HHS Public Access

Author manuscript

J Bioinform Comput Biol. Author manuscript; available in PMC 2015 August 07.

Published in final edited form as:

J Bioinform Comput Biol. 2013 October ; 11(5): 1350011. doi:10.1142/S021972001350011X.

A Peak Alignment Algorithm with Novel Improvements In Application to Electropherogram Analysis

Fethullah Karabiber

Department of Chemistry, University of North Carolina, Chapel Hill, NC 27599-3290, USA

Abstract

Alignment of peaks in electropherograms or chromatograms obtained from experimental techniques such capillary electrophoresis remains a significant challenge. Accurate alignment is critical for accurate interpretation of various classes of nucleic acid analysis technologies, including conventional DNA sequencing and new RNA structure probing technologies. We have developed an automated alignment algorithm based on dynamic programming to align multiple-peak time-series data both globally and locally. This algorithm relies on a new peak similarity measure and other features such as time penalties, global constraints, and minimum-similarity scores and results in rapid, highly accurate comparisons of complex time-series datasets. As a demonstrative case study, the developed algorithm was applied to analysis of capillary electrophoresis data from a Selective 2′-Hydroxyl Acylation analyzed by Primer Extension (SHAPE) evaluation of RNA secondary structure. The algorithm yielded robust analysis of challenging SHAPE probing data. Experimental results show that the peak alignment algorithm corrects retention time variation efficiently due to the presence of fluorescent tags on fragments and differences in capillaries. The tools can be readily adapted for the analysis other biological datasets in which peak retention times vary.

Keywords

Signal alignment; Dynamic programming; Peak similarity; Mobility shift; Electrophoresis

1. INTRODUCTION

Bioinformatics is the application of computer sciences and mathematics to the management and analysis of complex datasets to aid the solution of biological problems [1]. Alignment of time-scaled and time-shifted signals is often necessary in the analysis of datasets obtained in biological experiments. [2]. Time shifts can occur when a signal is measured as a function of time for two or more datasets with small or large-scale differences in experimental conditions across repeated samples, which could be due to factors including temperature or voltage changes, instrument imperfections, or variations in flow rates. Thus comparison of

different samples can be complicated by differences in their time scales or axes or the differences in the lengths of sample vectors.

Many different methods based on dynamic programming approaches have been used to correct drift in retentions times. One of the well-known algorithms used to compare two discrete signals is Dynamic Time Warping (DTW) [3]. DTW is a method of solving complex problems by breaking them down into simpler steps [12]. Although it was originally developed for speech recognition [3], classical DTW and its variations have also been applied to many other fields, and DTW is a fast and efficient method for alignment of time-dependent sequences. For example, dynamic programming has been applied to alignment of peaks in gas chromatography-mass spectrometry (GC-MS) spectra [9], and dynamic programming and a similarity function based on position, width, and amplitude are used to align nuclear magnetic resonance(NMR) spectra [10]. Methods for analysis of peak similarity have been reviewed in [2].

The data generated by capillary electrophoresis (CE) of nucleic acid fragments can be corrected automatically for shifts in retention time using a dynamic programming approach. The dynamic programming approach attempts to find an optimal alignment by considering three options for each location in the sequence and selecting the best option before considering the next location [3]. Each iteration considers the alignment of two bases (one from each sequence) or the insertion of a gap in either one. The best of the three is chosen and the system then moves on to the next element in the sequence. To handle this efficiently, the computer program maintains scoring and trace back matrices using a cost matrix. The Needleman-Wunsch (NW) [4] and DTW [5,6] approaches are widely used to align CE data, but neither approach optimally handles data collected over time.

In this work, we have developed a local and global peak alignment algorithm to solve the shift problems in capillary electrophoresis data obtained for samples collected over a time course. To improve the performance, the new algorithm combines properties of the NW and DTW algorithms. This algorithm creates mobility shift corrections that effectively solve the retention time shift problem. In addition, preprocessing tools and further optimizations were developed to improve the performance of the algorithm. To demonstrate the utility of the algorithm, it was applied to analysis of CE data from RNA structure-probing experiments using Selective 2′-Hydroxyl Acylation Analyzed by Primer Extension (SHAPE) [8].

1.1 Overview of the SHAPE Experiment

SHAPE substantially increases the accuracy of RNA secondary structure prediction [7, 17] by providing measurements of RNA flexibility at single-nucleotide resolution. SHAPE reagents are electrophilic small molecules that preferentially react to form 2′-O-adducts with flexible nucleotides that are typically found in single-stranded regions [17]. SHAPE experiments can be quantified using CE to resolve fluorescently labeled cDNA libraries. In order to quantify SHAPE experiments, cDNA libraries are prepared from RNA samples that have been treated with SHAPE reagent or left untreated. These cDNA libraries are mixed with a common dideoxy sequencing ladder and run in the capillary. The result is four individual channels of fluorescence intensity versus elution time data: plus SHAPE reagent (RX), without SHAPE reagent or background (BG), and two sequencing ladders (SL1 and

SL2) (Figure 1a). A sample of raw data is shown in Figure 1b. A plot of normalized SHAPE reactivities is shown in Figure 1c. Nucleotides with higher SHAPE reactivity (>0.7) are flexible and likely to be single-stranded, and those with lower SHAPE reactivity (<0.3) are constrained and likely to be base-paired or involved in tertiary contacts.

Analysis of CE data obtained from SHAPE experiments presents two challenges. First, each reaction is analyzed using a DNA primer labeled with a different fluorophore. The dyes alter electrophoretic migration rates such that cDNAs of the same length have slightly different elution times when labeled with different fluorophores. Secondly, experimental conditions in each capillary may differ, resulting in retention time differences for cDNAs of the same length.

ShapeFinder software [8] has been widely used to analyze SHAPE data. Mobility shift tools are an important part of the analysis of raw SHAPE capillary electrophoresis data in ShapeFinder. ShapeFinder has several mobility shift tools that can be combined serially to correct for time offsets. However, parameters for the mobility shift require manual intervention and the process is time consuming to implement (about 15–30 minutes per set of traces) and must be optimized for each RNA sequence. We set out to automate this process using a combination of dynamic programming and preprocessing tools.

2. METHODS

2.1 Preprocessing

2.1.1 Smoothing—Several preprocessing techniques are used to improve performance of the alignment algorithm. Since the raw data have some high frequency noise, a triangular smoothing method is applied. The triangular smoothing filter is similar to a boxcar (i.e., rectangular) filter, except that it uses a triangularly weighted smoothing function [11]. This filter more effectively reduces high-frequency noise than does the boxcar filter. The smoothing coefficients are symmetrically balanced around the central point. Since peaks are the most important measurement objective, it is important to preserve the peaks and their related features in the signal. Triangular smoothing reduces noise and preserves the peak shapes.

2.1.2 Enhancement—To localize the peaks more accurately and to detect low-quality peaks, a second-derivative-based resolution enhancement technique may be applied after smoothing. In this enhancement method, the second derivative of the input signal is subtracted from the input. A useful property of this procedure is that it does not change the total peak area, because the total area under the curve of the second derivative of a peak-shaped signal is zero [11].

2.1.3 Baseline Adjustment—The baseline adjustment algorithm is used to remove background signal and to normalize the baseline. In this algorithm, the minimum signal intensity points within the specified window size (typically ten times the average peak width) are found, and then the baseline signal is obtained by applying linear interpolation to these minimum points. Finally, the baseline drift is subtracted from the data signal to obtain the baseline-adjusted signal [7].

2.1.4 Normalization—Experimentally derived data commonly have experimental biases that stem from variations in concentrations of chemicals used in experiments or from imperfections in the detection equipment. In order to compare experiments, it is necessary to remove such biases; this process is usually accomplished through normalization. In this study, the commonly employed zero-mean, unit-variance statistical normalization is used. The mean of the data is subtracted from each data point and then these differences are divided by the standard deviation to obtain normalized data.

2.2 Peak Detection and Similarity

Peak detection is performed by looking for the downward zero-crossing in the first derivative of the time-series data sets. Peak maxima are identified as the points where derivatives transition from positive to negative. After detection of peak positions, peaks are represented in a list. Each time series is represented by its own peak list, in which peaks are ordered by their retention times. Retention time, amplitude, and shape characterize each peak in a peak list. The peak shape is obtained from interpolation (using cubic spline) of the intensity functions at the midpoints of several (usually 3) consecutive time points on each side of the peak center, thus deriving a peak shape function defined over a number of time points (N=2kn+1; here k is used as a factor to increase the number of points).

In order to represent the peaks, we employ two vectors $A = [a_1,a_2,\ldots, a_n]$and $B = (b_1,b_2, \ldots,b_n)$ both of length *n*. *A* and *B* vectors represent the points in the peaks. The point a_i is the i^{th} element of the time series A. The length of the vectors used to represent the peaks must be same to obtain similarity.

A number of different functions have been used to measure similarity of two peaks. For example, Robinson et al. [9] used cosine similarity, which measures similarity between two vectors as the cosine of the angle between them. The peak amplitude and width have also been used [2], and our approach is based on this latter method. Our approach differs from the previously described method in that we use the derivative of the peak shape function instead of the actual data points to find the similarity. Derivatives retain the information about peak shapes while reducing differences due to baseline drift. The peak similarity function is:

$$psim(A,B)=\frac{1}{2kn+1}\sum_{i=1}^{2kn+1}\left(1-\frac{\left|a_i'-b_i'\right|}{2\cdot\max\left(\left|a_i'\right|,\left|b_i'\right|\right)}\right) \quad (1)$$

Note that the range of the peak similarity value should be between −1 and +1. Additional mathematical operations may be used if necessary to correct the range. The result of equation (1) is multiplied by 2 and then subtracted by 1 to obtain same similarity range.

2.3 Global Peak Alignment

After applying preprocessing tools and peak detection to two different datasets, we have two peak lists $PL_A = [a_1, a_2,\ldots, a_N]$ and $PL_B = [b_1, b_2,\ldots, b_N]$ that contain *N* and *M* peaks, respectively. The alignment of the peak lists PL_A and PL_B refers to the establishment of a

one-to-one correspondence between the peaks from the two lists with the possibility that any peak from one list has no matching peak in the other list. The alignment between the peak lists PL_A and PL_B can be represented by a list of peak pairs where pairing implies peak-to-peak matching. For example,

$$PL_A \iff PL_B = [(a_1, b_1), (a_2, b_2), (a_3, -), (a_4, b_3) \ldots .] \quad (2)$$

where a_1 is matched with b_1 and a_2 is matched with b_2. In this example, the peak a_3 from PL_A does not have a matching peak in PL_B. The number of elements in the above list will depend on the optimal alignments, but cannot be less than the lesser of Nor M and cannot exceed $N+M$.

The developed global peak alignment algorithm is based on dynamic programming. Some features of NW sequence alignment approach [4] and of DTW [5] were adapted to improve the performance. The global peak alignment algorithm includes three main steps:

1. Construction of a cost matrix using peak similarity,
2. Calculation of the score and the traceback matrices,
3. Deduction of the alignment from the traceback matrix.

2.3.1 Construction of the Cost Matrix—The cost matrix (CM) size is $(N+1)\times(M+1)$, where N and M are the lengths of the first and second peak lists, respectively. The CM is obtained by applying a peak similarity function to PL_A and PL_B one peak at a time. In this way, the score between the peaks are obtained.

After measuring similarity of two peaks, a time penalty function can be applied:

$$TP(i, j) = e^{|i-j|T} \quad (3)$$

Here i and j are retention times of the two peaks. T is the retention time tolerance parameter, which determines the importance of retention time to the distance score. In other words, T determines the growth rate of the exponential function. T may be between 0 and 1. If $T=0$ there is no effect of the retention time on the distance. The time difference penalty will be more effective for higher values of T. For the examples shown here, T was 0.05.

2.3.2 The Score and Traceback Matrices—In this step, a two-dimensional score matrix (SM), with rows indexed with the peaks of one peak list and columns indexed with the peaks of the other peak list, is initialized. The cells of the score matrix are filled based on the peak similarity function and the gap penalty (GP). The SM maintains the current alignment score for the particular alignments. Since it is possible to insert a gap at the beginning of a sequence, the size of the scoring matrix is $(N+1)\times(M+1)$.

The peak similarity function gives the “cost” of matching any two peaks. In addition to this, the dynamic programming requires a GP to be defined. Since the peak similarity range is [−1,1], the meaningful GP value should be below 0. A low value of GP would favor the

insertion of gaps even when peaks are similar, a high value of GP would favor the alignment of peaks that are actually different. In this study, GP was −1.

The first step in the construction of SM is to establish the first column; the possibility of an initial gap in the alignment must be considered. The next step is to fill in each cell in a raster scan. There are three choices, and the selection is made by choosing the option that provides the maximum value,

$$SM_{m,n}=\begin{pmatrix} SM_{m-1,n}+GP \\ SM_{m,n-1}+GP \\ SM_{m-1,n-1}+CM_{m-1,n-1} \end{pmatrix} \quad (4)$$

$CM_{m-1,n-1}$ indicates the similarity of the peak a_{m-1} from PL_A and the peak b_{n-1} from PL_B, and GP is the gap penalty.

It is necessary to keep track of the choices made for each cell. Once the entire SM is filled, it will be necessary to use it to extract the optimal alignment. This process uses the traceback matrix (TM) to determine which cell was influential in determining the value of the subsequent cell. Traceback is the process of deduction of the best alignment from the TM. The traceback always begins with the last cell to be filled (i.e., the bottom right cell). One moves according to the traceback value written in the cell. There are three possible moves: diagonally (toward the top-left corner of the matrix), up, or left. As the SM is created, the TM is filled using the selected option in equation (4). If option 0 is selected, the corresponding cell in the TM has a value of 0; if option 1, 1; and if option 2, 2.

2.3.3 Extracting the Aligned Peak Lists—The final step is to extract the aligned sequences from the TM. The process starts at bottom-right corner of the TM and works toward the top-left corner. Thus the aligned peak lists are created from back to front. A value of 2 indicates that a letter from both peak lists is matched and the traceback moves up and to the left. Each time 1 is encountered, the peak from PL_A is aligned with a gap and the traceback moves to the left. Each time 0 is encountered, the peak from PL_B is aligned with a gap and the traceback moves up one location. The traceback is completed when the first (topleft) cell of the matrix (cell $SM_{0,0}$) is reached.

An example of a completed scoring matrix is shown in Figure 2. The aligned peaks are shown in gray. The quality of the alignment can be measured by using the cost matrix and gap penalty. In this case, the quality of this alignment is taken as the value of the bottom-right cell of score matrix.

In order to observe the alignment result, a figure is created to show the matched peaks. As can be seen clearly in Figure 3, the peaks are aligned correctly.

Some further modifications improved the performance of the peak alignment algorithm. Global constraint and minimum similarity scores were used to improve the accuracy of score matrix. The imposition of global constraints on the admissible warping paths is an often used dynamic programming variant. Such constraints speed up the alignment computations

and also prevent pathological alignments by globally controlling the route of the path. Two well-known global constraint regions are the Sakoa-Chiba band and the Itakura parallelogram [5,6]. In this application, The Sakoe-Chiba band, which runs along the main diagonal and has a fixed (horizontal and vertical) width, is used as a global constraint. Alignment of time points can be selected only from the defined region. Also, the distance function is not calculated for all data points but only in a defined region. This reduces execution time. Two peaks will not be aligned if they do not achieve the minimum acceptable similarity. In this application, the minimum score was 0.8. During the traceback, if the score of the matched peaks below the minimum score, these peak are not matched and a gap is inserted to the both peak lists.

2.4 Local Peak Alignment

A global alignment algorithm attempts to align two sequences from tip to tail. The traceback begins with attempts to align last two peaks in the strings and ends with attempts to align the first two peaks. In contrast, a local alignment algorithm – also known as the Smith-Waterman algorithm – attempts to find the best substring within the two strings that align. It accomplishes this through two modifications to the global alignment protocol. The first is to adjust the selection equations such that no negative numbers are accepted:

$$SM_{m,n}=\begin{cases} SM_{m-1,n}+GP \\ SM_{m,n-1}+GP \\ SM_{m-1,n-1}+CM_{m-1,n-1} \\ 0 \end{cases} \quad (5)$$

The other modification is done in the traceback algorithm. Instead of starting in the lower-right corner of the traceback matrix, the trace starts at the location with the largest value in SM. The trace continues until the fourth choice in above equation is reached. When the scoring matrix is created, the first line and column are not initialized.

Local alignment tools find an alignment describing the most similar region within the peak lists to be aligned. As can be seen in Figure 4, the shorter peak-based signal is aligned with the some part of the longer signal. Local alignment is useful analyses of time-series data to find the matched signals.

2.5 Signal Stretching and Compressing

After obtaining matching points, a cubic spline interpolation-based approach is used to compress or stretch the signals to obtain the same elution time scale. Cubic splines are readily implemented; they are constructed of piecewise third-order polynomials to produce a curve that appears to be perfectly smooth [11]. The objective is to fit a cubic spline to data points without oscillation, which is one of the common problems in curve fitting. In general, the cubic spline provides a good curve fit for arbitrary data points [11]. In case the number of points between the consecutive matching points differs (as in the case of gaps in the alignment), cubic spline interpolation is used to obtain same number of points.

3. RESULTS AND DISCUSSION

3.1 Mobility Shift Correction

In a SHAPE experiment, each reaction is analyzed using a DNA primer labeled with a different fluorophore. In any separation involving multiple dyes linked to DNA, as is done in dideoxy sequencing, the dye molecules alter the relative speed at which the attached DNA fragments travel. The overall effect is that DNA fragments of the same length labeled with different dyes elute at slightly different times. Correction for mobility shifts must be performed accurately to facilitate accurate location and linking of corresponding channels.

Alignment of traces obtained from a capillary gel separation must also correct for differences in the properties of dyes used to label each trace. In experiments using instruments from ABI, commonly used dyes are VIC, NED, FAM, and JOE. JOE- and FAM-labeled fragments have almost identical migration times, as do those labeled with VIC and NED, and the fragments labeled with JOE and FAM migrate faster than those labeled with the other two dyes. Our procedure for determining the necessary signal shift is explained in the following example using FAM for RX and VIC for the SL ladder. Since FAM fragments migrate faster than VIC-labeled fragments, the SL signal must be shifted left to match the left ends of the two time series. After shifting the SL point-by-point, dynamic programming is applied to obtain the match score. The score is calculated using the time penalty function (Eq. 3). After calculating the match scores for different shifts, the shift with the maximum score is selected and signals are aligned using the signal stretching/compression approach described in Section 2.3.4. An example of raw data and a resulting alignment are shown in Figure 5.

3.2 Capillary Alignment

A second challenge with electrophoresis data is to align data obtained from different capillaries. Since the parameters such as temperature and voltage may differ from run to run, the same sample may yield traces that vary in fragment retention times or peak intensities. Since nucleotide sequence traces (SL1 or SL2) are more similar than traces from samples treated with SHAPE reagents (RX and BG), they are used to align the data sets from different capillaries. The patterns of the sequence ladders with the same ddNTPs are similar, but elution times and intensities are generally not (Figure 6).

The sequence traces are aligned using the algorithm outlined above, combined with an additional optimization that controls the widths of the identified and aligned peaks. Since peaks are spaced at a fairly regular interval in SHAPE traces, the distance between the consecutive peak centers should be similar. Our enhanced algorithm produces excellent peak alignments of traces obtained from different capillaries. The aligned sequence traces are then used to align the other traces in the capillary.

3.3 Defining Regions of Interest Automatically

The other application of this algorithm is to define region of interest automatically. In fluorescent primer-based sequencing, sequence data is collected after the primer peak, which is the first wide peak in a trace (see Figure 7). In order to increase the accuracy of the next

steps, it is useful to remove the data that do not contain any sequence information. In order to find the same region of interest in traces from different capillaries, the peak alignment algorithm may be used.

Since each capillary has the same ddNTP-sequencing ladder, SL1 and SL2 signals are used to select the same region. After selecting the region of interest in the reference capillary manually, the selected data is used to find the same region in the other capillary. The local peak alignment algorithm is applied using the peaks in the reference SL1 data and sample SL1 data. The maximum value in the scoring matrix gives the best-matched point in both data. This maximum value is used to find the region of interest in the sample data. The first aligned peak positions are defined as a_0 and *b0*, and the last aligned peaks as a_1 and b_1 in reference SL1 and sample SL1, respectively. The region of interest in the sample data is between (b_0-a_0) and (b_1+(*N*- a_1)). Here *N* is the number of peaks in the sample data.

3.4 Comparison of peak similarly functions

Table 1 gives the performance of the peak similarity functions used in global peak alignment algorithm. Data corresponding to over 800 nucleotides from five different SHAPE experiments were evaluated. The peak alignment algorithm produced the best results when the derivative-mean similarity function was employed.

3.5 Comparison with DTW

As DTW is widely used, we compared the performance of DTW and our algorithm. DTW nonlinearly warps the two trajectories in such a way that similar events are aligned and a minimum distance between them is obtained. The DTW algorithm finds an optimal match between two sequences of feature vectors, which allows for stretched and compressed sections of the sequence. DTW works by warping the time axis iteratively until an optimal match between the two sequences is found. Accuracy and execution time were determined to compare the effectiveness of DTW and our peak alignment algorithm. The DTW approach was able to achieve only 87.48% accuracy, whereas our approach was 98.79% accurate. The DTW was much faster than the peak alignment algorithm, however. The DTW alignment of 5000 time-point, 393 peak traces was completed in 2.39 sec on 2 GHz, 2 GB RAM laptop computer; whereas the peak alignment algorithm required 4.28 sec on the same system. In practical terms, however, this difference in execution time is insignificant, and the accuracy of the new dynamic programming based algorithm is clearly superior. The other drawback of DTW approach is that the signals are aligned globally not locally.

3.6 Implementation

All preprocessing tools and signal alignment methods are implemented using version 2.6 of Python programming language [13]. *NumPy*[14] and *SciPy*[14] packages are used to manipulate array and data. *NumPy* is the fundamental package needed for scientific computing with Python. *Matplotlib*[15] is used to draw the figures. *Matplotlib* is a Python 2D plotting library, which produces quality figures in a variety of hardcopy formats and interactive environments across platforms. All packages are open-source software and can be downloaded at no cost from vendor websites.

4. CONCLUSION

In this study, a new global and local peak alignment procedure based on dynamic programming was developed to automatically align time-series data obtained in SHAPE, and other nucleic acid probing and sequencing experiments, resolved by capillary electrophoresis. This approach is significantly more accurate than the DTW method. The test results prove that the shift problems, which make SHAPE data analysis challenging, are solved correctly and efficiently using the algorithms developed in this work. The algorithms will be useful in solving time-shift problems in other types of datasets in which peak retention times and areas must be compared. These results have encouraged us to begin developing fully automated software with a graphical user interface based on the peak alignment algorithm described here.

Acknowledgments

This work was supported by a grant from the U.S. National Institutes of Health (AI068462). I especially thank Dr. Oleg Favorov and Dr. Kevin M. Weeks for their feedback and suggestions. I thank members of the Weeks laboratory for providing capillary electrophoresis data.

References

1. Kinser, J. Python for Bioinformatics. Sudbury, Massachusetts: Jones and Bartlett Publishers; 2008.
2. Last, M.; Kandel, A.; Bunke, H., editors. Data Mining in Time Series Databases. World Scientific; 2004.
3. Eddy SR. What is dynamic programming? Nature Biotech. 2004; 22:909–910.
4. Needleman, Saul B.; Wunsch, Christian D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of Molecular Biology. 1970; 48(3): 443–53. [PubMed: 5420325]
5. Müller M. Dynamic Time Warping. Information retrieval for music and motion. 2007:69–84.
6. Sakoe H, Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech and Signal Processing. 1978; 26(1):43–49.
7. Wilkinson KA, Merino EJ, Weeks KM. Selective 2′-hydroxyl acylation analyzed by primer extension (SHAPE): quantitative RNA structure analysis at single nucleotide resolution. Nature Protocols. 2006; 1:1610–1616. [PubMed: 17406453]
8. Vasa SM, Guex N, Wilkinson KA, Giddings MC, Weeks KM. ShapeFinder: A software system for high-throughput quantitative analysis of nucleic acid reactivity information resolved by capillary electrophoresis. RNA. 2008; 1:4, 1979–1990.
9. Robinson MD, De Souza DP, Keen WW, Saunders EC, McConville MJ, Speed TP, Likic VA. A dynamic programming approach for the alignment of signal peaks in multiple gas chromatography-mass spectrometry experiments. BMC Bioinformatics. 2007; 8(1):419. [PubMed: 17963529]
10. Staab J, O'Conell TM, Gomez SM. Enhancing metabolomic data analysis with Progressive Consensus Alignment of NMR Spectra (PCANS). BMC Bioinformatics. 2010; 11:123. [PubMed: 20214818]
11. Kiusalaas, J. Numerical Methods in Engineering with Python. 2. Cambridge University Press; 2010.
12. O'Haver, T. An Introduction to Signal Processing in Chemical Analysis. 2009. Available at: http://terpconnect.umd.edu/~toh/spectrum/
13. vanRossum, G.; Drake, FL. Python Reference Manual. Virginia, USA: 2001. [http://www.python.org/]
14. Jones, E.; Oliphant, T.; Peterson, P., et al. SciPy: Open source scientific tools for Python. 2001. [http://numpy.scipy.org/]

15. Hunter JD. Matplotlib: A 2D Graphics Environment. Computing in Science and Engineering. 2007; 9(3):90–95. [http://matplotlib.sourceforge.net/].

16. Keogh, Eamonn J.; Pazzani, Michael J. Derivative Dynamic Time Warping. First SIAM International Conference on Data Mining (SDM 2001); 2001.

17. Low JT, Weeks KM. SHAPE-directed RNA secondary structure prediction. Methods. 2010; 52:150–158. [PubMed: 20554050]

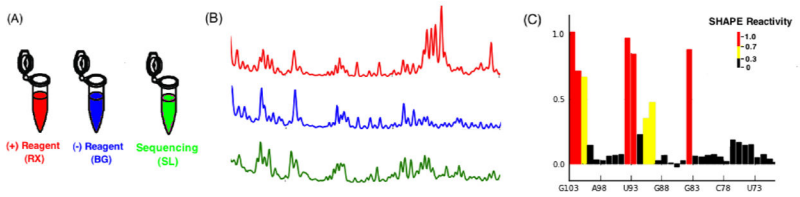

Figure 1.

Overview of SHAPE experiment. (A) Microtubes containing cDNA libraries prepared from RNA treated with SHAPE reagent or not and a sequencing reaction. cDNA libraries were prepared suing the same primer pairs of the same sequences but labeled with different fluorophores. (B) Raw SHAPE data obtained from CE. (C) Reactivity of each nucleotide after processing. Nucleotides are considered as unreactive, moderately, and highly reactive, and colored red, yellow, and black, respectively.

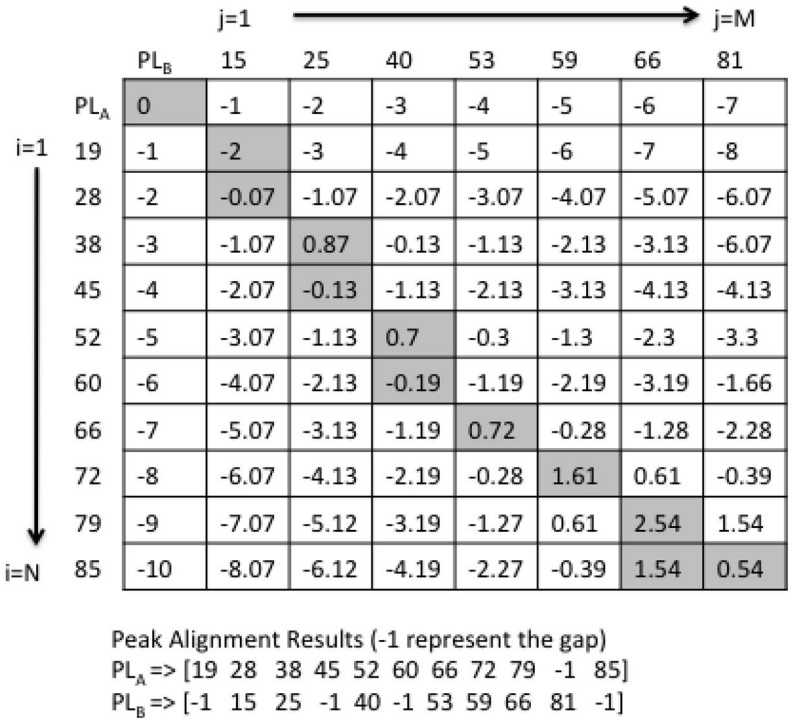

Figure 2.

An example of a completed scoring matrix. The first column and row of the matrix represent the peaks. Here peaks are labeled by their retention time. The result of the alignment is shown as a chromatogram in Figure 3.

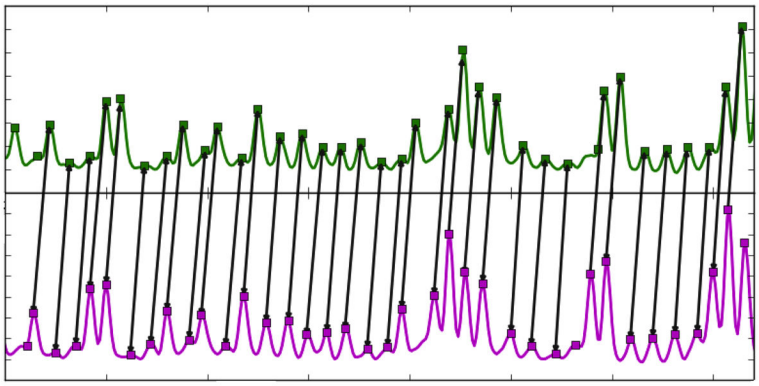

Figure 3.
Result of the peak alignment algorithm for representative RX and BG data. The arrows link the matched peaks in PL_A and in PL_B.

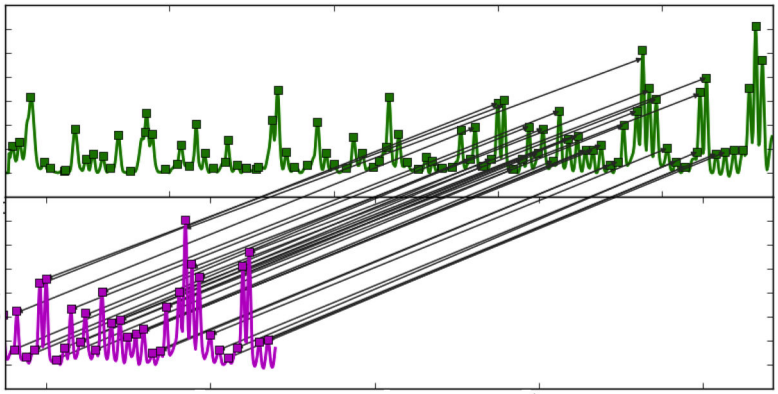

Figure 4.
Local alignment was used to match peaks in the shorter tract with those in the longer.

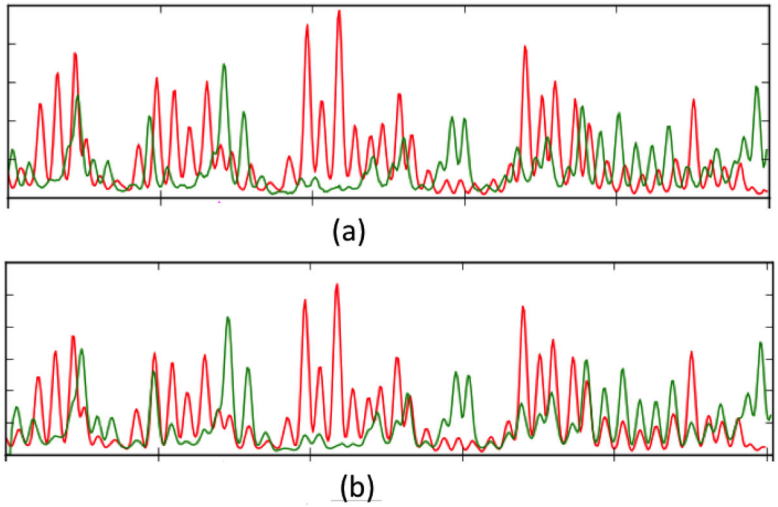

Figure 5.
Mobility Shift Correction. (a) The peaks in the same capillary traces (RX, SL) are not aligned. (b) Traces of signals in the same capillary after application of the mobility shift algorithm.

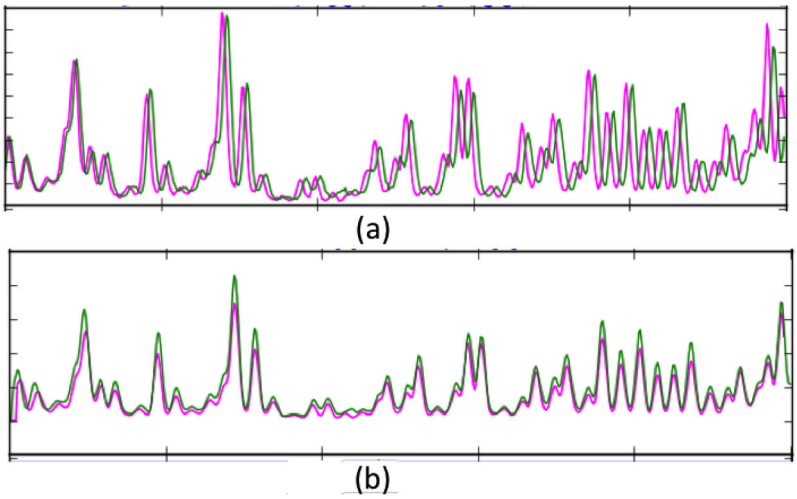

Figure 6.

Alignment of data obtained from different CE analyses.(a) The sequence traces obtained from different capillaries are not aligned. (b) Traces from different capillaries are aligned after application of the capillary alignment algorithm.

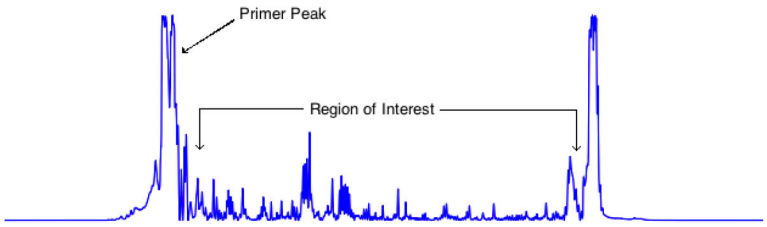

Figure 7.
Representation of a region of interest and identification of a primer peak in a raw trace.

Table 1

Comparison of peak similarly functions

Similarity Function	Accuracy
Amplitude [2]	96.78%
Mean [2]	97.99%
Correlation Coefficient [2]	92.87%
Cosines [9]	94.35%
Proposed Derivative-Mean	98.79 %