Biochimica et Biophysica Acta, 1023 (1990) 305–311
Elsevier

BBAMEM 74785

Fractal filtering of channel data

Roger R. Lew * and Charles L. Schauf

Indiana University - Purdue University at Indianapolis, School of Science, Department of Biology, Indianapolis, IN (U.S.A.)

(Received 16 August 1989)

Key words: Fractal dimension; Fractal filtering; Ion channel; Single-channel data

The fractal dimension of subsets of time series data can be used to modulate the extent of filtering to which the data is subjected. In general, such fractal filtering makes it possible to retain large transient shifts in baseline with very little decrease in amplitude, while the baseline noise itself is markedly reduced (Strahle, W.C. (1988) Electron. Lett. 24, 1248–1249). The fractal filter concept is readily applicable to single channel data in which there are numerous opening/closing events and flickering. Using a simple recursive filter of the form: $Y_n = w \cdot Y_{n-1} + (1 - w)X_n$, where X_n is the data, Y_n the filtered result, and w is a weighting factor, $0 < w < 1$, we adjusted w as a function of the fractal dimension (D) for data subsets. Linear and ogive functions of D were used to modify w. Of these, the ogive function: $w = [1 + p^{(1.5-D)}]^{-1}$ (where p affects the amount of filtering), is most useful for removing extraneous noise while retaining opening/closing events.

Introduction

The removal of unwanted noise in time series data is often necessary to uncover details in the data. It is common to use low pass filtering, either analog or digital, to remove noise. Unfortunately, low pass filtering can cause a decrease in signal amplitude and a phase shift that may result in the loss of large, rapid transients in the data. In channel data analysis, there has been a considerable effort to optimize the detectability of large transients, like opening/closing events, while minimizing baseline noise [1], to overcome this basic problem of data filtering.

To allow the retention of large amplitude transients which would be lost with typical filtering methods, Strahle [2] introduced an adaptive nonlinear filter modified by fractal geometry. This fractal filter would be quite useful in channel analysis since opening/closing events and flickering would be relatively unaffected by the filter due to the large fractal dimension of transient events. So they would be retained while baseline noise was extensively filtered. Using two techniques for weighting a digital filter as a function of the fractal dimension, we examined fractal filter behaviour and its usefulness in the analysis of single channel data.

Materials and Methods

The data chosen for analysis were examples of channel flickering from a loose cell-attached patch clamp recording [3] on the alga *Mougeotia* [4]. Clamping currents were measured with a List EPC-7 patch clamp amplifier and recorded on a video cassette recorder-based acquisition system using pulse code modulation (Model DAS-900, Dagan Corp., Minneapolis, MN 55407, U.S.A.). For frequency dependence of filtering, the data was a recording of a patch clamp electrode in solution. In both cases, the data was pre-filtered at an f_c (−3 dB gain response) of 2 kHz with an 8-pole Bessel filter (Frequency Devices, Haverhill, MA 01830, U.S.A.) and digitized at 500 μs/sample with a Labmaster analog/digital board (Scientific Solutions, Solon, OH 44139, U.S.A.) as implemented by Axon Instruments (Burlingame, CA 94010, U.S.A.) using their pClamp software. There was no significant problem with aliased high frequency noise in our data.

The fractal filter was written in the C programming language using an ANSI compatible compiler (Power C, Mix Software, Richardson, TX 75081, U.S.A.) The data was read/written in 16-integer groups from hard disk files to retain the flexibility of working with large data sets without using a large amount of computer memory. The local fractal dimensions were determined geometri-

* Permanent address: Department of Biology, York University, 4700 Keele street, North York, Canada, M3J 1P3.

Correspondence: R.R. Lew, Department of Biology, York University, 4700 Keele Street, North York, Canada, M3J 1P3.

cally [2,5]. For each 16-integer group, the number of boxes of size $\in$ required to cover the data, $N(\in)$, was determined. The fractal dimension (D), defined by:

$$D = -\frac{d \ln[N(\in)]}{d \ln[\in]} \quad (1)$$

was estimated by determining the slope of $\ln[N(\in)]$ versus $\ln[\in]$ using linear regression for $\in = 1$ to 16. This technique is a simple but rough estimate of fractal dimension in local regions of time series data.

For simplicity, the recursive filter was of the form [2,6,7]:

$$Y_n = w \cdot Y_{n-1} + (1-w) X_n \quad (2)$$

where X_n is the data, Y_n is the filtered result, and w is the weighting factor.

Other filtering techniques could also be used. The filtering window of a Gaussian non-recursive filter [1] could be narrowed or broadened as a function of the fractal dimension (see Discussion), but the calculations are time-consuming and thus difficult to implement in real time. By comparison, single coefficient recursive filtering allows rapid data manipulation and relative simplicity. Weighing of the recursive filter as a function of the fractal dimension was either linear, ogive, or inverse power, as described in Results.

To quantitatively assess the ability of the fractal filter to retain open/close events, a simulated channel burst was used. The simulation had an initial baseline (channel always closed) from which the extent of filtering was measured by calculating the standard deviation. Subsequent channel opening/closing transitions were determined by a random variate having an exponential distribution. This yields flickering quite similar to the real data shown in Fig. 1. Noise was added to the simulation with a random variate having a normal distribution (mean of 0 and standard deviation which was 25% of the open/close transition value). The random variates were generated by the SYSTAT statistical software (SYSTAT Inc., Evanston, IL 60201, U.S.A.) The generated noise was white and uncorrelated. Channel opening/closing events were scored as positive/negative transitions through the midpoint, i.e., 50% of the open/close transition value. Closed and open dwell times (which were both exponentially distributed) (only median closed times are shown) and false events were measured for the various filtering regimes.

Fourier transforms, to obtain power spectra, were performed on 1024 sample data sets using the SYSTAT statistical package. Prior to Fourier transformation, the mean and linear trends were removed from the data.

Results

The channel data chosen as an example is a worst case scenario, having both rapid flickering (at the time scale used) and high baseline noise due to low seal resistance. Under these circumstances, it is difficult to determine open times because the rapid channel closings are obscured by the high noise level. Fig. 1 shows the unfiltered channel data and calculated fractal dimensions per 16-value subset. The mean fractal dimension was 1.32 ± 0.15 ($\pm$S.D.); it increased when the channel was active. The calculated fractal dimension varies with the size of the data subset used for the calculation. For the data shown, 8-data value subsets yielded a fractal dimension of 2.58 ± 0.35, and a 32-value subset yielded a fractal dimension of 0.81 ± 0.09. In general, with larger data subsets, the fractal dimension is underestimated because the slope of $\ln[N(\in)]$ versus $\ln(\in)$ is no longer approximately linear. The use of 16-value subsets gave the most reasonable estimates of the fractal dimension, but these estimates could range beyond the expected values for D of between 1 and 2. The technique is inherently inaccurate but it is computationally simple and lends itself to estimates of local fractal dimension.

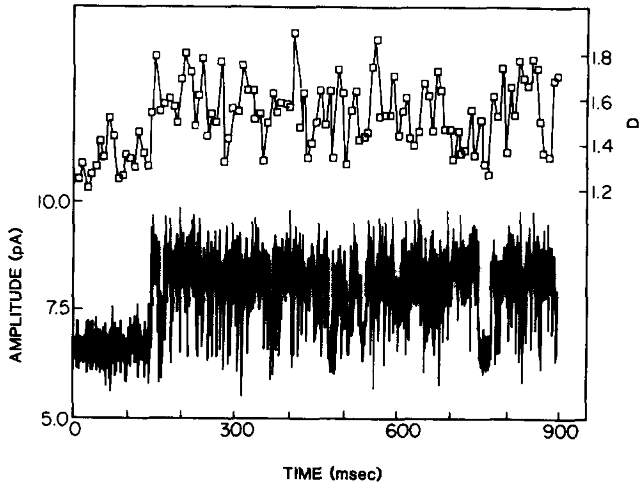

Fig. 1. Bursting activity of calcium-activated potassium channels from *Mougeotia* plasma membrane (lower trace) and fractal dimension of the channel data (upper trace). The record (lower trace) was digitized at 2 kHz after filtering with an 8-pole Bessel filter at 2 kHz. Fractal dimensions (upper trace) per 16 data value subset were calculated as described in the text.

Filtering with a recursive filter alone causes a decrease in amplitude and a loss of apparent closing events as the amount of filtering is increased (Fig. 2). Even though baseline noise is adequately removed, the loss of large transients means that open times can no longer be accurately measured.

The recursive filter is essentially a low-pass filter (Fig. 3; Ref. 6). The frequency dependence is defined by the sampling rate and the value of w:

$$|H(f)|^2 = \frac{(1-w)^2}{(1+w^2) - 2w \cdot \cos(2\pi \cdot f \cdot t)} \quad (3)$$

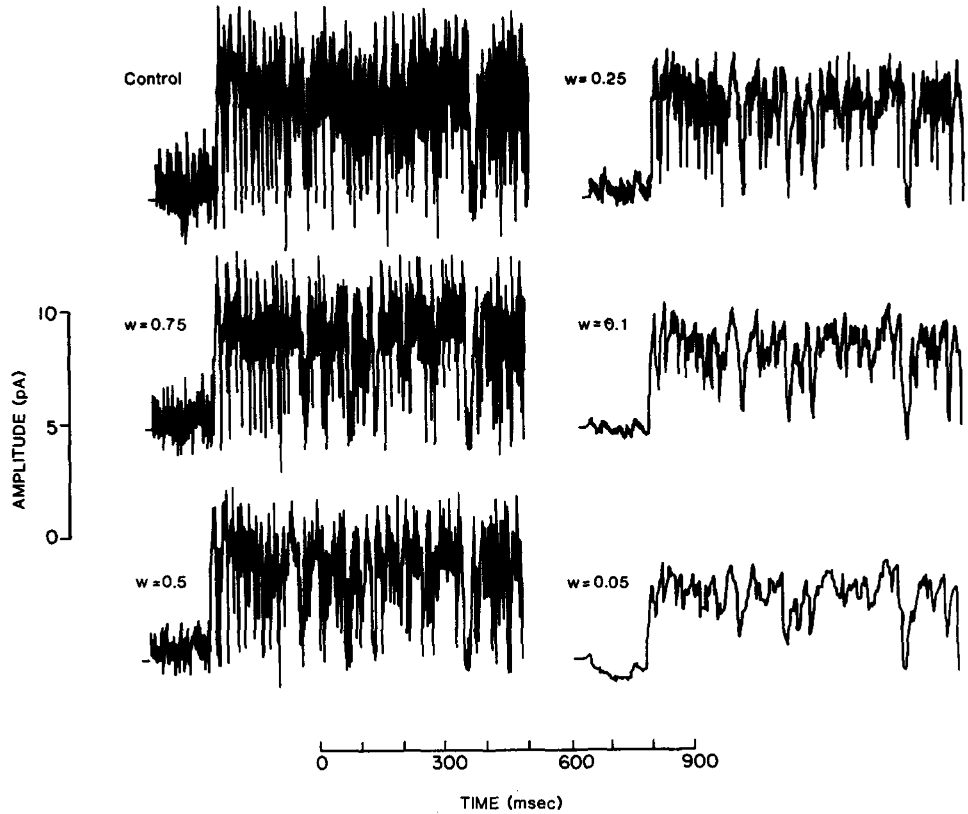

Fig. 2. Recursive filtering of channel data. The data from Fig. 1 was filtered at various values of w (which affects the cutoff frequency) using the recursive filter, $Y_n = w \cdot Y_{n-1} + (1-w) X_n$, described in the text.

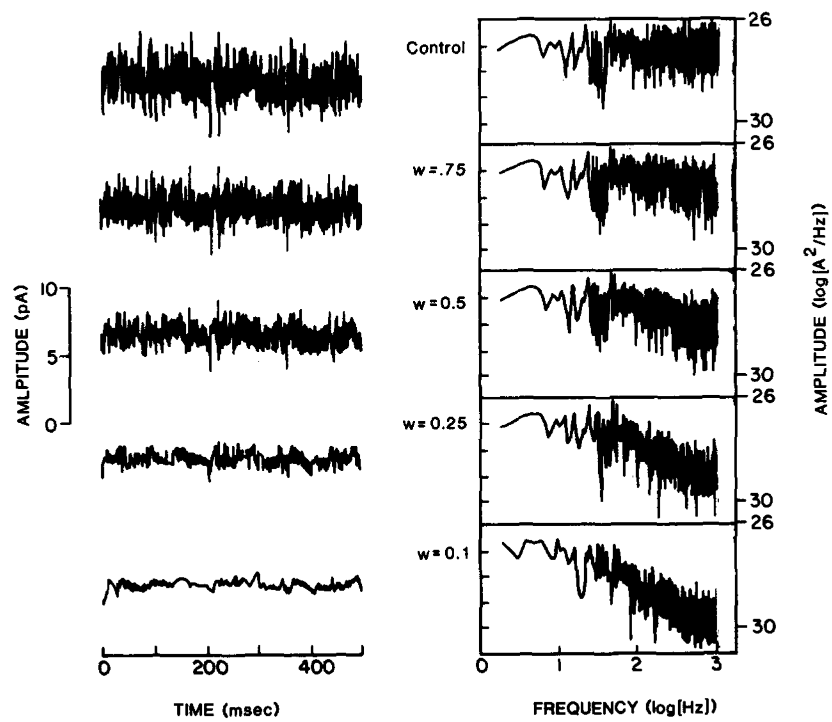

Fig. 3. Spectral dependency of recursive filtering, 2 kHz sampling. The left panels show data for an electrode in solution digitized at 2 kHz after filtering with an 8-pole Bessel filter at 2 kHz. The average fractal dimension was 1.29 ± 0.11. Samples (1024 data points) were Fourier-transformed and converted to power spectra, shown in the right panels. The weights (w) were varied from 1 (no filtering, uppermost panel) to 0.1 (lowest panel).

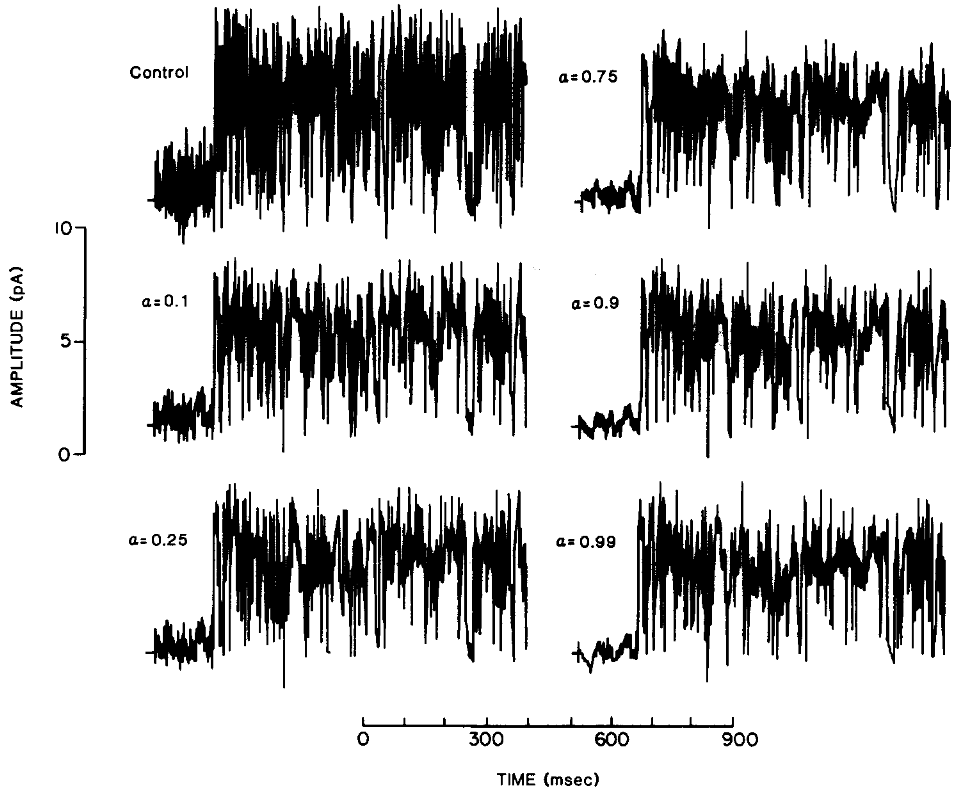

Fig. 4. Linear/fractal recursive filtering. Filtering of channel data was weighted by the fractal dimension, $w = \alpha(D - 1.5) + 0.5$, where α modulates filtering strength as shown.

where t is the time between samples, f is the sampling rate and $|H(f)|^2$ is the amplitude of the power spectra [6].

We chose to change the frequency dependence of filtering in response to changes in the fractal dimension by using two methods, linear and ogive, to modify w as a function of the fractal dimension. These two methods will give very different relationships between fractal dimension and w.

Linear / fractal recursive filter

The weighting factor was modified by:

$$w = \alpha(D - 1.5) + 0.5 \tag{4}$$

with α controlling the amount of filtering and 0.5 being the residual filtering when $D = 1.5$ (In the program, if $w < 0$, then w was set to 1 (i.e., no filtering). This corrects for instances where the calculated fractal dimension is less then 1). Compared to the recursive filter, baseline noise was noticeably less, while closing events were still retained (Fig. 4). Thus, linear/fractal filtering with $\alpha = 0.99$ (Fig. 4, last panel) decreases baseline noise to the same extent as the recursive filter at $w = 0.1$ (Fig. 2, last panel), but amplitude and closing events are retained when the recursive filter is modified by the fractal dimension.

Ogive / fractal recursive filter

The weighting factor was modified by:

$$w = [1 + p^{(1.5 - D)}]^{-1} \tag{5}$$

where p modifies the amount of filtering. There is residual recursive filtering of $w = 0.5$ when $D = 1.5$; this is an arbitrary choice which can be changed according to individual needs and applications by using some other value besides 1. The filtering behaviour is similar to the linear/fractal filter (Fig. 5). The advantage of the ogive/fractal filter is that filtering strength can be strongly modulated without causing w to move outside the range from 0 to 1. Dependent on the amount of filtering proscribed by the value of p, the relationship between fractal dimension and w varies from linear to a pronounced ogive curve.

The choice of a cutoff point of $1.5 - D$ (or $D - 1.5$ in the linear/fractal filter) is arbitrary. With the cutoff set to $1.1 - D$, the amount of filtering decreases as filtering strength is increased. When the cutoff is set to the mean fractal dimension for the data (1.3), there is very little filtering beyond that caused by the residual recursive filtering (set at $w = 0.5$) (data not shown), while a cutoff of $1.5 - D$, yields reasonable filtering with increasing filtering strength (Fig. 5). At higher filtering strengths,

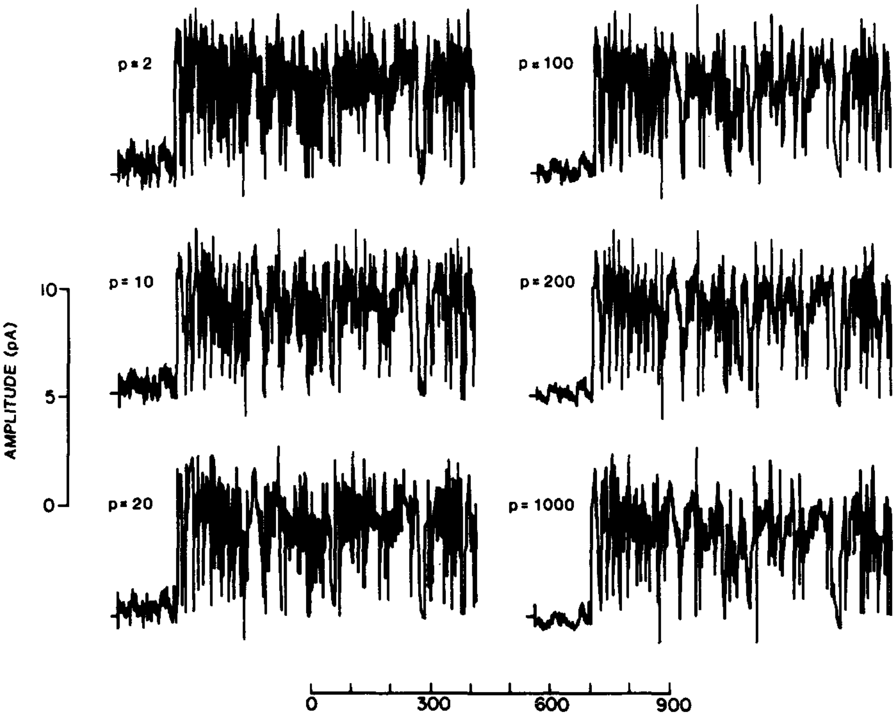

Fig. 5. Ogive/fractal recursive filtering. Filtering of channel data was weighted by the fractal dimension, $w = [1 + p^{1.5-D)}]^{-1}$, where p modulates filtering strength as shown.

the fractal weighting causes a slight reversal of the recursive filter frequency dependence (data not shown).

Power / fractal recursive filter

The weighting factor can also be modified by:

$$w = [p^{(2-D)}]^{-1} \qquad (6)$$

where p modulates the filtering strength. The cutoff has been set to 2 to avoid aberrant filter behaviour. This filter does not allow the retention of opening/closing events possible with either the ogive/fractal or the linear/fractal filters (data not shown).

Simulated data

The simulated data described in the Materials and Methods section was filtered recursively with w equal to 0.99, 0.95, 0.9, 0.75, 0.5, 0.25, or 0.1; ogive/fractal filtering was performed with residual filtering equivalent to the same values, and filtering strength (p) of 20 or 1000. The fractal cutoff point was $2.2 - D$ instead of $1.5 - D$ because the mean fractal dimension for the simulated data was 1.91.

Fig. 6 summarizes the effects of filtering: the extent of filtering, measured as the standard deviation of baseline noise, is the ordinate, and the number of closing event,s false events, and median close times are shown for the three filter types for various values of w. With recursive filtering, as the extent of filtering declines, more closing events are measured with a considerable increase in false events. The median close time (and open time) are not affected by the increase in false events in this simulated data. The noise level is consistently lower with fractal filtering, few false events are measured, and median close times are closer to the real value, even with a noise level considerably lower than that required for an accurate measure of median times with the recursive filter. In all cases, the closed and open times retain their exponential distribution (data not shown).

As an example, recursive filtering at $w = 0.1$ yields a standard deviation of 4.2, 84 closing events and median close time of 12; fractal filtering at $P = 1000$ and residual filtering of $w = 0.5$ yields a standard deviation of 2.0 and 166 closing events and median close time of 6.5. Essentially, there is 2-fold greater filtering and about 2-fold more open/close events are retained, and median close time is much closer to the real value when the fractal filter is used.

Discussion

The use of fractal geometry to modulate the filtering of channel data makes it possible to retain maximal information while optimizing baseline filtering. This is especially useful for channel bursting activity but would be equally effective for any channel data. The fractal dimension can be used to weight any digital filter, with

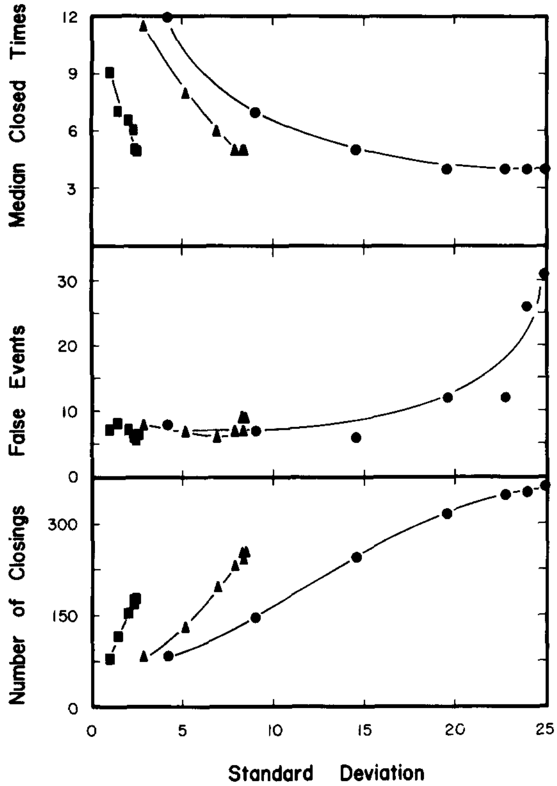

Fig. 6. Comparison of recursive and ogive/fractal filtering: noise level compared with retention of closing events, false events, and median closed time. The noise level was measured as the standard deviation of a section of the simulated time series data where the channel was inactive ($n = 499$). For the recursive filter (●), from left to right, the data points represent filtering weights, w, of 0.1, 0.25, 0.5, 0.75, 0.9, 0.95, 0.99. For the ogive/fractal filters, $p = 20$ (▲) and $p = 1000$ (■), the residual filtering weights are the same. The ordinate is given as the standard deviation to demonstrate that for the same noise level, the fractal filters measure more channel events, fewer false events, and more accurate median closed times (and open times, data not shown) than the recursive filter. The control (no filtering) values were: standard deviation, 26.03; number of closing events, 341; number of false events, 32; and median closed (and open) times: 4.

a variety of weighting functions. We have concentrated on a simple recursive filter because it is very effective when weighted by fractal dimensions, and it can be easily implemented due to its simplicity. Its bandpass filtering is modulated by the inclusion of weighting with the fractal dimension. The fractal weighting itself should be independent from frequency, but it will have a frequency response that depends upon the properties of the digital filter and, possibly, the nature of the time series data.

Linear/fractal and ogive/fractal recursive filtering yield similar results, but ogive/fractal filtering is better behaved, since w will always remain within the range 0 to 1, and it offers a variety of possible relationships between fractal dimension and w. As shown in Fig. 6, the ogive fractal filter can yield greater filtering of baseline noise (measured by the standard deviation), while retaining more open/close events compared with the recursive filter at the same noise level.

While the ogive/fractal recursive filter is a useful addition to the family of digital filters used in channel data analysis, the fractal filter concept can also be readily adapted to other digital filters. For example, fractal weighting of a gaussian filter would involve changing the width of the filtering window. For a non-recursive filter of the type:

$$Y_n = \sum_{j=-i}^{i} a_j \cdot X_{n+j} \tag{7}$$

where Y_n is the filter product, X_{n+j} the input data, and a_j the weighting coefficient [1], the values of a_j are determined by a width parameter, σ:

$$a_j = \left[(2\pi)^{1/2}\sigma\right]^{-1} \exp(-j^2/2\sigma^2) \tag{8}$$

The width parameter is defined by the desired cutoff frequency, $\sigma = 0.1325/f_c$, and can be readily modified by the inclusion of fractal weighting: $\sigma = (0.1325/f_c) \cdot [1 + p^{(D-1.5)}]^{-1}$. As is the case with the recursive filter, this would allow better retention of opening/closing events, although with considerably increased computation time.

To optimize the fractal filter for maximal noise removal and maximal retention of channel transitions requires four distinct choices: (1) The length of the time series data used to estimate local fractal dimension (For *Mougeotia* data, the 16-value subset was best, but the optimal value would presumably depend upon sampling frequency. There are alternative methods of estimating fractal dimension that are computationally simple [8] and less sensitive to sample size.) (2) A cut-off higher than the averaged fractal dimension (For *Mougeotia* data, the average was 1.32 and the cut-off was 1.5). (3) Residual filtering weight, w (a value of 0.75 was optimal; higher values were without significant effect). (4) Filtering strength, p (a value of 20 was optimal; smaller values would cause the filter to become more recursive in nature).

Acknowledgements

This study was supported by NSF grants BNS 87-15847 and DCB 87-15558 to C.L. Schauf. We would like to thank Dr. Cliff Christian of Axon Instruments for supplying us with a program for converting pClamp data files to an ASCII format, and the reviewers for their helpful comments.

References

1 Colquhoun, D. and Sigworth, F.J. (1983) in Single Channel Recording (Sakmann, B. and Neher, E., eds.), pp. 191–263, Plenum Press, New York.

2 Strahle, W.C. (1988) Electron. Lett. 24, 1248–1249.
3 Hamill, O.P., Marty, A., Neher, E., Sakmann, B. and Sigworth, F.J. (1981) Pflügers Arch. 391, 85–100.
4 Lew, R.R., Serlin, B.S., Schauf, C.L. and Stockton, M.E. (1989) Plant Physiol. 92, in press.
5 Mandelbrot, B.B. (1977) The Fractal Geometry of Nature, Freeman, New York.
6 Bendat, J.S. and Piersol, A.G. (1986) Random Data. Analysis and Measurement Procedures. 2nd Edn., pp. 365–368, John Wiley and Sons, New York.
7 Sachs, F. (1983) in Single Channel Recording (Sakmann, B. and Neher, E., eds.), pp. 265–285, Plenum Press, New York.
8 Katz, M.J. (1988) Comput. Biol. Med. 18, 145–156.