns that consist of alphabetical characters, we would like to generalize this to the matching of discrete curves. Figure 37 shows an example of a warping path that matches two discrete curves. Each point

in the curve has a value associated with it, which for well logs will be the readings from the tool responses. These values now take the place of the alphabetical characters in the word matching problem. Each point in the grid will have a value associated with it that represents the total cost of the match to that point. As in the word matching problem, the program must trace its way back through the grid, touching the points which represent the now monotonically decreasing minimum distance path. As the program traces through the matrix, it saves the discrete local motions in memory, and then uses these motions to draw the corresponding tie lines and reproduce the warping path. Note that in other applications the warping function is interpolated as a continuous curve, whereas in the dynamic depth warping algorithm the path is always saved as discrete moves. In this way, the user can retrieve information about the correlation from point to point in the global path.

Appendix B

Outline of the COREX system

The following is an outline of the flow of the COREX system as it proceeds from raw digitized logs to the final correlation (Figure 38). The inputs to the COREX system for a two well correlation are: (1) Two sets of digitized wireline logs containing any number of logs; (2) Seismic tie lines in each well; (3) Simple dipmeter information; and (4) The basic geometry of the problem including total depths, number of logs, and well spacing (Part 1). After the digitized logs are read, there are three pre-correlation steps which must be taken. A simple lithology inversion is performed which results in an alternating list of rock type and thicknesses for the entire correlating section in each well (Part 2a). This is the only step that cannot be performed by the program itself. Lithology information is either inverted from the wireline logs using TERRALOG software on the VAX 11/780, or input by hand from published stratigraphic correlation sections. Next, the logs are normalized on a scale from 0 to 100 based on each logs own minimum and maximum amplitude (Part 2b). The normalized logs are then filtered by Walsh transforms (Appendix C) to a minimum bed thickness of 1.2 meter (4 feet). Finally, these filtered logs are compared to the actual (normalized) logs to obtain a measure of noise content in the logs (Part 2c).

After the pre-processing has been completed, the actual correlation can begin. The first question the computer will ask is what type of depositional environment it is attempting to

correlate in (Part 3). If the environment is not stored in the knowledge base as a valid model, then the user is asked to supply some basic information about the problem. The user can also choose to run the correlation using unspecific geological rules, or no rules at all. After the environment is adequately determined, the program runs through three sets of rules concerning the lithologies in the wells, the geologic structure it can determine from the seismic tie lines, and the geometry of wells in relation to the model of the depositional environment (Part 4). The program uses these rules to determine how closely any unit in one well should match with a given unit in the other well. It uses this information to modify some general geologic rules on how continuous different units should be and hence how they should correlate. Once this data is stored, the program uses a simple dynamic programming routine to perform an initial match of the larger units in the wells (Part 5). The output from the initial match is displayed on the terminal and then stored as a LISP object that other portions of the program will use as tie points to complete the correlation.

At this point the seismic tie lines are included as additional constraints. Then the knowledge base takes over again to look at the individual sections between tie points. These sections will be from 10 to two-hundred points in length. The program will look at thickness differences and dip information to see if can further constrain the correlation with local continuity or maximum shift limits (Part 6). The program can now look at the lithology information and the noise analysis to decide how each of the logs should be weighted in the correlation. It will use this data to modify the local distance metric between points in each well (Part 7). Finally, the program has two series of data to correlate with a number of tie points, global restrictions, and local path constraints. At this point the problem should be reduced to a manageable number of calculations, on the order of 10^6 operations. It then uses a more powerful dynamic programming algorithm to match the wells point-to-point (Part 8). The final correlation is then displayed on the terminal for approval (Part 9). The user then has the option of keeping this result, or trying again with some of the parameters slightly modified. Assuming the pre-correlation steps are completed and stored in files, the entire matching process from start to finish takes roughly 5 minutes.

Appendix C

Noise analysis by Walsh Transform

In order to properly weight the influence a particular log should have on the correlation, we must obtain some measure of the noise content of the log. The technique used in this thesis is a running comparison over depth between the actual log and a smoothed version of the same log. For this purpose, the set of orthogonal Walsh functions is used to low-pass filter the logs to a common minimum resolvable bed thickness (MRBT). Walsh functions are a complete set of orthogonal functions which only take on the values +1 and -1 (Figure 39). Many of the characteristics of these rectangular functions, including discrete transitions in signal level, make them appropriate for analyzing borehole data.

The finite Walsh (or Hadamard) transform of any discrete series $f(t)$, $t = 0, 1, 2, \dots, N - 1$ is given by:

$$F_w(j) = \sum_{t=0}^{N-1} W(j, t) \frac{f(t)}{N}$$

Where $W(j, t)$ is the Walsh coefficient of j th order in sequency.¹

¹Sequency is defined as one-half of the number of zero-crossings per unit time (zeros per second or zps). It is analogous to frequency for sine waves, being the number of sign changes per unit time.

The original series can be recovered completely with the inverse of the Walsh transform:

$$F(t) = \sum_{j=0}^{N-1} W(j, t)$$

In order to smooth the logs to a common minimum resolvable bed thickness, the Walsh transformed logs must be low-pass filtered by convolution with a zero-phase box-car or pi filter. The number of terms to keep in the Walsh series is dependent on the sampling interval of the original logs, or the number of points in the log divided by the total depth covered by the log. By Walsh filtering the logs, we are effectively increasing the minimum resolvable bed thickness (MRBT), or decreasing the sampling interval. Thus for a given log spanning a depth D and with Z number of points we would like to find the maximum number of terms to keep in the Walsh sequence (maximum sequency component) that will reduce the logs to some value T_b , the resolution in feature thickness. For example, for well logs where the sampling interval is $s = 2$ pts/meter, the maximum sequency component with depth is 1 zpm (zero-crossings per meter), and the theoretical minimum resolvable bed thickness is .5 meters.

For a log spanning D meters in depth and containing Z points, the minimum resolvable bed thickness T_b for this log would then be:

$$T_b = \frac{D}{Z}$$

In order to low-pass filter the logs to a specified minimum bed thickness we must change the actual MRBT by some factor F , such that:

$$MRBT_{desired} = (F) MRBT_{actual}$$

Thus, if we wished to increase our MRBT by 2:

$$T_{desired} = 2 MRBT_{actual} = 2 \frac{D}{Z} = \frac{D}{\frac{1}{2}Z}$$

In other words, to increase our MRBT by a factor F , we must keep the first Z/F terms in the Walsh series in order to obtain the proper resolvable bed thickness. In this thesis, all the

logs are filtered to a MRBT of 1.5 meters. For a log which has an original sampling of 2 points / meter, the MRBT is .5 meters, and we wish to increase this by a factor of 3. Thus, we would wish to keep $Z/3$ points in the Walsh series to properly reconstruct the log.

C.1 The Noise Measure

The noise analysis for a given log can be summarized as follows:

If $l(z), z = 0, 1, 2, \dots, Z - 1$ is a log containing Z depth points, then

$$L_w(j) = \sum_{z=0}^{Z-1} W(j, z) \frac{l(z)}{Z}$$

would represent the Walsh-transformed log. The low-pass filtered version in the Walsh domain would thus be:

$$L_w^f = L_w(j) * \Pi$$

Where Π is a zero-phase, low-pass filter in the Walsh domain. The new depth-filtered series would be the inverse Walsh transform of the shortened Walsh series:

$$l^f(z) = \sum_{j=0}^{Z-1} W(j, z)$$

As mentioned above, an acceptable measure of the noisiness in a particular log would be the average distance between the log and some smoothed version of the same log. The normalized noise measure $N(l)$ over depth would thus be the running average of the difference between the filtered curve $l^f(z)$ and the actual log $l(z)$:

$$N(l) = \sum_{z=0}^{Z-1} \frac{|l^f(z) - l(z)|^2}{Z^2}$$

If the logs are all normalized to a scale from 0 to 100, the average noise value $N(1)$ will represent a noise percentage in the log. In the dynamic programming algorithm, the maximum allowable signal-to-noise ratio can be expressed as a percentage since the maximum deflection is artificially adjusted to be 100, and any ratio of smaller variations to this number is effectively a percent. In the dynamic depth warping algorithm, the maximum allowable noise (MAN) is defined as the point at which the noise influences the DDW to where it can no longer identify boundary deflections properly. Experiments performed with synthetic data on the Symbolics LISP machine show that an acceptable threshold is about 70%. Beyond that point, noise is no longer distinguishable from deflections in the trace.

Figure 40 shows the progression of a raw log to its Walsh low-passed filtered version. The original MRBT was 2 feet, and the log was filtered to a MRBT of 8 feet. It should be emphasized that it is imperative for the raw logs to be properly depth shifted when performing Walsh filtering. When a log has a MRBT of 2 meters, there will be a change in log value every 2 meters in the section. If the actual logs do not change boundaries at a multiple of 2 meters from the beginning of the logs, the location of a particular deflection can be in error by as much as one-half of the step length or bed width. Thus, the location of a deflection at a point D_i in the logs and that indicated by the Walsh low-passed version D_w are related by:

$$D_w = D_i \pm \frac{MRBT}{2}$$

Since the multiple log distance metric mentioned in Chapter 2 stacks all of the values at a particular depth point, smoothed curves which are not properly depth shifted can have a “smearing” effect on the distance function. This smeared distance will compare two points on either side of a boundary, instead of two points in a similar position, which inhibits large distance changes at bed boundaries. This effect can seriously deteriorate the resolution of the algorithm, and thus it is recommended that depth shifting be performed both before and after walsh filtering.

C.2 Noise Analysis by Dynamic Programming

The noise analysis described above is basically a running average of the difference between a trace (N) and its smoothed counterpart (N_w). This measure, however, is very similar to the total matching cost of a depth warp, as discussed in Chapter 2. In other words, the noise at any point $d(N, N_w)$ is the matching cost between the log's actual value and its smoothed version. The total matching cost, then, when normalized by the number of calculations n_{ave} , should be proportional to the average noise as determined by Walsh analysis:

$$n_{ave} \approx \frac{D(N, N_w)}{N}$$

Table C-1 shows how the logs from the Ghana example of Chapter 4 were ranked by “noisiness” from Walsh analysis as described above. Table C-2 shows the same ranking from the total costs of warping each log against its smoothed version. From the tables we see that, aside from scaling problems, the methods are nearly equivalent. We thus have another option for noise analysis to be used for log weighting or smoothing: namely, to match a log with its smoothed version, and look at the total cost.

Log	Ranking	Noise Measure %
GR	5	26.01
Sonic	4	32.52
Density	3	71.25
Neutron	2	143.22
Resistivity	1	144.05

Table C-1: Noise ranking by covariance.

Log	Ranking	Noise Measure: Total matching cost
GR	5	10.1
Sonic	4	12.3
Density	3	14.4
Resistivity	2	18.86
Neutron	1	22.05

Table C-2: Noise ranking by dynamic programming.

Results from noise analysis by depth warping show that for total normalized matching costs above 15-18, the logs are considered too noisy to be effective in the correlation. A normalized global cost of 16 results from matching logs at the noise threshold of 70%, as mentioned in section 2.7. The reason that this number is chosen so roughly is that an adequate noise average requires knowledge of the exact number of distance calculations. Since our best approximation to this number is the the number of points in the longer series, the average distance will be in error by a small amount in most cases.

Appendix D

Rules in the geologic Knowledge Base

The following section describes some typical rules that may operate when COREX solves a correlation problem. The rules do not appear as they do in LISP code, but instead as their English language translations. The rules in the COREX knowledge base fall into three categories:

1. **Lithologic Rules:** Rules fired as COREX performs the initial match of lithologies between the wells. These have three sub-classes, general lithologic rules, scale rules, and shoreline rules.
2. **Structural Rules:** Rules that translate structural information from seismic and dipmeter into dynamic programming constraints.
3. **Distance Metric Rules:** A miscellaneous category of general correlation rules which concentrate mostly on modifying the local distance metric.

D.1 Lithologic Rules

D.1.1 General Lithologic Rules

- If there is no other information about the depositional environment,
- **Then** shales are the most continuous units, sands are the next most continuous, and limestones the next.

- If there is no other information about the depositional environment,
- **Then** “thick” units are more continuous than “thin” units.

- If a lithology is designated as a marker bed,
- **Then** it will be the most continuous unit in the area, and it will be assigned a “continuity” ranking of 1, and be assigned a matching cost of zero.

- If a unit covers a large geographical area, and it is present in both of the wells,
- **Then** it will be ranked high in continuity.

- If a unit is spatially isolated, and it is not present in both of the wells,
- **Then** it will be ranked low in continuity.

- If two units are the equal in lithology and thickness,
- **Then** their matching distance is proportional to their ranking of continuity in the environment.

- If two units match in lithology, but not in thickness,

- **Then** their matching distance is proportional to .7 times their continuity.
- **If** two units do not match in thickness or lithology,
- **Then** their matching distance is the maximum of 100.
- **If** we are in a braided.stream environment,
- **Then** thick-sands are the most continuous units, and set the thin-shale matching distance to 100.

D.1.2 Scale Rules

- **If** a particular unit is less than 2 meters thick, or is only one or two data points in thickness,
- **Then** it is designated a “thin” bed.
- **If** a particular unit is greater than 50 meters thick, or greater than 1/5 the thickness of the entire section,
- **Then** it is designated a “thick” bed.
- **If** we are in a braided.stream environment, and the section is greater than 30 meters thick,
- **Then** thick-shales cap vertical sequences, and thick-shales should be marker beds.
- **If** we are in a meandering.stream environment, and the vertical section is greater than 40 meters,
- **Then** we should expect insertions and deletions of units.

- **If** we are in a meandering.stream environment, and the section is greater than 100 meters thick, and there are limestones present,
- **Then** limestones likely cap entire sections, and limestones should be marker units.

D.1.3 Shoreline Rules

- **If** correlation is in a braided.stream, meandering.stream, beach, delta, slope, or shelf environment,
 - **Then** ask the user for shoreline trends and invoke the shoreline rules.
-
- **If** we are in a meandering.stream environment, and the strike of correlation is perpendicular to shoreline, and the well spacing is less than 10 km.,
 - **Then** make sands the most continuous units.
-
- **If** the well spacing is greater than 10 km, and the paleostrike is parallel,
 - **Then** assign sand-sand matching distance to 100.
-
- **If** we are in a beach environment, and the strike of the wells is parralel,
 - **Then** sands are the most continuous units.
 - **Else** if the well spacing is greater than 2 km., then sands are the least continuous units.
-
- **If** we are in a reef environment, and the strike of the wells is parallel to shoreline,
 - **Then** limestones are the most continuous units.
-
- **Else** if the strike is perpendicular and the well spacing is greater than 10 km., then limestones are the least continuous units.

- **If** we are in a continental shelf environment, and the strike of the wells is perpendicular,
- **Then** allow sandy-shales to correlate with shaly-sands of the same dimension, and allow conglomerates to correlate with sands of the same dimension.

- **If** we are in a continental slope environment, and the strike of the wells is perpendicular,
- **Then** allow thin units to correlate with thick units of the same lithology down the slope, and allow sandy-shales to correlate with shaly-sands down the slope.

- **If** we are in a deltaic environment, and the strike of the wells is parallel, and the well spacing is less than 15 km.,
- **Then** sands are the most continuous units.

- **Else** if the well spacing is greater than 20 km.,
- **Then** set sand-sand matching distance to 100.

- **If** we are in a deltaic environment, and the strike of the wells is perpendicular to shore,
- **Then** allow thin units to correlate with thick units of the same lithology down the slope, and allow sandy-shales to correlate with shaly-sands down the slope.

- **If** we are in a deltaic environment, and there is coal in both wells,
- **Then** designate coal as a marker bed.

D.2 Structural Rules

- **If** the section in well 1 is at least 1.5 times as thick as the section in well 2,

- **Then** limit the warping path to the lower section of the global path.

- **If** two regions are forced to match by the interpreter,
- **Then** force the warping path to go through the region.

- **If** two points are tied from the lithologic match,
- **Then** impose a tie point with an error on each side equal to twice the minimum resolvable bed thickness.

- **If** there is a normal fault between the two wells, and it has an offset of x meters,
- **Then** start the warping ($x - 5$) meters in the downthrown well.

- **If** there is structural dip between the wells of D degrees, and the well separation is X meters,
- **Then** start the warping path ($X \sin D$) meters in the down-dip well.

- **If** there is no dip between the wells,
- **Then** assign a maximum shift constraint of 30 meters between the wells, and use a type II continuity constraint.
- **Else** assign a maximum shift between corresponding sections of (thickness + 10 meters) and use a type III constraint.

- **If** we expect insertions and deletions of rock units,
- **Then** use a weighted type I continuity constraint.

D.3 Distance Metric Rules

- If the two points are separated by a distance greater than the maximum allowable shift,
- Then assign a maximum matching cost of 100.

- If the matching cost is below the threshold of the smoothing operator,
- Then assign the points a matching cost of zero.

- If log A is determined to be beyond the noise threshold,
- Then weight it by zero in the local distance measure.

- If the caliper log at depth D has a spike,
- Then assign a very high matching distance at that depth.

Appendix E

Other geophysical applications of Dynamic Programming.

Although rarely used in geophysical analysis, dynamic programming methods have many potential extensions into the areas of full waveform acoustic logging, seismic event tracing and detection, and multi-well log correlation. Anderson and Gaby (1983) give a good overview of potential applications of dynamic waveform matching. During this thesis, a number of other possible applications have arisen. In general, any time a cost function needs to be minimized, dynamic programming principles may apply.

In the processing of full waveform acoustic logs, it may be possible to perform a very quick moveout calculation using dynamic time warping on the four adjacent traces of the full waveform tool. If the P-wave onset can be adequately picked, the moveout or formation p-wave velocity may be calculated from the slope of the warping path. In this case, range constraints as mentioned in chapter 2 could be used to for vast savings in computation time. Dynamic programming could also be used as a pattern classification scheme to provide direct lithologic interpretations from full waveform logs. The technique of Hoard (1982) could be used, except that a total minimum cost function would replace cross-correlation as the classifying scheme.

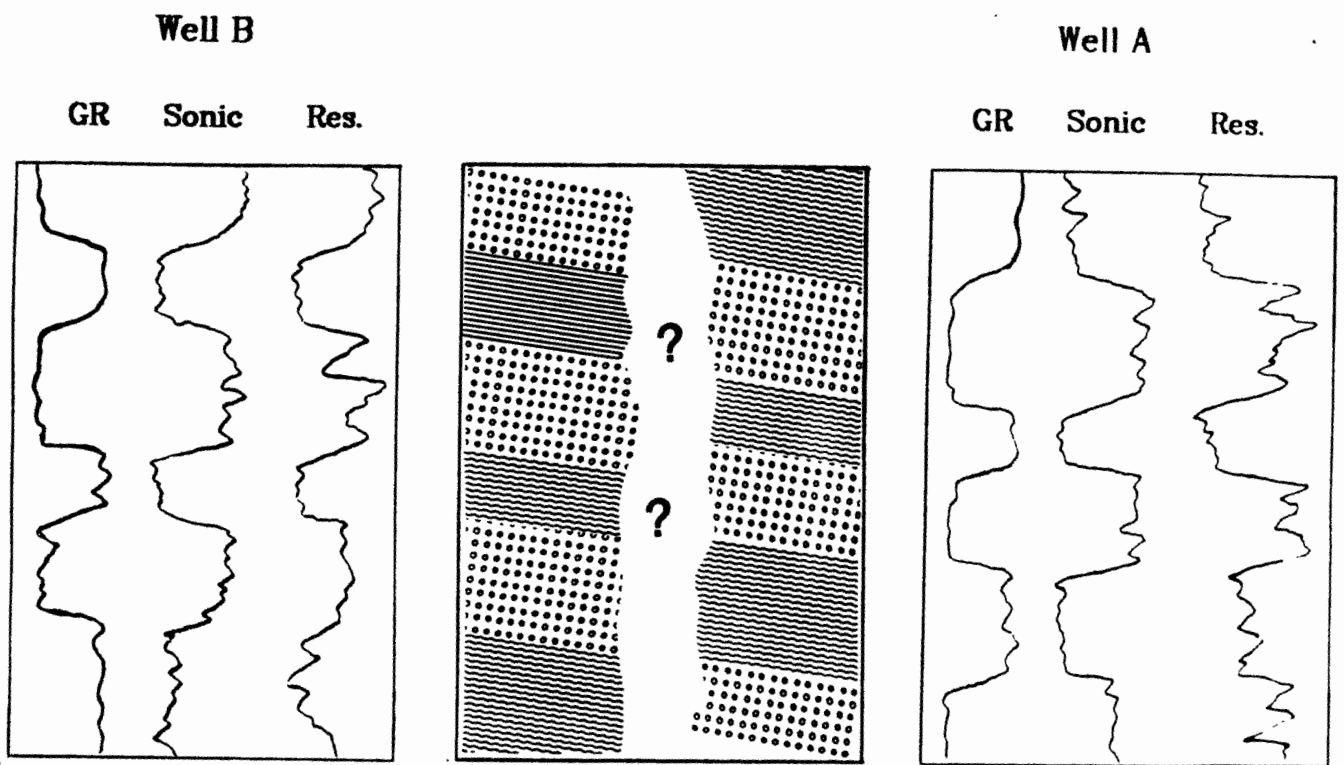
Dynamic waveform matching could also be used on surface seismic for reflection event tracing. Present methods use a trace-to-trace local cost minimization function only, with no total cost minimization over the entire seismic section. Dynamic programming could be used to adequately explore the very large search space required for event tracing with conventional seismic data.

Finally, dynamic programming could be used to extend the well correlation problem into three or perhaps more wells (Smith and Waterman, 1980). Interpreting depth warping as a pathfinding problem would merely require the global area to be extended to a three-dimensional solid, with the minimum distance warping path being a curve in three-dimensional space.

Although the geologic knowledge base of COREX is at present somewhat limited, it should serve as a prototype for more advanced systems. Natural extensions for the geologic knowledge base would include rules in three main areas. First, interpreting depositional facies based on electric log patterns and using these in conjunction with known depositional models. Second, rules concerning the elaborate changes in log sensitivity as a function of invasion parameters and formation fluid content. Thirdly, automatically incorporating more detailed structural information into the correlation through seismic sections and dipmeter logs.

- Anderson, K.R. and Gaby J.E. (1983) Dynamic Waveform Matching: Inform. Sciences. vol. 31, pp. 221-242.
- Anderson, K.R., Rosenberg, S.T., and Lanan, D. (1982) Automatic Association Using Expert Systems Techniques, in Seismic Discrimination Semiannual Technical Summary, Lincoln Labs, MIT, Cambridge, MA. pp. 27-32.
- Cheng, Yao-Chou and Lu, Shin-Yee (1985) Waveform Correlation by Tree Matching: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-7, no. 3, pp. 299-305.
- Delcoigne, A. and Hansen, P. (1975) Sequence Comparison by Dynamic Programming: *Biometrika*, vol. 62 no. 3, pp. 661-664.
- Findler, N.V. and Van Leeuwen (1979) A family of Similarity Measures Between Two Strings: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-1 no. 1, pp. 116-118.
- Gordon, A.D. and Reyment, R.A. (1979) Slotting of Borehole sequences: Journ. Soc. of Mathematical Geology. Vol. 11 No. 3 pp. 309-327.
- Kwon, Byung-Doo and Rudman, A.J. (1979) Correlation of Geologic Logs with Spectral Methods: Math. Geol. Vol. 11 No. 4 pp. 373-390.
- Lang, W.H. Jr. (1984) Conductivity and Interval Transit Time as Correlation Tools: The Log Analyst, May-June, pp. 21-33.
- Mann, C.J. and Dowell, T.P. Jr. (1978) Quantitative Lithostratigraphic Correlation of Subsurface Sequences: Comp. & Geosciences V. 4 pp. 295-306
- Myers, C.S. (1980) A comparative study of several dynamic time warping algorithms for speech recognition: Master's Thesis, MIT, Cambridge, MA.

- Robinson, J.E. (1978) Pitfalls in Automatic Lithostratigraphic Correlation. *Comput. and Geo.* Vol. 4, pp. 273-275.
- Rubel, M. and Pak, D.N. (1984) Theory of stratigraphic correlation by means of ordinal scales. *Comput. and Geo.* Vol. 10 No. 1, pp. 97-105
- Rudman, A.J. and Lankston, R.W. (1973) Stratigraphic Correlation of Well Logs by Computer Techniques: AAPG V. 57 No. 3, pp. 577-588.
- Sakoe, H. and Chiba, S. (1978) Dynamic Programming Optimization for Spoken Word Recognition: *IEEE Trans. Acoust. Speech. Signal Processing.* Vol. ASSP-26, pp. 43-49.
- Sankoff, David and Kruskal, J.B. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison.* Addison-Wesley Publishing Company, INC., London, 1983.
- Scholle, P.A. and Spearing, Darwin. *Sandstone Depositional Environments.* A.A.P.G. Publications, Tulsa, Oklahoma, 1982.
- Smith, T.F. and Waterman, M.S. (1980) New Stratigraphic Correlation Techniques *Jour. Geol.* vol. 88, pp. 451-457.
- Startzman, R.A. and Kuo, T.B. (1986) An Artificial Intelligence Approach to Well Log Correlation. SWPLA Twenty Seventh Annual Logging Symposium, Houston, June 1986.
- Whittaker, A.H. and J.D. Macpherson. (1986) "Assistant Expert" Log Data Analysis Systems: Applications of Artificial Intelligence Programming Techniques in Formation Evaluation. SWPLA Twenty Seventh Annual Logging Symposium, Houston, June 1986.
- Wu, X. and Nyland, E., (1986) Well Log Interpretation Using Artificial Intelligence Technique. SWPLA Twenty Seventh Annual Logging Symposium, Houston, June 1986.



THE CORRELATION PROBLEM

Figure 1: The correlation problem. Two wells, each containing a number of logs, are to be matched over some distance D . Any number of factors can change from well to well, including geologic structure, fluid properties, and lithologies.

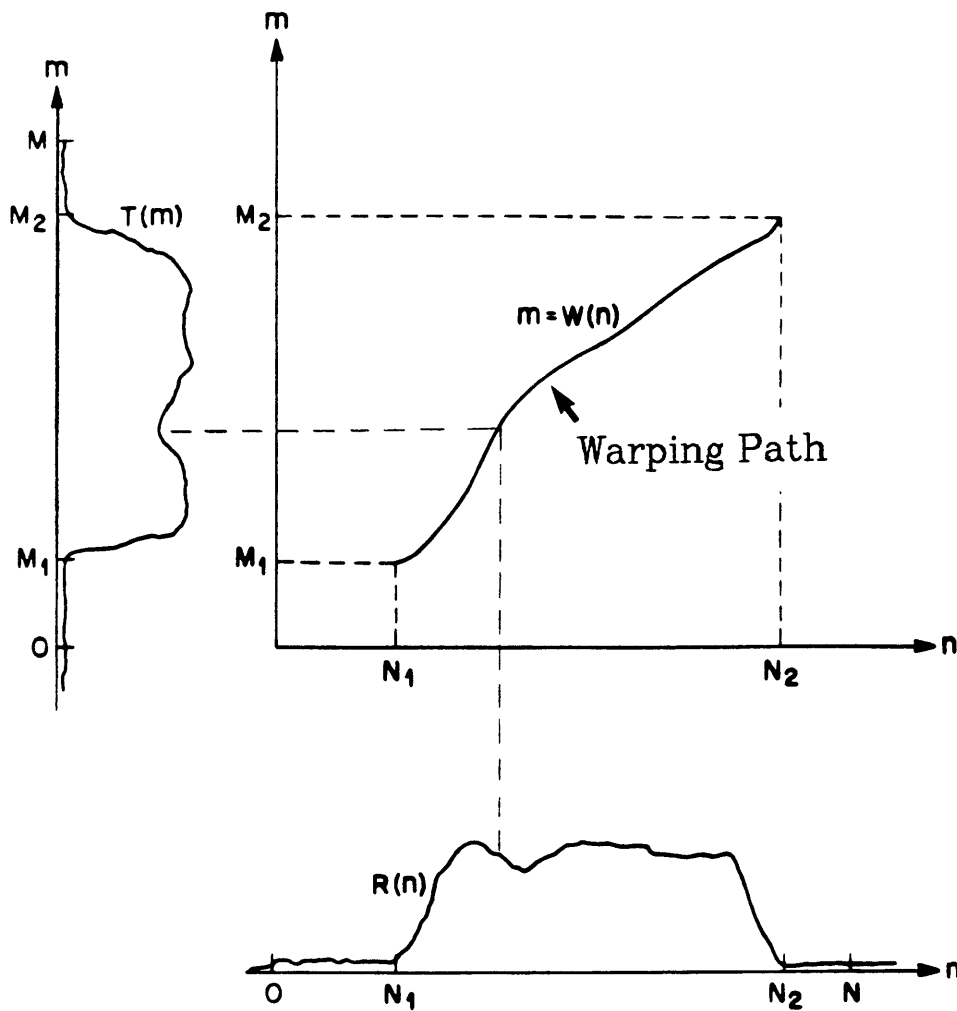
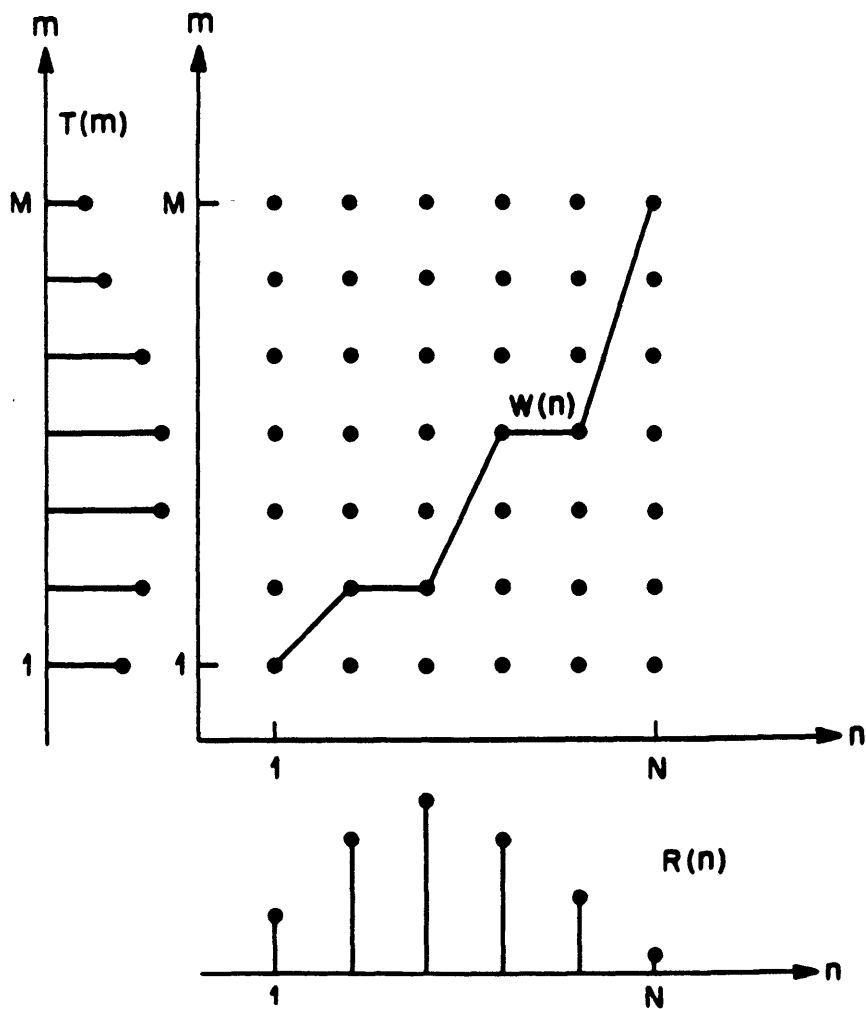


Figure 2: An example of a warping function which matches two curves. Corresponding points in each curve are found by travelling from one curve to the warping function (dashed lines), and then perpendicular toward the other curve.



DISCRETE CURVE MATCHING

Figure 3: A discrete warping path. Each point in the matrix represents a portion of the global area, where the warping path is allowed to wander. The value stored at each point (n,m) is the total cost of matching the waveforms up to the points n and m respectively. The optimal warping path runs through the matrix points with the cheapest matching costs.

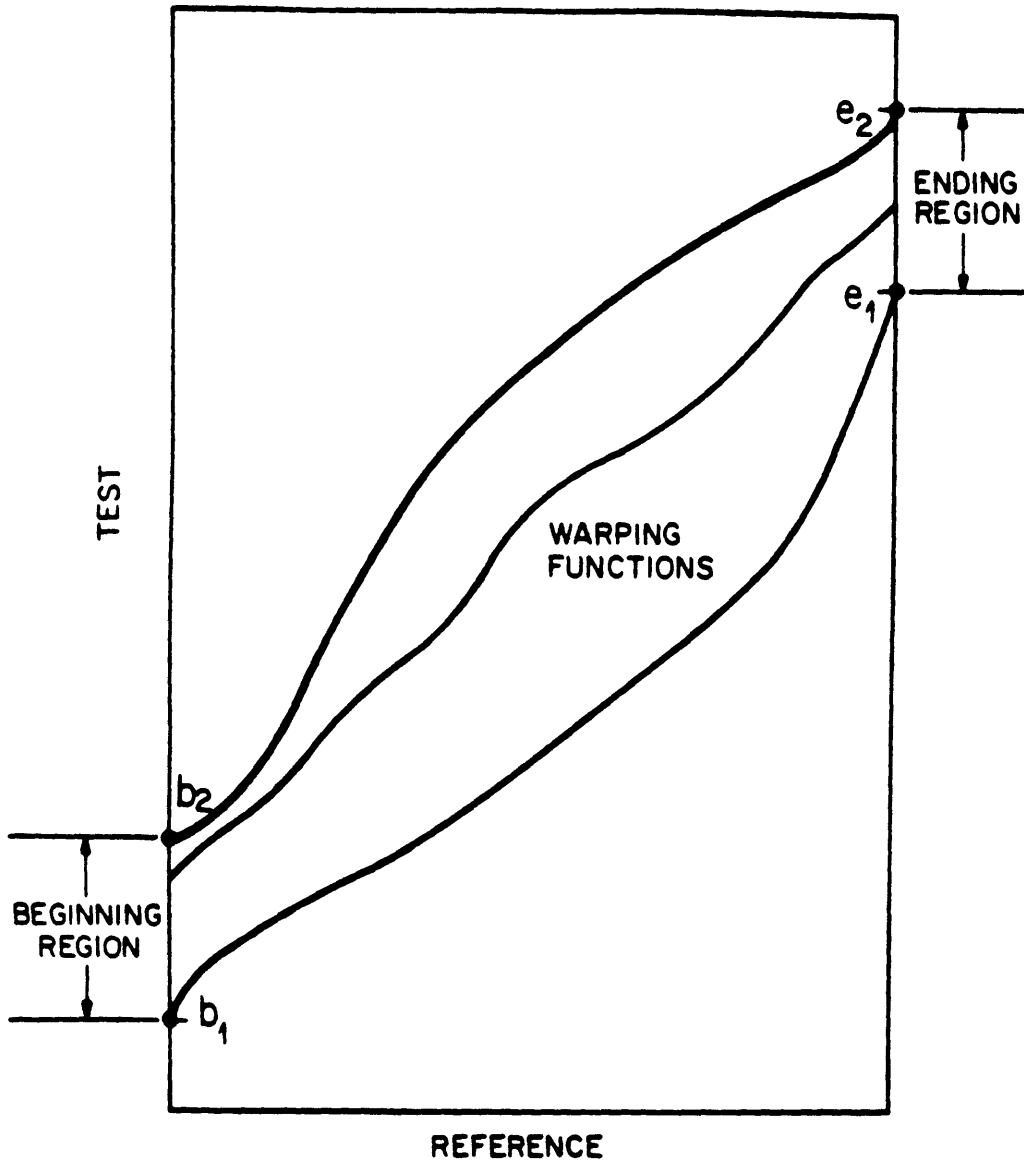


Figure 4: Relaxed endpoint constraints on a depth warp. A matching sequence would consist of $E = (b_2 - b_1)$ total warps, where E is the width (in points) of the beginning region. Each warp would start at a different point in the beginning region and proceed as usual. The final match would be the one with the minimum total cost over all of the E warps.

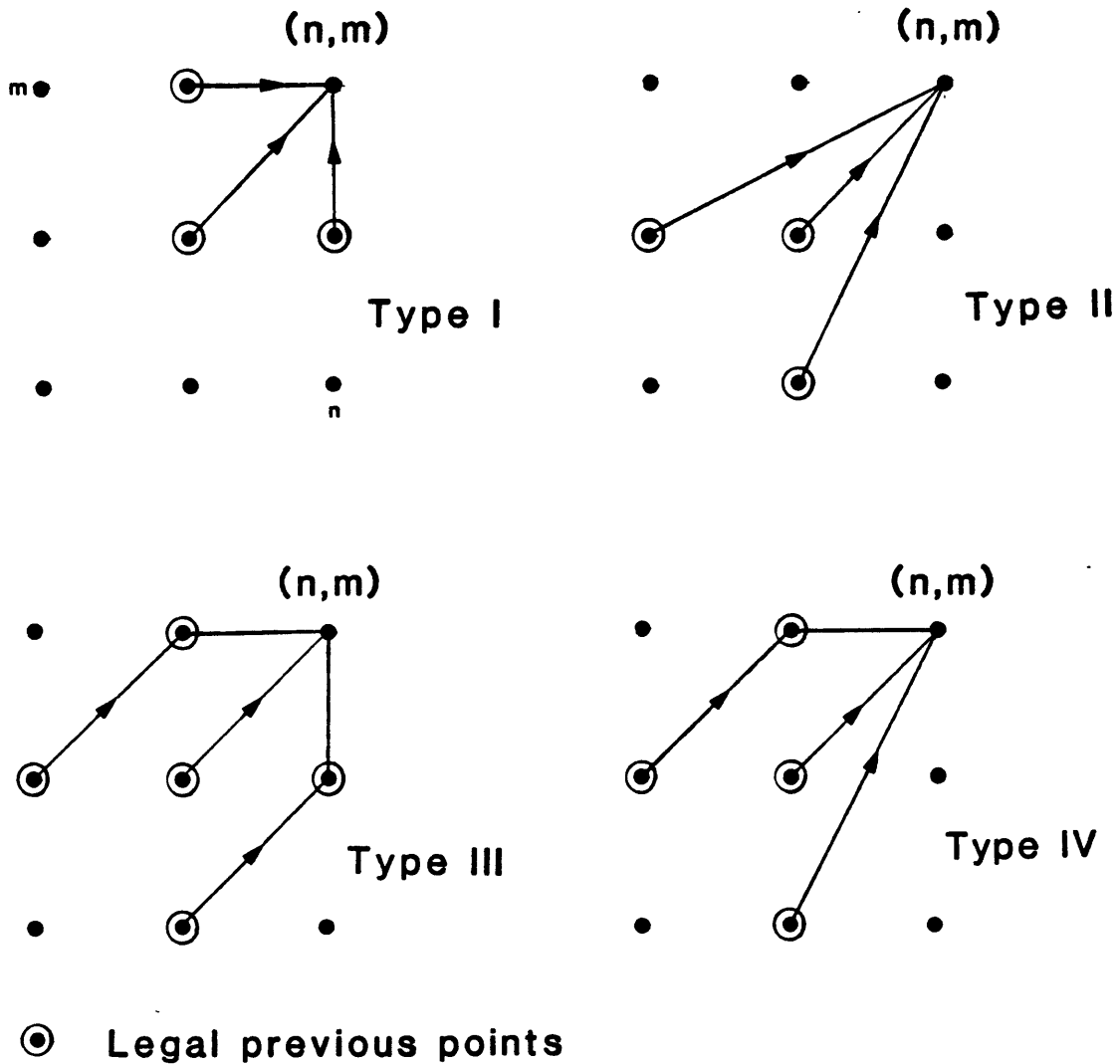


Figure 5: Four different types of local continuity constraints. When the warping path is proceeding to the point (n,m) in the grid, the circled points represent the previous indices which the path may have crossed. Weighting indicates that the cost of a motion along a particular path will be multiplied by the weighting coefficient. (a) Simple Type I, where horizontal and vertical motions are allowed. (b) Type II where motions are restricted to slopes of $1/2$ and 2 . (c) Type III where motions are combined horizontal and vertical and diagonal, with smoothed weighting. (d) Type IV, where horizontal motions are favored over vertical motions.

REGIONAL TIES (TIE POINTS WITH ERRORS)

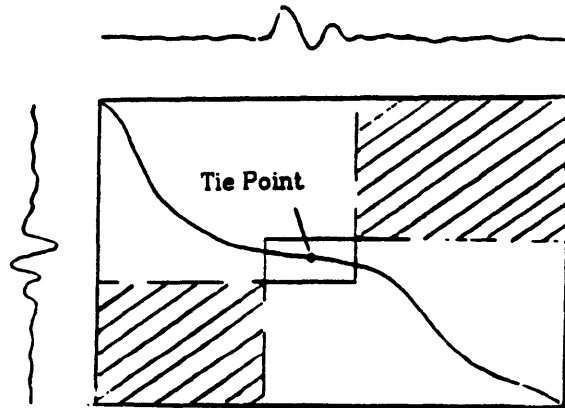


Figure 6: Limiting the global area from a regional tie point. When two regions in each curve are forced to match, this forces the path to travel through the rectangle indicated in the figure. When this type of constraint is imposed, large sections of the global area are eliminated as possible minimum-distance paths (shaded area).

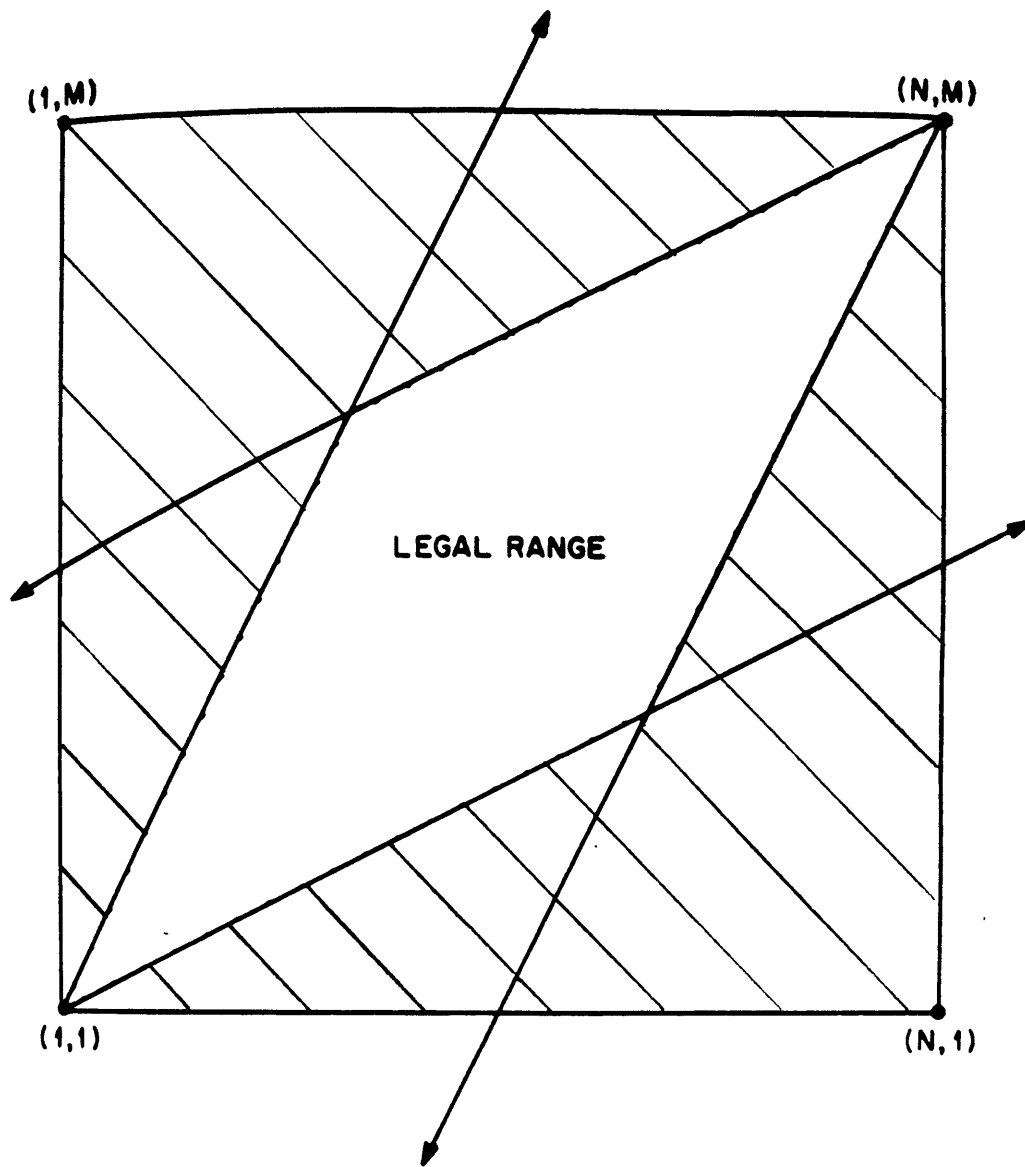


Figure 7: Limiting the global area from continuity constraints. The type II constraint of figure 5 allows only motions which have slopes of $1/2$, 1 , and 2 respectively. This manifests itself in the global path as a parallelogram whose edges leave the ending regions with the slopes $1/2$ and 2 . Again, shaded regions no longer need to be considered as legal paths.

SLOPE CONSTRAINTS (MAXIMUM ALLOWABLE SHIFT)

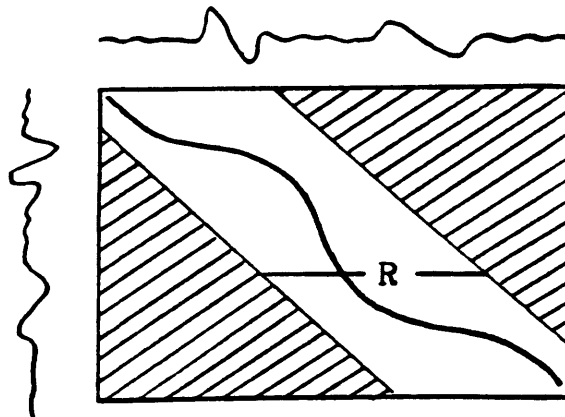


Figure 8: Limiting the global area with a range constraint. Any two points separated by a distance greater than R are assigned a very high matching cost. This shows in the global area as a two parallel lines separated by a horizontal distance R , their slopes equal to the ratio N/M . This type of constraint is very common, and is very efficient for the algorithm to calculate.

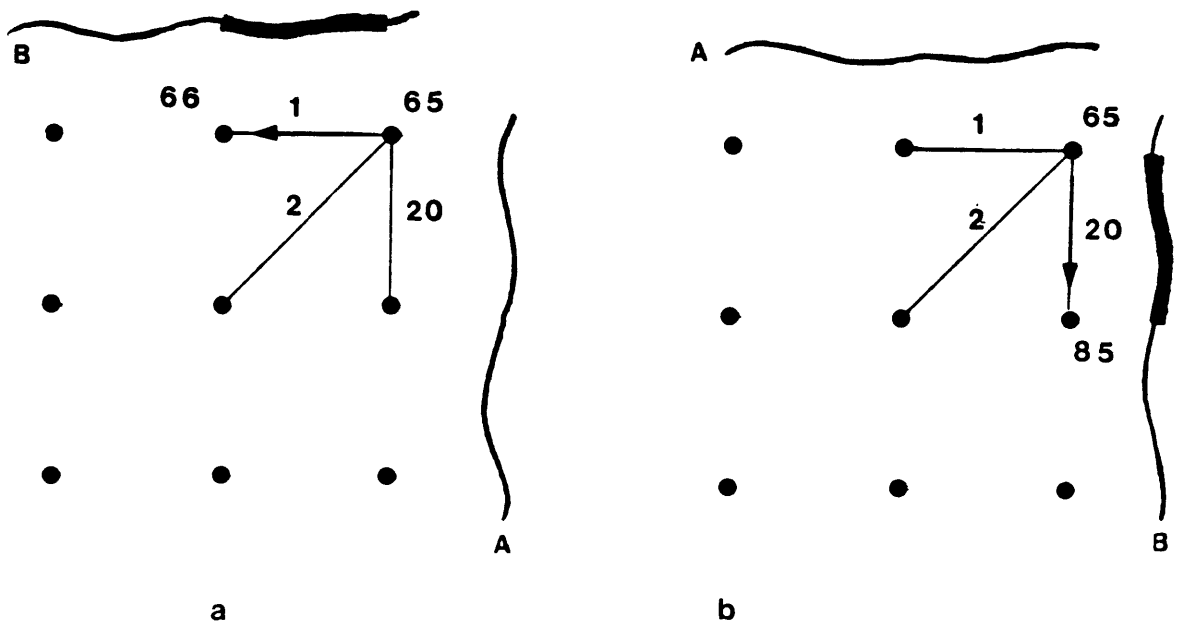
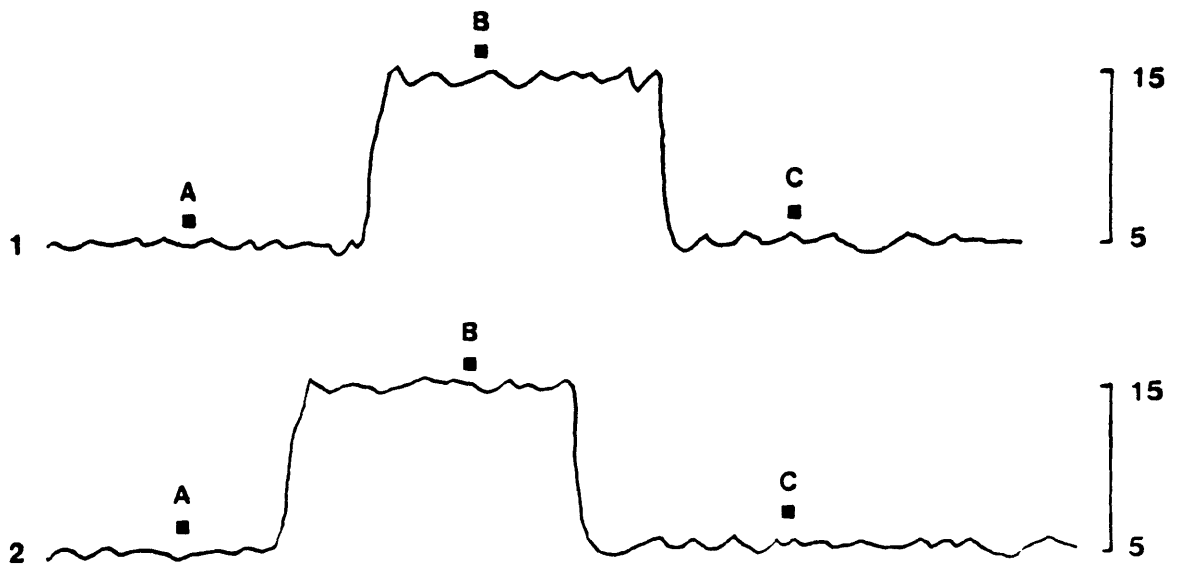


Figure 9: Interchanging the axis can many times drastically change the cost of a particular edit operation. A simple type I continuity constraint with asymmetric weighting (part a), will bias the minimum distance path toward horizontal motion. With this configuration, the shaded section in curve B can be deleted with an increase in total cost of only 1. If we interchange the two curves (part b), the cost of deleting the shaded region in curve B is increased to 20.



- a $d(A_1, A_2) = 0 \pm \text{noise}$
- b $d(A_1, B_2) = 10 \pm \text{noise}$
- c $d(B_1, B_2) = 0 \pm \text{noise}$

Figure 10: Effects of a bed boundary on the local cost or distance measure $d(a,b)$. As the calculation proceeds across a boundary, sudden changes in amplitude and local distance occur between points on opposite sides of the deflection. Shown are local distances for (a) like points, (b) points on opposite sides of a boundary, (c) like points within the bed boundary.

DISTANCE FOR MULTIPLE (K) LOGS

	GR_N	$Sonic_N$	SP_N	$Density_N$	$Resist._N$. . .
GR_M	5	-	-	-	-	
$Sonic_M$		12				
SP_M			23			
$Density_M$				8		$DEPTH (n, m)$
$Resist._M$					45	

$$d(n, m) = \sum_{k=1}^K w_k f [A_{kn} - B_{km}]$$

where $\{w_k (k = 1, 2, \dots, K)\}$ are weighting coefficients

Figure 11: Diagram of the multiple logs distance measure. Points along the diagonal represent “auto-correlations” or distances between two logs of the same type, whereas off-diagonal values would represent “cross-correlations”. The weighting coefficients, as explained in the text, can alter which logs are used in the local distance measure.

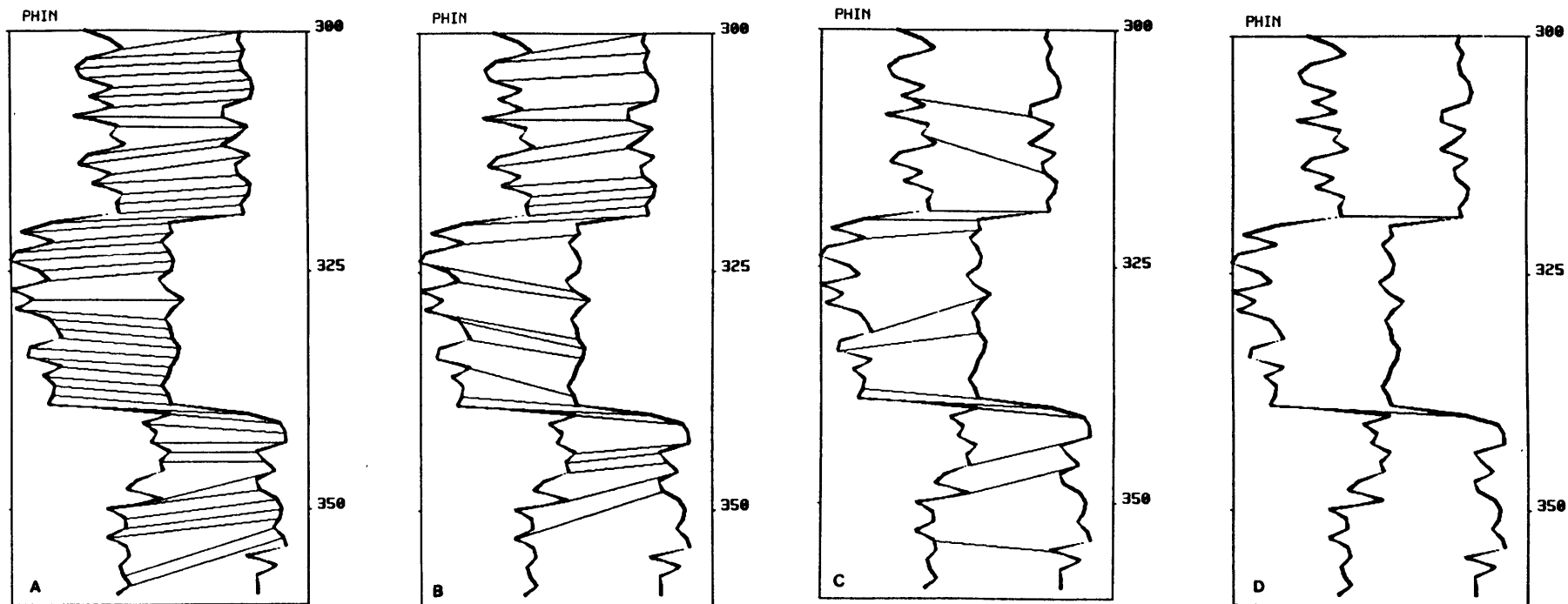


Figure 12: The effect of the smoothing operator on correlations of logs with 50% noise, where the maximum signal deflection is 50. Correlations resulting from smoothing of (a) zero, (b) 8, (c) 15, and (d) 25. When we apply smoothing which is less than the noise amplitude (a,b), the total matching costs are high and boundaries are not accurately picked. Once the smoothing value approaches the noise value (c,d), only boundary signals are picked as correlating points, and the total matching cost is reduced.

CORRELATED 5% NOISE WITH:

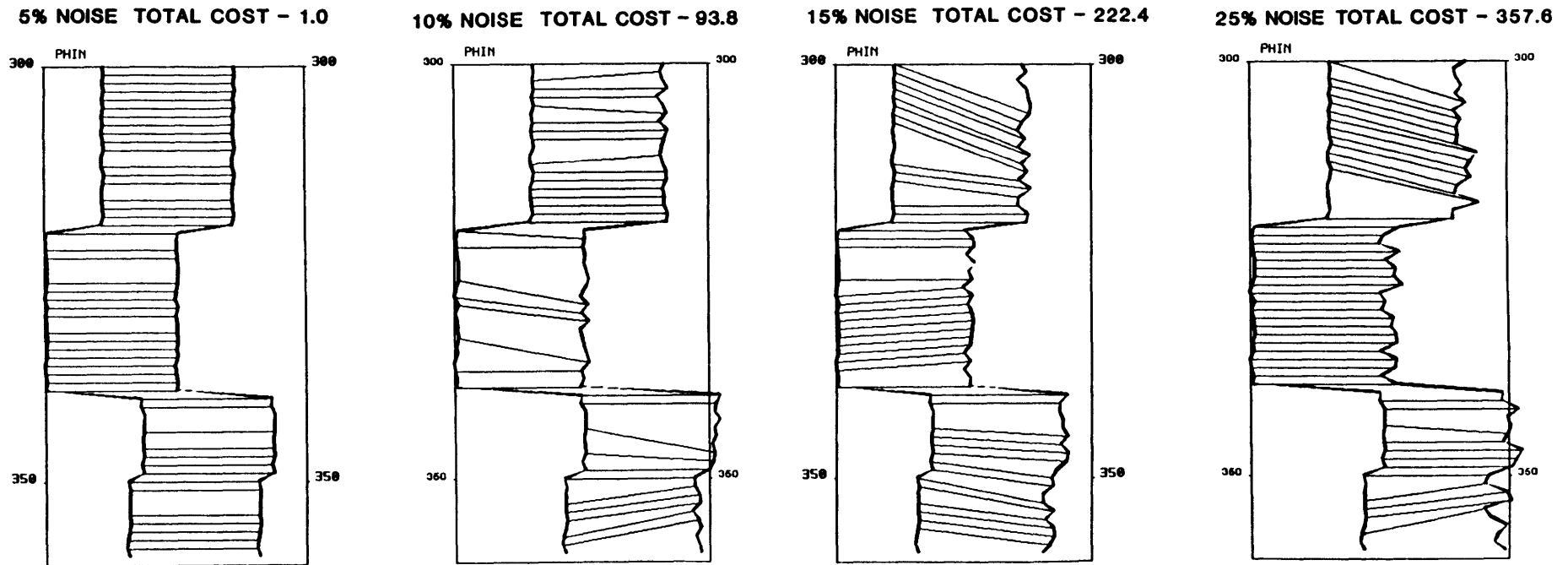


Figure 13: The effect of noisy logs on the global matching cost. Correlation of noisy logs means a poor quality match, and a corresponding increase in the total matching cost. Logs which match identically (part a), will require a total cost no greater than the initialized cost. As the noise content of each log increases, so does the total cost of a successful match. The total cost is analogous to the amplitude of the peak in the cross-correlation function.

The COREX System

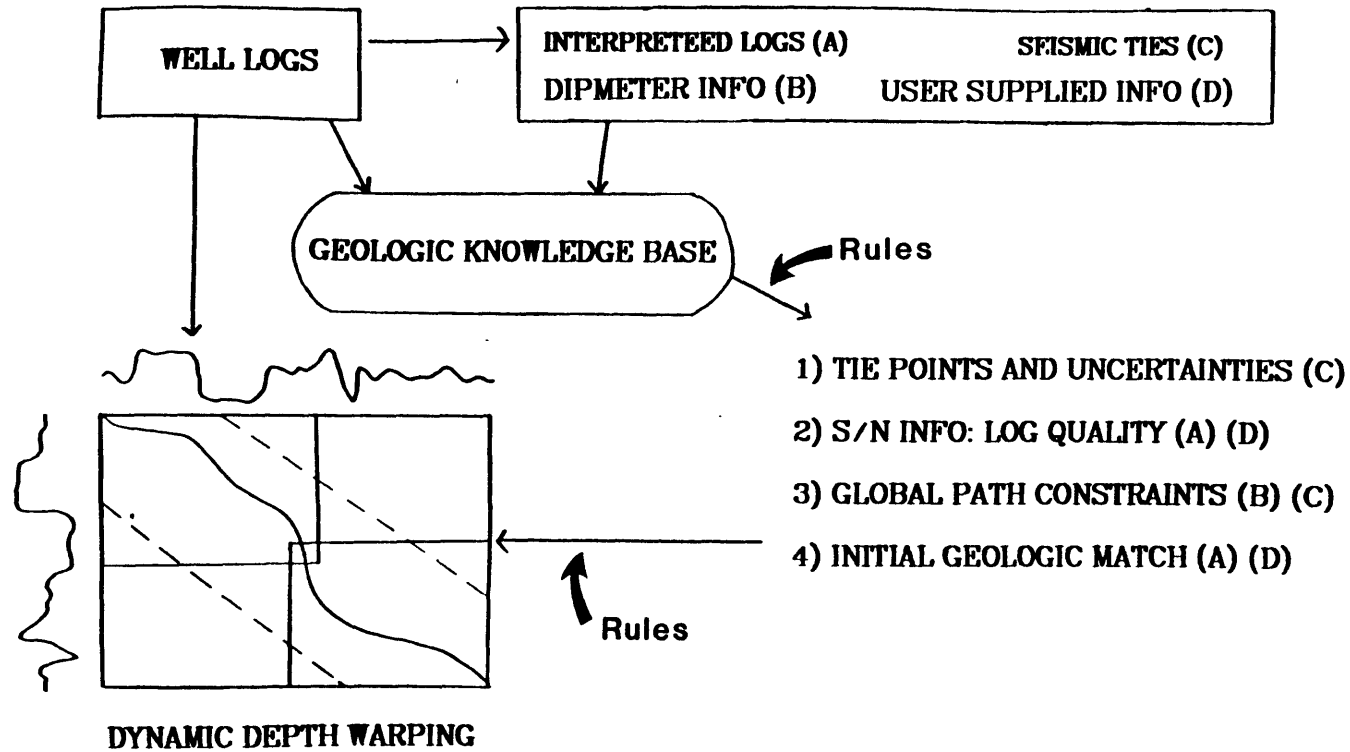


Figure 14: Diagram of the COREX well log correlation system. The well log database and the geologic knowledge base are connected via rules which translate the geologic knowledge into constraints on the dynamic depth warping algorithm.

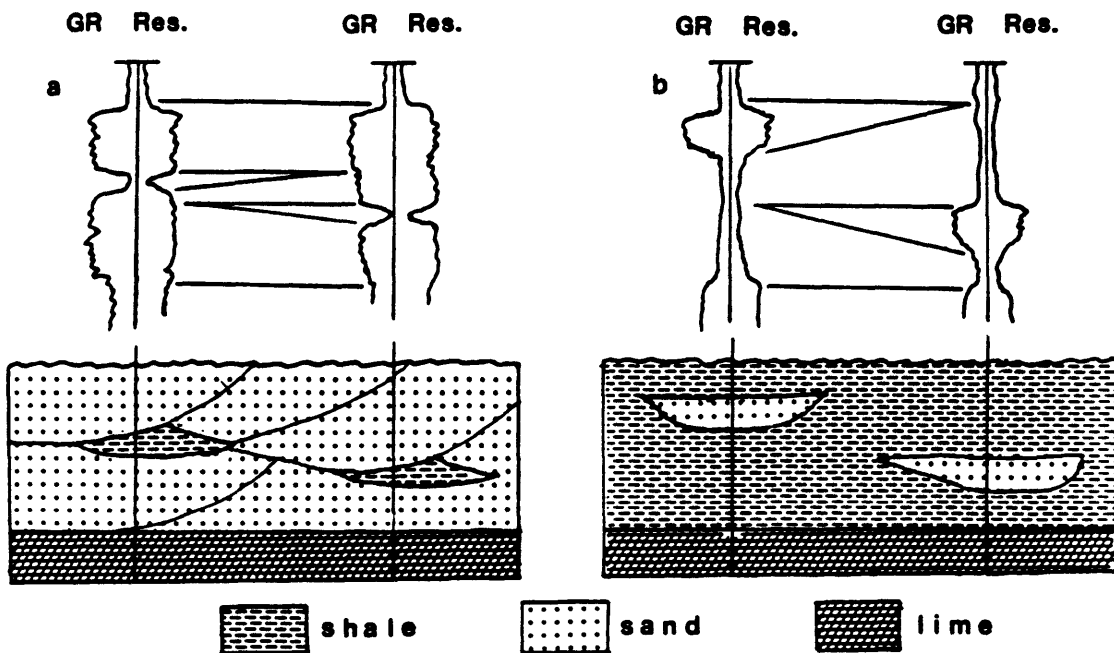


Figure 15: The changes in lateral continuity of shale beds with depositional environment. (a) Braided stream: Shales are isolated, discontinuous lenses. (b) Meandering stream: Shales are thick, persistent marker beds.

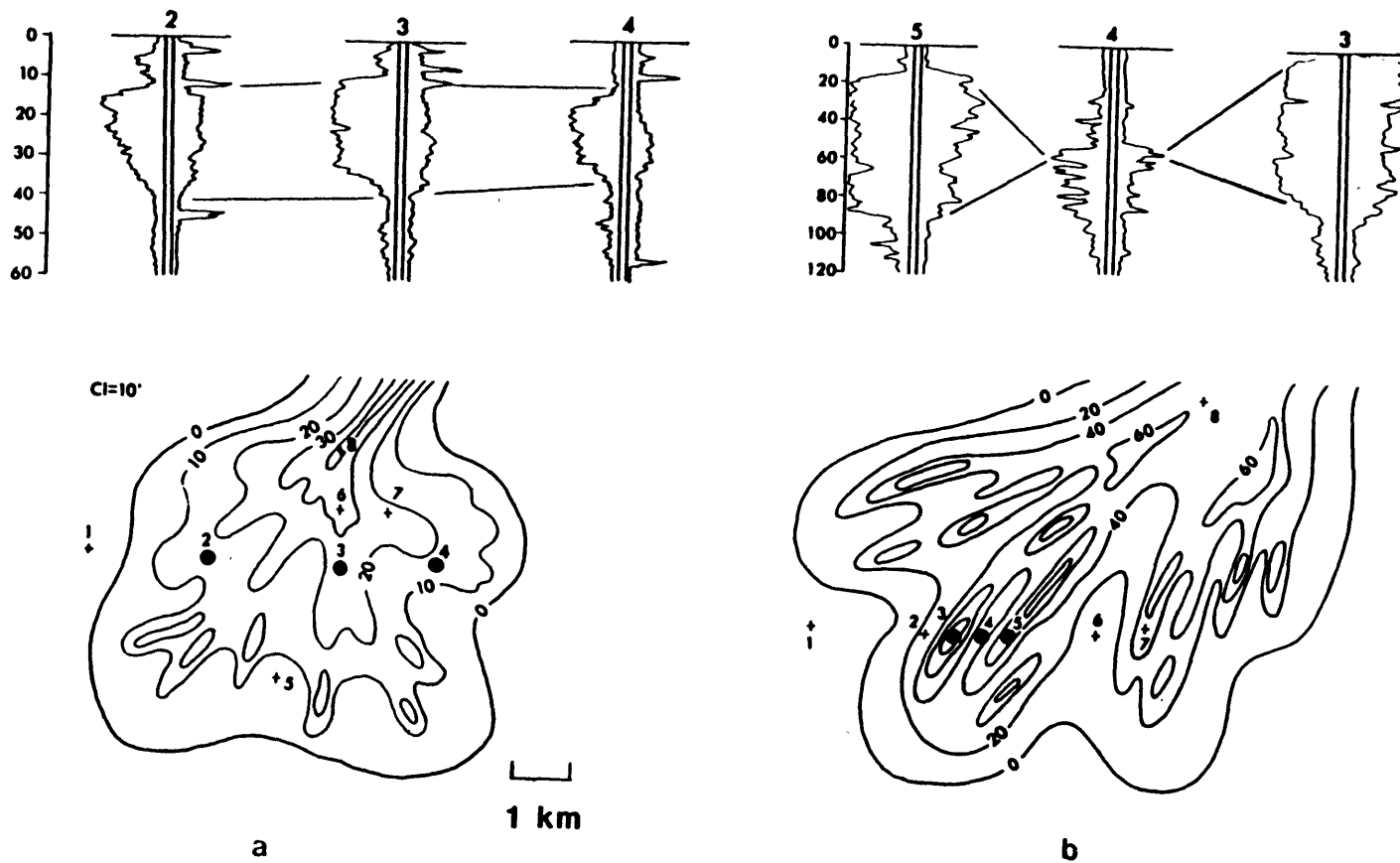


Figure 16: Changes in lateral continuity of sand bodies systems based on the strike of the paleoshoreline. In part (a), correlation in a submarine delta lobe with a well spacing on the order of 1-5 km. would indicate continuous sand bodies parallel with shoreline. In a tidal-dominated river mouth (b), however, wells 3 and 5 are separated by only a few kilometers, yet sand bodies should not be correlated from one well to the next.

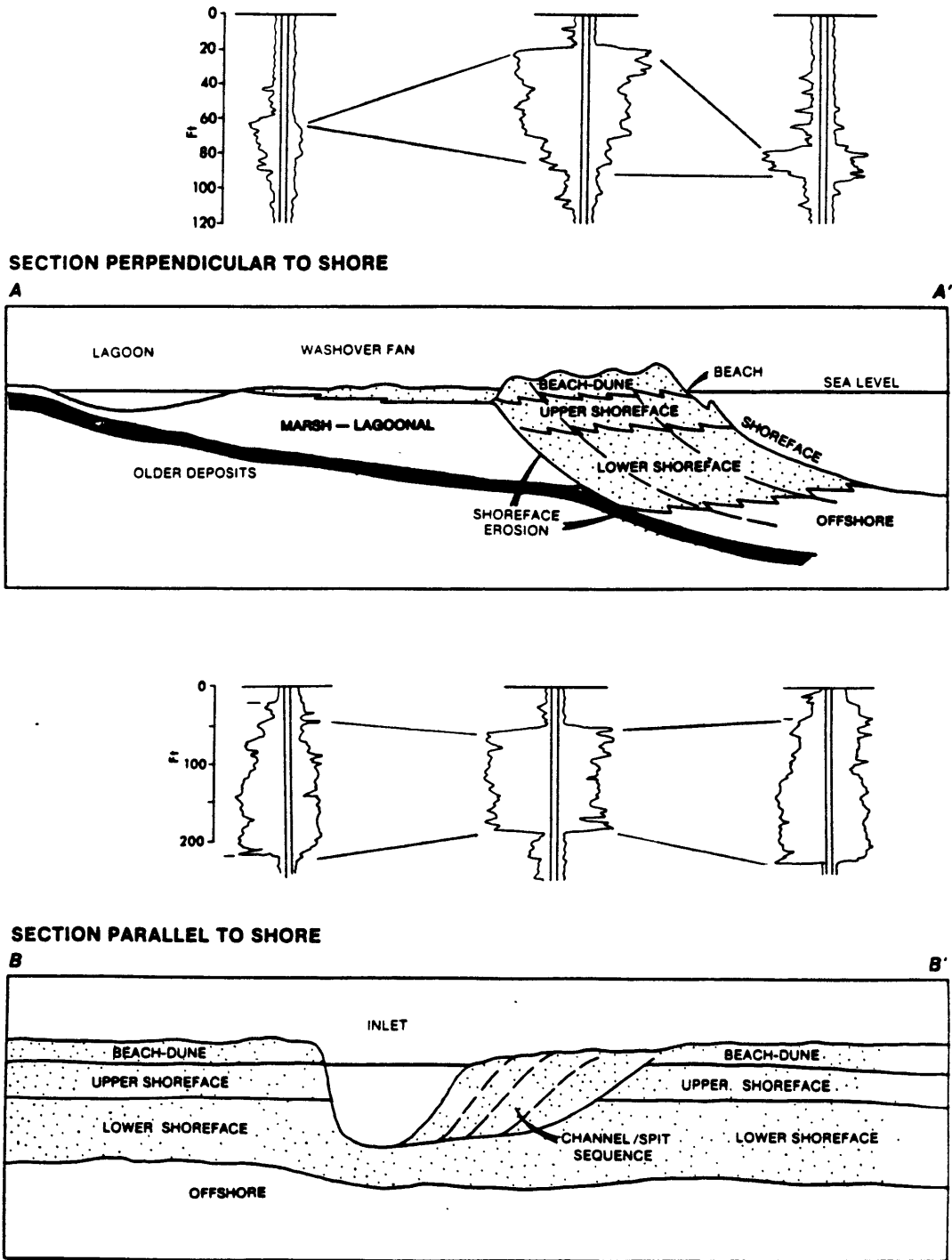


Figure 17: Generalized cross-sections showing changes in continuity of beach facies in a barrier island-lagoon system. When correlating wells which strike perpendicular to shoreline (section A - A'), beach sands appear as discontinuous bodies. When correlating parallel with the shoreline (section B - B'), beach sands can be traced laterally for many kilometers.

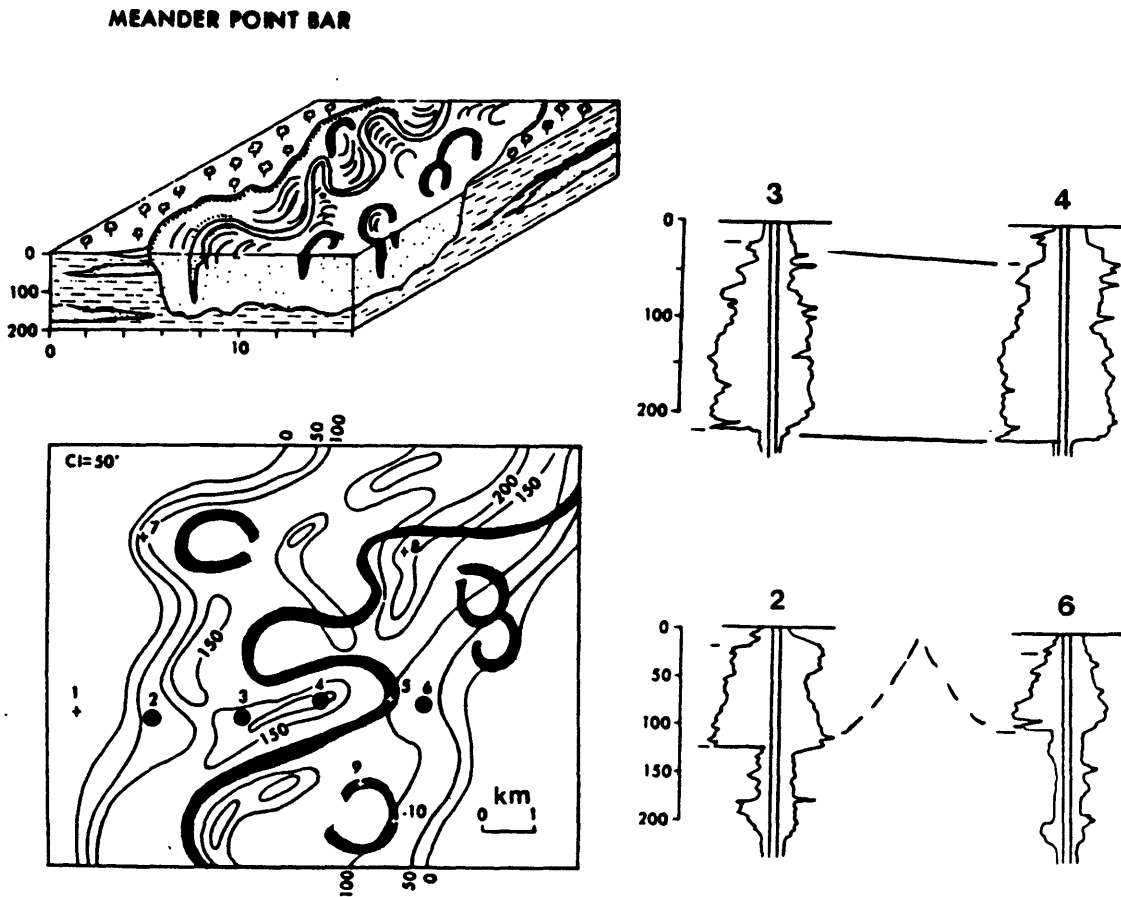


Figure 18: The effects of spatial aliasing on a correlation strategy. Because the continuity of facies can change from the environment and the strike of the paleoshoreline, these factors will affect the minimum well spacing needed to avoid aliasing. In a meandering stream point-bar deposit, correlation with well spacing of roughly 1-2 km. can match thick sands (Wells 3 and 4). If the well spacing is as much as 8-10 km., however, almost identical sand signatures in two wells should not be matched (Wells 2 and 6).

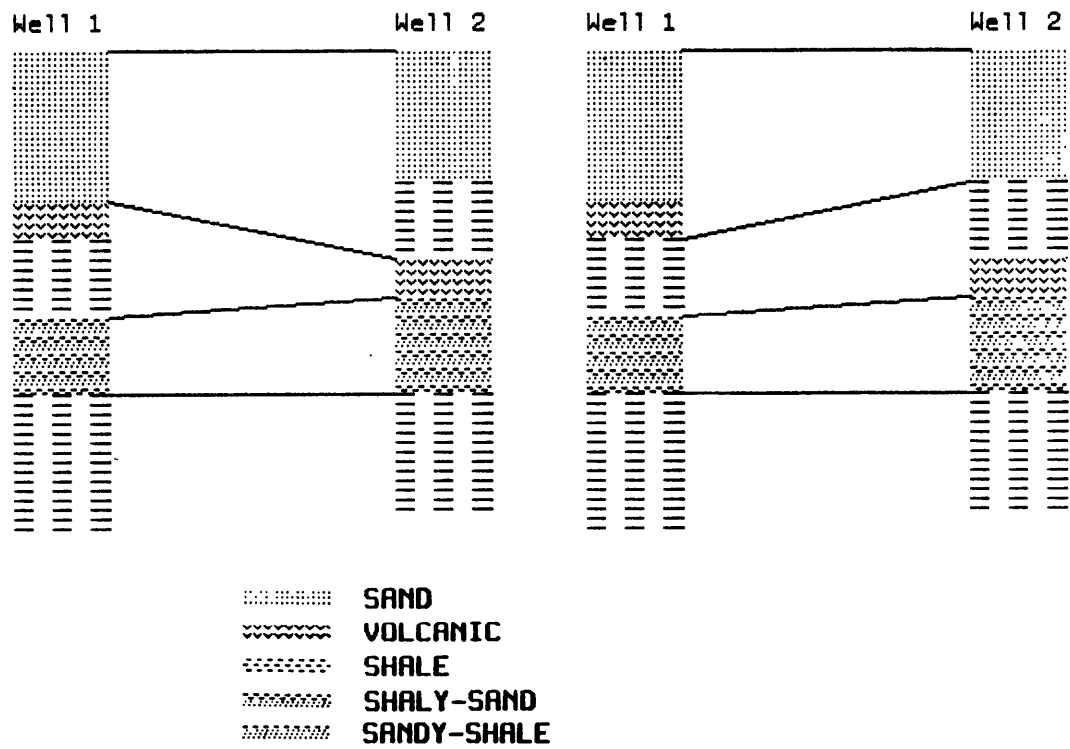


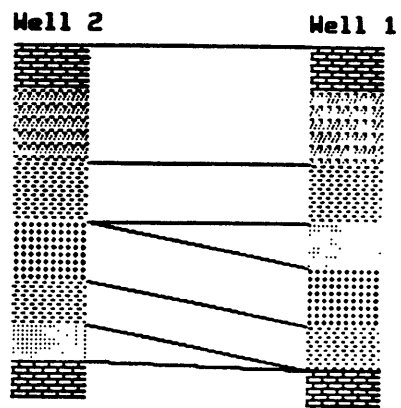
Figure 19: Schematic correlation section where (a) thick shales and sands are the most continuous units (low matching costs), and (b) where volcanics have been designated as marker beds, and assigned the cheapest possible matching cost.

	COAL	SHALE	SAND	LIME	SANDY-SHALE	SHALY-SAND	SALT
COAL	0	100	100	100	100	100	100
SHALE		10	100	100	100	100	100
SAND			20	100	100	100	100
LIME				30	100	100	100
SANDY-SHALE					40	60	100
SHALY-SAND						50	100
SALT							60

LOCAL DISTANCES
FOR LITHOLOGIC MATCH

Figure 20: Diagram of the cost matrix for lithologic correlation. Thick shale units, generally the most continuous units, have the cheapest matching cost. Units which are to have no correspondence are assigned a matching cost of 100, the highest possible value on the normalized scale. Intermediate matching costs are proportional to the continuity of the units with respect to the depositional environment. Lithologies designated as marker beds will automatically be assigned a cost of zero. These values are altered by the knowledge base with rules about paleoshoreline strike and depositional models.

0	100	200	300	400	500	600	700
100	30	130	230	330	430	530	560
200	130	130	230	330	430	530	630
300	230	230	135	235	335	340	440
400	330	330	235	235	275	375	440
500	430	430	240	335	335	280	380
600	530	530	340	285	385	380	380
700	560	630	440	385	385	480	410



Lithologic Match

Warping Path

.....	SAND	SALT
-----	SHALE	=====	LIMESTONE
-----	SHALY-SAND	~~~~~	SANDY-SHALE

Figure 21: Output from the COREX program after the initial match between two wells showing (a) the resulting correlation, (b) the computed global distance matrix, and (c) the warping path that produced the match. The correlation proceeded with no information about depositional environments.

SOME GEOLOGIC SITUATIONS AND THEIR CORRESPONDING WARPING PATHS

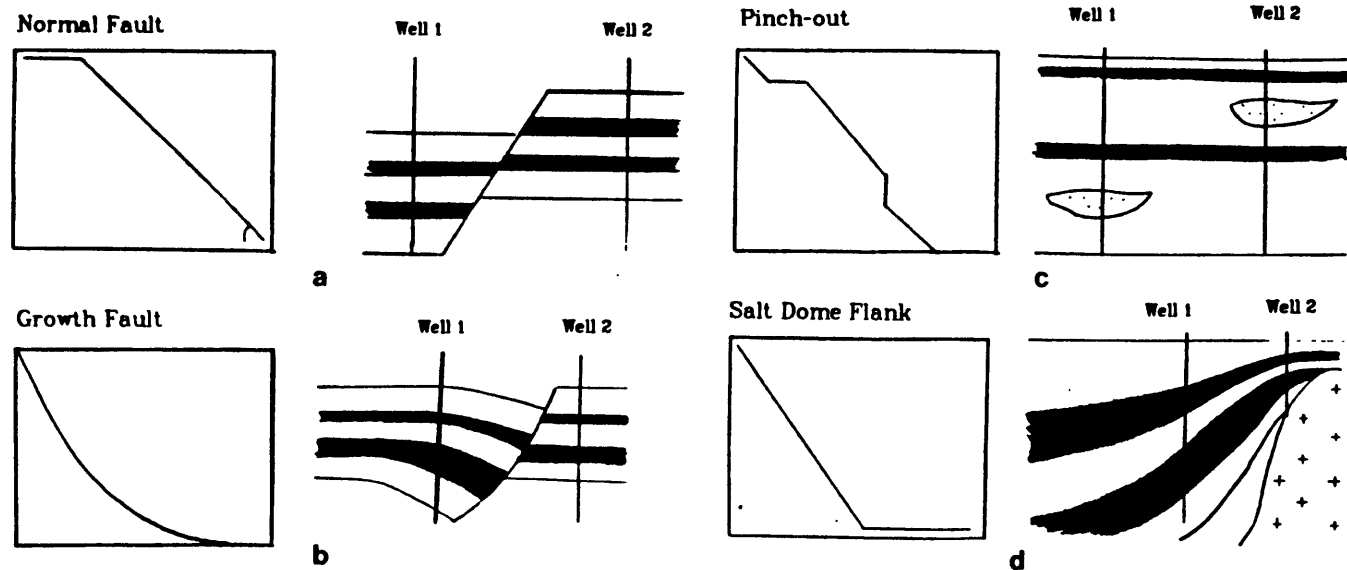


Figure 22: Geological expression in the warping path. Some common geologic settings and the corresponding depth warping paths that would result from a correct correlation across the feature. (a) Simple offset, resulting from a normal fault. (b) Non-linear stretching with depth, from a growth fault. (c) Pinch-outs, from isolated sand lenses. (d) Linear stretching, from flanks of a salt dome.

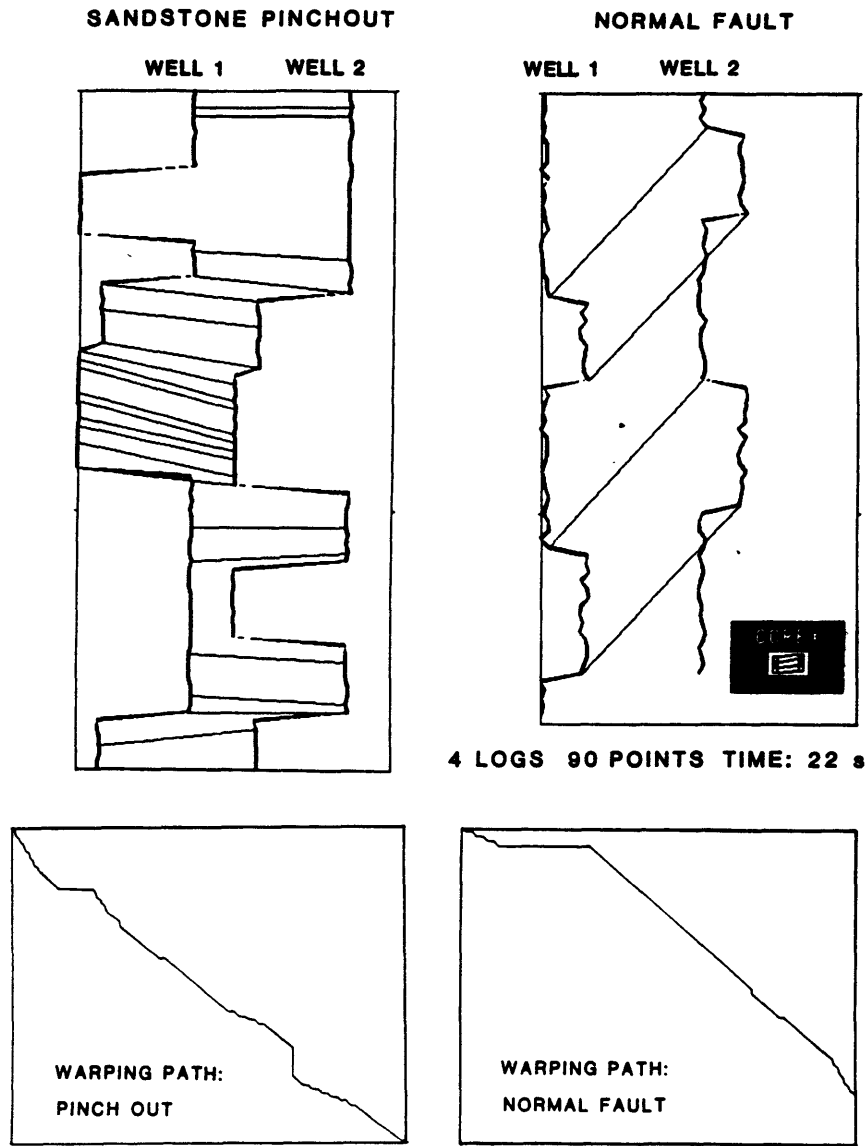


Figure 23: Results of COREX correlation on synthetic examples of geologic structure. Upper left diagram shows correlation across a normal fault with simple depth shift between corresponding points, and the right diagram shows correlation where sandstone units pinch-out between wells, and must be deleted or inserted from one well for a correct correlation. The lower diagram shows the warping paths for each of these situations. Note the correspondence with the theoretical diagrams of figure 22.

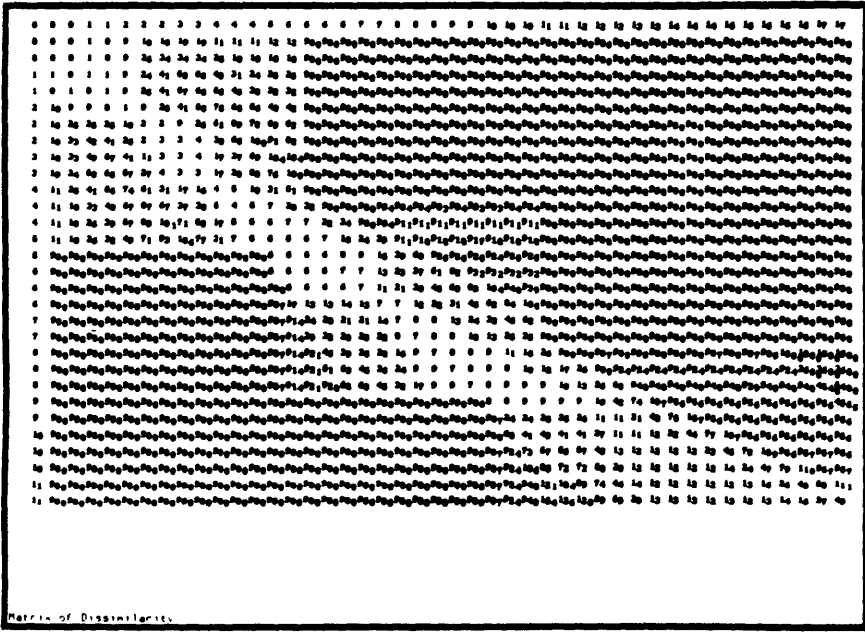
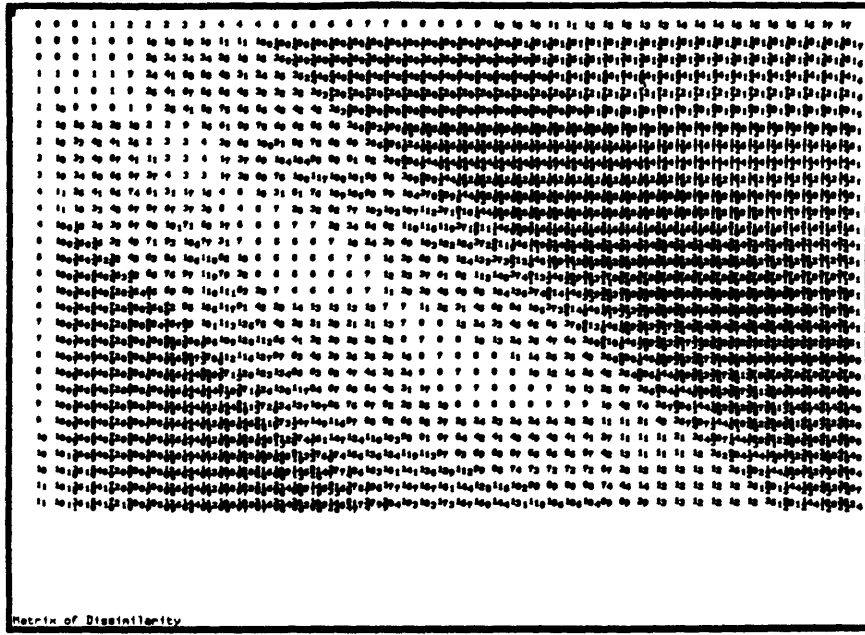


Figure 24: Limiting the possible warping paths with global path constraints. Global cost matrices for a correlation across a listric fault, with 90 points in well 1 and 60 in well 2. In the first matrix, warping paths are limited by a maximum shift constraint of 20 points. In the lower matrix, warping paths are limited by two tie lines. Large cost values (darker areas) represent areas that are eliminated from the warping function search space.

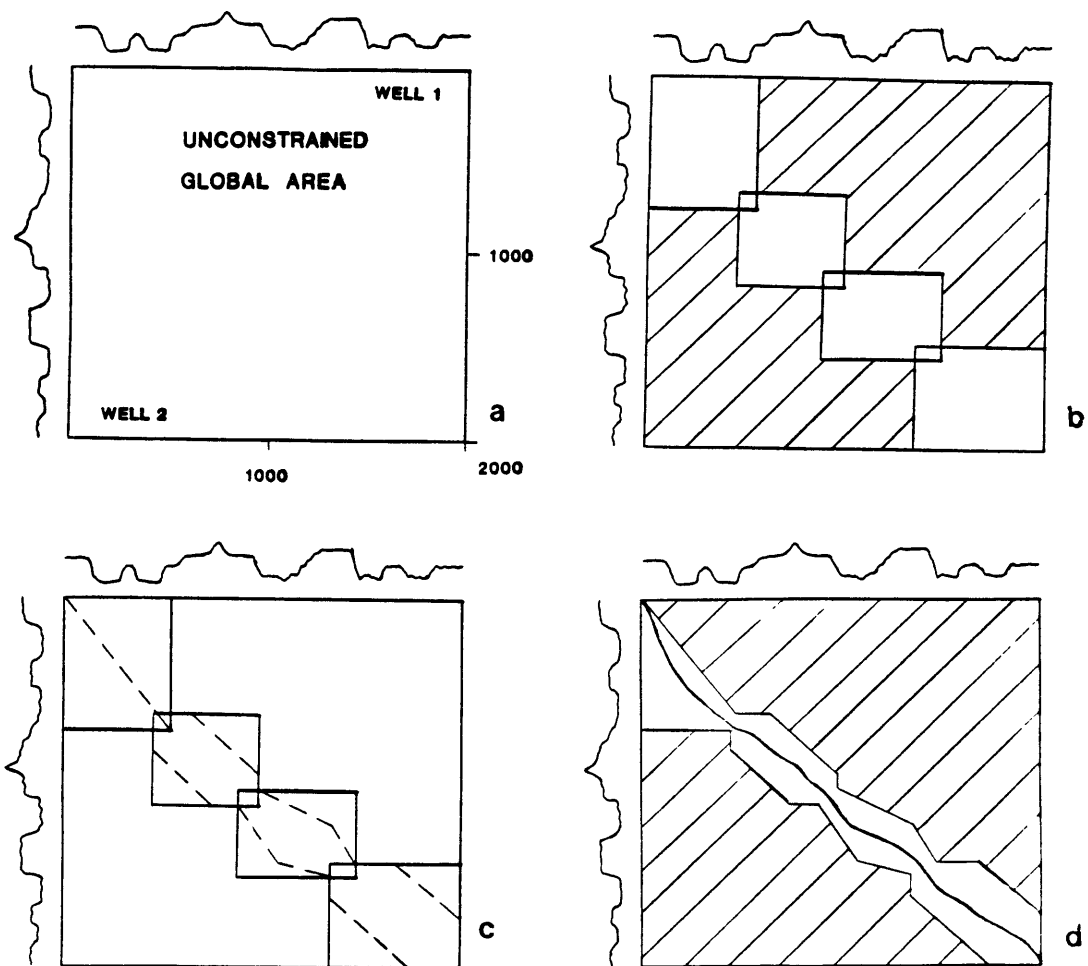
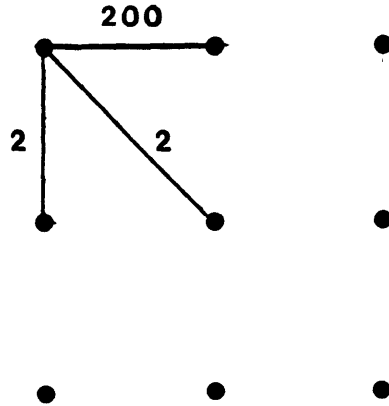


Figure 25: The breakdown of a large correlation into sub-problems with global path constraints.
 (a) The simple correlation problem with no constraints. (b) The reduced problem after tie points from the initial match. (c) Further global and local reduction after analysis of individual sections. (d) The final correlation for dynamic depth warping to solve. Most of the geologically unreasonable correlations have been eliminated, and the cost of the correlation drastically reduced.



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	
1	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
2	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
3	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
4	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
5	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
6	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
7	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
8	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
9	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
10	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
11	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
12	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
13	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
14	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
15	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
16	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
17	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
18	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
19	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
20	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
21	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
22	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
23	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200

Matrix of Dissimilarity

Figure 26: Adjusting the global path with local continuity constraints. Part (a) shows the possible local motions and their corresponding weights. Part (b) shows the manifestations of this asymmetric weighting in the global path.

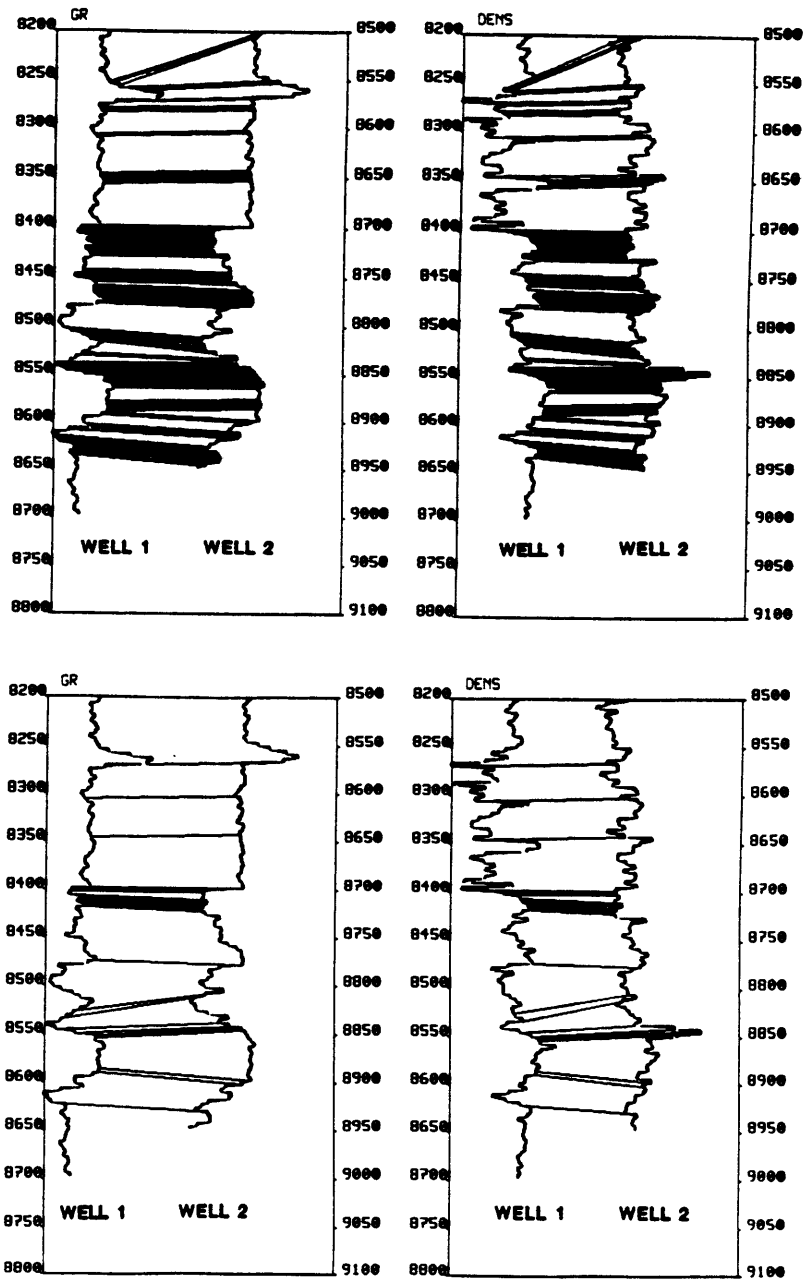
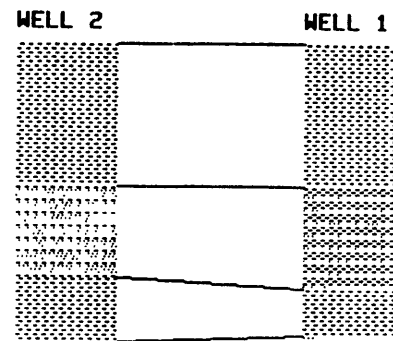


Figure 27: Improved correlation from noise analysis. In the upper diagram, correlation proceeded using five logs, some of them noisy, giving poor results. After noise analysis, the resistivity and neutron logs were omitted from the calculation, resulting in the improved correlation shown below. The major improvement occurred between the depths 8500 feet and 8600 feet.

0	100	200	300	400
100	5	105	112	212
200	105	85	185	212
300	119	185	95	195
400	219	219	195	110



WARPING FUNCTION FOR WELLS:

Well 1 Well 2

■■■■■ SAND
 ■■■■■ SHALE
 ■■■■■ SHALY-SAND
 ■■■■■ SANDY-SHALE

Figure 28: Results from offshore wells in West Africa, showing the initial lithologic correlation, and the cost matrix and warping path which matched the lithologies. The correlation proceeded with the following rules: (1) Environment is continental shelf; (2) Correlation strike is perpendicular to shoreline; (3) Thick-shales are the most continuous units, sands the next continuous; and (4) Sandy-shales can correlate with shaly-sands down the paleoslope.

CORRELATION: WEST AFRICA

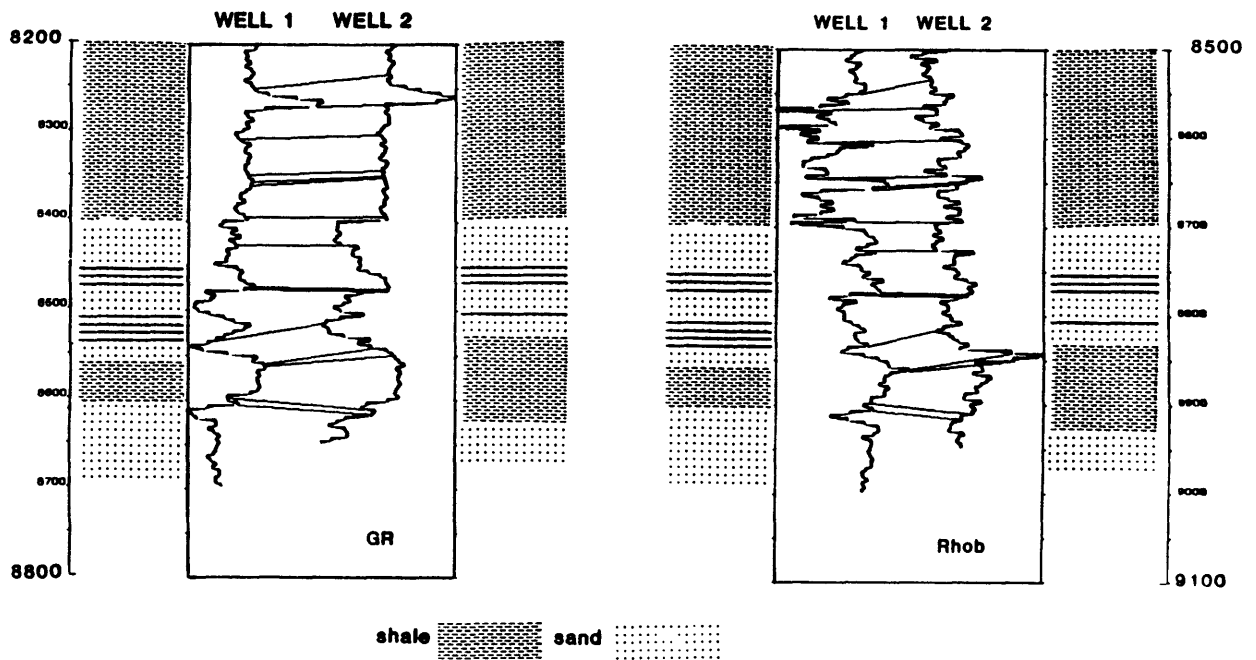


Figure 29: Final correlation for West African wells displaying gamma ray and bulk density logs, with lithologic inversion displayed next to the logs. Note the successful matching of the lithologic boundaries, and the expansion and contraction of various sands and shales in the lower section of the wells. The advantage of a multiple-log correlation shows in the display of the various logs. There appears to be correlation of meaningless features in the thick upper shale section of the gamma ray log, but the density log reveals that lithology changes are taking place, even though they are not reflected in the gamma ray log.

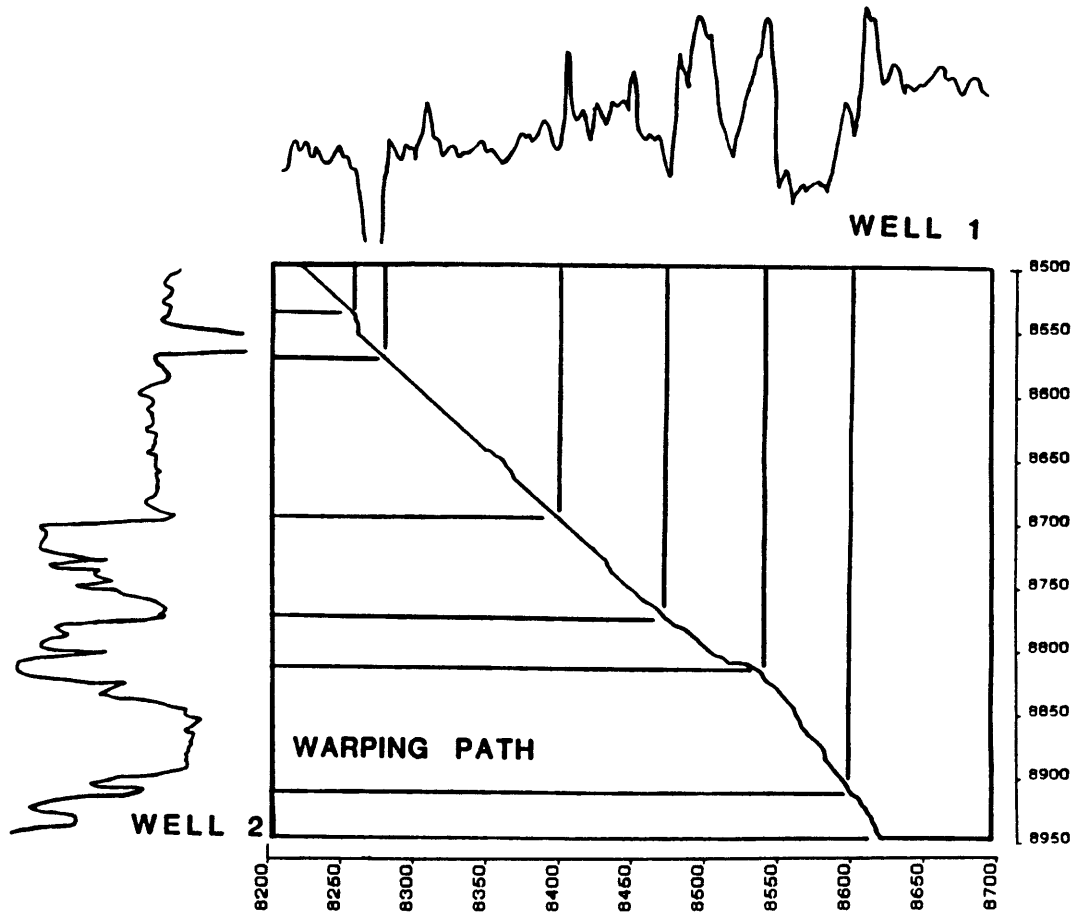


Figure 30: The warping path which matched the West African wells. Note the reflection of the geologic features in the warping path: 1) Slight expansion of the highly radioactive zone (around 8260 feet) from well 1 relative to well 2 shows up as a vertical line in the path; (2) Expansion of a sandy-zone at 8525 from well 2 relative to well 1 shows up as a horizontal section in the warping path; (3) The superposition of a curved section onto a rough 45 degree diagonal, representing the fact that the sections correlate well with depth at the beginning and end points (a diagonal line), but that non-linear stretching is required in a few areas (the curved sections). Note the similarity with parts (a) and (c) of figure 22.

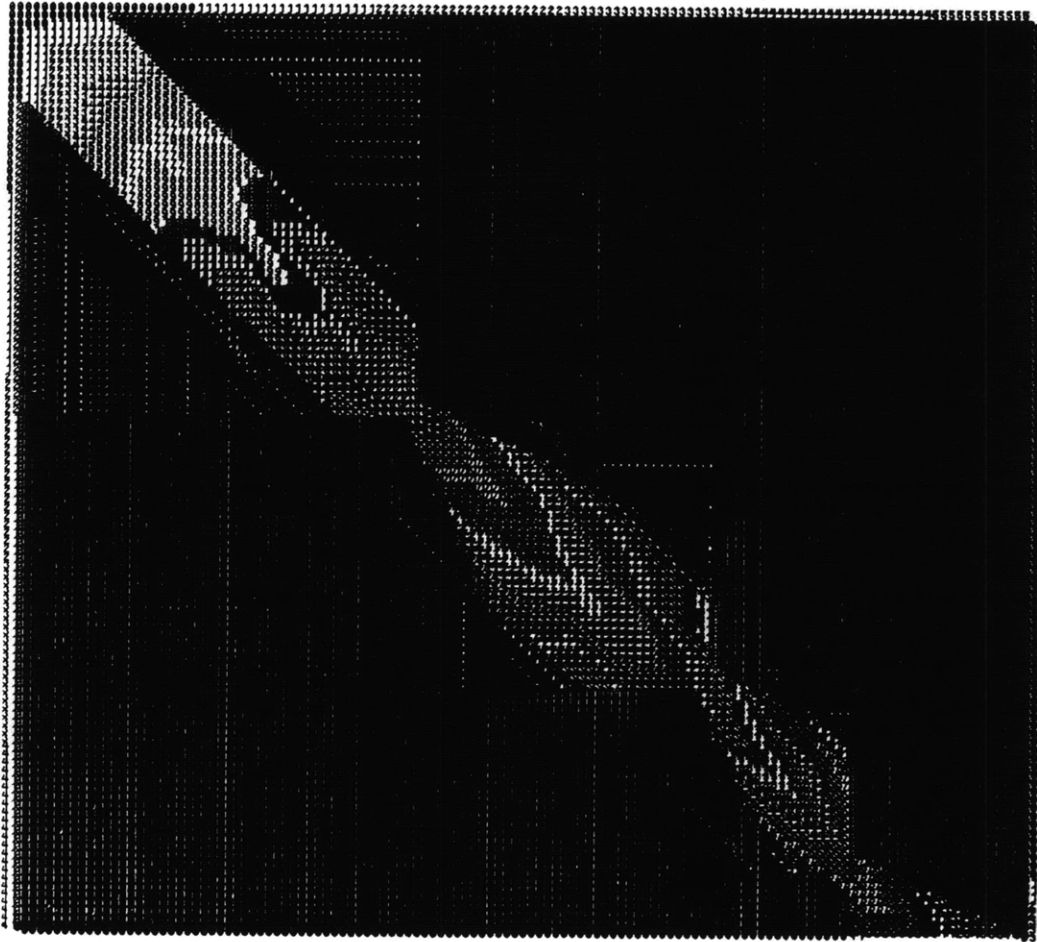


Figure 31: The global cost matrix that resulted from the correlation of the West African wells. Darker regions mean higher cost values. Black areas represent portions of the global area that were eliminated by tie lines and maximum shift constraints. Notice the “propagation” of the lower cost values through the diagonal of the matrix, and how this corresponds with the warping path of figure 30.

CORRELATION: THRACE BASIN, TURKEY

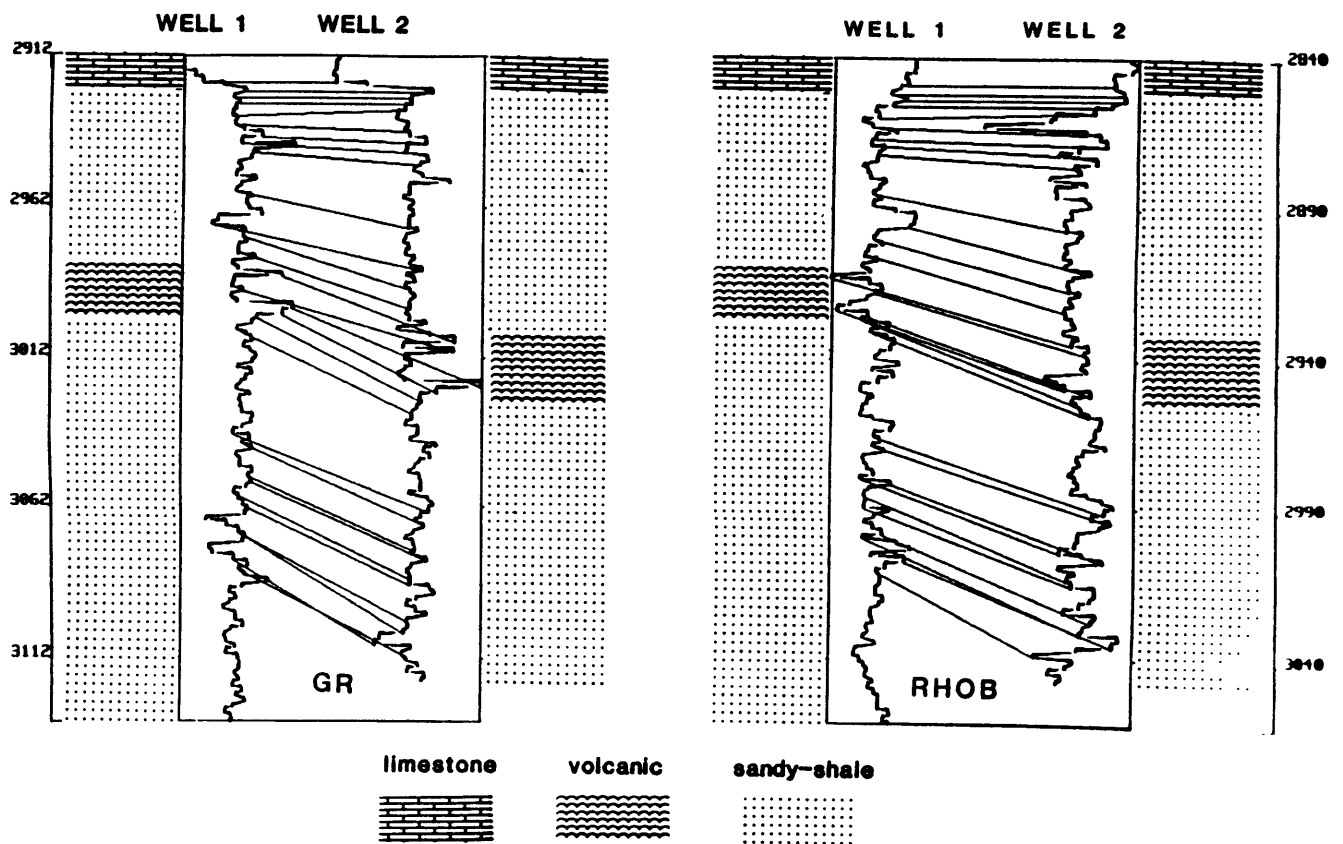


Figure 32: Final correlation from wells in the Thrace Basin, Turkey. No lithologic correlation was performed. Part (a): Correlation with lithologic inversion displayed next to the wells. Notice the successful match of the formation onset and the section of volcanic tuffs in each well. The tie lines display some non-linear stretching of well 1 relative to well 2, which may represent a thickening of section down the correlation line.

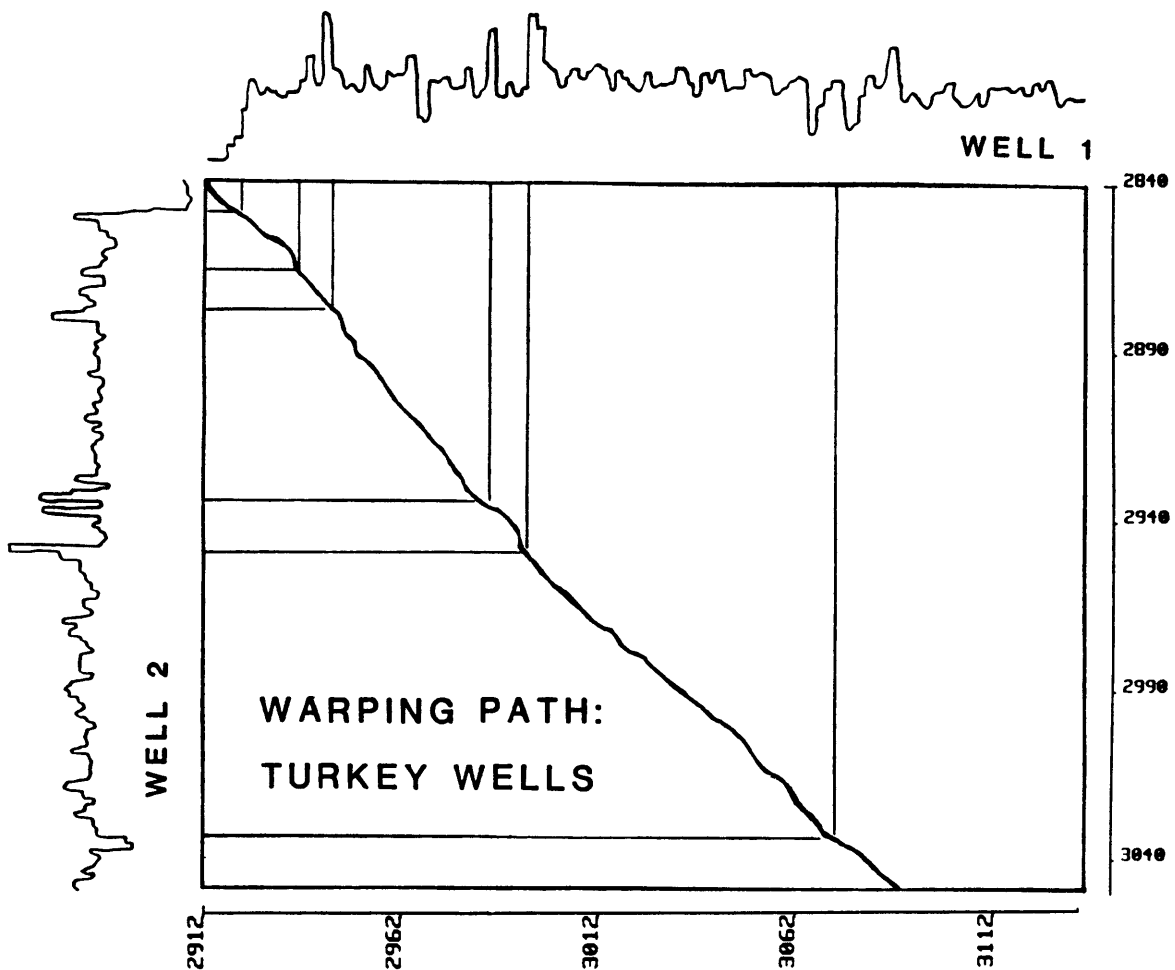


Figure 33: Warping path which matched the wells in the Thrace Basin. Notice the superposition of the different geometric properties on the warping path, and how these correspond to the theoretical paths discussed in Figure 22. (1) A diagonal path toward the bottom of the matrix represents the constant stretching of section in well 1 relative to well 2; and (2) a curved portion is superimposed on this general trend to account for the non-linear stretching with depth that occurs from well 1 to well 2, as is shown in the tie lines of figure 32. Note the similarity with parts (b) and (d) of figure 22.

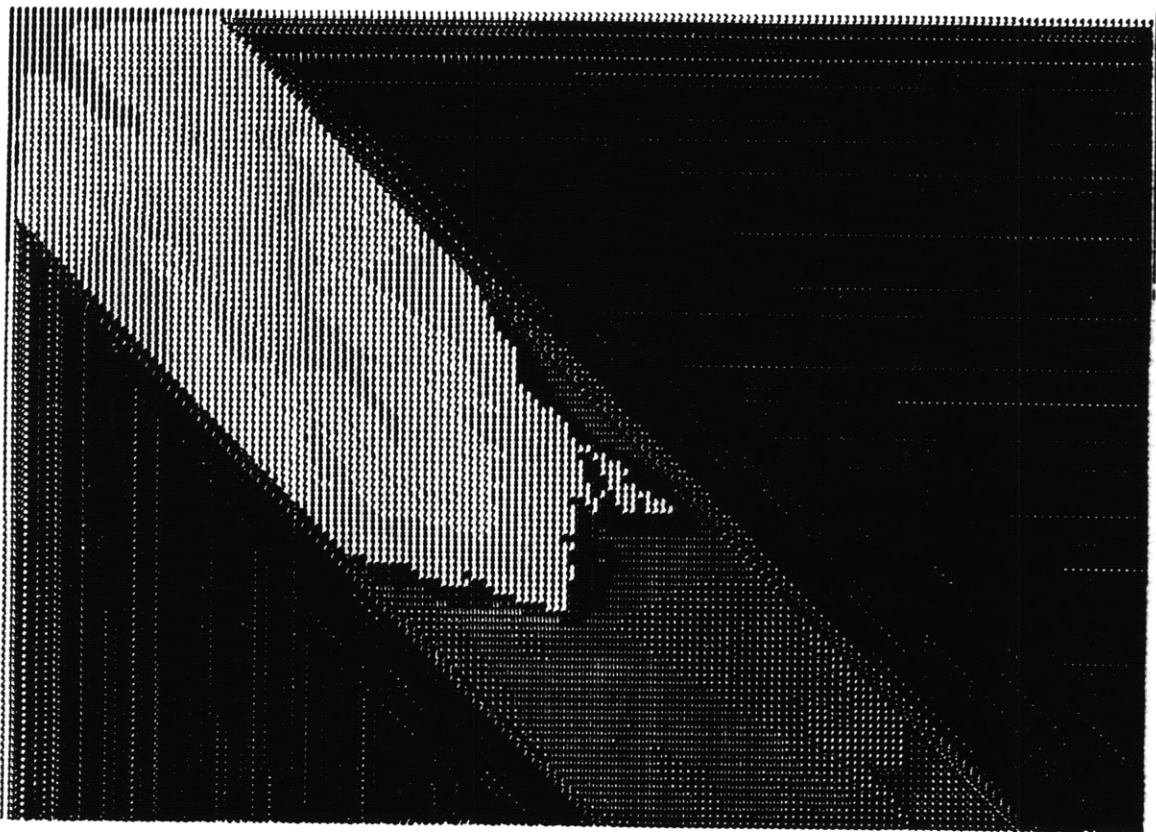


Figure 34: Global cost matrix for Thrace Basin correlation. Darker areas represent high matching costs. Black areas are sections that were eliminated from calculation by a maximum shift constraint. Compare the minimum distance areas with the warping path seen in figure 33.

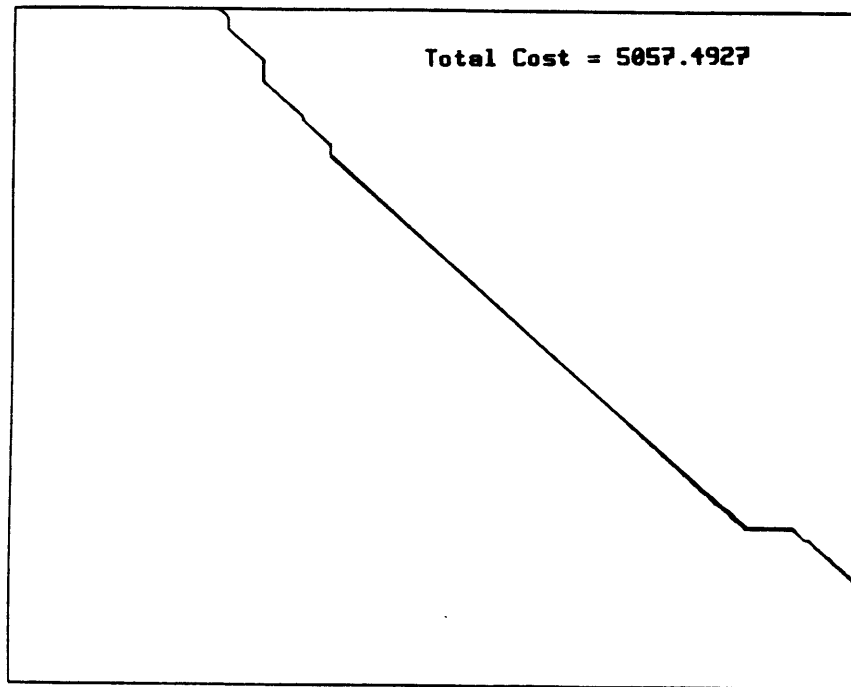
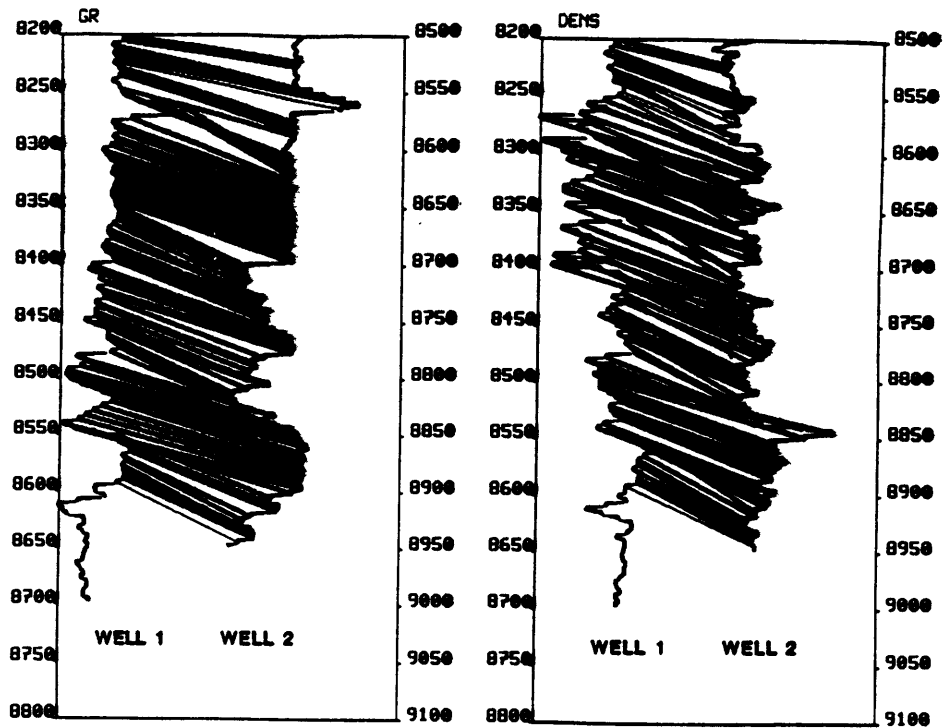
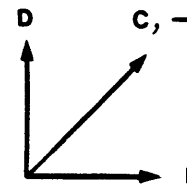


Figure 35: An incorrect correlation resulting from a warp that proceeded through a “local” instead of a global minimum path. Local minimums are created in the global path by two objects which are similar in character (ie. two sands), but do not match because they are separated by too large a distance in depth. (a) Final correlation, (b) Warping Path.

STRING MATCHING = PATH FINDING

$D(N,M)$

R	7	6	5	4	3	2
E	6	5	4	3	2	3
I	5	4	3	2	2	3
L	4	3	2	1	2	3
L	3	2	1	2	3	4
I	2	1	3	4	5	6
H	1	2	3	4	5	6
-	M	I	L	L	E	R



(After Anderson and Gaby, 1983)

A MINIMUM EDIT PATH

Figure 36: String matching as a path finding problem. Matching two character sets is equivalent to find the minimum-cost path through a matrix. (a) Each value in the grid represents the total cost of matching the words to that point. At any point in the match, the warping path is restricted to one of the three motions shown in (b). Each of the three possible motions horizontal, vertical, or diagonal – correspond to the three possible edit operations – insertion, deletion, or substitution.

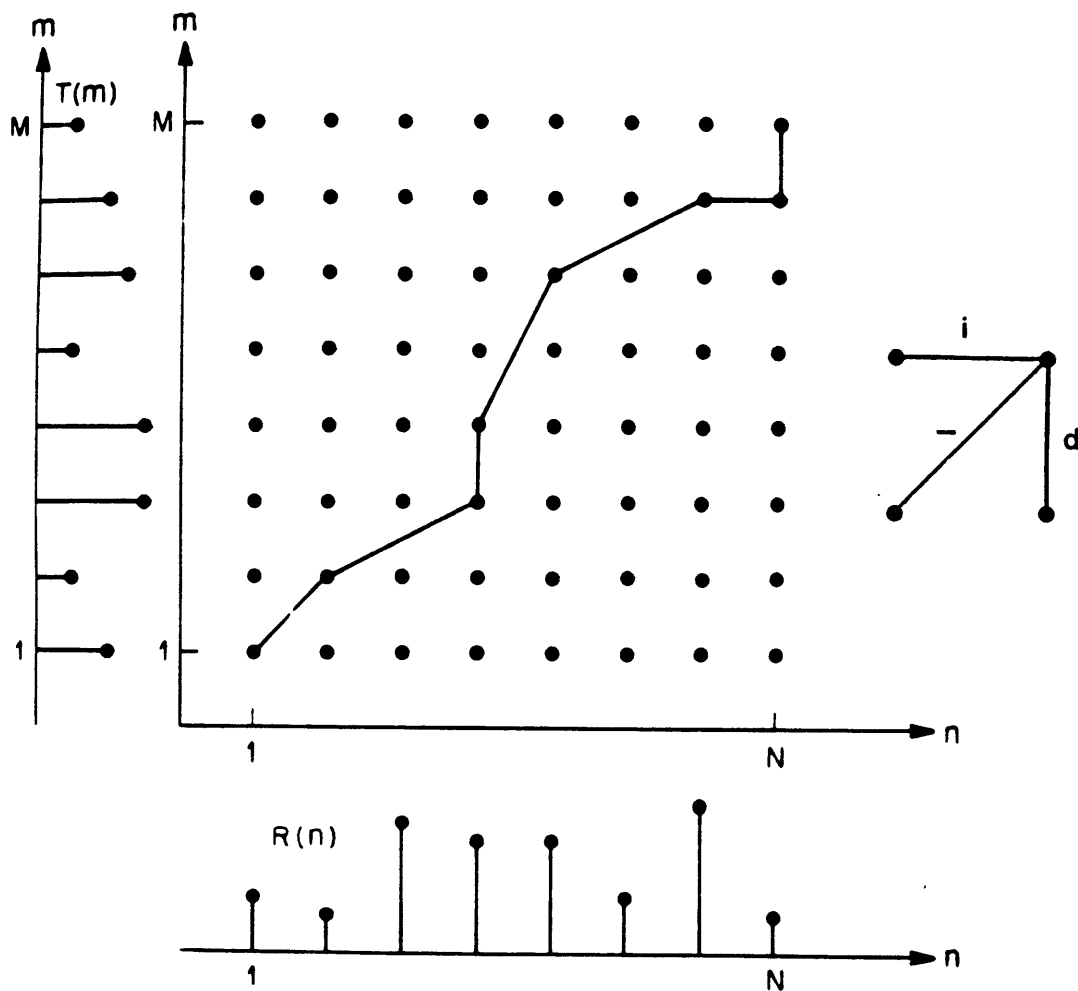


Figure 37: Discrete warping as path finding. The same dynamic programming principles apply as in the string matching problem, but the characters are replaced by discrete values from a digitized curve. The same path restrictions apply, representing insertion, deletion, and substitution, as shown in figure 36.

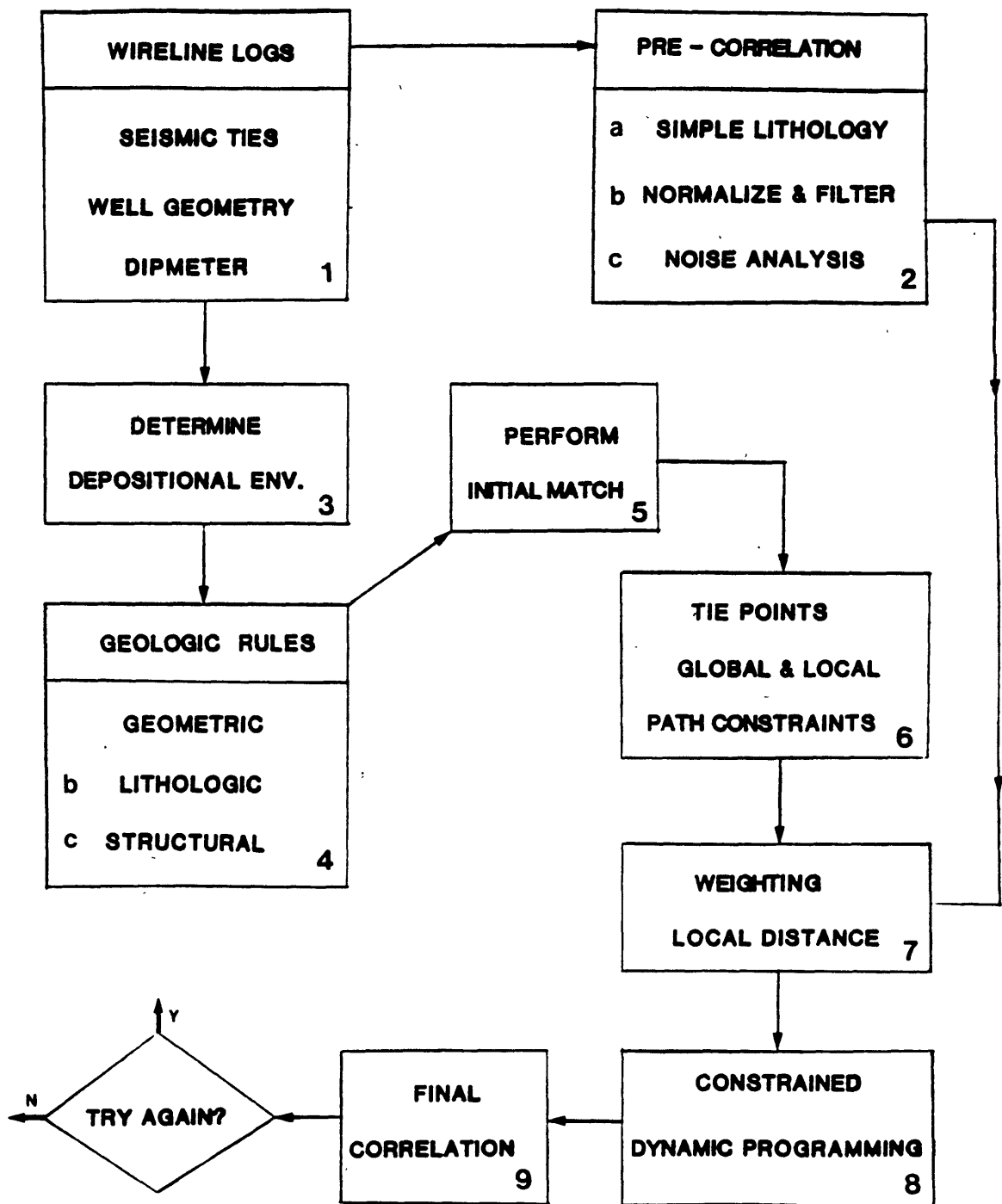


Figure 38: Flow chart of the COREX system.

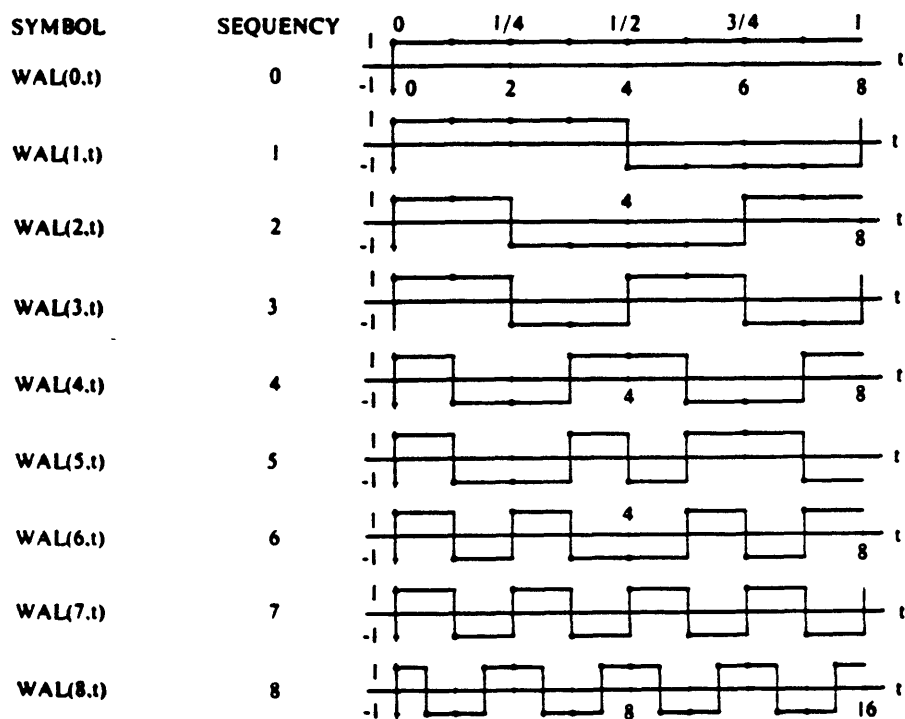


Figure 39: The first eight members of the set of orthogonal Walsh functions. Each function assumes only the values +1 and -1 with discrete transitions between the two.

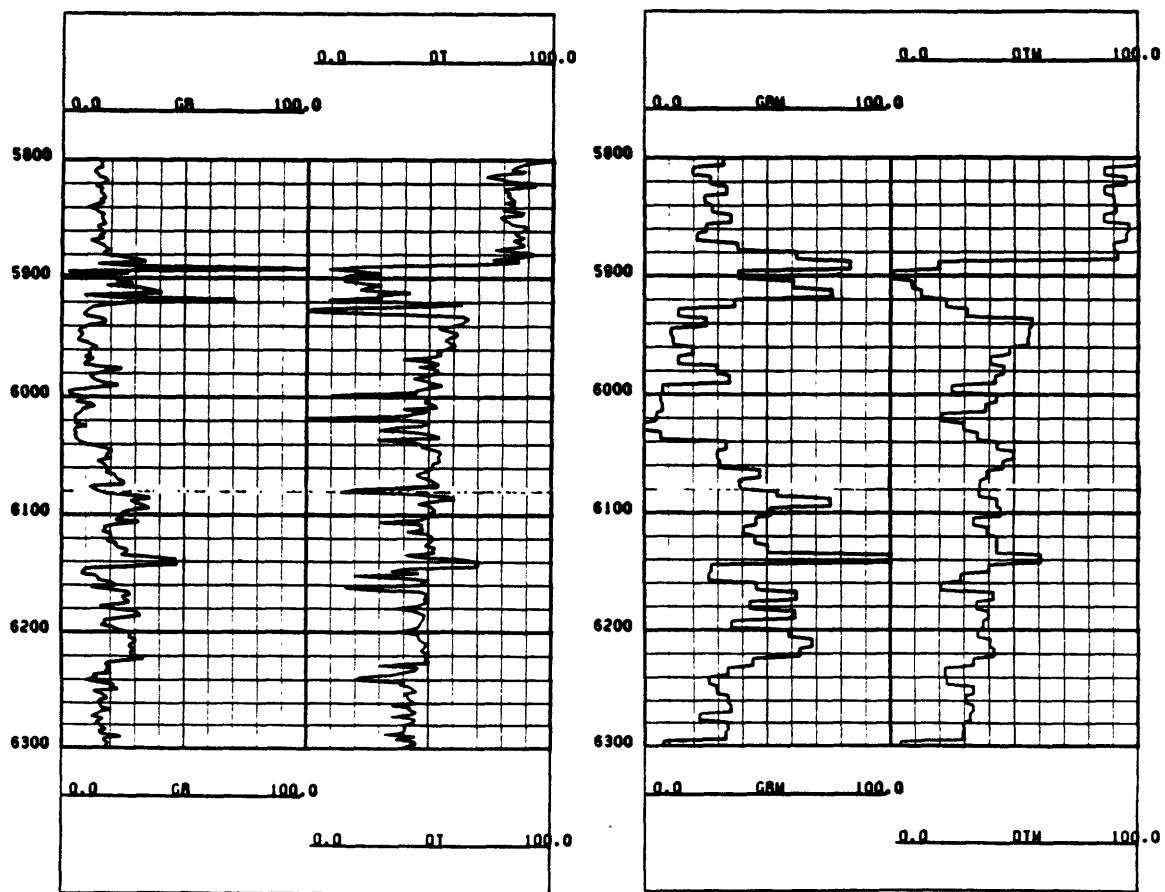


Figure 40: Smoothing logs by Walsh filtering. (a) Original gamma ray and sonic logs, sampled every 2 feet. (b) The same logs after filtering to a minimum bed thickness of 8 feet.