Stanford Exploration Project, Report 103, April 27, 2000, pages 205–312

Stanford Exploration Project, Report 103, April 27, 2000, pages 205–312

Short Note

The Burg Method on a Helix?

Jon Claerbout[1]

We have not yet put the Burg method of PEF estimation on a helix. The first reason to do so is that the Burg method assures us a stable PEF. The second reason to do so is that the Burg method should be much faster than conjugate gradients. For data length ND and filter length NF, the PEF estimation costs are

```
Conjugate gradients    ND*NF**2
Levinson               ND*NF + NF**2
Burg                   ND*NF
```

Estimating PEF's on a helix with the Burg method does not seem difficult: Terms in sums that involve missing data can simply be omitted from certain averages. We could probably proceed much as we now do with conjugate gradients (CG).

PEF estimation is not our main problem, however. Our main problem is missing data. The Burg method has not yet been adapted to missing data estimation but we should try.

It remains to be seen how we can estimate missing values, both off the ends of the data and internal to it. As with CG, polynomial division seems to be an important part of the solution.

BURG PEF ESTIMATION REVIEW

Burg PEF estimation should work fine on a helix. Full details along with the 1-D code are found at (Claerbout, 1976). I will quickly review the theory from memory (partly to see how simple I can make it).

First is the notion that PEFs can be built up from this recursion

$$A^{N+1}(Z) \quad = \quad A^N(Z) + cZ^{N+1}A^N(1/Z) \tag{1}$$

where c is in the range $-1 \leq c \leq +1$. There is a theorem from Algebra that is easy to prove that if c is in the required range, then $A^{N+1}(Z)$ will be minimum phase if $A^N(Z)$ is minimum phase. Since $A^0 = 1$, all are minimum phase.

Burg's PEF calculation begins from two copies of the data $X(Z)$. One, called $F(Z)$ will be turned into the forward prediction error $A(Z)X(Z)$. The other called $B(Z)$ will be turned

[1]**email:** claerbout@stanford.edu

into the backward prediction error $A(1/Z)X(Z)$. At each stage of the calculation, we compute c with this formula

$$c \quad = \quad -2\,\frac{b\cdot f}{b\cdot b + f\cdot f} \tag{2}$$

The triangle inequality shows that for arbitrary f and b, c is in the required range. Given c we now form an upgraded F and B with $f \longleftarrow f + cb$ and $b \longleftarrow b + cf$. [At successive iterations, increasing time lag is introduced between f and b. Details in (1976).] When you are finished, you have $F = A(Z)X(Z)$ and $B = A(1/Z)X(Z)$. Why is that? To understand that requires delving into 1-D theory, in particular, the Levinson recursion, and we won't do that now. [Maybe I can think up an easier explanation later. Perhaps by a sequence of orthogonality arguments.] I recall if you append a tiny impulse function off the end of X before you start, when you finish, you will see it has turned into the PEF.

Now let us think about missing data off the ends of the Burgian one-dimensional data set. Given that we have computed $F(Z) = A(Z)X(Z)$, then we should find that $F(Z)/A(Z)$ matches $X(Z)$ until its end, and it is a logical continuation thereafter. Likewise $B(Z)$ could be used for extensions before the beginning of $X(Z)$. Thus it remains to think about how to handle gaps in the middle.

BURG PEF ESTIMATION ON A HELIX

Now, how does the Burg method fit on a helix? There is nothing new except for the huge gap while we wind around the back of the helix. In this gap, we would simply presume $c = 0$ and we do nothing there. We compute the PEF and the prediction error simply by omitting steps that we would ordinarily do.

If, however, we intend to use the PE filter, then we have some details to attend to, and this begins to get complicated. Reviewing the Levinson recursion, we find that gaps internal to the filter tend to fill as the recursion proceeds. The filter is not as sparse as the reflection coefficients. We'll need to keep track of the nonzero filter coefficients. We need to keep track of them in order that we have a PEF that we can use in polynomial division because polynomial division is an essential part of finding missing data with the Burg method and polynomial division is a part of preconditioning the conjugate-gradient method.

A promising thought is that perhaps the Burg recursion can be run backwards. Since this would take a PEF (or its reflection coefficients) and white inputs (forward and backward prediction errors) and create a colored outputs, it seems analogous to polynomial division.

When I began multidimensional filtering studies I was ignorant of the helix and thus had not the opportunity to use the Burg or the Levinson methods. Stability was not an issue until we began to do preconditioning using polynomial division.

CONTINUOUSLY VARIABLE PEF'S

Inevitably, we get involved with nonstationarity and we become interested in continuously variable PEF's. Again, 1-D theory guides us. We can do regional averaging of c's and that preserves minimum phase. Likewise, we can independently average across micropatches the numerator and denominator of (2). Many people did this in the old days when filter theory was basically one dimensional.

Polynomial division by nonstationary PEFs made up from an assemblage of stationary ones need not, however, be necessarily stable, as shown by (Rickett, 1999). He demonstrated instability when two stable PEFs alternated at alternate time points. Whether this kind of instability would arise in practice remains to be seen.

ACTION ITEMS

Regretably, I do not recognize any immediately manageable action items. Any Burg-helix PEF estimation method must include a method for polynomial division or it cannot provide fast solutions to the missing data problem. Even with a polynomial division method, we'll still have some thinking to do.

WEB REFERENCE

```
http://sepwww.stanford.edu/sep/jon/trash/helburg/                (This paper)
http://sepwww.stanford.edu/sep/prof/fgdp/c7.ps.gz                (FGDP chapter)
http://sepwww.stanford.edu/sep/prof/fgdp/c7/paper_html/node3.html (html)
```

ACKNOWLEDGEMENT

I'd like to acknowledge helpful conversations with Sergey Fomel, James Rickett, Francis Muir, and John Burg.

REFERENCES

Claerbout, J. Fundamentals of Geophysical Data Processing:. http://sepwww.stanford.edu/sep/prof/, 1976.

Rickett, J., 1999, On non-stationary convolution and inverse convolution: SEP–**102**, 129–136.