

Solving for seismic mistie corrections in time, amplitude, and phase

Bill Harlan

December, 2016

Abstract

Seismic images of different vintages often show inconsistent waveforms that complicate mapping and interpretation. A popular workflow cross-correlates overlapping traces wherever any two datasets intersect. From correlations we can find the time shifts, amplitude scaling, and phase rotations that tie datasets at these intersections.

For global consistency, we need to adjust each dataset independently by absolute corrections that best reproduce the relative corrections. Independent time shifts and amplitude scalars are easily estimated with a damped least-squares optimization, which we review.

Phase rotations cannot be optimized in the same fashion without ambiguity in cyclical phase angles. Instead, we begin with phase-shifted impulses as simplified representations of cross-correlations. We solve for simplified transfer functions, which then unambiguously determine the best phase correction for each dataset. A revised objective-function is optimized with a Gauss-Newton optimization.

INTRODUCTION

This paper describes a robust generalized inversion for time, amplitude, and phase corrections of overlapping and intersecting seismic datasets.

In a well-developed area, we may have overlapping 3D seismic surveys, 2D seismic lines, and even 4D repeated surveys that attempt to illuminate the same seismic reflections. Different acquisition parameters and different vintages result in very different wavelet characteristics. Such differences complicate the correlation of seismic sections with sonic logs. Interpreters find it more difficult to track formation boundaries from dataset to dataset.

We will consider approaches that begin by cross-correlating selected traces from different surveys near intersection points. Early work by Harper, 1991 and Henry and Mellman, 1988 used these cross-correlations to estimate convolutional wavelets, or transfer functions, that best adjusted each dataset. Later

simplifications were independently implemented by Mackidd, 1998 at EnCana and by Bishop and Nunns, 1994. From each cross-correlation they extract relative time shifts, phase rotations, and amplitude scale factors. The time shift is generally located at the peak of the analytic envelope of the cross-correlation. Comparing the strength of the cross-correlation to the strength of a dataset determines a relative amplitude scaling. The phase rotation is the constant phase bias remaining in the frequency domain after removing the time shift. (Equivalently, the best time lag and phase rotation correspond to the slope and intercept of a line fitting phase in the frequency domain.)

Each of the three corrections are inverted separately to find independent adjustments for each dataset. Interpreters find such corrections easy to view for quality control, with minimal risk of damage to spectral information. If the inversion is robust, then repeating the procedure will not result in any further improvements.

Mackidd, 1998 preferred to construct explicit systems of equations with specific constraints to remove non-uniqueness. Bishop and Nunns, 1994 used a recursive algorithm (essentially Gauss-Seidel) that converged to the same solution for time shifts and amplitudes.

Both inverted phase rotations with modified versions of their algorithms for time shifts. Phases, however, require “unwrapping” that repeatedly adjusts by full cycles of 360 degrees to minimize errors. Such approaches cannot guarantee that they avoid local minima or that they find a globally optimum solution.

We define all solutions with objection functions that best minimize errors, with damping to handle non-uniqueness. For optimization, we choose a Gauss-Newton optimization as described in Luenberger, 1973, and as implemented by Harlan, 2004 for related problems in Harlan, 2014 and Harlan, 1989. In the case of time shifts and amplitudes, we obtain the same least-squares solution as solved by Bishop and Nunns, 1994.

We prefer, however, to avoid fitting spectral phases directly. Instead we return to the time domain and fit the corresponding phase-shifted impulses. Instead of adding or subtracting phases, we convolve or correlate these simplified transfer functions. In the time domain, we avoid any need to adjust phase by an ambiguous number of cycles. This optimization is a specialization of the cross-balancing algorithm in Harlan, 2015, which is very similar to a surface-consistent deconvolution as described in Levin, 1989.

FITTING TIME SHIFTS

We begin by reviewing the numerically simplest optimization of relative time shifts.

Let us index independent seismic lines or datasets by an integer index i or j . To align all the datasets, we assume that each line i needs to be shifted by a time t_i , in milliseconds (or meters or feet if in depth). A trace amplitude

previously appearing at zero time will appear at the sample with this time, after correction. A negative time correction will reduce the times at which a particular amplitude appears.

If a pair of lines cross, then we could make a relative correction t_{ij} to line i to make it better tie a line j . If shifts are globally consistent, then relative corrections can be computed from absolute corrections:

$$t_{ij} = t_j - t_i. \quad (1)$$

We can directly measure or estimate these relative corrections, but not the absolute corrections. Let us use an integer index m to identify each intersection at which we measure a relative shift from the data. Each m will index a triplet of indices $\{m, i, j\}$. At each of these intersections m , we will measure a relative time shift t_{ij}^m that adjusts line i to match line j better. Selected traces from each line are cross-correlated so that this lag can be measured from the analytic peak.

We will assume that errors in measured t_{ij}^m follow an uncorrelated Gaussian distribution with zero mean. This assumption would imply a maximum likelihood solution for t_i that minimizes the least-squares error between the measured and computed corrections:

$$\min_{\{t_i\}} \sum_{\{m,i,j\}} (t_{ij}^m - t_{ij})^2 \quad (2)$$

$$\text{or } \min_{\{t_i\}} \sum_{\{m,i,j\}} (t_{ij}^m - t_j + t_i)^2. \quad (3)$$

Unfortunately, this function has a flat bottom with more than one minimum. We can add a constant to all absolute shifts and still compute exactly the same relative shifts.

Worse, there may be some linear components that are poorly conditioned, where large changes in some combination of $\{t_i\}$ result in very small changes to the total error. A larger area of the objective function may have almost the same minimum value within the precision of our measurements.

To handle such poorly conditioned solutions, we add a damping term on the magnitude of absolute corrections. Damping is equivalent to assuming that inverted corrections also have a Gaussian distribution, then maximizing the a priori probability (known as a MAP estimate).

$$\min_{\{t_i\}} \sum_{\{m,i,j\}} (t_{ij}^m - t_j + t_i)^2 + \epsilon \sum_i t_i^2. \quad (4)$$

We use a damping factor $\epsilon = 0.0001 \cdot M/N$, where M is the number of intersections indexed by m , and N is the number of independent datasets indexed by i and j . Additionally we add a factor of 0.0001, which is equivalent to assuming measurement errors have about one hundredth the magnitude of typical

shifts. This factor can be reduced even more without significantly altering the result. Only a little damping is required to eliminate non-uniqueness and poorly conditioned components.

This objective function (4) is a purely quadratic function of the unknown $\{t_i\}$ values. We use a low-memory iterative conjugate-gradient algorithm as in Luenberger, 1973 and Harlan, 2004.

Damping ensures that the average absolute correction will be zero. Instead, we can choose certain lines as reference lines and force their corrections to be any predetermined value, such as zero. To do so, we simply omit these corrections from the optimization. (Programatically, we can remove elements from our solution vector, or we can apply conditioning that zeros out these elements.)

FITTING AMPLITUDE SHIFTS

With a few tweaks, we can fit amplitude changes with the same algorithm we used for time shifts.

We assume each line i needs to be scaled by an absolute amplitude factor of a_i to best match all other lines. If a pair of lines cross, then we could make a relative amplitude correction a_{ij} to line i to make it better tie a line j . If amplitude corrections are globally consistent, then relative corrections can be computed from absolute corrections:

$$a_{ij} = a_j/a_i. \quad (5)$$

At each intersection m , we also measure a relative correction a_{ij}^m that makes line i more consistent with the amplitudes in line j . Once again, we want to fit measured corrections a_{ij}^m by estimating the best absolute corrections $\{a_i\}$.

In most cases, lines will have already been normalized to some degree, perhaps to make their root-mean-square (RMS) amplitudes all the same. Normalization does not result in the best overall ties because one side of a dataset may be stronger in amplitude than in another. But we do expect remaining amplitude scale factors to be reasonably distributed around a value of 1, with no large outliers.

To turn this problem into the same linearized problem we just solved, we will assume that the distributions of scale factors are all log-normal. That is, the logarithm of a_i follows a Gaussian distribution. Multiplying or dividing log-normal variables results in more log-normal variables, including our computed a_{ij} . The resulting MAP objective function is

$$\min_{\{a_i\}} \sum_{\{m,i,j\}} [\log(a_{ij}^m) - \log(a_j/a_i)]^2 + \epsilon \sum_i \log(a_i)^2. \quad (6)$$

This function is not quadratic in the unknown $\{a_i\}$ and is more difficult to solve. Instead, if we define the following,

$$b_i \equiv \log(a_i) \text{ and } b_{ij}^m \equiv \log(a_{ij}^m), \quad (7)$$

then the objective function becomes

$$\min_{\{b_i\}} \sum_{\{m,i,j\}} (b_{ij}^m - b_j + b_i)^2 + \epsilon \sum_i b_i^2. \quad (8)$$

This is exactly the same form as our previous quadratic objective function (4) for time shifts. Here we optimize for $\{b_i\}$, and as a final step recompute $\{a_i\}$.

FITTING PHASE ROTATIONS

Superficially, the problem of fitting phase rotations is very similar to those of previous sections. Relative phase rotations are measured at line intersections, and are modeled as differences of absolute phase rotations for each line.

Some authors (Bishop and Nunns, 1994, Mackidd, 1998) solve for absolute phase rotations with least-squares approaches very similar to those just described for time and amplitudes. Unfortunately phase rotations are not uniquely invertible. A phase rotation of ϕ has the same effect on a trace as $\phi + 2\pi$. As a result, phases are usually limited to a branch such as $-\pi$ (exclusive) to π (inclusive), or equivalently -180° (exclusive) to 180° (inclusive).

When minimizing least-squares errors in phases, a small perturbation can easily cross over a branch cut. Such approaches must update both measured and estimated phases repeatedly during optimization to allow values outside the original branch range. Such “phase unwrapping” does not converge well for larger numbers of intersections and datasets; there is no guarantee of finding a global minimum.

Phase rotations

We should first clarify the definition of a phase rotation (also known as a constant phase shift). We will represent all necessary operations in the time domain without Fourier transforms.

We will write a seismic trace $f(t)$ as a continuous function of a time t . (Discretization is equivalent to assuming a bandlimited “sinc” function as a basis for the continuous function.)

We first compute an analytic trace $\tilde{f}(t)$ of complex amplitudes by convolv-

ing our real trace $f(t)$ with an analytic function $g(t)$:

$$\tilde{f}(t) \equiv g(t) * f(t) \equiv \int g(t - \tau) f(\tau) d\tau, \quad (9)$$

$$\text{where } g(t) \equiv \delta(t) + ih(t) \quad (10)$$

$$\text{and } h(t) \equiv \frac{1}{\pi t}. \quad (11)$$

The analytic function $g(t)$ is the sum of a Dirac delta function $\delta(t)$ and an imaginary Hibert transform $h(t)$, which are well explained in Bracewell, 1978.

The real part of an analytic trace restores the original trace.

$$f(t) \equiv \text{Re}[\tilde{f}(t)]. \quad (12)$$

The new imaginary component is always orthogonal to the original function, mapping a cosine to a sine, and a sine to a negative cosine, like a derivative, but without affecting the amplitude spectrum.

To apply a constant phase rotation of angle ϕ to a trace $f(t)$, we multiply the complex analytic trace $\tilde{f}(t)$ by a complex number of unit magnitude $\exp(i\phi)$, then keep just the real part.

$$f(\phi, t) \equiv \text{Re}[e^{i\phi} \tilde{f}(t)] \quad (13)$$

$$= \text{Re}\{e^{i\phi} [g(t) * f(t)]\} \quad (14)$$

$$= r(\phi, t) * f(t) \quad (15)$$

$$\text{where } r(\phi, t) \equiv \cos(\phi)\delta(t) - \sin(\phi)h(t). \quad (16)$$

A phase rotation is equivalent to convolving with the phase rotation function $r(\phi, t)$, which is purely real. We can see a few simple cases easily. A zero-degree rotation has no effect. A 180° rotation reverses the sign. A 90° rotation is the same as applying the Hilbert transform with the opposite sign, so that a sine becomes a cosine, and a cosine becomes a negative sine.

Successive convolutional phase rotations can be combined merely by adding their corresponding phase rotations:

$$r(\phi_1, t) * r(\phi_2, t) = r(\phi_1 + \phi_2, t). \quad (17)$$

Even better, a phase rotation is a unitary operation, so we can perform the inverse simply by reversing the sign of the phase rotation.

$$r(-\phi, t) * r(\phi, t) = \delta(t). \quad (18)$$

Because of various symmetries, we also see that

$$r(-\phi, t) = r(\phi, -t). \quad (19)$$

And so correlation with a phase rotation will invert a convolutional phase rotation.

$$r(\phi, t) \star r(\phi, t) \equiv \int r(\phi, \tau) r(\phi, t + \tau) d\tau \quad (20)$$

$$= r(\phi, -t) * r(\phi, t) = \delta(t). \quad (21)$$

If we have an instance of the phase rotation function, we can reestimate its angle:

$$\text{Define } x \equiv r(\phi, t = 0) \text{ and} \quad (22)$$

$$y \equiv - \int r(\phi, t) h(t) dt; \quad (23)$$

$$\text{then } y = x \tan(\phi) \text{ and} \quad (24)$$

$$\phi = \text{atan2}(y, x) \equiv \text{Arg}(x + iy). \quad (25)$$

Here we use a two-argument version of an arc-tangent to get the angle in the correct quadrant.

Optimizing phase rotation functions

Rather than optimize phase rotations directly, we can equivalently construct and work with their corresponding phase rotation functions. Phase rotations that differ by exactly 360° have exactly the same representation as functions, so we avoid any non-unique phase unwrapping problems.

We assume each line i needs to be adjusted by an absolute phase rotation of ϕ_i to best match all other lines. If a pair of lines cross, then we could make a relative phase correction ϕ_{ij} to line i to make it better tie a line j . If rotations are globally consistent, then relative corrections can be computed from absolute corrections:

$$\phi_{ij} = \phi_j - \phi_i. \quad (26)$$

At each intersection m , we also measure a relative rotation ϕ_{ij}^m that makes line i more consistent with line j . Once again, we want to fit measured corrections ϕ_{ij}^m by estimating the best absolute corrections $\{\phi_i\}$.

Instead of fitting these phases directly, we will substitute phase rotation functions and model them with correlations rather than differences.

For each line i we want to estimate an absolute phase rotation function $r(\phi_i, t)$. We can then compute a relative phase correction function that best corrects line i to line j by correlating their respective absolute functions.

$$\begin{aligned} r(\phi_{ij}, t) &= r(\phi_i, t) \star r(\phi_j, t) \\ &= r(\phi_i, -t) * r(\phi_j, t) \\ &= r(-\phi_i, t) * r(\phi_j, t) \\ &= r(\phi_j - \phi_i, t). \end{aligned} \quad (27)$$

We can fit the measured rotations at intersections with the following damped least-squares objective function.

$$\min_{\{\phi_i\}} \sum_{\{m,i,j\}} \int [r(\phi_{ij}^m, t) - r(\phi_i, t) \star r(\phi_j, t)]^2 dt + \epsilon \sum_i \int r(\phi_i, t)^2 dt. \quad (28)$$

In fact, this objective function distributes errors in a very plausible fashion. These phase rotations, and their correlations, are equivalent to the transfer functions that would result from an idealized dataset with no inconsistencies other than phase rotations.

This optimization is a specialization of the cross-balancing algorithm by Harlan, 2015, which is very similar to a surface-consistent deconvolution as described by Levin, 1989.

We can optimize for phases in two steps. First we estimate the absolute rotation functions, then we estimate phase from those functions as in equation (25).

$$\min_{\{r_i(t)\}} \sum_{\{m,i,j\}} \int [r(\phi_{ij}^m, t) - r_i(t) \star r_j(t)]^2 dt + \epsilon \sum_i \int r_i(t) dt, \quad (29)$$

$$\text{where } r_i(t) \equiv r(\phi_i, t). \quad (30)$$

Notice that this objective function, unlike our previous versions, is quartic rather than quadratic in the unknown $\{r_i(t)\}$. Nevertheless, the objective function is well behaved and convex with a single global minimum.

We use a low-memory Gauss-Newton optimization as described in Luenberger, 1973, and as implemented by Harlan, 2004. We iteratively approximate the objective function by linearizing the correlation in perturbations of the unknown rotation functions, which approximates the objective function as a quadratic:

$$\begin{aligned} \min_{\{\Delta r_i(t)\}} \sum_{\{m,i,j\}} \int [r(\phi_{ij}^m, t) - r_i(t) \star r_j(t) - \Delta r_i(t) \star r_j(t) - r_i(t) \star \Delta r_j(t)]^2 dt \\ + \epsilon \sum_i \int \Delta r_i(t) dt. \end{aligned} \quad (31)$$

We initialize all unknown $\{r_i(t)\}$ with delta functions and solve for perturbations $\{\Delta r_i(t)\}$ with a standard quadratic conjugate gradient algorithm. Then we use a line-search optimization to find the best scale factor for all perturbations before updating the reference $\{r_i(t)\}$ functions. Then we relinearize and solve for perturbations again. Because the objective function is convex, with a single global minimum, convergence is guaranteed.

From these $\{r_i(t)\}$ functions, we use equation (25) to estimate the best phase rotations $\{\phi_i\}$ for each dataset. We also can use the simple differences

in equation (26) to recompute the expected phase shifts ϕ_{ij}^m at each intersection m , after adjusting to the standard branch cut. Any remaining residual errors are removed by the quadratic least-squares of previous sections.

We find 23 samples more than adequate for correlations and transfer functions, reconstructing phases with less than one degree of error. Because time lags are handled separately, the envelopes of these functions remain centered, reducing the effect of truncations.

We do find that estimated phases can drift somewhat when tens of thousands of lines are inverted without any constraints. That is, the reconstructed differences at intersections are well within measurement error, but the absolute phase shifts can vary more substantially at distant degrees of connectedness. If adjusted lines are no more than a hundred intersections from a constrained line, then inverted phases avoid such drift.

CONCLUSIONS

We began with the premise that time shifts, amplitude scaling, and phase rotations are an adequate correction to tie overlapping datasets. These corrections are orthogonal and can be optimized independently.

We first reviewed a least-squares approach to time and amplitude corrections, with few differences from previous work. We do, however, prefer damping to explicit rank reduction. A low-memory iterative optimization scales well and more conveniently incorporates constraints.

We believe phase corrections are best solved as phase-rotated delta functions representing simplified transfer functions. Such an approach avoids ad-hoc phase unwrapping and guarantees convergence to a global minimum.

Data may not always agree with these simplifying assumptions. Phase distortions can change more arbitrarily than just linear time shifts and constant phase rotations. Amplitudes may also scale differently with frequency. Additional preprocessing with deconvolution can help, but does not eliminate all such issues.

It is also possible to fit cross-correlations directly as in Harlan, 2015. The three independent corrections can be extracted from full transfer functions estimated for each dataset. Users may find it more difficult to audit and edit such corrections at intersections.

Ultimately, the popularity of the three independent corrections is the best testimonial for their effectiveness.

REFERENCES

- Bishop, T. N., and A. G. Nunns, 1994, Correcting amplitude, time, and phase mis-ties in seismic data: *Geophysics*, **59**, 946–953.
- Bracewell, R., 1978, *The fourier transform and its applications*, second ed.: McGraw-Hill Kogakusha, Ltd.
- Harlan, W. S., 1989, Simultaneous velocity filtering of hyperbolic reflections and balancing of offset-dependent wavelets: *Geophysics*, **54**, 1455–1465.
- , 2004, An implementation of generalized inversion. (https://github.com/dhale/jtk/blob/master/docs/opt_package/opt.pdf).
- , 2014, Wavelet balancing for residual moveouts: *Geophysics*, **79**, V217–V225.
- , 2015, Cross-balancing intersecting surveys. (<http://billharlan.com/papers/crossbal.html>).
- Harper, M. D., 1991, Seismic mis-tie resolution technique: *Geophysics*, **56**, 1824–1830.
- Henry, M., and G. R. Mellman, 1988, Linearized simultaneous inversion for source wavelet equalization and mis-tie adjustment: 58th Annual International Meeting, SEG, Expanded Abstracts, Soc. of Expl. Geophys., 953–955.
- Levin, S., 1989, Surface-consistent deconvolution: *Geophysics*, **54**, 1123–1133.
- Luenberger, D. G., 1973, *Introduction to linear and nonlinear programming*: Addison Wesley.
- Mackidd, D., 1998, Grid balance: How it works. (Unpublished manuscript.).