

## On non-stationary convolution and inverse convolution

James Rickett<sup>1</sup>

*keywords: helix, linear filtering, non-stationary deconvolution*

### ABSTRACT

Recursive inverse filtering with non-stationary filters is becoming a useful tool in a range of applications, from multi-dimensional inverse problems to wave extrapolation. I formulate causal non-stationary convolution and combination and their adjoints in such a way that it is apparent that the corresponding non-stationary recursive filters are true inverse processes. Stationary recursive inverse-filtering is stable if, and only if, the filter is minimum-phase. I show that recursive inverse-filtering with a filter-bank consisting of minimum-phase two-point filters is also unconditionally stable. However, I demonstrate that, for a more general set of minimum-phase filters, stability of non-stationary recursive inverse-filtering is not guaranteed.

### INTRODUCTION

In the past, applications of non-stationary inverse filtering by recursion have been limited to problems in 1-D, such as time-varying deconvolution (Claerbout, 1998a). Theory presented no way of extending polynomial division to higher dimensions.

With the development of the helical coordinate system (Claerbout, 1998b), recursive inverse filtering is now practical in multi-dimensional space. Non-stationary, or adaptive (Widrow and Stearns, 1985), recursive filtering is now becoming an important tool for preconditioning a range of geophysical estimation (inversion) problems (Fomel et al., 1997; Clapp et al., 1997; Crawley, 1999), and enabling 3-D depth migration with a new breed of wavefield extrapolation algorithms (Fomel and Claerbout, 1997; Rickett et al., 1998; Rickett and Claerbout, 1998).

With these applications in mind, it is important to understand fully the properties of non-stationary filtering and inverse-filtering. Of particular concern is the stability of the non-stationary operators.

---

<sup>1</sup>email: james@sep.stanford.edu

## THEORY

### Stationary convolution and inverse convolution

The convolution of a vector,  $\mathbf{x} = (x_0 \ x_1 \ x_2 \ \dots \ x_{N-1})^T$ , with a causal filter,  $\mathbf{a} = (a_0 \ a_1 \ a_2 \ \dots \ a_{N_a-1})^T$ , whose first element,  $a_0 = 1$ , and whose length,  $N_a < N$ , onto an output vector,  $\mathbf{y} = (y_0 \ y_1 \ y_2 \ \dots \ y_{N-1})^T$ , can be defined by the set of equations:

$$y_k = x_k + \sum_{i=1}^{\min(N_a-1, k-1)} a_i x_{k-i}. \quad (1)$$

This can be rewritten in linear operator notation, as  $\mathbf{y} = \mathbf{A}\mathbf{x}$ , where  $\mathbf{A}$  is a lower-triangular Toeplitz matrix representing convolution with the filter,  $\mathbf{a}$ .

The adjoint operator,  $\mathbf{A}'$ , which describes time-reversed filtering with filter  $\mathbf{a}$ , can similarly be expressed by considering the rows of the matrix-vector equation,  $\mathbf{x} = \mathbf{A}'\mathbf{y}$ , as follows,

$$x_k = y_k + \sum_{i=1}^{\min(N_a-1, N-1-k)} a'_i y_{k+i}. \quad (2)$$

The `helicon` Fortran90 module (Claerbout, 1998a) exactly implements the linear operator (and adjoint) pair described by equations (1) and (2).

Equation (1) explicitly prescribes internal boundary conditions near  $k = 0$ ; however, since  $\mathbf{a}$  is causal, no particular care is needed near  $k = N - 1$ . On the other hand, equation (2) explicitly imposes internal boundary conditions near  $k = N - 1$ , and no care is needed near  $k = 0$ . It is possible to rewrite equations (1) and (2) in a more symmetric form; however, as written, the equations lead naturally to recursive inverses for operators,  $\mathbf{A}$  and  $\mathbf{A}'$ .

Rearranging equation (1), we obtain

$$x_k = y_k - \sum_{i=1}^{\min(N_a-1, k-1)} a_i x_{k-i}. \quad (3)$$

This provides a recursive algorithm, starting from  $x_0 = y_0$  for solving the system of equations,  $\mathbf{y} = \mathbf{A}\mathbf{x}$ . Equation (3) describes the exact, analytic inverse of causal filtering with equation (1). In principle, given a filter,  $\mathbf{a}$ , and a filtered trace,  $\mathbf{y}$ , the above equation can recover the unfiltered trace,  $\mathbf{x}$  exactly; although in practice, with numerical errors, the division may become unstable if  $\mathbf{a}$  is not minimum phase. Similarly, if we inverse filter a trace with equation (3), we can recover the original by causal filtering with equation (1) subject to the stability of the inverse filtering process.

Equation (3) appears very similar to polynomial division. However, the output of polynomial division is an infinite series, while equation (3) is defined only in the range,  $0 \leq k \leq N - 1$ . As such, equation (3) describes polynomial division followed by truncation.

Equation (2) can also be rewritten as

$$y_k = x_k - \sum_{i=1}^{\min(N_a-1, N-1-k)} a'_i y_{k+i}, \quad (4)$$

which provides an exact recursive inverse to adjoint operator,  $\mathbf{A}'$ , that can be computed starting from  $y_{N-1} = x_{N-1}$ , and decrementing  $k$ .

### Non-stationary convolution and combination

There are several possible approaches to generalizing convolution described by equation (1) to deal with non-stationarity. The simplest approach (Yilmaz, 1987) is to apply multiple stationary filters and interpolate the results. This approach, however, gives incorrect spectral response in the interpolated areas (Pann and Shin, 1976).

Following Claerbout (1998a) and Margrave (1998), I extend the concept of a filter to that of a filter-bank, which is a set of  $N$  filters: one filter for every point in the input/output space. I identify  $\mathbf{a}_j$  with the filter corresponding to the  $j^{\text{th}}$  location in the input/output vector, and the coefficient,  $a_{i,j}$ , with the  $i^{\text{th}}$  coefficient of the filter,  $\mathbf{a}_j$ .

Margrave (1998) describes two closely related alternatives which both reduce to normal convolution in the limit of stationarity. The first approach is to place the filters in the columns of the matrix,  $\mathbf{A}$ . This associates a single filter with a single point in the output space, and defines *non-stationary convolution*:

$$y_k = x_k + \sum_{i=1}^{\min(N_a-1, k-1)} a_{i, (k-i)} x_{k-i}. \quad (5)$$

In contrast, the second approach is to place individual filters in the rows of the matrix,  $\mathbf{A}$ , associating a single filter with a single point in the input space. This defines what Margrave (1998) refers to as *non-stationary combination*:

$$y_k = x_k + \sum_{i=1}^{\min(N_a-1, k-1)} a_{i, k} x_{k-i}. \quad (6)$$

The advantage of non-stationary convolution over non-stationary combination is that the response of equation (5) to an impulse at the  $j^{\text{th}}$  element of  $\mathbf{x}$ , is  $\mathbf{a}_j$ . A more general output is then a scaled superposition of filter-bank filters, which fits with Green's function theory for linear, constant coefficient, partial differential equations. Again, in contrast, the response of equation (6) to an impulse at the  $j^{\text{th}}$  element of  $\mathbf{x}$ , is the  $j^{\text{th}}$  column of non-stationary combination matrix, which bears no direct relationship to the filter,  $\mathbf{a}_j$ , or any other individual filter for that matter.

As an illustration, consider the differences between matrices,  $\mathbf{F}_{\text{conv}}$  and  $\mathbf{F}_{\text{comb}}$  below, which represent, respectively, non-stationary convolution and combination with a causal three-point

( $N_f = 3$ ) filter-bank,  $\mathbf{f}$ , to vectors of length,  $N = 5$ . The two are equivalent in the stationary limit; however, while the columns of  $\mathbf{F}_{\text{conv}}$  contain filters,  $\mathbf{f}_j$ , the columns of  $\mathbf{F}_{\text{comb}}$  do not.

$$\mathbf{F}_{\text{conv}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ f_{10} & 1 & 0 & 0 & 0 & 0 \\ f_{20} & f_{11} & 1 & 0 & 0 & 0 \\ 0 & f_{21} & f_{12} & 1 & 0 & 0 \\ 0 & 0 & f_{22} & f_{13} & 1 & 0 \\ 0 & 0 & 0 & f_{23} & f_{14} & 1 \end{bmatrix}, \quad \mathbf{F}_{\text{comb}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ f_{11} & 1 & 0 & 0 & 0 & 0 \\ f_{22} & f_{12} & 1 & 0 & 0 & 0 \\ 0 & f_{23} & f_{13} & 1 & 0 & 0 \\ 0 & 0 & f_{24} & f_{14} & 1 & 0 \\ 0 & 0 & 0 & f_{25} & f_{15} & 1 \end{bmatrix}$$

It is also clear that while  $\mathbf{F}_{\text{conv}}$  and  $\mathbf{F}_{\text{comb}}$  are related, they are not simply adjoint to each other.

### Adjoint non-stationary convolution and combination

The adjoint of non-stationary convolution can be written as

$$x_k = y_k + \sum_{i=1}^{\min(N_a-1, N-k-1)} a'_{i,k} y_{k+i}, \quad (7)$$

and the adjoint of non-stationary combination can be written as

$$x_k = y_k + \sum_{i=1}^{\min(N_a-1, N-k-1)} a'_{i,(k+i)} y_{k+i}. \quad (8)$$

For many applications, the adjoint of a linear operator is the same operator applied in a (conjugate) time-reversed sense. For example, causal and anti-causal filtering, integration, differentiation, upward and downward continuation, finite-difference modeling and reverse-time migration etc.

For non-stationary filtering, it is important to realize this is *not* the case: the adjoint of non-stationary convolution is time-reversed non-stationary combination, and vice-versa. Therefore, the output of adjoint combination is a superposition of scaled time-reversed filters,  $\mathbf{a}'_j$ . So for anti-causal non-stationary filtering, it may be advantageous to apply adjoint combination, as opposed to adjoint filtering.

### Inverse non-stationary convolution and combination

As with the stationary convolution described above, formulae for non-stationary recursive inverse convolution and combination follow simply by rearranging the equations in (5) and (6). Similarly, their adjoints can be obtained by rearranging the equations in (7) and (8). The recursive formulae describing these inverse processes are given in Table 1.

Inverse NS convolution:	$x_k = y_k - \sum_{i=1}^{\min(N_a-1, k-1)} a_{i, (k-i)} x_{k-i}$	(9)
Inverse NS combination:	$x_k = y_k - \sum_{i=1}^{\min(N_a-1, k-1)} a_{i, k} x_{k-i}$	(10)
Adjoint inverse NS convolution:	$y_k = x_k - \sum_{i=1}^{\min(N_a-1, N-1-k)} a'_{i, k} y_{k+i}$	(11)
Adjoint inverse NS combination:	$y_k = x_k - \sum_{i=1}^{\min(N_a-1, N-1-k)} a'_{i, (k+i)} y_{k+i}$	(12)

Table 1: Recursive formulae for non-stationary (NS) inverse operators.

As with the stationary inverse convolution described above, it is apparent that subject to numerical errors, non-stationary inverse filtering with these equations in Table 1 is the exact, analytic inverse of non-stationary filtering with the corresponding forward operator described in equations (5) through (8): they are true inverse processes. If operator  $\mathbf{A}$  represents filtering with a non-stationary causal-filter, and  $\mathbf{B}$  represents recursive inverse filtering with the same filter then

$$\mathbf{AB} = \mathbf{BA} = \mathbf{I} \quad \text{and} \quad \mathbf{A}'\mathbf{B}' = \mathbf{B}'\mathbf{A}' = \mathbf{I}.$$

The `nhelicon` module (Claerbout, 1998a) implements the non-stationary combination operator/adjoint pair, described by equations (6) and (8), while `npolydiv` implements the corresponding inverse operators, described by equations (10) and (12).

### The stability of non-stationary inverse filtering

A filter is stable if any bounded input produces a bounded output (Robinson and Treitel, 1980). Therefore, to prove that inverse filtering with a class of filters is stable, we have to demonstrate that all possible members of the class have bounded outputs for all bounded inputs. On the other hand to show that a class of filters is not stable, we just need to find a single example where a bounded input produces an unbounded output.

The stability of stationary recursive inverse filtering depends on the phase of the causal filter: if (and only if) the filter is minimum phase, then its inverse filter is stable. This raises the question: is non-stationary inverse filtering stable if all filters contained in the filter-bank are minimum-phase?

For the case of inverse filtering with a two-point filter ( $N_a = 2$ ), equation (9) reduces to  $x_0 = y_0$ , and the following formula for  $k > 0$ :

$$x_k = y_k - a_{1, (k-1)} x_{k-1}. \quad (13)$$

Recursive substitution then produces an explicit formula for elements of  $\mathbf{x}$  in terms of elements of  $\mathbf{y}$ :

$$x_N = y_N - a_{1, (N-1)} y_{N-1} + a_{1, (N-1)} a_{1, (N-2)} y_{N-2} \dots + (-1)^N \left( \prod_{i=0}^{N-1} a_{1, i} \right) y_0. \quad (14)$$

For stability analysis, we need to understand how the above series behaves as  $N \rightarrow \infty$ . If the filters,  $\mathbf{a}_i$ , are all minimum phase, and there exists a real number,  $\kappa$ , such that  $|a_{1,i}| \leq \kappa < 1$  for all  $i$ , then

$$\prod_{i=0}^{N-1} a_{1,i} \leq \kappa^N. \quad (15)$$

The above series will therefore converge, and stability is guaranteed. Furthermore, this proof can easily be extended to gapped two-point minimum-phase non-stationary filters, which correspond to matrices with ones on the main diagonal, and variable coefficients whose magnitude is less than one on a secondary diagonal.

There is a larger class ( $N_a \geq 3$ ) of stable non-stationary recursive filters that can be obtained by repeatedly multiplying stable bidiagonal matrices. However, given a general non-stationary filter matrix, there is no straightforward way to determine whether it is a member of this stable class. In fact, it is relatively easy to find an example filter-bank consisting of minimum-phase individual filters whose recursive output is unbounded for finite input. Consider the filter-bank,  $\mathbf{f}$ , consisting of minimum-phase filters,

$$\begin{aligned} \mathbf{f}_{0,2,4\dots} &= (1 \ -0.9 \ 0), \text{ and} \\ \mathbf{f}_{1,3,5\dots} &= (1 \ 0.8) * (1 \ 0.8) = (1 \ 1.6 \ 0.64). \end{aligned} \quad (16)$$

Figure 1 shows the impulse response of non-stationary inverse filtering with this filter: clearly an unstable process.

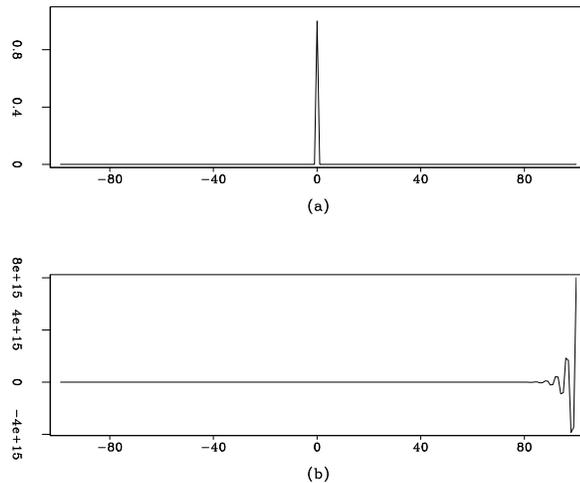


Figure 1: Impulsive input (a) and response (b) to non-stationary filtering with filter-bank defined in equation (16). james1-one [ER]

The instability stems from the fact that as  $N$  increases, so does the number of boundaries between different filters. Such rapid non-stationary filter variations, as in the example above, are pathological in the context of seismic applications, where filters are typically quasi-stationary. For these applications instability is rarely observed; however, we must be aware that we are not dealing with an unconditionally stable operator, and instability may rear its ugly head from time-to-time.

## CONCLUSIONS

There are three important points to draw from this paper. Firstly, I have formulated causal non-stationary convolution and combination and their adjoints in such a way that it is apparent that the corresponding non-stationary recursive filters are true inverse processes. If you think of causal non-stationary filtering as a lower triangular matrix, then recursive inverse filtering applies the inverse matrix.

The second important point is that recursive inverse-filtering with a filter-bank consisting of minimum-phase two-point filters is *unconditionally stable*, and as such it is totally safe to apply in any circumstance.

However, the final point is that for a more general set of minimum-phase filters, stability of non-stationary recursive inverse-filtering is *not* guaranteed: use with care.

## REFERENCES

- Claerbout, J. F., 1998a, Geophysical estimation by example: available on the World-Wide-Web at <http://sepwww.stanford.edu/prof/gee/>.
- Claerbout, J. F., 1998b, Multidimensional recursive filters via a helix: *Geophysics*, **63**, 1532–1541.
- Clapp, R. G., Fomel, S., and Claerbout, J., 1997, Solution steering with space-variant filters: *SEP-95*, 27–42.
- Crawley, S., 1999, Interpolation with smoothly nonstationary prediction-error filters: *SEP-100*, 181–196.
- Fomel, S., and Claerbout, J. F., 1997, Exploring three-dimensional implicit wavefield extrapolation with the helix transform: *SEP-95*, 43–60.
- Fomel, S., Clapp, R., and Claerbout, J., 1997, Missing data interpolation by recursive filter preconditioning: *SEP-95*, 15–25.
- Margrave, G. F., 1998, Theory of nonstationary linear filtering in the Fourier domain with application to time-variant filtering: *Geophysics*, **63**, no. 1, 244–259.
- Pann, K., and Shin, Y., 1976, A class of convolutional time-varying filters: *Geophysics*, **41**, no. 1, 28–43.
- Rickett, J., and Claerbout, J., 1998, Helical factorization of the Helmholtz equation: *SEP-97*, 353–362.
- Rickett, J., Claerbout, J., and Fomel, S., 1998, Implicit 3-D depth migration by wavefield extrapolation with helical boundary conditions: *SEP-97*, 1–12.
- Robinson, E. A., and Treitel, S., 1980, *Geophysical signal analysis*: Prentice-Hall.

Widrow, B., and Stearns, S. D., 1985, Adaptive signal processing: Prentice Hall.

Yilmaz, O., 1987, Seismic data processing: Soc. Expl. Geophys.

