Getting it right: source-receiver offsets in the radial trace transform

David C. Henley

ABSTRACT

Our original radial trace (RT) transform was intended to be applied only to 2D data, for which the distribution of source-receiver offset values for traces within a typical source/receiver ensemble is nearly always linear. Hence, we constructed the inverse RT transform using a simplification in which the source-receiver offset values used by the inverse transform to populate the trace headers of the output X/T source gather are interpolated from the values of XMIN and XMAX, stored in the radial trace headers by the forward transform. Eventually, however, we extended the use of the RT transform to 3D receiver line gathers. Here, the linear offset approximation is no longer accurate, since offset distributions become hyperbolic as the source position is displaced laterally from a receiver line.

We recognized this problem and have recently updated our RT transform module for SeisSpace so that it now offers the option of restoring original offset values to the headers of the inverse transform, in addition to the original offset interpolation options implemented as diagnostics. We document the changes here, describe the steps necessary in order to use the updated module, and show examples of the new module applied to 2D data with source positions offset from the line (3D applications have not been tested yet, but are expected to work properly, as long as the 3D equivalent of the described database operations are successfully performed). We also review the original 2D offset interpolation options available in the current version of the inverse transform.

INTRODUCTION

The radial trace (RT) transform was first introduced by Jon Claerbout, primarily as a processing shortcut for a particular variety of wave equation migration (Claerbout, 1975, 1983, Ottolini, 1979). Unlike the better-known F-K transform or the Tau-P transform, the RT transform is not an integral transform, but simply a point-to-point remapping of samples from the familiar X/T domain to the slowness-transit time domain. Because the remapping from a uniformly sampled X/T trace ensemble to the RT domain results in a non-uniform array of samples, the mapped points must be interpolated to uniform sampling in the new domain. The RT transform is actually similar to the “NMO transform”, in which an X/T trace ensemble is mapped to an X/T_0 gather. In the NMO transform, a so-called ‘moveout velocity’ function guides the mapping via the normal moveout, or Dix formula, while in the RT case, a radial fan function of linear trajectories with common origin and incremental slowness values is used to guide the mapping of samples from the X/T domain to the RT domain. In the NMO case, X/T domain traces are interpolated *parallel* to the T axis to provide the uniform sampling (but with non-uniform stretch) required to simulate moving a trace recorded at offset distance X to a trace recorded at offset distance X_0; whereas in the RT case, X/T domain traces are interpolated *perpendicular* to the T axis (parallel to the X axis) to provide uniform sampling (with uniform stretch) required to populate the traces of the RT transform.

Either transform, as a 1D interpolation operation, can be inverted exactly, by mapping and interpolation back into the original domain.

The header information problem

2D representations of seismic data are two-dimensional matrices, with values organized in rows and columns. Seismic data are usually acquired in the X/T domain, with X, the column identifier, representing the distance of a particular receiver or geophone from the source position, and T, the row identifier, representing the elapsed time of the recording. Hence, it is natural that information about the location of the receiver, and other acquisition parameters, be associated with each column (or trace) of samples—hence 'trace headers'. If the number of rows and columns were the same for each 2D representation of the data, we could simply keep the trace headers intact with the columns and add to each trace header vector whatever parameters are necessary to allow transformation from that domain to another (or back). In the case of the NMO transform, this is exactly what happens—the number of traces (columns) does not change, and the moveout velocities used to remove NMO are stored in the trace headers for each column (trace) and can be used to restore (de-stretch) the NMO. For most other transforms, however, including the RT transform, the number of columns *does* change; hence retaining original trace header values for use in transform inversion becomes a problem.

One possible solution for the header problem is to append the original data to the output traces of a given 2D transform, and to flag these traces as auxiliary traces, not to be processed further in the flow unless the auxiliary flags are reset. This solution works as long as the transform output matrices remain intact (retain their rows and columns) through subsequent processing steps. For many interesting processing flows, however, the traces in a set of transform output matrices are sorted across ensemble boundaries to form a new data set with a different number of matrices of different dimension (Henley, 2008, 2012a, 2012b, 2014, Cova et al, 2013, 2014, 2015a, 2015b). There is no clear way to handle the 'auxiliary traces' with appended header information in this case, since each set of auxiliary traces is associated with a particular input ensemble number. Examples of this type of sorting operation are the creation of 'constant ray parameter' ensembles from the Tau-P transforms of a series of source ensembles, or the similar 'constant angle' ensembles from the RT transforms of a series of source ensembles.

We encountered this dilemma early in the development of our version of the RT transform and decided to implement an approximation which would allow us to restore a few critical trace headers to a source gather upon being inverted from the transform. Instead of trying to preserve a complete set of trace headers, we identified a few key parameters that would suffice to ensure the inversion of the RT transform and allow further processing. Under the assumption that any ensemble to be transformed would be a 2D trace gather (typically source or receiver gather) with sources collinear with the receiver spread, we determined that the maximum and minimum signed source-receiver offset value, the CMP number of the first trace, the CMP increment, and the number of traces in the original ensemble were required for RT transform inversion (Henley, 1999c). We made the additional assumption that a typical 2D trace ensemble would have uniformly spaced receiver stations, which enabled us to approximately recover the signed source-receiver offset values by knowing only the minimum offset, maximum offset, and the number of traces in the original ensemble. The uniform spacing assumption allowed us to recover a full set of offset values by linear interpolation of the minimum and maximum values, as well as the appropriate CMP bin numbers. In order to have these parameters available for RT transform inversion, the RT forward transform stores each of the four (minimum offset, maximum offset, first CMP, and CMP increment) in particular trace headers that are not used in the RT domain (Henley, 1999c, Henley and Wong, 2013). The parameter values are duplicated in the headers of *all* radial traces so that they can be retrieved from any individual radial trace.

Further benefits of the approximation

As we demonstrate later, the linear offset interpolation approximation is quite accurate for inverting most 2D RT transforms, even those with somewhat irregular trace spacing, or those obtained from source gathers whose source point is only slightly displaced from the receiver spread; this accuracy justifies our choice in most cases.

In addition to simplifying the trace header retention problem, this strategy motivated the implementation of several features of the RT transform module that can be useful for diagnostics or various unconventional processing applications (Henley and Wong, 2013). Among them are:

- From an input gather with slightly irregular trace spacing (+/- 1/4 station interval), a forward RT transform followed by an inverse will yield an output gather with regular spacing.
- By overriding the intrinsic output trace number, an input gather can be interpolated or decimated during the forward/inverse RT transform to provide a gather with the same offset range, but different trace spacing.
- In addition to the linear interpolation option for offset values, the offsets may also be quadratically interpolated to provide an X^2/T gather after forward/inverse RT transform.
- A further diagnostic output option is the X^2/T^2 panel, on which all events with NMO become approximately linear, regardless of moveout velocity. This output may be reverted to X/T by a further application of the RT inverse.

Regardless of the usefulness of these features of the approximate inverse RT, there are instances where the linear offset approximation is not accurate enough. In particular, anytime the source position is not collinear with the line of receivers, due to acquisition geometry irregularity, or because the receiver line is part of a 3D survey (Henley, 2015, 2016a, 2016b, 2017a, 2017b), the offset distribution is hyperbolic and is not well-approximated by linear interpolation. In a later section, we describe our modification to the RT transform module which allows it to retrieve and use the original offset values from the input X-T domain data, a procedure which requires additional processing steps prior to applying the forward RT transform to the original input data.

Why isn’t RT filtering affected?

Our original application for the RT transform was removing coherent, source-generated noise from seismic trace gathers. In this application, the RT transform is stored in an internal array in the filter module, and is never output as traces. All parameters needed to invert the transform exactly are always retained in arrays inside the operation module. The module we developed for filtering in the RT domain, first for ProMAX, then for SeisSpace, is fundamentally different from the stand-alone RT transform module in that while it does transform input traces to the RT domain, it never outputs the RT traces. The filter module internally stores input X/T traces and headers as well as the RT traces and their headers, simultaneously for each application. In the most often-used mode, the input X/T traces are transformed to radial traces in the RT trace array, a filter is applied to the RT array, and this filtered array is inverse transformed and either subtracted point-by-point from the input trace array, or used to replace the array, depending upon the type of filter applied (Henley, 1999a, 1999b, 2000a, 2000b, 2003a, 2003b). The interpolation within the RT transform and inverse ensures accurate mapping to the original offsets associated with the input traces.

The stand-alone RT transform module, on the other hand, outputs the RT trace array as actual traces, including trace headers, so that any information needed to invert the RT trace ensemble, like min and max offsets and CMP, must be included in the RT trace headers, since it is not otherwise available within the RT transform as originally deployed (Henley, 1999c).

SOLVING THE HEADER PROBLEM

When we first developed the RT transform algorithm for ProMAX, it was intended mainly as a diagnostic tool, requiring only rudimentary seismic geometry information for its successful application. Hence, the original algorithm we deployed required only the source-receiver offset values placed in the trace headers of the raw trace ensembles. Thus, it was easy to test a single trace ensemble, or a small group of ensembles, without building a database for an entire seismic line. Our standard practice was to process most 2D data from a file conventionally named ‘shots with geometry’, or some other similar file. During this early development we adopted the strategy of storing a handful of geometry parameters for each input gather in the trace headers of every single output radial trace in the transform of that gather. Thus we could invert any RT transform, approximately, from this augmented RT trace header information alone, and we could output not only the original X/T gather, but also an interpolated or decimated gather, or a gather with a different offset distribution for diagnostic purposes. This scheme worked well, as long as the input data were strictly 2D, with sources collinear with receiver spreads.

If we always kept RT transform ensembles intact after their creation, we could have chosen to accompany each RT ensemble with its original X/T input data, flagged as auxiliary traces, and we would then always have access to the exact trace header values needed to invert any given RT ensemble. One application, however, which requires RT transform ensembles to be reorganized by sorting is raypath interferometry (Henley, 2008, 2012a, 2012b), in which a set of RT transform gathers corresponding to the source (receiver) gathers of a seismic line are sorted to ‘common-ray-parameter’ or ‘common-angle' gathers (analogous to common-offset gathers in the X/T domain). During this rearrangement, there is no obvious way in which to keep a set of auxiliary traces representing an original input data ensemble intact; hence our shortcut, storing only a few key X/T ensemble parameters in RT trace headers. When the RT traces are subsequently re-sorted into RT source (receiver) ensembles, these rudimentary parameters can then be used to approximately invert the recovered RT ensembles to X/T source (receiver) gathers. As illustrated later, the approximation is sufficiently accurate for seismic data which are strictly 2D. As we discovered, however, our version of the RT transform is not adequate when applied to any data whose geometry deviates significantly from a 2D plane, like most receiver line gathers in a 3D land survey. In this case, the linear reconstruction of source-receiver offset values does not accurately reflect the hyperbolic offset distribution actually present, and the inverse RT transform is badly distorted (Henley, 2015, 2016a, 2016b, 2017a, 2017b). We illustrate this, as well, in a subsequent section.

The solution—database creation

Since it appears impractical, or at least very tedious to store enough X/T header information in the headers of RT traces to allow accurate RT inversion, we adopt the approach used in commercial software for other 2D transforms in which the dimensions of the transform array differ from those of the input array. Hence, we create a database compatible with the original input data, which allows the original geometric trace header information to be retrieved at such a time as the inverse RT transform recreates an output ensemble with the same number of traces and other characteristics as the input X/T gather, and we use the trace index number to match the proper header information to each output trace.

Most processing projects using commercial software packages such as SeisSpace involve the creation of a database in which are stored all the pertinent geometric and other parameters associated with every seismic trace in a data set. The database creation process usually happens early in the processing project so that most subsequent operations can reference the database, if needed.

Because there is a well-established procedure for database creation, the task of data preparation for the RT transform and its inverse is relatively straightforward, involving three SeisSpace operations. The details are presented in the updated RT transform documentation included as an appendix; but in brief, the steps are as follows:

- Begin with a set of 2D trace ensembles with proper survey geometry trace headers appended, typically labelled 'shots + geometry'. Survey geometry must include, but is not limited to: X and Y coordinates of source and receiver, signed source-receiver offset distance values and the like.

- Apply the SeisSpace operation 'Extract database files'.

- Use the '2D land geometry spreadsheet' interactively to enter key survey parameters, reconcile specific geometry features, and finalize the database.

- Apply the operation 'Inline geom header load' to 'shots + geometry' and output a new file of trace ensembles in which all traces match their database descriptions. When the new file is read by a subsequent operation, the SeisSpace preparation phase will confirm whether the trace headers match the database.

The resulting data set will consist of trace ensembles with an associated database, from which may be drawn various trace header parameters as required,

Using the database

The RT transform module, radtran, in SeisSpace has been modified to include a new option for inversion. We have added to the original parameter choices, under 'Method for offset increment computation', a choice labeled 'original offsets', and have made it the default selection. This selection assumes that a database has been created for the survey or group of ensembles to be processed, and that the message 'Geometry loaded in the trace headers matches the database--Trace numbers can be used to reference the database' has appeared in the execution log during the SeisSpace execution preparation phase. When these conditions have been met, the inverse RT transform opens the database and retrieves the exact source-receiver offset values required to invert each RT ensemble, hence making the inversion exact. Any of the other choices for the 'Method' parameter use the minimum and maximum offset values stored in the RT trace headers to interpolate approximate offset values for the inversion, as in the original RT transform algorithm.

Users should be aware, however, that although the values appearing in the offset headers of an X/T trace ensemble output from the inverse RT transform will be correct, most of the other header values are dummies. Hence, two more SeisSpace operations are required to completely restore original header information to the output trace gathers; 'Trace header math' and 'Inline geom header load'. The first operation is required to restore the channel numbers to the traces of each output gather, since these were lost during the forward RT transform. Trace header math is thus used in 'sequence renumber mode' to renumber 'traces' with 'recording channel number', using 'source index number' as the reset index. Once this has been accomplished, 'Inline geom header load' is used to load all the trace headers to each output trace gather using 'source index number' and 'recording channel number'. The resulting trace ensembles should then be indistinguishable from the original input X/T ensembles, except for whatever processing has occurred in the RT domain.

If the database has been created, the 'trace header math' and 'Inline geom header load' can also be used to restore headers to output ensembles for which the 'linear' or 'quadratic' interpolation options were used in the inverse RT transform, as long as the number of output traces equals the number of input traces for the corresponding original input ensembles. In this case, however, the offset values will not correspond to the offsets actually used in the inversion.

DEMONSTRATIONS

In this section, we illustrate various aspects of the preceding discussion with field data examples. We start first with radial trace filtering, since this technique is unaffected by the problem of retrieving header information. We show both 2D and 3D examples. We then illustrate the application of the forward and inverse RT transform, first to data that are strictly 2D, and we show each of the four ‘offset increment’ selections on these data. We also demonstrate the trace interpolation/decimation capability of the RT inverse while using the linear offset interpolation method. Finally, we demonstrate the linear method on 2D data with significantly offset source point and show the data distortion resulting from offset values that depart from the correct values by more than a fraction of the offset increment.

RT filtering

Even in the earliest days of its deployment as a tool to assist in efficient application of wave equation migration algorithms, the radial trace (RT) transform was recognized for its ability to effectively separate linear coherent noise, particularly ground roll, from seismic reflections on source ensembles of seismic traces (Claerbout, 1983). The reason for its effectiveness becomes intuitively apparent when a map of its sampling trajectories (a fan of straight lines with origin placed at the source point) is overlaid on a source gather. For a typical source gather, the RT trajectories are very nearly parallel to the various modes of linear noise radiating from the source point, direct arrivals as well as ground roll. This means that linear noises are mapped into very low-frequency events in the RT domain, regardless of their frequency content in the original X/T domain, making them easily separable from desired reflection events, which do not align with RT sampling trajectories. We developed a ProMAX (later, SeisSpace) operation, radfilt, which utilizes this separation to remove undesirable source-generated noises from seismic trace ensembles. The separation and removal can be accomplished by one of several techniques, most often by estimating the coherent noise, then subtracting it from the original data traces. Regardless of the chosen method, however, the noise removal occurs totally inside the radfilt algorithm, which always retains the input X/T trace ensemble to re-use as the output ensemble. Thus, the input ensemble trace headers are used directly for output and are always correct. The RT domain exists only inside the operation as the filtering work array, whose content, via the inverse RT transform either replaces the samples of the input ensemble, or is subtracted from them to produce the output X/T.

Figure 1 shows a generic source gather with an overlay of radial trace sampling trajectories, showing that coherent noises, which tend to be linear with source-receiver offset, are quite conformable to the trajectories. Figure 2 shows the resulting RT transform; and it is apparent that the coherent noises map into low-frequency traces, while reflections are nearly untouched. Figure 3 shows a low-pass filter applied to the RT transform of Figure 2; all the reflections have been removed. Transforming Figure 3 back to the X/T domain results in Figure 4, the estimated coherent noise. Subtracting this noise estimate from the original trace gather yields the filtered gather shown in Figure 5.

To verify that this works just as well on 3D data, where source-receiver offset distances on receiver line ensembles are hyperbolically distributed, rather than linear, we show a 3D receiver line gather in Figure 6, as well as the noise estimate in Figure 7, where it can be seen that the hyperbolic offset distribution is faithfully reproduced in the noise estimate trace gather. Hence, subtracting the noise estimate from the original gather results in Figure 8, the correctly filtered receiver line gather, with correct trace headers, since they are identically those of the input gather (Henley, 2007).

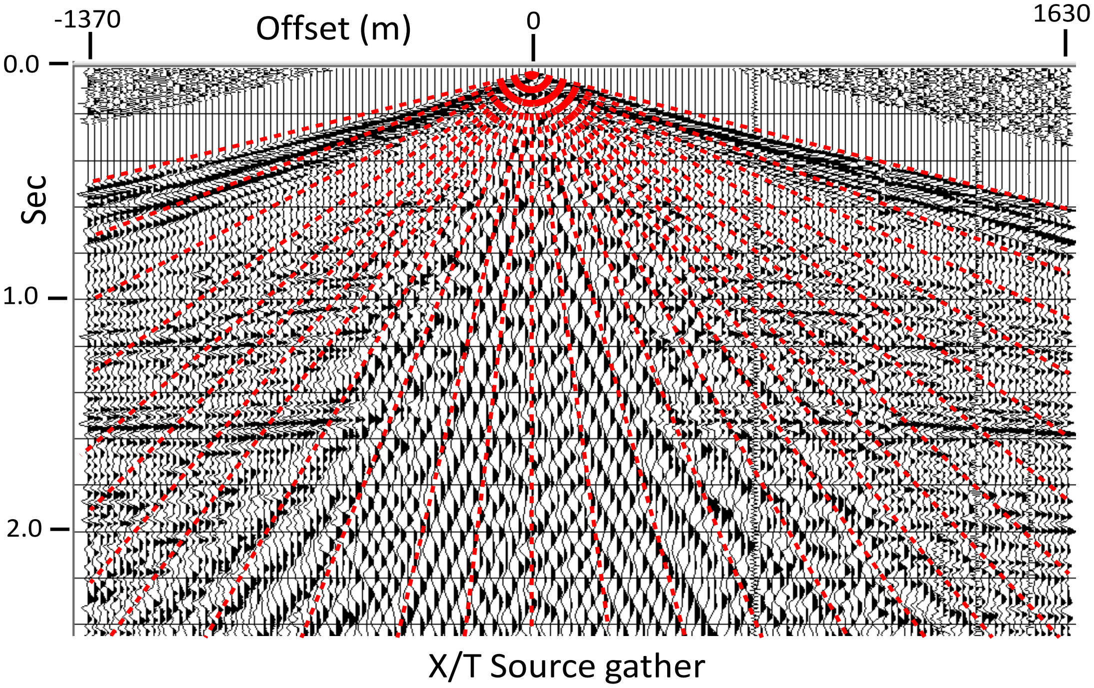


FIG. 1. A generic, X/T domain source gather of seismic traces, overlaid with a fan of radial trace (RT) sampling trajectories. The X/T domain samples lying along each RT trajectory become a trace in the RT domain. Typically, there would be 2000 or more trajectories to avoid aliasing.

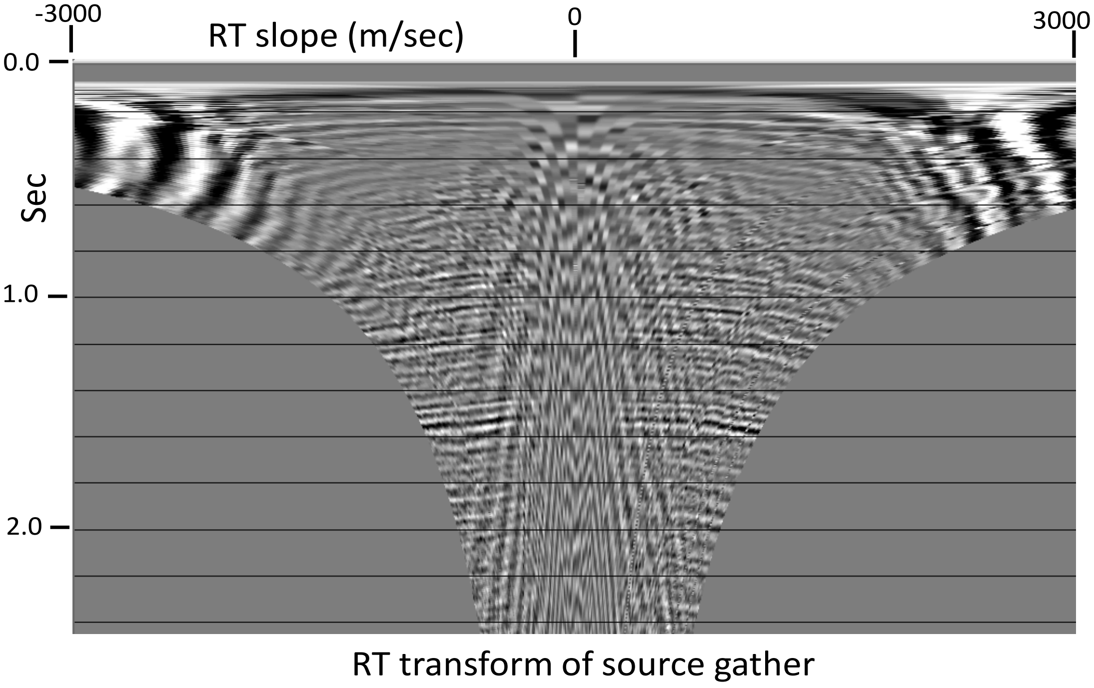


FIG. 2. Radial trace (RT) transform of the X/T source gather in Figure 1. In this domain, direct arrivals are nearly vertical towards the outer edges of the gather, and the ground roll maps to nearly vertical events near the centre. In either case, the events are much lower in apparent frequency in this domain than in the X/T domain, allowing us to isolate them using a low-pass filter.

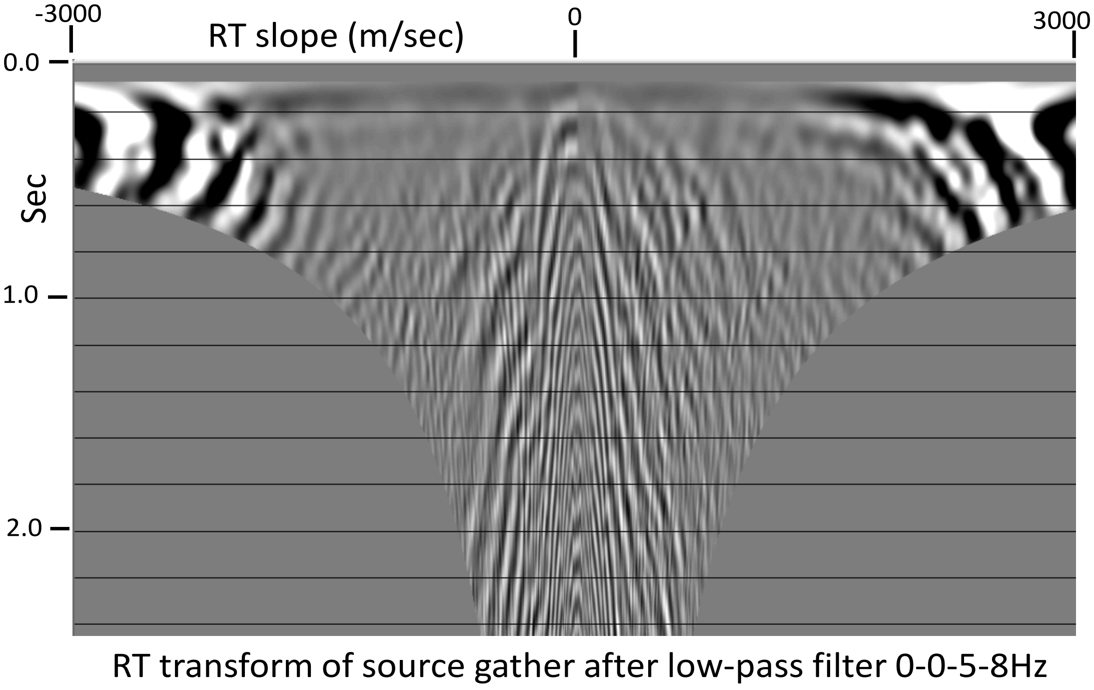


FIG. 3. Radial trace transform of the X/T source gather in Figure 1 after application of a low pass filter. Reflections have been removed from the gather, leaving only the direct arrival and ground roll coherent noise. This filtered RT transform is the coherent noise estimate.

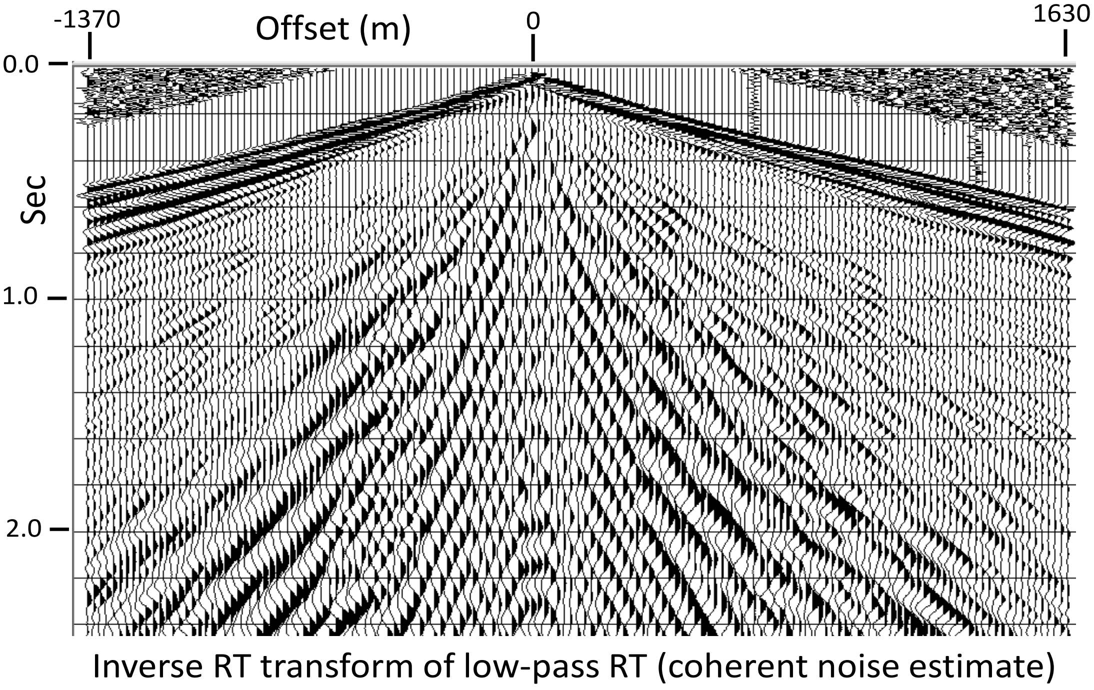


FIG. 4. Inverse RT transform of the low-passed RT gather in Figure 3. This is the coherent noise estimate in the X/T domain, which may be subtracted directly from the raw X/T source gather in Figure 1.

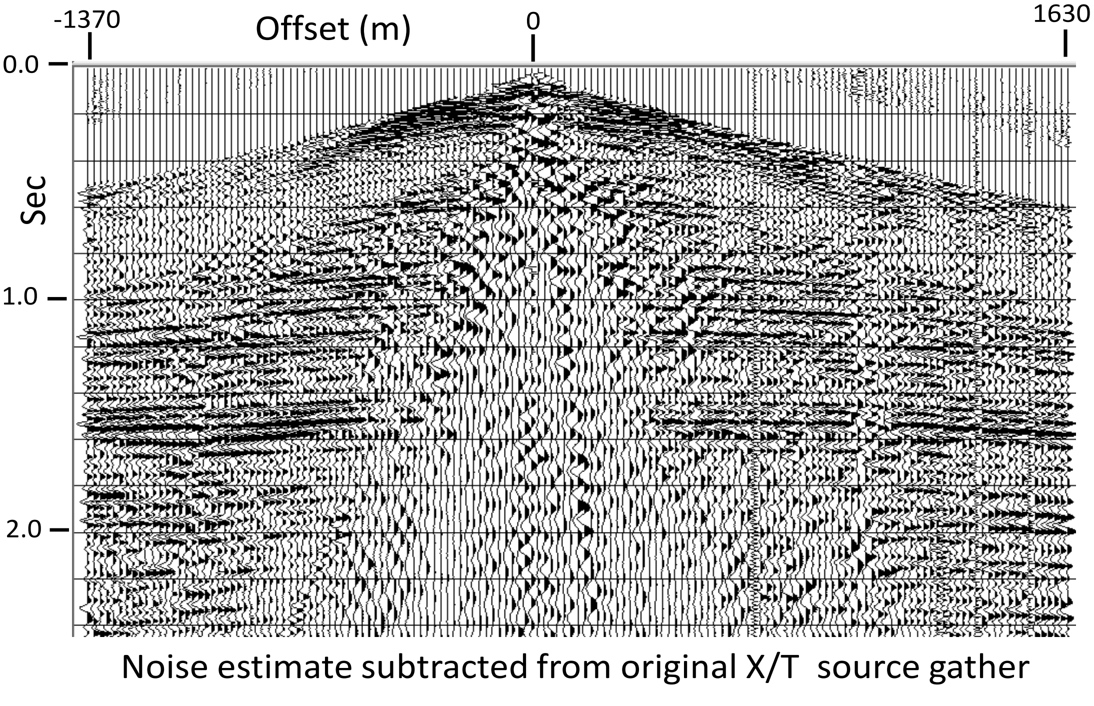


FIG. 5. The X/T source gather from Figure 1 after subtracting the coherent noise estimate in Figure 4. Most of the direct arrival and ground roll events have been greatly attenuated.

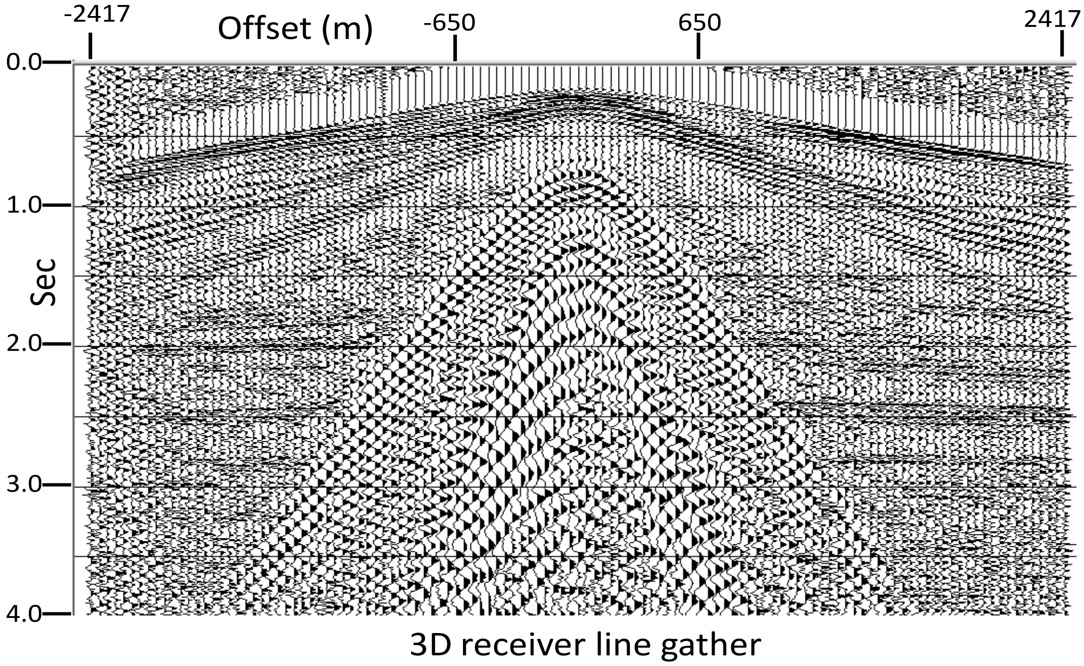


FIG. 6. A generic X/T 3D receiver line gather, where the source position is displaced laterally from the receiver line by several stations. This alters the source-receiver offset distances to a hyperbolic distribution, rather than the linear distribution of a 2D line. Hence, the coherent, source-generated surface noise has a hyperbolic pattern on this gather, even though the event moveouts are actually linear with source-receiver offset values.

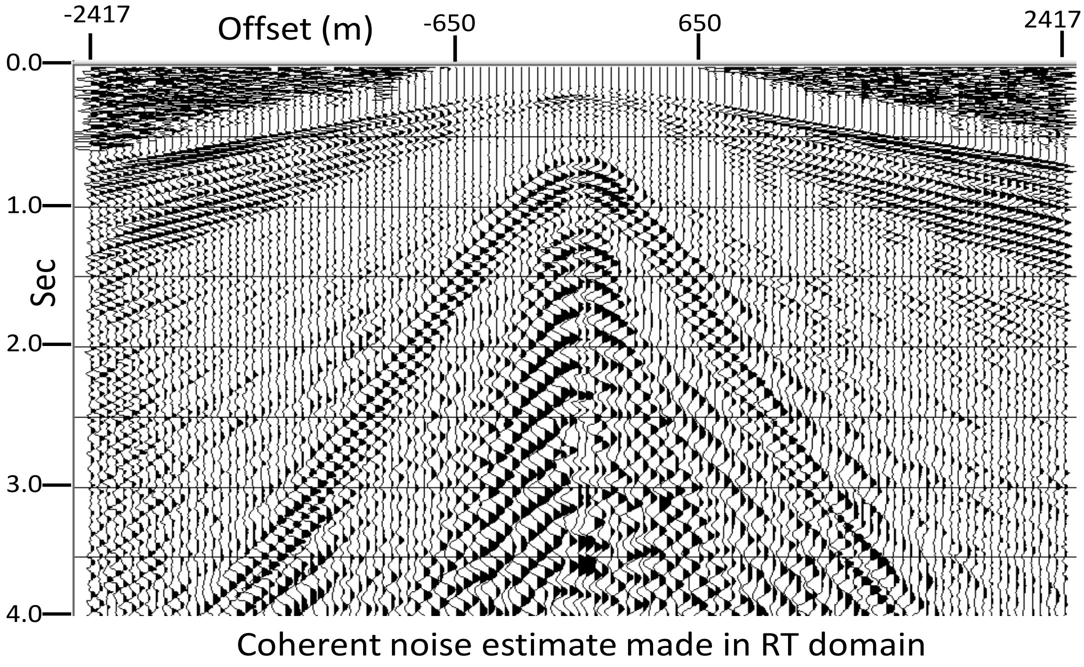


FIG. 7. Coherent noise estimate for the receiver line gather in Figure 6, obtained from a low-pass filter in the RT domain. The noise events have a hyperbolic appearance because of the hyperbolic distribution of source-receiver offsets for any 3D receiver line gather. Because the event shapes are faithfully represented, this gather may be subtracted from the original gather in Figure 6 to accomplish coherent noise attenuation.

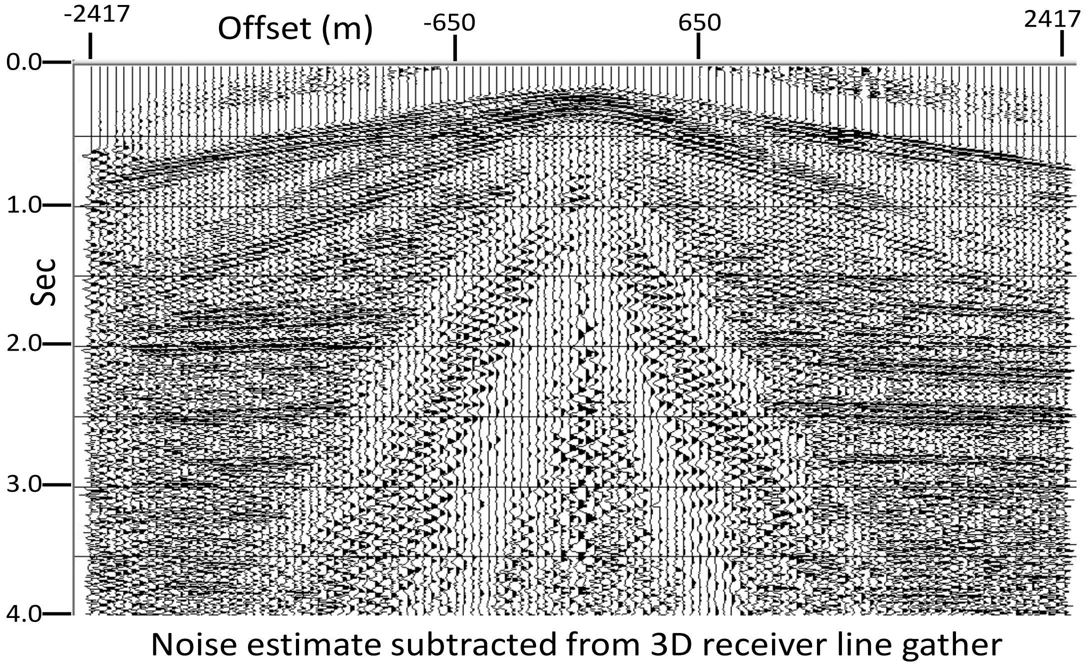


FIG. 8. Coherent noise attenuated on the 3D receiver line gather from Figure 6 by subtracting the noise estimate in Figure 7.

RT transform—2D trace ensembles

Since our version of the RT transform was originally developed to apply to seismic trace ensembles whose acquisition geometry is strictly collinear, we show here that the forward/inverse RT transform using linear offset interpolation is sufficiently accurate for routine use. Figure 9 shows a generic source gather with a typical split-spread geometry; and Figure 10 shows the same gather after forward/inverse RT transform, using the linear offset interpolation. Comparing the two figures, the trace details are indistinguishable, and the offset headers are identical, as well. Comparing a zoom view in Figures 11 and 12 confirms the accuracy of this approach.

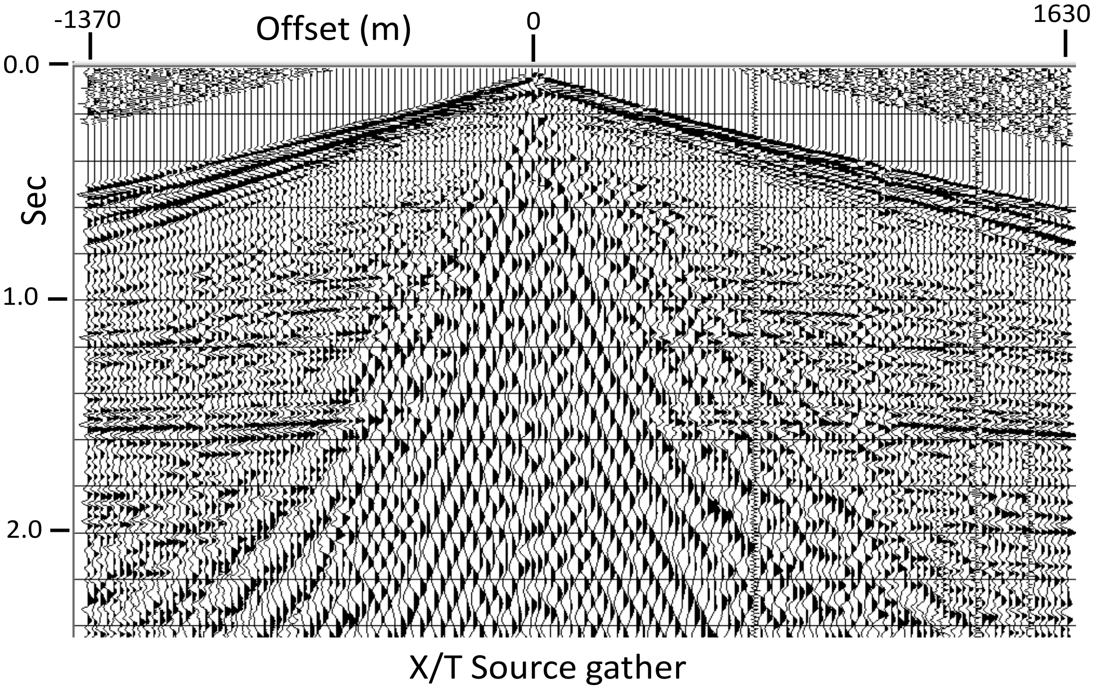


FIG. 9. Original X/T source gather from a 2D seismic line.

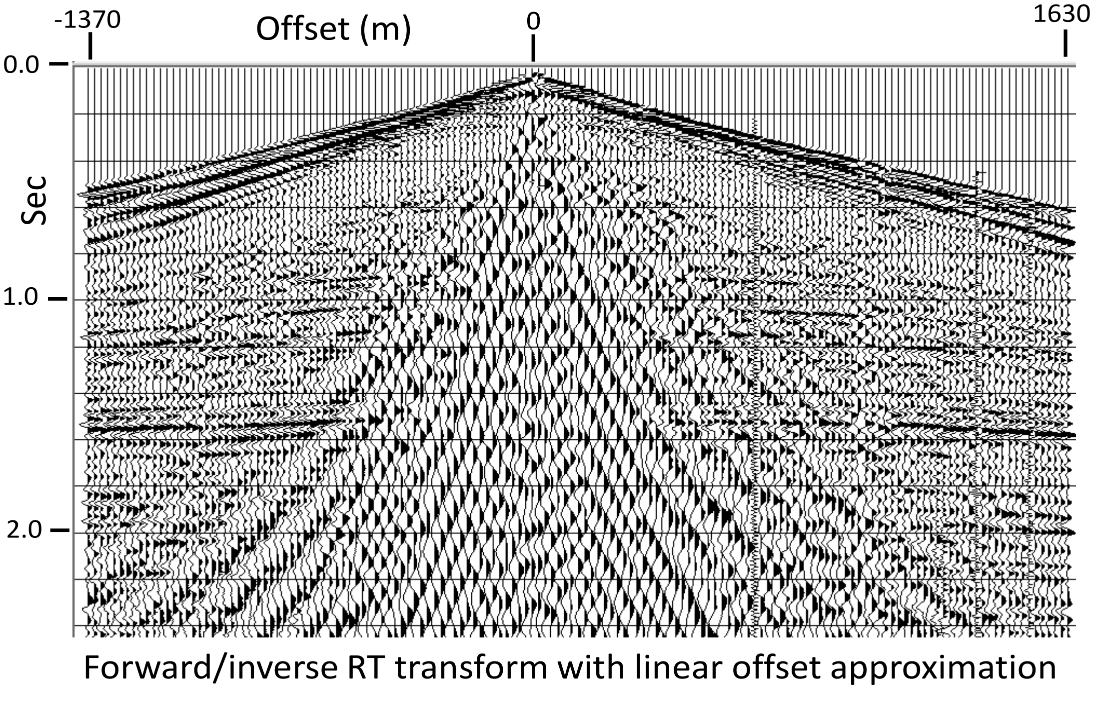


FIG. 10. For 2D lines, the forward/inverse RT transform provides quite adequate accuracy using the linear interpolation option for source-receiver offset values. Compare with Figure 9.

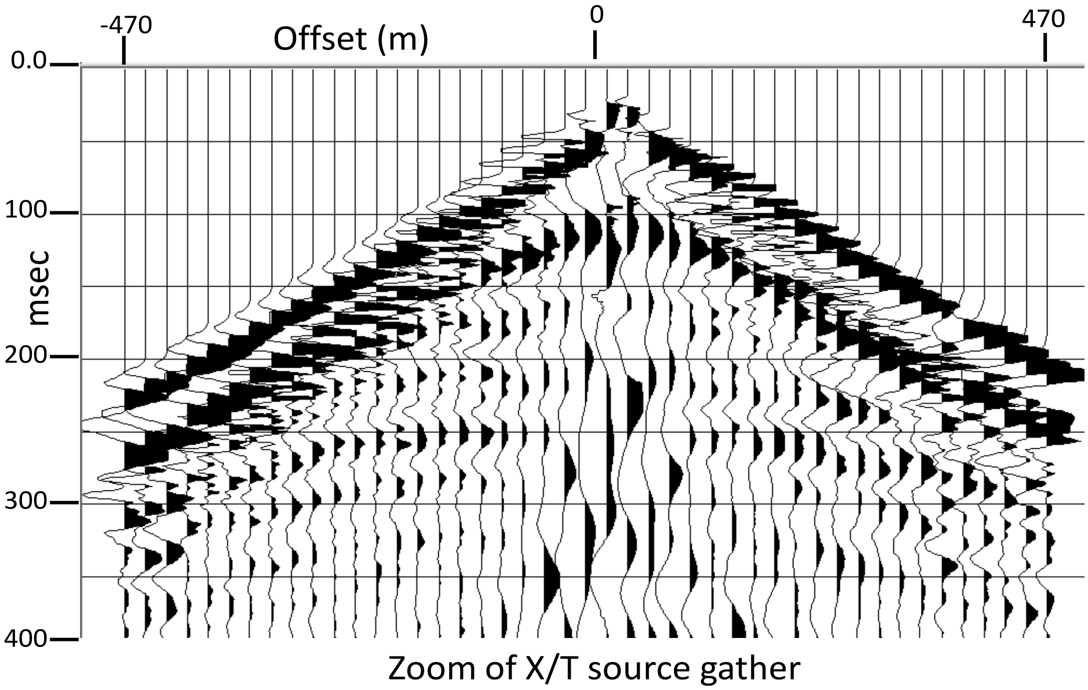


FIG. 11. Close-up view of source gather in Figure 9.

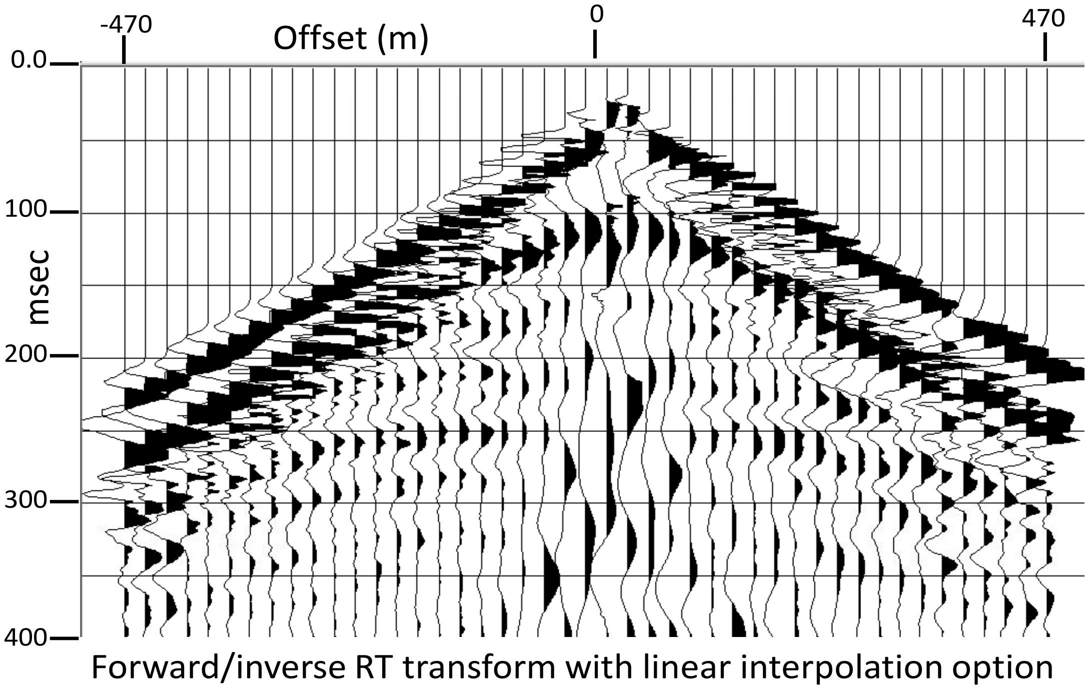


FIG. 12. Close-up view of the forward/inverse RT in Figure 10; the linear interpolation of source-receiver offsets provides quite adequate accuracy for the inversion. Compare with Figure 11.

When using the RT transform in a diagnostic mode, or for various image distortions (Henley and Wong, 2013), we illustrate the trace interpolation/decimation feature in Figures 14 and 15. First, in Figure 13, we show the source gather from Figure 9 after a pass of RT filtering and mild trace mixing to attenuate steeply dipping events which might alias. In Figure 14, we show the trace gather from Figure 13 after forward/inverse RT transform with twice the number of X/T traces in the output as in the input, to demonstrate trace interpolation. Figure 15 shows the gather from Figure 13 with half the number of traces, after the forward/inverse RT transform. To show the relative fidelity of the interpolation/decimation processes, Figure 16 shows the result of decimating the ensemble in Figure 14, while Figure 17 shows the interpolation of Figure 15. Both these figures should be compared with Figure 13 for accuracy. In our experience, neither interpolation nor decimation should exceed factors of 2, or ½, respectively; other, rational factors can certainly be used, however. For example, a 100 trace source ensemble can certainly be interpolated to 125, 150, etc. with little loss of fidelity; and the same gather can be decimated to 75, 60, etc. also with little loss of fidelity.

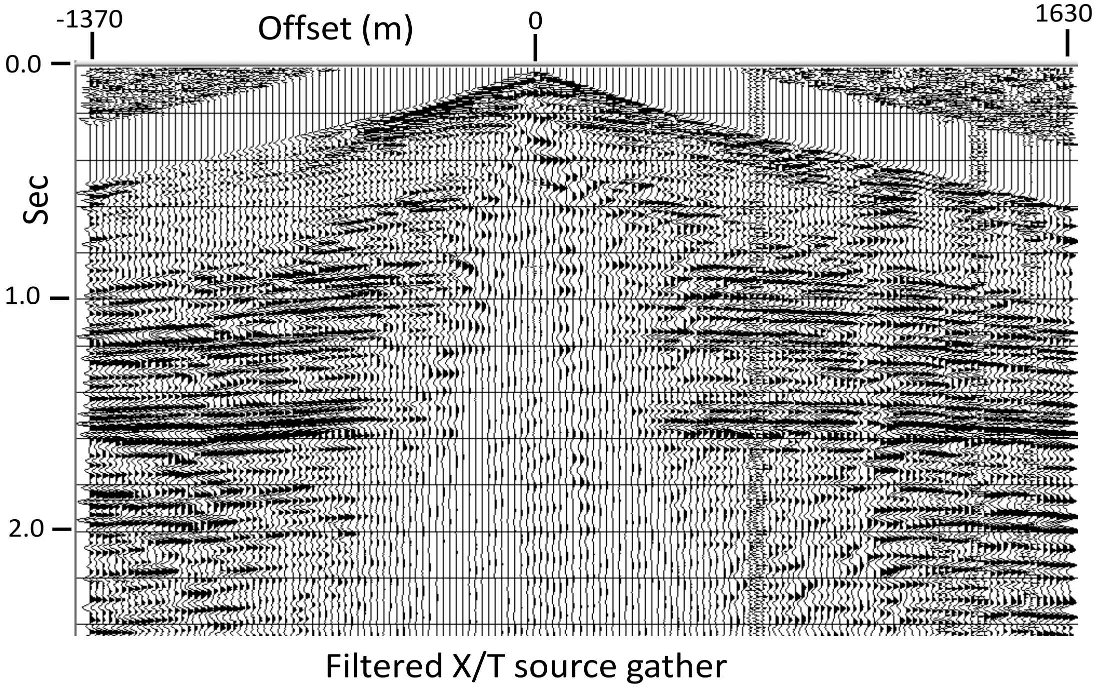


FIG. 13. The X/T source gather in Figure 9 after being RT filtered and trace mixed to remove steeply dipping events that might alias during interpolation or decimation.

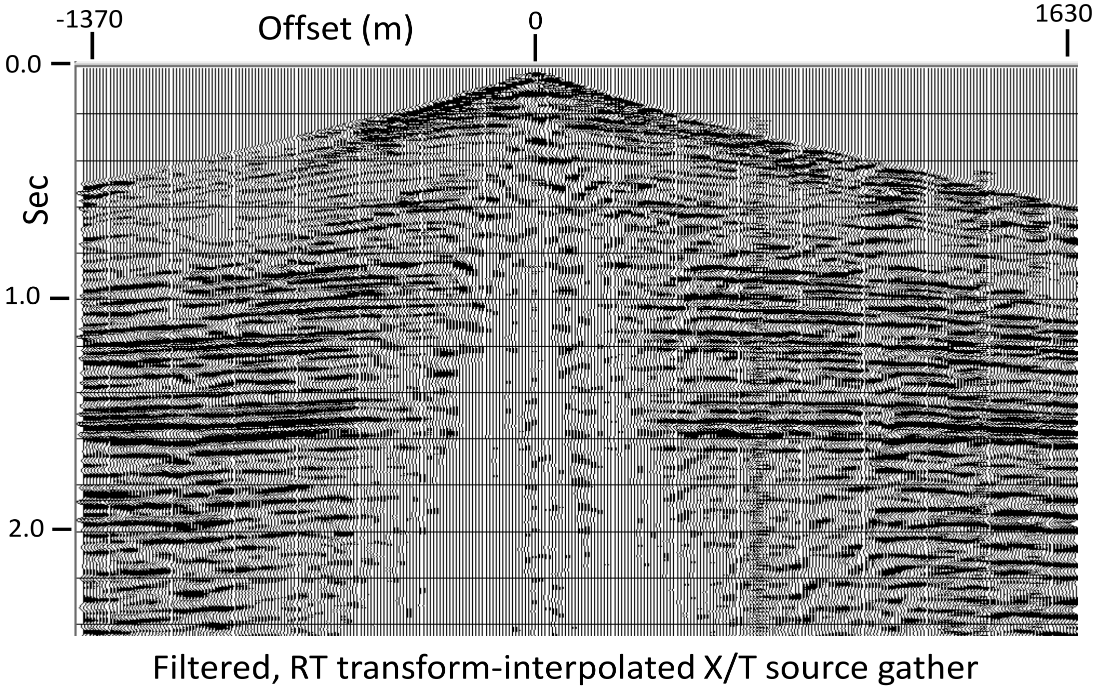


FIG. 14. Filtered X/T source gather from Figure 13 after being interpolated by a factor of 2 using the RT forward and inverse transform.

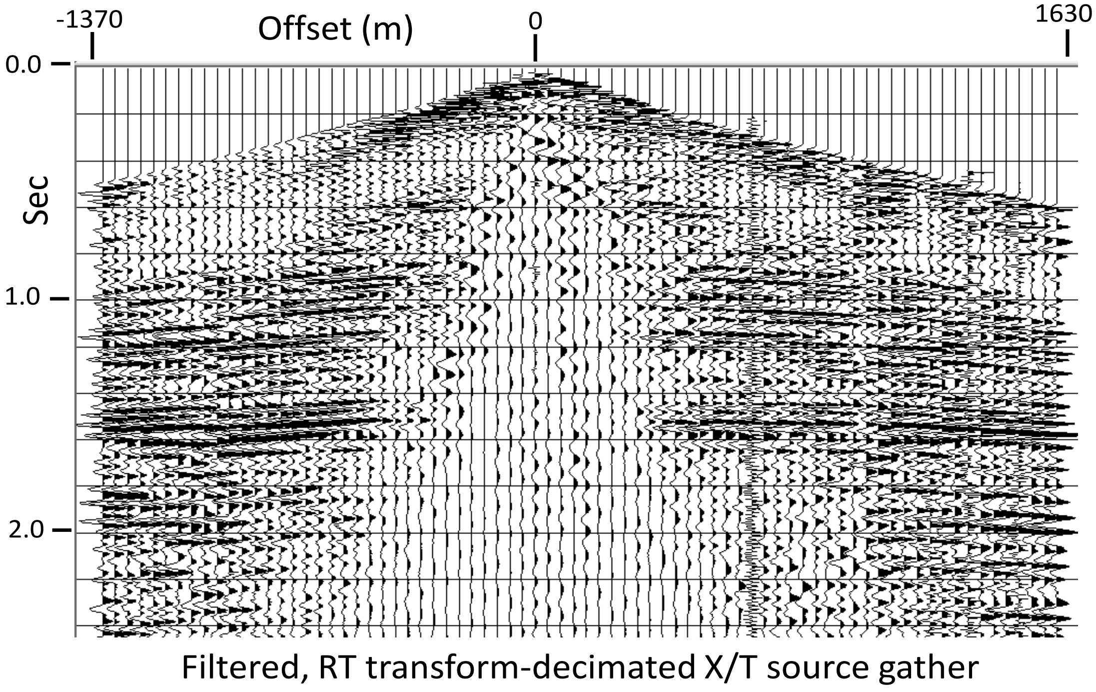


FIG. 15. Filtered X/T source gather from Figure 13 after being decimated by a factor of 2 using the RT forward and inverse transform.

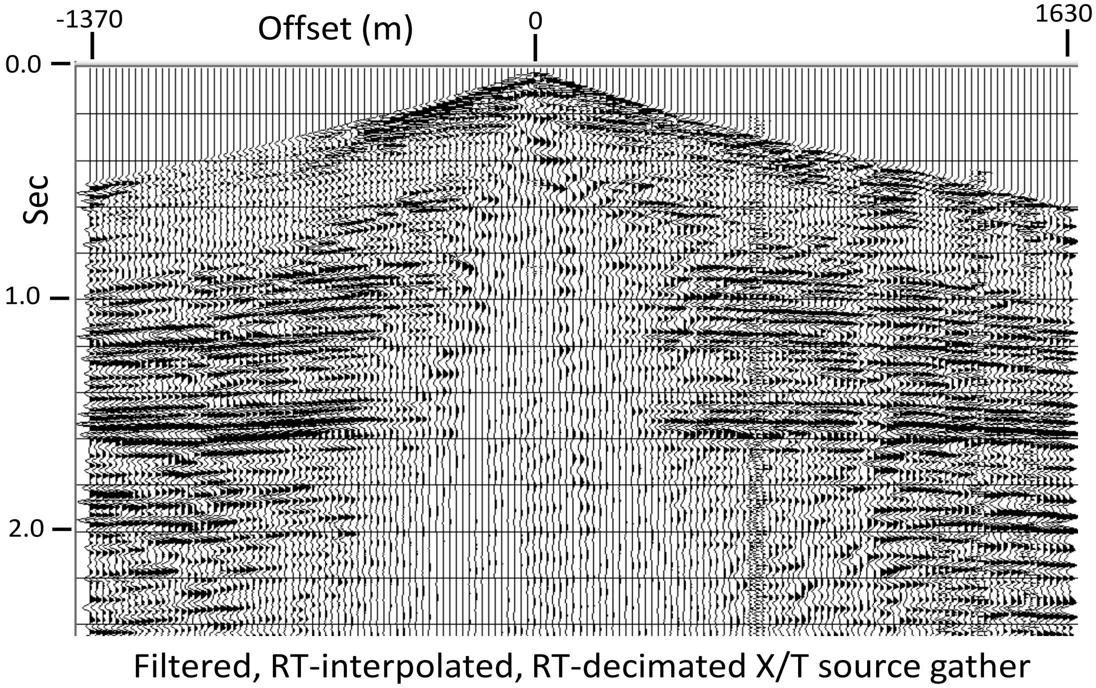


FIG. 16. The interpolated gather from Figure 14 after being decimated by a factor of 2 using RT forward and inverse transform. Compare with Figure 13.

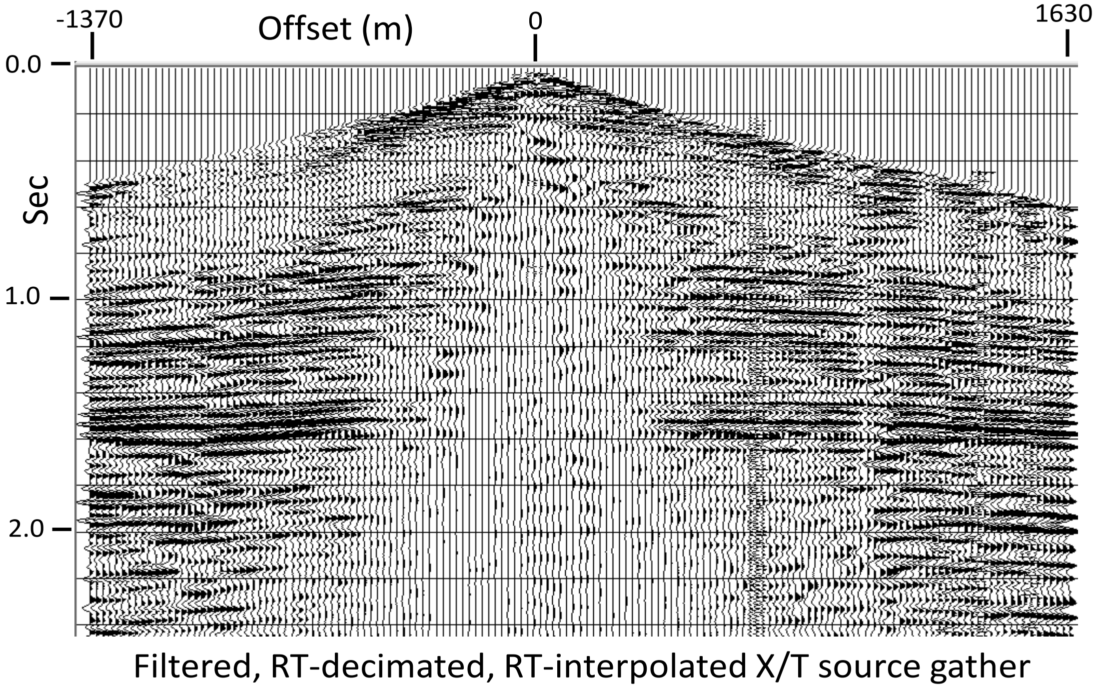


FIG. 17. The decimated gather from Figure 15 after being interpolated by a factor of 2 using RT forward and inverse transform. Compare with Figures 13 and 16.

The non-linear interpolation options offered by the RT transform are intended primarily for diagnostic purposes, and there is no guarantee of data fidelity. Figure 18 shows the gather of Figure 13 after being interpolated to quadratic offset values upon RT inversion, and Figure 19 shows the same gather transformed to an X^2/T^2 panel upon RT inversion. While either of these outputs could be interpolated or decimated on output, as described above, the resulting ensembles could well show aliasing and/or undersampling as a result of the non-uniform offset interpolation.

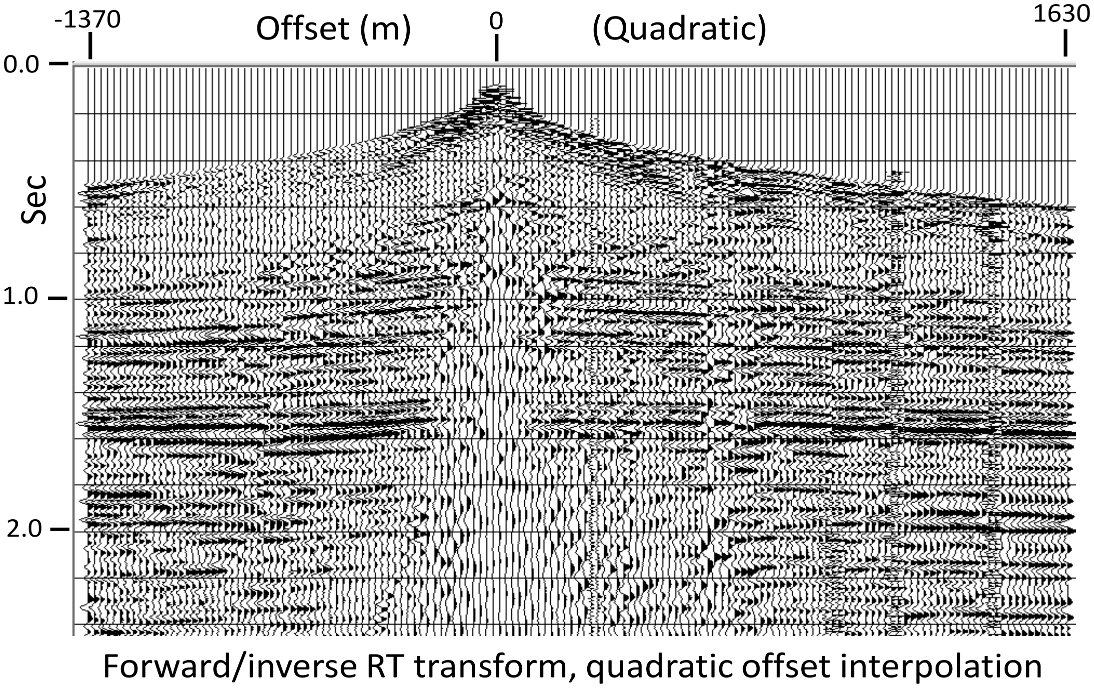


FIG. 18. The X/T source gather from Figure 9 after being RT filtered, then re-mapped to quadratic offset distribution using the RT forward/inverse transform with quadratic offset interpolation option.

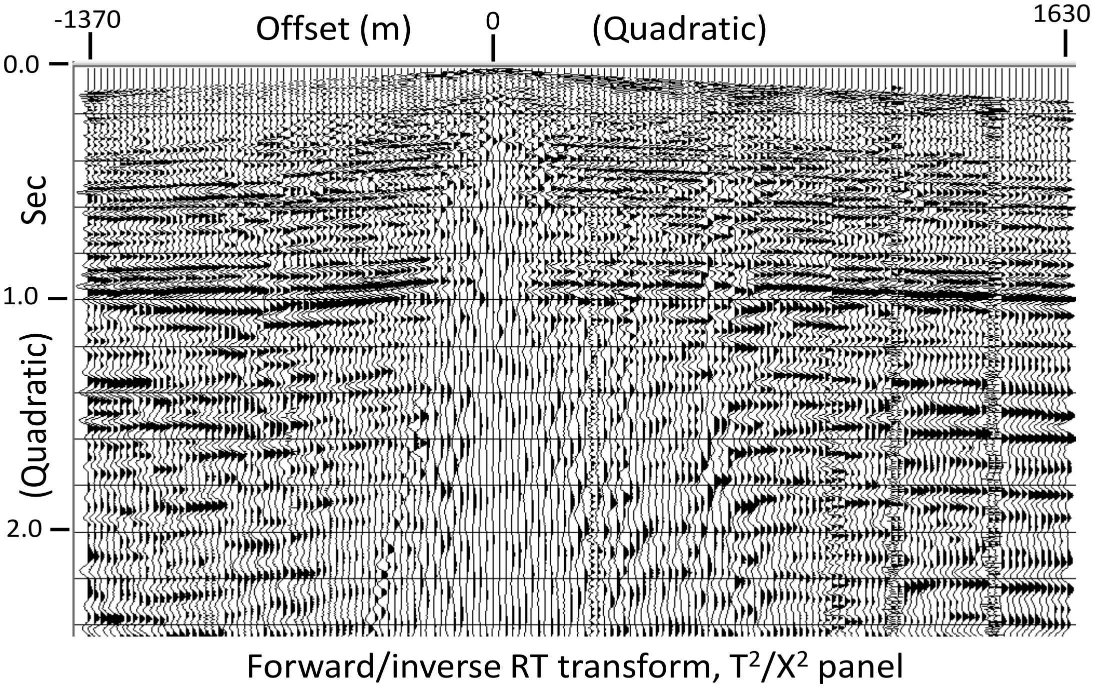


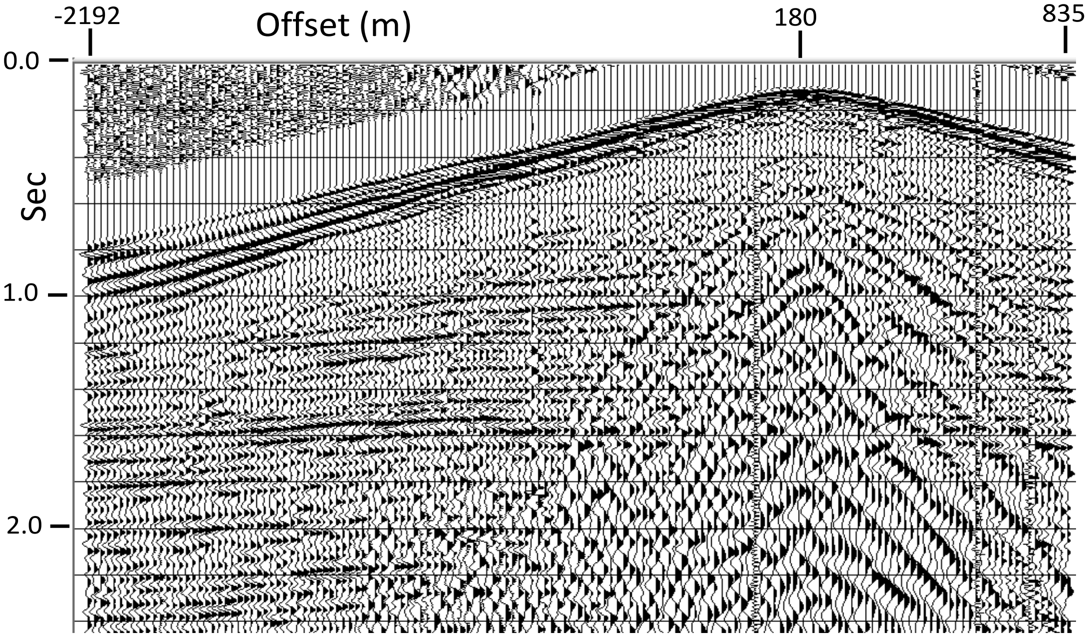
FIG. 19. The X/T source gather from Figure 9 after being RT filtered, then re-mapped to the X^2/T^2 domain using the RT forward/inverse transform with quadratic offset interpolation, as well as quadratic travel-time mapping. All reflections and other hyperbolic events in the X/T domain should appear linear in the X^2/T^2 domain.

RT transform—2D ensemble with non-collinear source point

To illustrate the offset header problem with the original RT transform algorithm, we display first, in Figure 20, a source ensemble from the same 2D line as the ensemble in Figure 9. It is obvious from the pattern of first arrivals that the source is displaced perpendicular to the receiver spread by several station intervals (approximately 6 stations), making the distribution of offset values hyperbolic. Even though this is not a particularly large source displacement, replacing the actual offsets with the linearly interpolated values results in the distortion seen in Figure 21, clearly unacceptable for data fidelity. It is clear from this display that we cannot use interpolation to reconstruct the source-receiver offsets for any RT inverse transform where the input data geometry is not very nearly collinear. For 3D, we most often apply the RT transform to receiver line gathers for which the source displacement is often a distance comparable to a sizeable fraction of the receiver spread length.

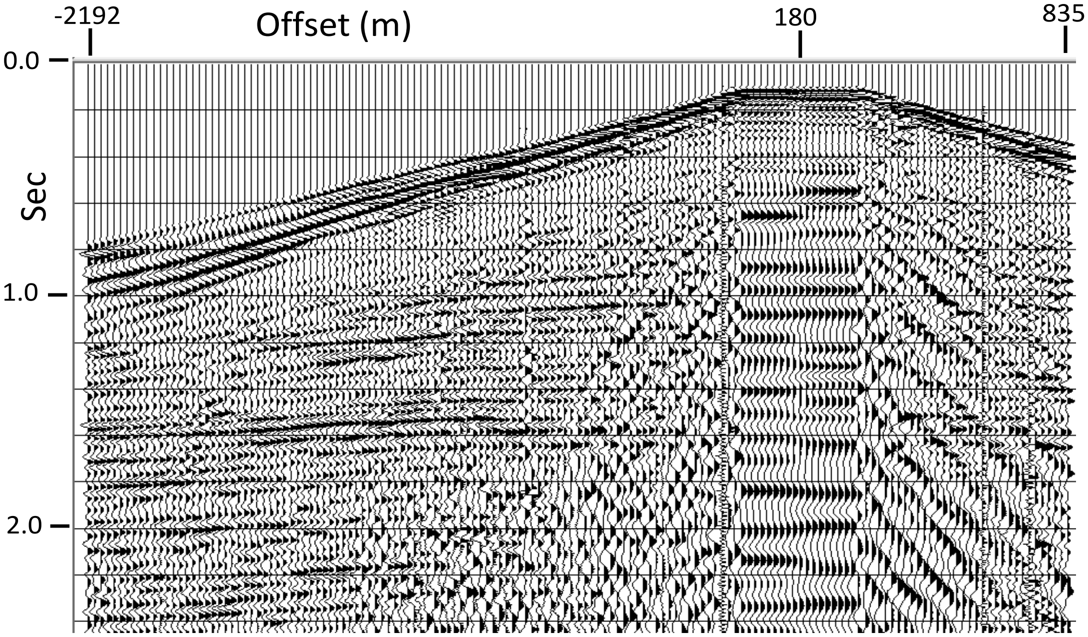
The solution

As we described earlier, the most reasonable solution to the offset header issue is to transfer trace header values from a database to the output array in the inverse RT transform; this is what we display in Figure 22. In comparison with Figure 20, it is clear that the accuracy of the RT transform and inverse is retained. Figures 23 and 24 show zoomed portions of the input and output for comparison and are nearly indistinguishable.



X/T source gather, source point displaced 6 stations from the line

FIG. 20. An X/T source gather from the same survey as Figure 9. In this case, the source point is displaced perpendicular to the 2D line by 6 stations. Because the source-receiver offset values are hyperbolic for this geometry, the direct arrivals and ground roll assume a hyperbolic pattern when trace spacing is plotted linearly.



Forward/inverse RT transform, linear offset interpolation option

FIG. 21. Linear interpolation of source-receiver offsets during the RT forward/inverse transform is not sufficiently accurate, when the acquisition geometry deviates from the 2D plane, as when the source is displaced perpendicular to the line by a few stations.

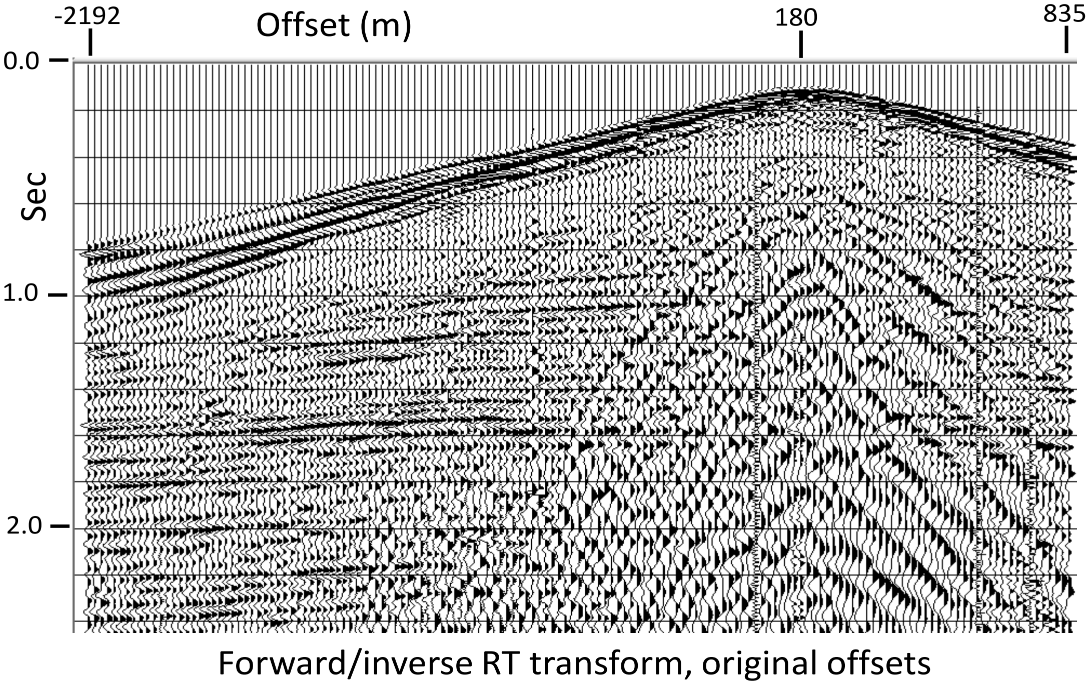


FIG. 22. Using a database to recover the original source-receiver offsets from the input X/T source gather results in an accurate recovery of the original X/T gather, regardless of the displacement of the source point from the 2D line.

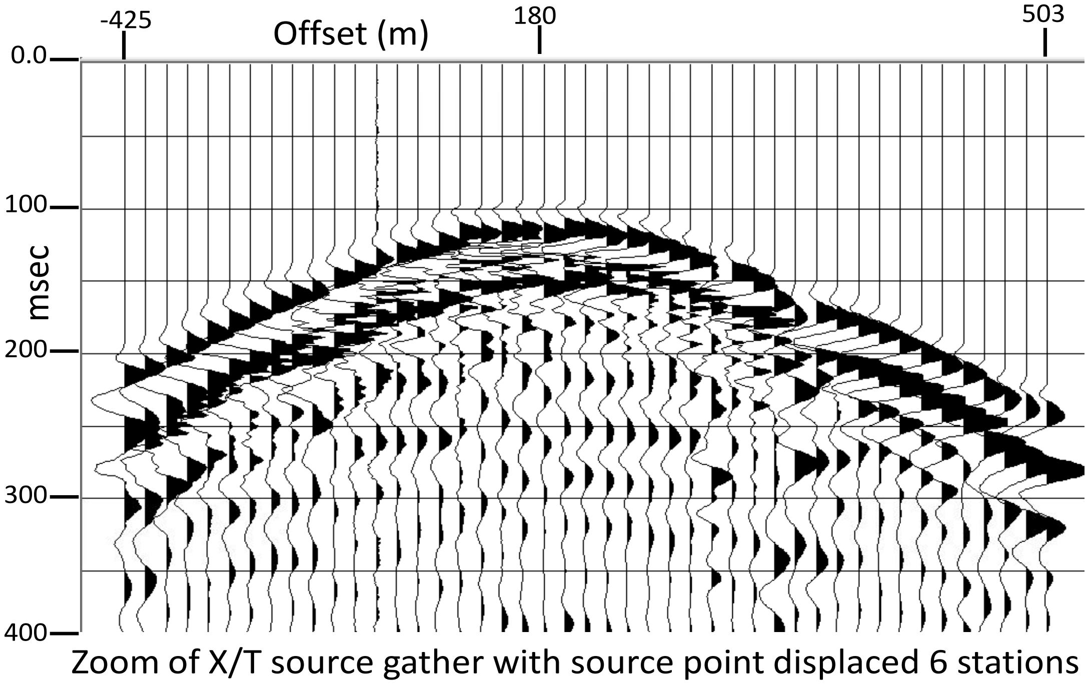


FIG. 23. Close-up of part of the X/T source gather in Figure 20.

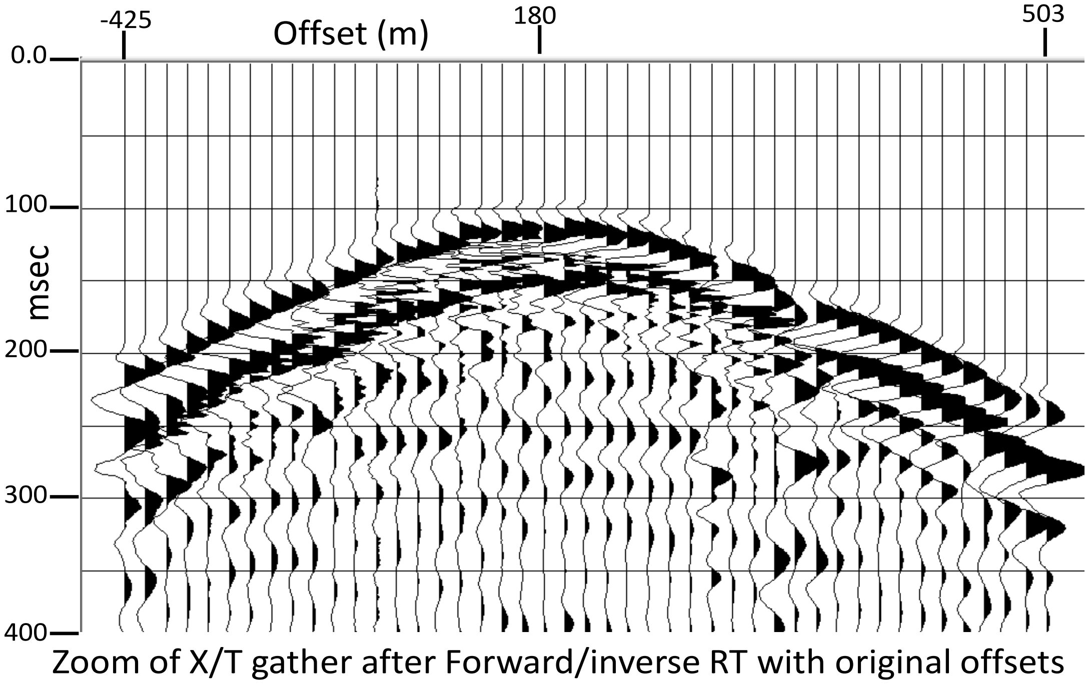


FIG. 24. Close-up of the X/T source gather after forward/inverse RT transform using the original source-receiver offset values retrieved from a database built for the survey from the input X/T gathers. Details in Figure 23 are faithfully reproduced in this figure.

DISCUSSION

The intent of this report is obviously not to exhibit new or innovative technology, but just to document a long-overdue software fix and why it was necessary. At the same time, the report is intended as a review of the radial trace transform and a reminder of some of its applications, particularly in diagnostic mode. Because it is a relatively simple and unsophisticated operation, the RT transform is often overlooked in favour of more complex operations, in spite of its proven utility.

Considering that our RT transform was first released nearly 20 years ago, the big question is why it took so long to correct a major flaw. The honest answer is sheer reluctance on the part of the programmer (the author) to tackle the project. As well, we only ventured into the 3D domain about 3 years ago, where the shortcoming became hard to ignore. The main impetus to do the project was provided by the author's recent use of Vista, which has an RT inverse transform which correctly handles output headers. The awkwardness of attempting to emulate in Vista some of the processing flows used in SeisSpace finally prompted the author to 'bite the bullet' and invest the necessary time to fix our RT transform so that it could properly and accurately transport data to the RT domain and back. Mea culpa!

ACKNOWLEDGEMENTS

The author gratefully acknowledges the support of CREWES and NSERC for funding.

Kevin Hall was very helpful in assisting with programming environment tasks, and in boosting the author's knowledge of database operations.

REFERENCES

Claerbout, J.F., 1975, Slant-stacks and radial traces, Stanford Exploration Project Report, SEP-5, 1-12.

Claerbout, J.F., 1983, Ground Roll and Radial Traces, Stanford Exploration Project Report, SEP-35, pp 43-53.

Cova, R., Henley, D.C., and Innanen, K.A.H., 2013, Non-stationary shear wave statics in the radial trace domain, CREWES Research Report, 25.

Cova, R., Henley, D.C., and Innanen, K.A.H., 2014, Making shear wave statics actual statics using radial trace and tau-p transforms, CREWES Research Report, 26.

Cova, R., Wei, X., and Innanen, K.A.H., 2015a, Shear wave near-surface corrections in the tau-p domain: a case study, CREWES Research Report, 27.

Cova, R., Henley, D.C.., and Innanen, K.A., 2015b, Addressing shear wave static corrections in the ray parameter domain: a non-stationary interferometric approach, SEG Technical Program Expanded Abstracts 2015.

Cova, R., Henley, D.C., Wei, X., and Innanen, K.A., 2017, Receiver-side near-surface corrections in the Tau-P domain: a raypath-consistent solution for converted wave processing, Geophysics, 82, No. 2, pp U13-U23.

Henley, D.C., 1999a, Coherent noise attenuation in the radial trace domain: introduction and demonstration, CREWES Research Report 11.

Henley, D.C., 1999b, The radial trace transform: an effective domain for coherent noise attenuation and wavefield separation: 69th Ann. Int. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1204-1207.

Henley, D.C., 1999c, Radial trace computational algorithms at CREWES, CREWES Research Report 11.

Henley, D.C., 2000a, Wavefield separation in the radial trace domain, Geo-Canada 2000 Expanded Abstract No. 131

Henley, D.C., 2000b, More radial trace domain applications, CREWES Research Report, Vol. 12.

Henley, David C., 2002, Radial trace filtering revisited: current practice and enhancements, CREWES Research Report, Vol. 14.

Henley, David C., 2003a, More coherent noise attenuation in the radial trace domain, CREWES Research Report, Vol. 15.

Henley, D.C., 2003b, Coherent noise attenuation in the radial trace domain: Geophysics, 68, No. 4, pp 1408- 1416.

Henley, D.C., 2007, Radial filtering 3D data, CREWES Research Report, 19.

Henley, D.C., 2008, Raypath interferometry: statics in difficult places, 2008, SEG Technical Program Expanded Abstracts 2008.

Henley, D.C., 2012a, Interferometric application of static corrections, Geophysics, 77, No. 1, pp Q1-Q13.

Henley, D.C., 2012b, Interference and the art of static correction: raypath interferometry at Hussar, CREWES Research Report 24.

Henley, D.C., and Wong, J., 2013, Through the looking glass: using X-T plane distortions for wavefield separation, CREWES Research Report, Vol. 25

Henley, D., 2014, Static corrections via raypath interferometry: recent field experience: Geoconvention 2014, CSEG, Expanded abstracts 2014.

Henley, D.C., 2015, 3D or not 3D, that is the question: raypath interferometry in 3D processing, CREWES Research Report, Vol. 27.

Henley, D.C., 2016a, To boldly go into the next dimension; 3D raypath interferometry issues, CREWES Research Report, Vol. 28.

Henley, D.C., 2016b, 3D or not 3D, that is the question: raypath interferometry in 3D processing, Geoconvention Expanded Abstracts, 2016.

Henley, D.C., 2017a, A tale of two transforms: 3D raypath interferometry, CREWES Research Report, Vol. 29.

Henley, D.C., 2017b, To boldly go into a new dimension: 3D raypath interferometry issues, Geoconvention Expanded Abstracts 2017.

Ottolini, R., 1979, Migration of Radial Trace Sections, Stanford Exploration Project Report, SEP-20, pp 97-115.

APPENDIX

As an appendix, we have attached the updated documentation for the SeisSpace operation radtran (pod version).

```
=head1 Radial Trace Transform

This module provides both the forward and inverse radial transform.
As with the other radial trace modules, two modes of transform are
provided, the 'radial fan' mode and the 'radial dip' mode.  Since
the 'radfilt' module is specifically designed for linear noise
attenuation and wavefield separation, this RADTRAN module is
primarily for research and data analysis purposes.  One interesting
application, however, is in using the inverse transform to form an
X-T panel containing a significantly different set of source-
receiver offsets than the original X-T panel transformed by the
forward transform. In fact the module can be used to regularize the
offsets, or to interpolate the panel to a finer or coarser offset
increment.  The inverse transform also has the capability of
generating non-linear offsets.  The only requirement for the inverse
is that the range of offsets provided fall within the range of the
original offsets...extrapolation is not performed.

A new option for the inverse transform is the capability to
transform to or from a panel quadratic in both offset and travel
time. On  a X**2/T**2 panel all events having hyperbolic moveout in
```

the original X-T domain acquire linear moveout. In fact, all primary reflection events from flat layers have virtually the SAME linear moveout, enabling moveout correction by static shift alone.

A feature added to the radial trace transform in 2000 is the ability to create Snell traces, which are mapped from the X-T domain by following raypath trajectories through an interval velocity model according to Snell's law.

In 2004, the algorithm was modified so that the forward transform stores key geometry parameters in trace headers which are otherwise unused in the radial trace domain. The headers used are sou_x, rec_x, sou_y, and rec_y; and they store the minimum and maximum offsets of the input gather, the number of traces in the gather, and the cdp number of the first trace. These headers are retrieved by the inverse transform and used to approximately reconstruct the offset and cdp headers of the original input gather, except that the offsets are exactly uniformly spaced between the minimum and maximum of the input gather. The reason for these changes is to enable a processor to routinely transform gathers to the R-T domain, perform an operation on the radial traces, then inverse transform to yield a gather which can be processed and stacked in the same way as the original input X-T gather. Previous versions of this module were intended primarily for diagnostics to be applied to single gathers, where the particular header values could be explicitly entered as parameters for the inverse transform. The companion module, radial trace filter, did not share the same restrictions, since it carries the full set of input trace headers and sample values as input/output data arrays, and the radial traces are only stored internally in the module.

Late in 2004, a small bug was repaired in RADTRAN. This bug led to nonzero data values outside the boundaries of the XT trace gather when the transform was inverted. Now, the sample values outside the boundary are explicitly zeroed during the inversion.

In 2018, a fundamental shortcoming in the inverse transform mode was fixed...the inability to recreate the exact offsets contained in the trace headers of the original input gather. This fix was necessitated by the requirement that the radial trace transform be applied to trace gathers of 3D data where the source position is typically offset from the receiver line, thus introducing nonlinearity into the offset distribution, and rendering the linear interpolation used for 2D lines inappropriate. The fix was placed in the subroutine ‘OFFVECT’, which is called only when RADTRAN is in the inverse transform mode. In addition to the original options for restoring offsets to the output X/T panel, a new option, selected in the menu, is to use the ‘original offsets’. This is now the default option in the menu. When this option is selected, the subroutine OFFVECT opens the TRC order database file and extracts the original stored source-receiver offsets to place in the offset vector used for the radial trace inversion.

IMPORTANT: In order to use the new option, the original X/T data, with correct geometry in the trace headers, MUST have been processed through the ‘EXTRACT DATABASE FILES’ operation. Then the ‘2D LAND GEOMETRY SPREADSHEET’ must be used to establish the basic geometry parameters and finalize the database. Finally, the ‘INLINE GEOM HEADER LOAD’ must be applied to the original X/T data file to create a new output file whose trace headers agree with those stored in the database. This new file, when subjected to a ‘DISK DATA INPUT’ should yield the message “Geometry loaded in the trace headers matches the database: Trace numbers can be used to reference the database”.

During the Extract Database Files operation, the parameter options will typically be:

- Data type – land
- Source index method - FFID
- Receiver index method – STATIONS
- Mode of operation – OVERWRITE
- Pre-geometry extraction – YES

Other choices may work, but have not been tested.

The 2D Land Geometry Spreadsheet is beyond the scope of this document, but is relatively easy to use to finalize the database.

For the Inline Geom Header Load, the key parameter is the first one: “Match by valid trace number” should be set to ‘Yes’, all others can be defaulted.

NOTE: Although the ‘offset’ trace headers in the trace ensembles output from the inverse radial trace transform will be correct, the other headers will need to be restored using the ‘Inline Geom Header Load’ before using the ensembles in further processing, since the transform to the radial trace domain essentially destroys almost all geometry and trace-specific header information. Specifically, the operation ‘Trace header math’ should be applied immediately after the RT inverse, in the ‘sequence renumber mode’, with ‘traces’ selected for renumbering, and ‘recording channel number’ chosen as the header to restore, with ‘1’ as the starting number, ‘1’ as the increment, and ‘SIN’ as the ‘reset header’.

The ‘Inline geom header load’ operation should then be applied with ‘SIN’ selected as the ‘primary header to match’. The complete set of original trace headers from the database will be restored to each trace ensemble, matched by SIN and CHAN.

Presumably, the operations described above for preparing the database and input data file are similar, but more extensive for 3D data (but have not been tested as of 30June18). One 3D complication, that also has not been tested, is that the ‘offset’ trace headers typically placed in the traces of 3D source ensembles are positive values only, whereas the radial trace transform requires signed source-receiver offsets, with the sign indicating the relation of a receiver position to the source position projected normally to the receiver line. The CREWES operation RAD3D, applied to a raw 3D source gather prepares the proper ‘offset’ trace headers, and presumably must be used before the ‘Extract Database

Files’ operation. ***All 3D operations remain untested at this time, however.***

For those dealing only with 2D data, or those desiring to test the other offset interpolations offered with the inverse radial trace transform, it is unnecessary to perform the database operations described above, as long as the user is aware that any offsets appearing in the inverse radial trace transform output are not the original offsets (though they may be arbitrarily close in value for data with very regular 2D geometry.
The other trace header values will generally be incorrect, as well, and can only be restored from a database, if one has been prepared. The ‘Trace header math’ and ‘Inline geom header load’ operations described above can be used to accomplish this, but only if a database actually exists.

=head2 Theory

The radial trace transform is a re-mapping of the normal X-T seismic domain with co-ordinates of source-receiver offset and two-way travel time into a domain whose co-ordinates are apparent velocity and two-way travel time. Traces in this domain all share the same X-T origin and hence are "radial" with respect to that origin (often the shot origin).

Because the radial transform has the same time co-ordinate as the original X-T domain, the transform operation can be posed as a simple interpolation of trace samples from X-T time slices to R-T time slices. A major effect of re-mapping seismic data into the R-T domain is that linear events which have apparent velocity and origin in common with those of radial trace trajectories have their apparent frequencies dramatically lowered in the radial domain; while events, such as reflections, which do not share apparent velocity and origin with any radial traces, are unaffected.

An experimental mode of the radial transform is available in this module. Since the samples of the radial traces are extracted from the original X-T panel by interpolation, the spatial direction of the interpolation greatly influences which portions of the wavefield are best represented in the radial domain. The standard radial transform interpolates samples to the radial traces horizontally in the iso-time direction, which best preserves horizontal reflection-like events. The experimental radial transform interpolates samples in the iso-velocity direction from the original X-T traces to the radial traces, which best preserves events parallel to the velocity direction (like linear noise). Because of this interpolation direction, even linear events which are spatially aliased can be at least partially recovered, if they are precisely aligned with the radial trace trajectories. Because this form of the radial transform does not preserve horizontal events, it should be used WITH CAUTION. The V-interpolated radial transform mode is invoked by selecting V-interpolation for the 'Interpolation method' parameter.

```
=head2 Usage

RADTRAN is intended to be applied to seismic trace ensembles
representing shot gathers, receiver gathers, or CDP gathers,
although other ensembles are also possible to process.  The one
requirement for any ensemble to be processed by RADTRAN is that it
have legitimate offset (or velocity)
values in the 'OFFSET' trace header field for each trace, and that
all ensembles be sorted according to 'ascending' OFFSET value.  Any
panel of seismic traces can be made to conform to this standard by
appropriate use of 'trace header math' operations, and some
operation to properly define ensembles.  The notes at the end of
this help file disclose a set of operations that will convert a
stacked section into an ensemble acceptable to RADTRAN.

The default parameters in RADTRAN have been chosen to give
reasonable results for arbitrary input, but close attention to the
parameter descriptions below can lead to better performance and more
appropriate parameter choices.

=over 4

=item transform switch

This switch chooses either the forward radial transform from the X-T
domain to the R-T domain, or the inverse transform from the R-T
domain to X-T.

=item number of traces

This is the number of traces in the transform domain...R-T for the
forward transform, X-T for the inverse. The default value is
2000...meaningful for the forward transform only. If this value is
set to 1 for the inverse transform, the number of traces in the
original X-T gather is retrieved from a trace header and used to
reconstruct an output X-T gather with the same number of traces.
Setting this parameter larger than the original number of X-T traces
causes the output X-T gather to be interpolated to a smaller offset
interval.

=item switch for dip transform

This parameter allows choice of either the conventional fan radial
trace transform mode or the dip transform mode.  If fan is chosen,
the appropriate fan description parameters are solicited by the
menu...if dip is chosen, the solicited parameters are those
appropriate for dip transform.

=item nominal transform velocity

This is the apparent velocity of the dipping linear events to be
enhanced. This parameter is only used in the dip-transform mode.

=item velocity range for dip transform
```

```
This parameter defines the total width in velocity units of the
narrow fan used in the radial dip-transform mode.  It is not used in
the fan-transform mode.

=item minimum radial trace velocity

This is the apparent velocity of the first trace of the radial fan
used to extract samples from the X-T panel. It is usually .le. 0.0.
For a transform of an entire split-spread shot gather, it is
typically the apparent velocity of the first arrivals of the
trailing spread. In the dip-filter mode, this parameter is computed
by the module. For the inverse transform, this parameter is not
used, since velocities are read from input trace headers.

=item maximum radial trace velocity

This is the apparent velocity of the last radial trace in the
transform.  It is usually .ge. 0.0 and MUST be .gt. the minimum
velocity above. For a split-spread gather, it is typically the
apparent velocity of the first arrivals of the leading spread. In
the dip-transform mode, this parameter is computed by the module.
For the inverse transform, this parameter is not used, since
velocities are read from input trace headers.

=item minimum source-receiver offset

This is the minimum offset required in the X-T panel to be generated
by the inverse transform from an input R-T panel.  It is not used in
the forward transform, since offsets are read from trace headers.
While this value may be less than the original minimum offset, no
extrapolation will occur for values less than the original value. If
this parameter and the following one are both set to zero, the
values will be extracted from the sou_x and rec_x header fields of
each input radial trace gather, where they are placed by the forward
transform.

=item maximum source-receiver offset

This is the maximum offset required in the X-T panel generated by
the inverse transform from an input R-T panel.  It is not used in
the forward transform since offsets are read from trace headers.
While this value may be greater than the original maximum offset, no
extrapolation will occur for values greater than the original value.
If this parameter and the preceding one are both set to zero, the
values will be extracted from the sou_x and rec_x header fields of
each input radial trace gather, where they are placed by the forward
transform.

=item offset increment method

This parameter determines the method for generating the set of
offsets from the maximum and minimum offsets above.  The original
choices are linear offsets, quadratic offsets, the X**2/T**2 domain
in which both offsets and travel time are quadratic, and the inverse
```

X**2/T**2 domain (ordinary X-T recovered from X**2/T**2. Offset increment is defaulted to the linear method for the radial dip mode. It is not used in the forward radial transform.

A new choice, added in 2018, is the option to place the original offset values from the input trace ensemble into the offset trace headers for the output. In order for this option to work, the database operations described above MUST have been performed on the data input to the forward transform, AND the ‘number of traces’ parameter above MUST have been set to 1. The retrieval of original offset values is now the default for this parameter, so if the database and data set have NOT been prepared as described earlier, the default must be changed to ‘linear’ or one of the other options.

=item time co-ordinate for radial trace origin

This is the two-way travel time of the APPARENT origin of the RT transform. Negative values place the origin above the time zero axis of conventionally displayed seismic traces, while positive values place it below (down into the gather). In the dip-transform mode, this parameter is computed by the module.

=item offset co-ordinate for radial trace origin

This is the offset distance of the radial trace origin from the origin of the input trace panel. Negative values are to the left of the origin, and positive values to the right. The ability to specify the co-ordinates of the radial trace origin allow the transform to be placed at any 'virtual source' point on a record in order to efficiently capture linear data events from that point. In the dip-transform mode, this parameter is computed by the module.

=item nominal offset increment

This is just the station spacing for a shot gather, shot spacing for a receiver gather, etc. This value is not used in the inverse transform, as it is computed from other parameters for the inverse.

=item time-reverse switch for X-T traces

This switch causes the input panel traces to be time-reversed before transformation to the R-T domain and restored after transformation back to X-T. Thiseffectively allows a radial trace transform to be applied as if itsorigin were below the original X-T panel at a 'virtual source'for 'back-scattered' noise.

=item interpolation method for radial transform

This parameter allows choice of the interpolation method used in the radial tracetransform and its inverse. The linear method provides the smoothest transform with the fewest artifacts, but at some cost in lateral definition. The nearest-neighbor method, on the other hand, preserves lateral definition, but leads to the introduction of high-frequency artifacts into some portions of the resulting X-T panel. The 'soft neighbor' method provides a reasonable compromise by using interpolation weights proportional to exponential powers of the distances to the two nearest input samples. The V-interpolation method linearly interpolates the input along the direction of the radial traces. This method favours linear noise modes at the expense of reflection-like events.

=item exponent for 'soft neighbor' interpolation

This parameter specifies the exponent to be used for computing the weights in the 'soft neighbor' interpolation method described above. A choice of 1 for the exponent is equivalent to linear interpolation, and the higher the exponent, the more closely the algorithm approaches 'nearest-neighbor' behaviour.

=item refractive index computation method

This version of the radial trace transform allows the generation of curved radial traces by the simple expedient of supplying a 'refractive index' value for every instant of two-way travel time. Values greater than one cause the radial traces to curve divergently outward, while values less than one cause convergent curvature. At present, there are two simple methods supplied for generating curved radial traces...linear refractive index increment or decrement, and exponential increment/decrement. The default for this parameter is to specify refractive index to be unity everywhere (straight radial traces); and this is the preferred option, as the other two are experimental at this time. DEFAULT THIS UNLESS YOU KNOW WHAT YOU'RE DOING! Added in April 2000 is a new method, which computes refractive index from a layercake velocity model. If this method is chosen, an edit window appears in the menu, allowing editing of a default rms velocity function of up to 20 time/velocity pairs. The rms function is converted to interval velocities, optionally smoothed, and converted to refractive index by dividing by the initial velocity value.

=item start time for refractive index computation

This is the travel time at which either the linear or exponential methods of refractive index computation pass through unity.

=item starting value for refractive index

This is the value used as the beginning value for either method of refractive index computation.

=item refractive index exponent or slope

This is the exponent used to generate refractive index with the exponential method, or the slope in percent increment per second for the linear method.

=item velocity function

This is a character string of up to 128 characters representing up to 20 pairs of time/velocity values in standard velocity function format (time1-vel1,time2-vel2,...time20-vel20/), which may be replaced or edited in the edit window presented to the user. This parameter only appears if refractive index computation method (above) is set to 'velocity'.

=item smoothing length

This parameter only appears when the 'velocity' option is chosen for refractive index computation method. The default value of 1 leads to no smoothing of the interval velocity function, while any other value causes smoothing by a running average whose length is specified by this parameter.

=back

=head2 General

When displaying a radial trace transform consisting of several hundred or several thousand traces, the grayscale mode of display available in trace display may be more appropriate than wiggle trace.

In order to compute the inverse transform of a radial dip transform, the exact minimum and maximum offsets of the original panel must be supplied to the respective parameters, as well as the same dip velocity and velocity range. If these parameters are not exact, the internal computations will mis-position the transform origin, and the inverse will be incorrect. There are, however, circumstances in which particular geometric distortions of the input data may be deliberately imposed by a creative choice of parameters both in the forward and inverse transforms. Since the forward transform automatically stores the minimum and maximum offsets for each gather in the radial trace headers sou_x and rec_x, respectively, setting the 'Minimum offset' and 'Maximum offset' parameters both to zero for the inverse transform causes these header values to be substituted automatically. In addition, the cdp of the first input X-T trace is retrieved, the cdp increment computed, and cdp numbers computed for the new output X-T gather.

REMEMBER: The preceding paragraph does not apply if the option chosen for ‘offset increment method’ is ‘Original offsets’...in that case, the database and data set must have been prepared as described earlier.

In order to apply a radial transform to a panel of stacked traces, it is necessary to make the panel into an ensemble (shot gather, usually), complete with certain essential trace headers. The following sequence of ProMAX operations can be used to easily accomplish this, requiring only that the user supply the nominal CDP spacing in metres, xx, and the total number of CDP traces in the panel, nn:

```
    'Trace Header Math' -- sequence renumber mode; ensembles; SEQNO;
    start value = 1; increment value = 1
    'Trace Header Math' -- fixed equation mode; SIN = 1
```

```
    'Trace Header Math' -- fixed equation mode; REC_SLOC = SEQNO
    'Trace Header Math' -- fixed equation mode; TRACENO = SEQNO
    'Trace Header Math' -- fixed equation mode; OFFSET =
    FLOAT( xx * (SEQNO - nn/2))
    'Trace Header Math' -- fixed equation mode; CDP_X = OFFSET
    'Trace Header Math' -- fixed equation mode; CDP_Y = 0.0
    'Inline Sort'       -- primary key = SIN; ascending;
    secondary key = OFFSET; ascending;
    traces per ensemble = nn;
    traces in buffer = nn;
    'Trace Header Math' -- fixed equation mode; LAST_TRC = END_ENS
```

The inline sort does not physically rearrange the traces, but is used only to remove the end_of_ensemble flags from the stacked traces and re-flag only the last trace in the panel, thus re-defining the ensemble.

A macro has been constructed to accomplish the above tasks. It is called 'Convert stack to ensemble'; and it essentially adds the above ProMAX tools to a flow, supplying the three requested parameters. This basically turns a stack panel into a split-spread shot ensemble.

=head2 References

D.C. Henley -- henley@crewes.org