

Seismic data exhibit fundamentally non-stationary variations in attributes such as frequency and amplitude content. For example, Figure 1 shows a sample seismic image and its local frequency content (?). Data matching problems involve finding some transformation, or set of transformations, that can be applied to one data set to best match some attribute with another data set. In data matching applications, in particular, it is crucial to acknowledge the non-stationary variations present in seismic data. Therefore, any transformations we might apply in a data matching problem must be variant in all data dimensions to account for these non-stationary variations.

Local seismic attributes are useful in the analysis of these non-stationary variations (?). The calculation of these attributes uses iterative inversion with shaping regularization (?) to measure signal characteristics in local regions of data, rather than specifying windows or looking at instantaneous attributes. For the procedures described in this thesis, local attributes are more appropriate than instantaneous attributes. Take, for example, local frequency (Figure 1), as opposed to instantaneous frequency. Local frequency allows the comparison of frequency content in a local region of samples, as opposed to instantaneous frequency, which attempts to provide a point by point comparison of frequency values between images. Because the corresponding reflections between images might not be precisely aligned in time, using the local frequency attribute to balance frequency content between images would allow for more accurate frequency balancing than instantaneous frequency. Additionally, local frequency is a more geologically accurate attribute than instantaneous frequency because it honors time-frequency uncertainty and does not contain physically unrealistic negative or unrealistic high frequency values (?).

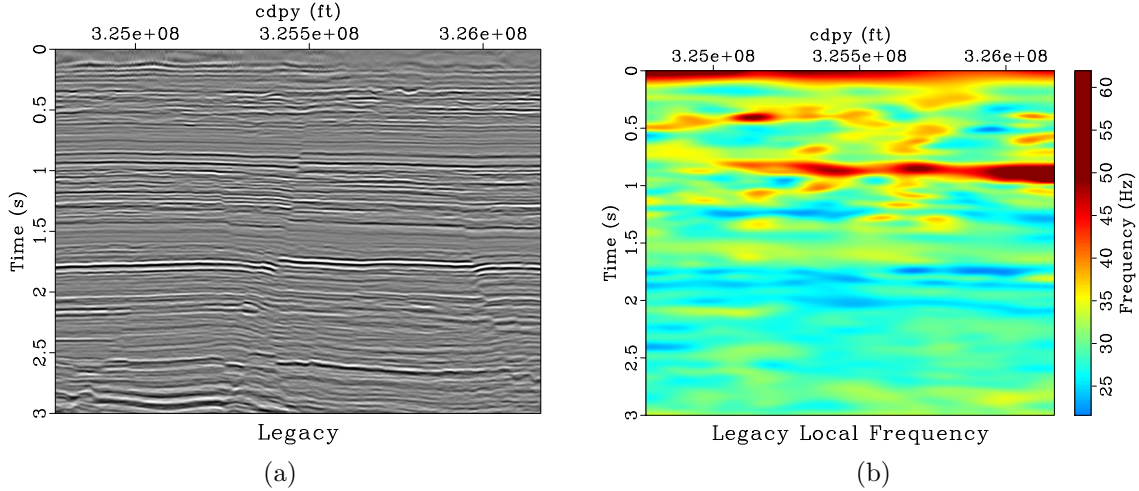


Figure 1: A seismic image (a) and its local frequency content (b). Because its local frequency varies both spatially and temporally, we can say its frequency content is smoothly *non-stationary*.

In this thesis, I propose solving data matching problems by balancing smoothly varying non-stationary attributes, such as local seismic attributes, between two data sets. This is done using three primary data matching operations—filtering, scaling,

and shifting—to effectively balance these attributes across seismic data sets. These operations are applied to different data sets to match them correctly for analysis or further processing.

To demonstrate the three matching operations in action, I apply them consecutively to an example trace shown in Figure 2a. The data used in this example comes from a line from the *P-cable* data sets. In this example, we want to find a transformation, or a set of transformations, to apply to the red trace to match with the black trace. I use the three data matching operators by first filtering (Figure 2b), then scaling (Figure 2c), and finally shifting (Figure 2d). These three operations are described in more detail in the following sections.

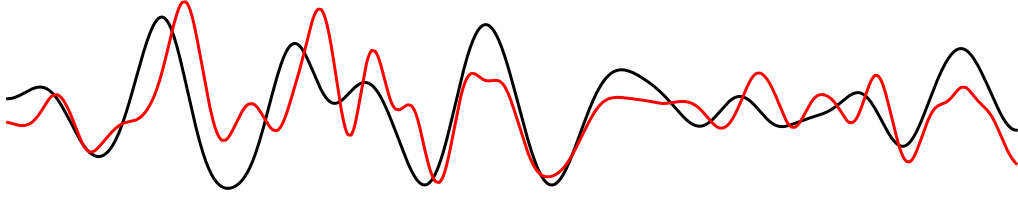
## FILTERING

Perhaps one of the most visibly obvious differences between data sets comes from their difference in frequency content. For example, Figure 2a shows two traces that need to be matched; the red trace has visibly higher frequencies than the black trace which makes it difficult to observe that they may both contain data from the same model. This makes the correlation of these two signals, both visually and computationally, difficult as they have information in different frequency bands. In this situation, we want to remove the higher frequency variations from the red trace to match the lower frequency content of the black trace.

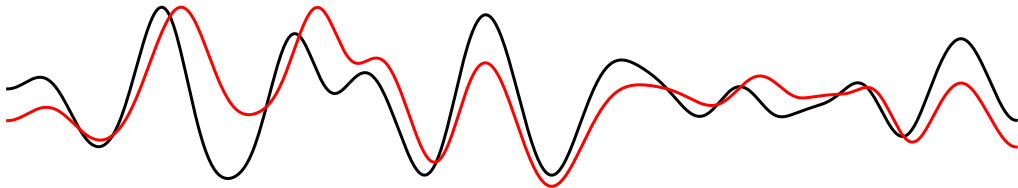
There are several ways to do this. A naïve first approach would be to apply a bandpass filter to the red trace such that the passband covers only frequencies that are present in the black trace. However, this stationary operation does not take into account the non-stationary frequency variations that may be naturally present in the data. In order to properly match the data in frequency content, we propose balancing local frequency content between data sets.

We can do this by applying a non-stationary triangle smoothing operator to the red trace to match the local frequency content of the black trace. We define the *radius* of this operator as the number of samples in a specified dimension that are averaged over in a triangle weight. We allow the radius to vary in all dimensions to account for the potential non-stationary frequency variations present in the data. This is a linear operation—if we represent the filtering operation by a matrix applied to the trace represented by a vector, the matrix would be diagonally banded, where the width of the band at a particular index is related to the size of the smoothing radius at that point.

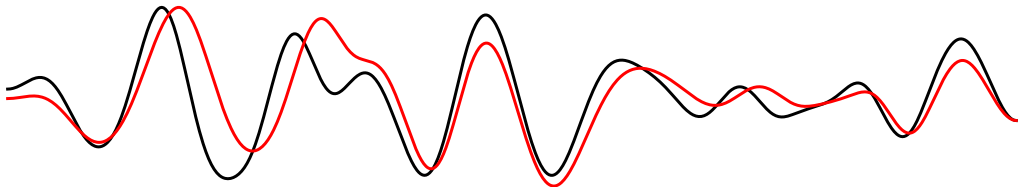
Chapter 3 is dedicated to the discussion of how to find this frequency balancing operator. Figure 2b shows the two traces after balancing the local frequency content of the red trace to match the black trace.



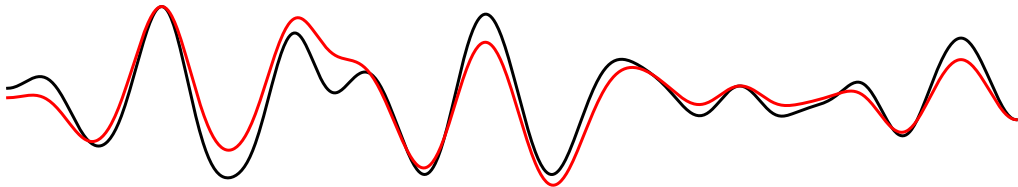
(a)



(b)



(c)



(d)

Figure 2: (a) Two traces that need to be matched—we will match the red trace to the black trace. (b) Red trace after filtering. (c) Red trace from (b) after scaling. (d) The final result: red trace from (c) after shifting, which now better matches with the black trace. The amount and order in which these operations is applied affects the final result (Figure 3).

## SCALING

The second primary data matching operation is scaling. Amplitudes of the data we are trying to match may not be initially equivalent—scaling attempts to balance the amplitudes between these data. Scaling is also a linear operation and can be thought of as a diagonal operation; if the scaling operator is represented by a matrix multiplication to the data vector, this matrix only contains terms along the diagonal.

In this thesis, the diagonal scaling operator is found by first calculating the amplitude envelope of the data that need to be matched. Then, the scaling weight is calculated by smoothly dividing the amplitude envelopes of the two traces to find the diagonal elements of the scaling operator.

## SHIFTING

If two events are slightly misaligned in time, finding the transformation that correctly aligns these events is referred to as *shifting*. Dynamic time warping is a common algorithm used for the alignment of events of two time series, both for geophysical and other applications, as diverse as speech recognition and finance (????).

In this thesis, I calculate the shifting operator by finding the time shifts that maximize local similarity between the two data sets. Local similarity extends the concept of local seismic attributes to global correlation coefficients and is effective for measuring the match of seismic events in a local region of samples (?). Unlike filtering and scaling, shifting is a non-linear transformation.

In certain situations, these data must be correctly aligned in space instead of time; for example, in a depth-migrated image. In these cases, *time* shift is a misnomer. However, because the data we are typically dealing with are time series, the shifting operation that we find and apply is considered a function of time.

The result after finding and applying the time shift to the red trace to match it to the black trace is in Figure 2d. This shifting operation is discussed in more detail in chapters 3 and 4.

## REPRESENTATION OF OPERATORS

Data matching problems are applicable for data sets that have the same physical model, yet have different characteristics that make their comparison difficult. For example, two seismic traces that were acquired over the same area but with different acquisition methods would benefit by applying data matching operations before their direct comparison. These three operations can be applied to one data set to match it with the other.

When applying one operation, we assume the other two operators have already been applied, when this may not be the case. This is why we balance *local* seismic

attributes instead of instantaneous ones—we want the match to be accurate in a region of data points rather than a point-by-point match. These operators must be smooth enough such that any misalignment of one attribute does not affect the result of balancing the others. For example, if a scaling operator is applied before the traces have been correctly aligned in time by shifting, too precise of an amplitude balancing operation could inadvertently balance amplitudes to incorrect events.

The order and amount of these operators affects the final result. After correcting for frequency variations, amplitude variations, and time shifts, additional corrections can be applied to further refine the match if necessary. In some cases, multiple rounds of applying these operators and in different orders may be beneficial for the best match. These three operators are noncommutative, so the order in which they are applied matters. Applying filtering before scaling produces a different result from applying scaling before filtering. An illustration of this property is shown in Figure 3.

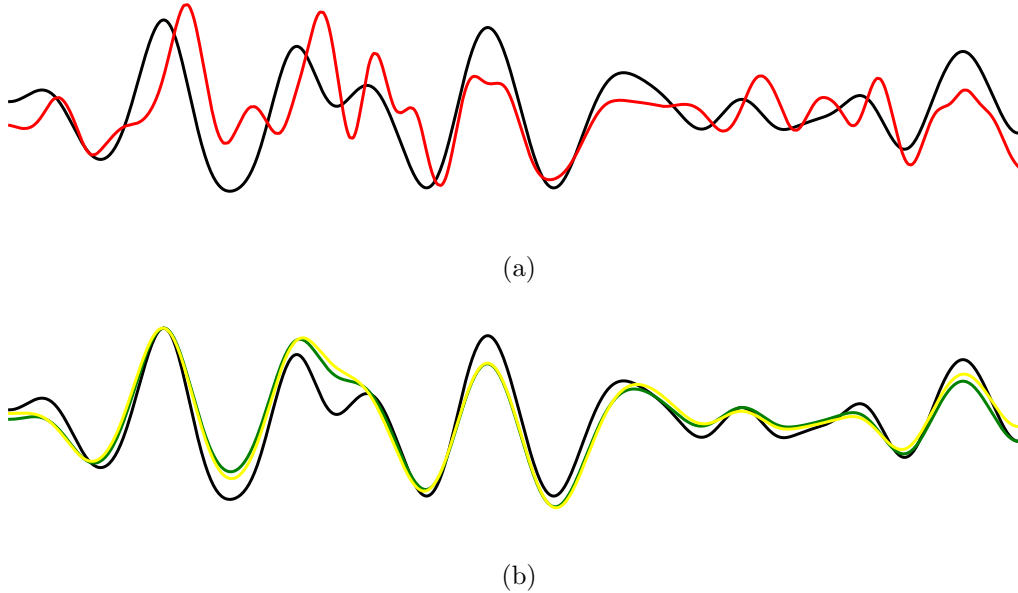


Figure 3: (a) A lower-resolution seismic trace (black) and a higher-resolution seismic trace (red) acquired over the same area. (b) Data matching operations are noncommutative—the order in which they are applied matters. When matching the red trace to the black trace, the green trace had first smoothing, then amplitude balancing, and finally shifting; the yellow trace had first amplitude balancing, then smoothing, and finally shifting. The operation order matters and affects the final result.

A summary of the three data matching operations is shown in Table 1. Both scaling and filtering are linear operations, whereas shifting is generally a non-linear operation. Additionally, as scaling is to weighting in the time domain, filtering is to weighting in the Fourier domain. The next chapter discusses a few methods and examples of filtering, or frequency balancing, between two data sets.

<b>Operator</b>	<b>Representation</b>
Shifting	$\mathbf{d}_2(x) = \mathbf{d}_1(x + \mathbf{s}(x))$
Scaling	$\mathbf{d}_2(x) = \mathbf{w}(x)\mathbf{d}_1(x)$
Filtering	$\mathbf{D}_2(k) = \mathbf{W}(k)\mathbf{D}_1(k)$

Table 1: Mathematical representation of the three data matching operators.