Journal of Neuroscience Methods 86 (1998) 25–34

JOURNAL OF NEUROSCIENCE METHODS

Software-based correction of single compartment series resistance errors

Stephen F. Traynelis *

Department of Pharmacology, Emory University, Room 5025 Rollins Research Center, 1510 Clifton Road, Atlanta, GA 30322-3090, USA

Received 3 February 1998; received in revised form 19 August 1998; accepted 23 August 1998

Abstract

Resistance across a patch pipette following seal formation and attainment of the whole cell recording configuration can introduce considerable and sometimes non-intuitive errors into voltage clamp recordings. These errors can be corrected actively on most commercially available amplifiers, although decisions during the experiment often must be made concerning the degree of correction to be employed, and the decision is essentially irreversible once the data is recorded. Amplifier-based corrections assume a single compartment, and thus any degree of compensation could be applied after data collection with a delay of only a single digitization interval. This report describes computer algorithms that correct capacitative filtering that results from pipette series resistance as well as the voltage error for current responses with either linear or non linear current–voltage curves. The algorithms are designed to operate on data that are recorded at a single holding potential rather than in response to voltage steps. The simplest algorithm for correction of responses with linear IV consists of about a dozen lines of C code and can easily be incorporated into data analysis programs, spreadsheets, and mathematical analysis packages. Code, sample programs, and spreadsheets that implement these algorithms are available from ftp.pharm.emory.edu. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Series resistance; Patch clamp; Voltage clamp; Electrotonic filtering; Ligand-gated ion channel

1. Introduction

Series resistance errors exist in patch clamp recordings because the current that flows through the headstage into the cell must cross the resistance of the pipette tip, which is usually enhanced from intracellular debris that becomes lodged within upon seal formation and membrane breakthrough. This resistance in the pipette tip (R_S) is in series with the membrane resistance (R_M), and therefore creates a voltage divider. There has been extensive discussion of the series resistance problem in the literature (e.g. Marty and Neher, 1995) as well as in manuals provided by manufacturers of patch clamp amplifiers. Briefly, Ohm's law dictates that, even though the imposed potential difference from the amplifier headstage to ground will be the holding potential set on the amplifier (V_{HOLD}), there will be a small voltage drop across the pipette resistance ($V_{PIPETTE}$) and a second voltage drop across the cell's membrane ($V_{MEMBRANE}$). The sum of these two voltage drops equals the holding potential set on the amplifier. The pipette and membrane voltages can be calculated, assuming a one compartment cell, from

$$V_{PIPETTE} = V_{HOLD}[R_S/(R_S + R_M)] \quad (1)$$

and

$$V_{MEMBRANE} = V_{HOLD}[R_M/(R_S + R_M)] \quad (2)$$

The speed with which a voltage step can be imposed on the cell membrane follows an exponential time course according to

* Tel.: +1 404 7270357; fax: +1 404 7270365; e-mail: straynelis@pharm.emory.edu

0165-0270/98/$ - see front matter
PII S0165-0270(98)00140-X

$$V_{\mathrm{MEMBRANE}}(t+\Delta t) = V_{\mathrm{MEMBRANE}}(t) + V_{\mathrm{STEP}}\{1-\exp[-\Delta t(R_{\mathrm{S}}+R_{\mathrm{M}})/(R_{\mathrm{S}}R_{\mathrm{M}}C_{\mathrm{M}})]\} \quad (3)$$

where t is time and C_{M} is the membrane capacitance (Marty and Neher, 1995). If a current I that flows across the pipette is recorded, then the voltage drop across the pipette and membrane can also be calculated from Ohm's Law at any moment according to

$$V_{\mathrm{PIPETTE}} = IR_{\mathrm{S}} \quad (4)$$

and

$$V_{\mathrm{MEMBRANE}} = V_{\mathrm{HOLD}} - V_{\mathrm{PIPETTE}} \quad (5)$$

For example, if the holding potential of a patch clamp amplifier is set to −100 mV during recording from a 20 pF cell with series resistance of 10 MΩ and membrane resistance of 1 GΩ (i.e. membrane conductance is 1000 pS), then the voltage drop across the pipette will be −0.99 mV. This means the cell's transmembrane voltage will be −99.01 mV and a leak current of −99.0 pA will flow at rest. If at some time enough ion channels on the cell open to create a new 10000 pS conductance pathway in response to a stimulus (e.g. rapid application of an agonist), the total cell resistance would drop to 0.091 GΩ (i.e. conductance will be 11000 pS). Accordingly, the relative size of the voltage drops across the pipette and membrane will change. Because R_{M} is reduced to 0.091 GΩ at the instant that the channels open, the voltage drop across the series resistance will change from −0.99 to −9.90 mV and the transmembrane potential will change from −99.01 to −90.10 mV with a time course described by Eq. (3) as the amplifier-based resistive current that flows changes (nearly instantly) from −99.0 to −990 pA. Two types of errors are associated with this change in transmembrane potential subsequent to ion channel opening. First, the cell is no longer effectively held at the same potential as before, and the resistive current changes accordingly. This can become an even more serious error when larger currents flow or series resistance is high. For instance, a 2 nA current in our example creates about a +20 mV voltage error. Second, the change of the intracellular transmembrane potential creates a time-dependent capacitative current, $I_{\mathrm{CAPACITOR}}(t)$. That is, if the ion channels open within the time it took to measure a single point (one digitization interval or Δt), then the capacitative current would be

$$I_{\mathrm{CAPACITOR}}(t) = C_{\mathrm{M}}\,\mathrm{d}V/\Delta t \quad (6)$$

where $\mathrm{d}V$ is the voltage step that occurs at the membrane within a time period Δt. Because voltage changes are not instantaneous, small voltage steps are applied to the cell over several time intervals following the change in membrane resistance. In the first 10 μs period after channels open in our example, the voltage will change by +0.48 mV, and the magnitude of the net change will reach +8.91 mV with a time constant of 0.18 ms. For our 20 pF cell described above, $\mathrm{d}V$ is +0.48 mV and Δt is 10 μs immediately after channel opening. Accordingly, $I_{\mathrm{CAPACITOR}}(\Delta t)$ will be +960 pA within the 10 μs in which channels open. This current nearly completely offsets (for an instant) the net resistive current that would flow with no series resistance. If no more channels open or shut, the capacitative current would exponentially relax, again, with a time constant of 0.18 ms. As the membrane capacitor becomes charged, the ionic current becomes unopposed, and eventually reaches its full amplitude. Thus, the capacitative current has siphoned off the initial current flowing into the cell to charge the membrane capacitor; the capacitor will discharge this current when the channels close, causing some extension of the current response. This capacitative charging and discharging of the membrane in effect applies a 1 pole (RC) filter to the data (in our example the cut off frequency is 0.4 kHz). However, if R_{M} also changes moment to moment (as it does in real life), the small internal voltage shift also will fluctuate along with the magnitude of the capacitative current, the corresponding filter cut off frequency, as well as the error in the holding potential.

Modern patch clamp amplifiers can correct both errors through their circuitry. The first error can be corrected by monitoring the current that flows, calculating in real time the voltage drop if series resistance is known, and simply adding a potential to the imposed potential to force the transmembrane potential to equal the experimentally set potential. This correction makes no assumptions about the rectification properties of the current–voltage curve of the ion channel(s) underlying the current being studied, which is ideal. The second error can also be corrected, but is slightly more complex, even for a single compartment cell. The capacitative current that flows can also be calculated within the amplifier circuitry if the cell capacitance and series resistance are known. However, the correction is usually passed through a 1 pole filter so that the amount of current immediately flowing is used to correct the current that flowed at a previous instant. Inherent noise and the fact that some stray current flows across the pipette creates the possibility for an escalating positive feedback loop that can lead to current oscillations and loss of voltage control when high levels of compensation are employed. The decision about how much compensation to employ in a given experiment usually is made quickly and on the spot. It can be tempting to apply a suboptimal degree of correction to guard against instability that might develop at some point during the experiment, which would require compensation to be changed mid-experiment. The decision is nearly irrevocable, as the data recorded by the amplifier is corrected at whatever level is specified prior to analogue or digital storage of the signal.

An alternative method to amplifier-based series resistance compensation is to perform some or all series resistance correction off-line or within the acquisition program after the signal is recorded. This way the highest degree of compensation tolerated by the data record can be applied, differing degrees of compensation can be employed, and the uncorrected and corrected signals can be compared—something that cannot easily be done with amplifier-based correction without repeating the experiment with and without series resistance compensation circuitry switched on. A straightforward computer algorithm is presented below to correct both series resistance errors in either simulated or recorded current data from a fixed holding potential (i.e. responses from ligand-gated ion channel).

2. Theory

The correction of the voltage error caused by the series resistance acting as a voltage divider in a one compartment cell can be made for channel responses with linear current–voltage curves for a given digitized current (recorded or simulated), provided the series resistance is known. This correction, with a lag time of one digitization interval, is

$$I^*_{i-1} = I_{i-1}(V_{\mathrm{HOLD}} - V_{\mathrm{REV}})/(V_{\mathrm{HOLD}} - V_{\mathrm{REV}} - I_i R_{\mathrm{S}}) \tag{7}$$

or

$$I^*_{i-1} = I_{i-1} - I_{i-1} f[1 - (V_{\mathrm{HOLD}} - V_{\mathrm{REV}})/(V_{\mathrm{HOLD}} - V_{\mathrm{REV}} - I_i R_{\mathrm{S}})] \tag{8}$$

where I^*_{i-1} is the corrected current for the previous data point, I_{i-1} is the current recorded in the previous data point, I_i is the current recorded in the present data point, V_{REV} is the reversal potential of the current being corrected, and f is the fraction of correction desired (ranging from 0 to 1). The correction for the capacitative current is also straightforward, and is

$$I^*_{i-1} = I_{i-1} - f I(\mathrm{cap})_i \tag{9}$$

where f again is the fractional compensation desired (ranging from 0 to 1), and $I(\mathrm{cap})_i$ is the capacitative current calculated from a modification of Eq. (6),

$$I(\mathrm{cap})_i = C_{\mathrm{M}}[(V_{\mathrm{HOLD}} - I_i R_{\mathrm{S}}) - (V_{\mathrm{HOLD}} - I_{i-1} R_{\mathrm{S}})]/\Delta t \tag{10}$$

or

$$I(\mathrm{cap})_i = C_{\mathrm{M}}[I_{i-1} R_{\mathrm{S}} - I_i R_{\mathrm{S}}]/\Delta t \tag{11}$$

where $\mathrm{d}V$ has been replaced by the difference in the transmembrane voltage (i.e. the holding potential corrected for R_{S}) for the current and previous data point determined using the digitized current record and Ohm's law. This correction is also applied with a lag time equal to the digitization interval. Longer lag times for Eqs. (8) and (10) can be achieved by applying a 1 pole RC filter to the capacitative current correction, which delays and smooths this correction before it is added to the raw current. The effect of this filter on the capacitative current at each time point can be calculated according to the equation

$$I(\mathrm{cap})^*_i = I(\mathrm{cap})_i(1 - \exp(-\Delta t 2\pi Fc) \tag{12}$$

where Fc is the -3 dB cutoff frequency of the 1 pole filter applied to the corrected current and $I(\mathrm{cap})^*_i$ is the filtered capacitative current. The degree of such filtering is sometimes referred to as a lag time where Fc corresponds to $1/(2\pi\tau_{\mathrm{LAG}})$.

3. Implementation

Corrections of series resistance errors can be made in a computationally efficient manner for a one compartment cell if the membrane capacitance and series resistance are known. These parameters can be determined from the compensation circuitry on the amplifier, or better yet, from a series of capacitative transient currents recorded in response to a known imposed voltage step applied at various times throughout the experiment. The C function discussed below (and listed without interrupting text in Appendix A; see Appendix B for an example of FORTRAN code) corrects an array (`rawdata[]`) containing simulated or recorded current (in amps) for the voltage errors and filtering associated with series resistance and cell capacitance assuming a single compartment cell. The function requires that R_{S} (ohms), C_{M} (farads), V_{HOLD} (volts), V_{REV} (volts), Δt (s), and the number of data points are globally defined elsewhere in the program; these variables can be included as external variables in a file containing the function `RsCorrection()`, as shown below.

```
extern int i_numdatapts;             // number of data points
extern float f_rawdata[];            // recorded current data in Amps
extern float f_adinterval;           // seconds
extern float f_holdingpotential;     // volts
extern float f_reversalpotential;    // volts
extern float f_Rs;                   // ohms
extern float f_Cm;                   // farads
```

In the implementation here of Eqs. (8)–(12), the corrected output current replaces the input current array (`f_rawdata[]`), but could just as easily be redirected to a new output array. The correction is applied with a delay of one digitization interval and the fractional compensation (0.0–1.0) for the steady state voltage correction (`f_frac_V_compensation`) and capacitative filtering correction (`f_frac_C_compensation`) are passed to the function `RsCorrection()` as arguments. The capacitative current correction can be filtered prior to adding it to the uncorrected current to introduce a further lag in the correction, and this filtering is controlled by a -3 dB cutoff frequency for a 1 pole filter (`f_correction_Fc`) that is also passed to the function. The code is ANSI compatible, and can be implemented on 16 or 32 bit compilers. The function `RsCorrection()` uses five internal variables, which are `i` (a loop counting integer), `f_Icap` (the capacitative current), `f_volt_this_pt` (the transmembrane voltage for the current data point), `f_volt_last_pt` (the transmembrane voltage for the previous data point), and the fractional voltage correction (`f_V_correct`).

```
void RsCorrection(float f_frac_V_compensation, float f_frac_C_compensation, float
f_correction_Fc)
{
int i=0;
float f_Icap=0.0, f_volt_this_pt=0.0, f_volt_last_pt=0.0, f_V_correct=0.0;
```

The voltage drop within the intracellular compartment is initially calculated for the first data point in the input array using Ohm's Law. The measured current at the first data point must have flowed across the series resistance, so that the real membrane potential within the cell will be the difference of the holding potential set on the amplifier and the small voltage drop across R_S. The code guards against divide-by-0 errors in lines 1 and 6.

```
if  ((f_volt_last_pt=(f_holdingpotential-f_rawdata[0]*f_Rs))!=f_reversalpotential)
                                                                          line 1
   f_V_correctf_frac_V_compensation *(1-(f_holdingpotential-f_reversalpotential)
   /(f_volt_last_pt-f_reversalpotential));                                line 2
else
   f_V_correct=0.0;                                                       line 3
   f_rawdata[0]=f_rawdata[0]-f_rawdata[0]*f_V_correct;                    line 4
```

Next the full dataset is corrected starting at the second data point. The correction has two parts. The correction required for the resistive current is calculated for later use in lines 5–8 for a fractional portion (`f_frac_V_compensation`) of the voltage using Ohm's Law, which of course assumes a linear current–voltage relationship for the response. At this stage (i.e. lines 2 and 7) a more complex function could easily be introduced if the current–voltage curve was nonlinear and known (see Appendix C). The transmembrane potential between the first and second data point is then determined by calculating voltage drop that occurs as a result of any change in current flow across the series resistance, as described above. The transient error due to the fraction of the current flowing that charges the membrane is subsequently calculated. That is, Eqs. (6), (9) and (10) are used to calculate the capacitative current that flows the instant the current response (and hence membrane conductance) changes (line 9). The membrane capacitance (`f_Cm`) is defined or measured by the experimenter. To obtain dV/dt, use the difference between the exact transmembrane potential (corrected for series resistance as above) between the previous point and the current point (i.e. dV is the difference between `f_volt_this_pt` and `f_volt_last_pt`). This capacitative current correction is then filtered in line 10 (1 pole) using the -3 dB cut off frequency passed to the function (see above and Eq. (3))

```
for (i=1; i<i_numdatapts; i++)                                           { line 5
  if ((f_volt_this_pt=(f_holdingpotential-f_rawdata[i]*f_Rs))
  !=f_reversalpotential)                                                  line 6
    f_V_correct=f_frac_V_compensation *(1-(f_holdingpotential-f_reversalpotential)/
    (f_volt_this_pt-f_reversalpotential));                                line 7
else
  f_V_correct=0.0;                                                        line 8
  f_Icap=f_Cm*(f_volt_this_pt-f_volt_last_pt)/f_adinterval;               line 9
  f_Icap*=(1-exp(-2.0*3.141592654*adinterval*f_correction_Fc));           line 10
```

Next, the correction for capacitative current is applied to the raw current by subtracting the capacitative current multiplied by some fraction of the desired correction (line 11). Alternatively, the capacitative current could have been divided by two and spread out to the present and previous data points (not shown here).

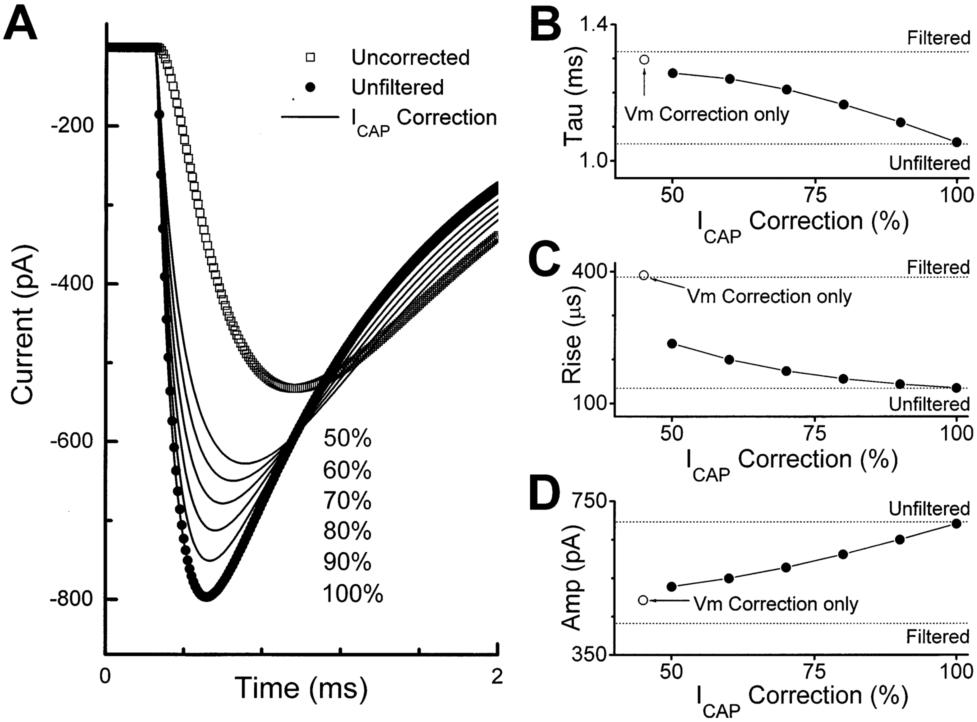

Fig. 1. (A) Small amplitude rapidly desensitizing current responses were simulated using an exponentially rising and decaying function. The filled circles show an unfiltered current response and open squares show the same response subject to series resistance filtering ($R_S = 20$ MΩ, $C_M = 20$ pF, $R_M = 1$ GΩ, $V_{HOLD} = -100$ mV, $V_{REV} = 0$ mV). The solid lines show the filtered response with full correction of the voltage error and variable correction of the capacitative filtering error (50–100%). (B)–(D) The fitted time constant of decay, the 10–90% rise time, and the amplitude are shown for the filtered and unfiltered simulated currents (broken lines). These parameters are also shown for currents with full voltage correction alone (denoted 'Vm Correction only', open circles) or full voltage correction plus a variable correction for capacitative filtering (filled circles). The capacitative filtering dominates the effects of series resistance for these small amplitude and transient responses.

Either method works; there is little difference observed between them when the AD interval is brief. Line 12 corrects the net current for the instantaneous voltage drop associated with the series resistance, which was previously calculated in line 7 and 8. Finally, in line 13 the transmembrane voltage for the current data point is set to the transmembrane potential for the last data point, in anticipation of evaluating the next data point. Fig. 1 shows an example of full voltage correction and varying degrees of capacitative current correction applied to a filtered rapidly desensitizing current response.

```
f_rawdata[i-1]-=(f_frac_C_compensation*f_Icap);      line 11
f_rawdata[i-1]-=(f_rawdata[i-1]*f_V_correct);        line 12
f_volt_last_pt=f_volt_this_pt;                       line 13
}
}
```

This C code has been implemented in the DOS-based program (NPM051) and a Windows95 program (MODEL), and sample FORTRAN code (Appendix B) has been implemented in a DOS-based program (RSCORR). The code, these programs, and a spreadsheet version of this correction algorithm are available from ftp.pharm.emory.edu.

4. Results and discussion

The series resistance correction algorithm presented here is useful in a number of situations. This algorithm requires that the cell behaves as a single compartment and that series resistance be diligently measured at appropriate intervals throughout the experiment. The algorithm has the advantage that compensation can be applied at a variety of levels after the experiment is completed, the effects of correction can be directly evaluated without repeating experiments, and the relative contribution of capacitative filtering and voltage errors can be assessed. Finally, the calculations are computationally efficient and can be nearly exact (100% compensation, lag time of about one digitization interval in the absence of noise). A disadvantage of this algorithm is that, for simplicity, the voltage correction assumes current–voltage curves are linear, a point

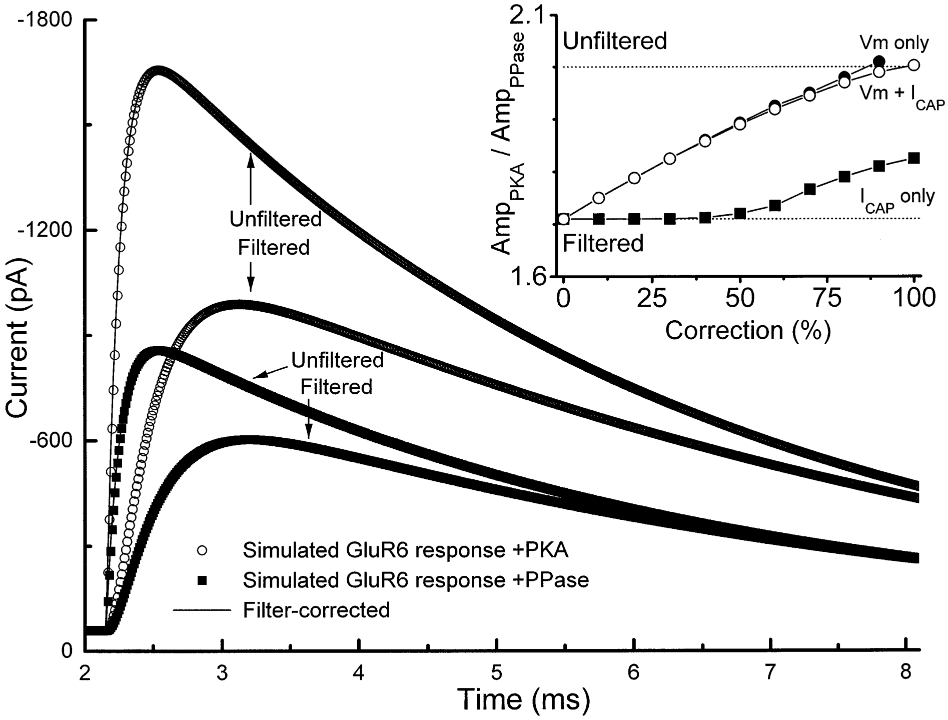

Fig. 2. Filtered and unfiltered simulated peak responses are shown for a large (30 nS, open symbols) and smaller (15 nS, closed symbols) desensitizing response ($R_S = 20$ MΩ, $C_M = 20$ pF, $R_M = 1$ GΩ, $V_{HOLD} = -60$ mV, $V_{REV} = 0$ mV, $\tau_{DECAY} = 4$ ms), with an appropriate time course and relative amplitude for PKA-treated (open symbols) or calcineurin-treated (PPase, closed symbols) responses for homomeric GluR6 receptors (Traynelis and Wahl, 1997). The filtering reduces the apparent difference between peak response amplitude simulated for PKA vs PPase. The inset shows the ratio of large (+ PKA) to small (+ PPase) current peaks for currents with variable correction for capacitative filtering (filled squares), series resistance voltage error (filled circles), and both (open circles). The broken lines show the ratio of currents for unfiltered and filtered (uncorrected) responses. The filter-reduced difference between the large and small currents is primarily a result of the voltage error rather than the capacitative filtering.

which can introduce difference between the amplifier-based and software-based corrections of data obtained from rectifying currents. Whereas, non-linearity in a current–voltage curve is straightforward to introduce into the voltage-correction part of the algorithm (see Appendix C for example and explanation), it does mean that a stable and accurate current–voltage curve must be obtained for each cell to be corrected. This problem is more easily circumvented for amplifiers that can provide only a circuitry-based correction of the voltage error independent of correction for capacitative filtering, such as the Axon Axopatch 200 series and Warner PC-501A and PC-505A amplifiers. In this situation, the data can be recorded with a voltage correction applied on the amplifier, and correction of capacitative filtering can be applied off line using methods described here.

In many experiments, suboptimal series resistance compensation is employed, partially correcting the time course of a recorded response, but leaving the true waveform unknown. By applying varying degrees of series resistance corrections to the same response off line, it is possible to estimate full correction of the peak amplitude or decay time constants of the response by extrapolation (Fig. 1) without using 100% compensation, which can be impossible for the algorithm described above as well as circuitry-based corrections if there is considerable noise associated with the response (see below).

Some investigators have sought to determine the exact conductance waveform underlying current responses in an effort to understand and model temporal aspects of the signal under consideration. In the absence of circuit-based series resistance compensation, elaborate methods have been used to extract the true waveform underlying the filtered current (e.g. Silver et al., 1992). The present algorithm can be adapted to directly and rapidly correct the series resistance errors associated with any whole cell current response for which the current–voltage curve is known, series resistance and membrane capacitance can be determined, and electrotonic structure is compact.

This algorithm, in addition to being a useful experimental tool, can also be used didactically for evaluating the relative contribution of non-intuitive series resistance errors to current recordings. For example, series resistance voltage errors are amplitude-dependent, which means they can minimize the observed

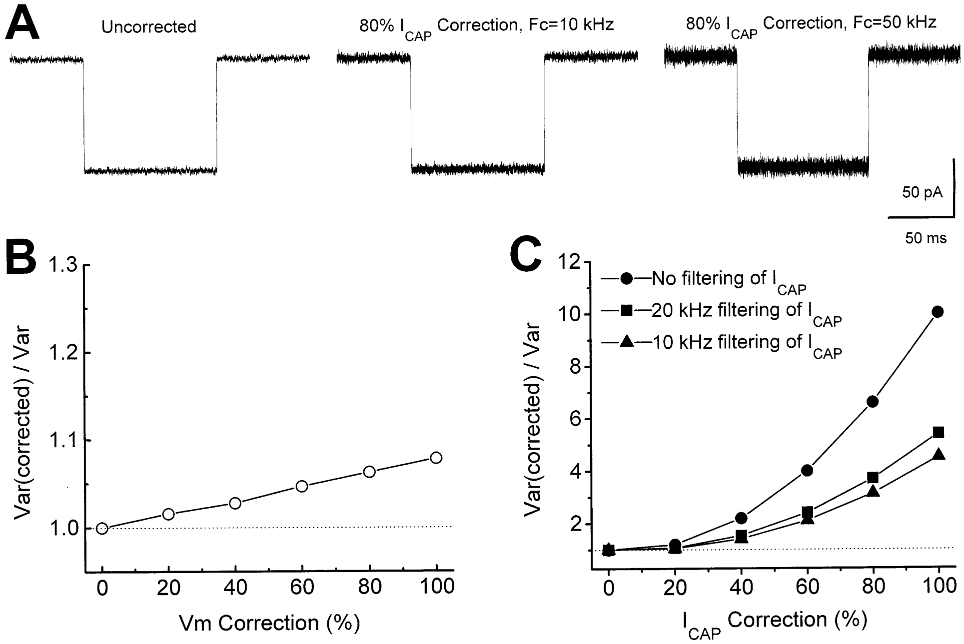

Fig. 3. (A) Simulated whole cell step current responses are shown uncorrected and corrected for capacitative filtering but not for series resistance voltage errors. This cell had a series resistance of 10 MΩ, an input resistance of 1 GΩ, and a whole cell capacitance of 10 pF (V_{HOLD} was -100 mV; time interval 10 μs). The uncorrected current arose from a 1000 pS conductance change with Gaussian noise added (STD 20–60 pS, which corresponds to uncorrected RMS values of 0.8–1.7 pA). 80% corrections are shown with 10 (τ_{LAG} 16 μs) or 50 kHz filtering (τ_{LAG} 3.2 μs) of the capacitative correction current. (B) Correction of the series resistance voltage error in isolation only marginally increases the variance of the current response for the same simulated responses shown in (A). The ratio of the variance of the corrected signal to the variance of the uncorrected signal was obtained from the average result of correction of Gaussian noise of 20, 40, and 60 pS added to the step conductance change underlying the current response. (C) Correction of the capacitative filtering error in isolation increases the variance of the current response about 10-fold ($\sim$ threefold increase in root mean square noise) when the correction is applied with a lag time of a single AD unit. Filtering the capacitative correction at 20 or 10 kHz (i.e. introducing τ_{LAG} of 8 or 16 μs) reduces this increase in current variance.

changes in current amplitudes that experimental treatments may induce. One such example is the protein kinase A (PKA) potentiation of recombinant glutamate receptor responses (e.g. Traynelis and Wahl, 1997). PKA phosphorylation can double the response of homomeric GluR6 receptors by increasing P(open) without changing response time course. Because the voltage error associated with the series resistance can be larger for large amplitude whole cell currents than for smaller responses, it can minimize the changes in amplitude that result from potentiation, for example, of GluR6 responses by PKA. That is, as the current increases in PKA, so too will the voltage error, obscuring the true effects of PKA (Fig. 2). In contrast, for small but transient currents (Fig. 1), the capacitative filtering can cause more serious errors than series resistance-induced alterations in the voltage.

Finally, this algorithm allows an evaluation of the relative contribution of corrections of the voltage error and capacitative current to the enhancement of noise that is known to occur with circuit-based series resistance corrections. Fig. 3 shows that correction of the voltage error has virtually no effect on the variance of simulated responses. In contrast, correction of the capacitative current can increase the root mean squared noise level several fold, depending on the cell characteristics. This enhancement of noise also occurs in circuit-based corrections, and follows directly from the idea that the cell filtering to be corrected has itself already decreased the noise level. Circuit-based corrections apply a 1 pole filter to the capacitative current correction, and introduction of this manipulation into algorithm described here can also effectively decrease the noise level while only modestly compromising the correction.

Acknowledgements

I am grateful for Drs D. Bowie, R. Dingledine, and M. Farrant for helpful comments on the manuscript. This work was supported by NINDS, NIMH, The John Merck Fund, and The Epilepsy Foundation of America.

Appendix A

C code is shown below for correction of series resistance errors for current responses with linear current–voltage relationships.

```
extern int i_numdatapts; // number of data points
extern float f_rawdata[]; // recorded current data in amperes
extern float f_adinterval; // seconds
extern float f_holdingpotential; // volts
extern float f_reversalpotential; // volts
extern float f_Rs; // ohms
extern float f_Cm; // farads
// f_correction_Fc in Hz
void RsCorrection(float f_frac_V_compensation, float f_frac_C_compensation, float f_
correction_Fc)
{
int i=0;
float f_Icap=0.0, f_volt_this_pt=0.0, f_volt_last_pt=0.0, f_V_correct=0.0;
if ((f_volt_last_pt=(f_holdingpotential-f_rawdata[0]*f_Rs)) !
=f_reversalpotential)
  f_V_correct=f_frac_V_compensation*   (1-(f_holdingpotential-f_reversalpoten-
tial)/ (f_volt_last_pt-f_reversalpotential));
else
 f_V_correct=0.0;
f_rawdata[0]=f_rawdata[0]-f_rawdata[0]*f_V_correct;
for (i=1; i<i_numdatapts; i++) {
 if ((f_volt_this_pt=(f_holdingpotential-f_rawdata[i]*f_Rs)) !
=f_reversalpotential)
   f_V_correct=f_frac_V_compensation*   (1-f_holdingpotential-f_reversalpoten-
tial)/ (f_volt_this_pt-f_reversalpotential));
else
   f_V_correct=0.0;
 f_Icap=f_Cm*(f_volt_this_pt-f_volt_last_pt) / f_adinterval;
 f_Icap*=(1-exp(-2.0*3.141592654*f_adinterval*f_correction_Fc));
 f_rawdata[i-1]-=(f_frac_C_compensation*f_Icap);
 f_rawdata[i-1]-=(f_rawdata[i-1]*f_V_correct);
 f_volt_last_pt=f_volt_this_pt;
}
}
```

Appendix B

FORTRAN code is shown below for correction of series resistance errors for current responses with linear current–voltage relationships.

```
    PROGRAM RSCORR
    INTEGER i
    DIMENSION raw(10240)
    REAL adint, hold, rev, rs, cm
    REAL icap, vthis, vlast, vcorr
    REAL fracv, fracc, fc
    i=1
    DO 50, i=1, 10240, 1
    raw(i)=0.0
50  CONTINUE
```

```
      adint=0.00001
      hold=-0.1
      rev=0.0
      rs=10.0e6
      cm=10.0e-12
      icap=0.0
      vthis=0.0
      vlast=0.0
      vcorr=0.0
      fracv=1.0
      fracc=1.0
      fc=100000
      OPEN (UNIT=1,FILE='INPUT.DAT',STATUS='OLD')
      READ (1,*) (raw(i), i=1, 10240)
      CLOSE (1,STATUS='KEEP')
      vlast=(hold-1e-12*raw(1)*rs)
      IF ((vlast-rev).EQ.0.0) THEN
      vcorr=0.0
      ELSE
      vcorr=fracv*(1-(hold-rev)/(vlast-rev))
      END IF
      raw(1)=raw(1)-(raw(1)*vcorr)
      DO 100, i=2, 10240, 1
      vthis=(hold-1e-12*raw(i)*rs)
      IF ((vthis-rev).EQ.0.0) THEN
      vcorr=0.0
      ELSE
      vcorr=fracv*(1-(hold-rev)/(vthis-rev))
      END IF
      icap=cm *(vthis-vlast) / adint
      icap=icap*(1-EXP(-2.0*3.141592654*adint*fc))
      raw(i-1)=raw(i-1)-(fracc*icap*1e12)
      raw(i-1)=raw(i-1)-raw(i-1)*vcorr
      vlast=vthis
100   CONTINUE
      OPEN (UNIT=1,FILE='OUTPUT.DAT',STATUS='NEW')
      DO 150, i=1, 10240, 1
      WRITE(1,*) raw(i)
150   CONTINUE
      CLOSE (1,STATUS='KEEP')
      END
```

Appendix C

C Code is shown that can correct series resistance errors for current responses that arise from channels with non-linear current–voltage relationships. This is done by replacing lines 2–3 and 6–8 described above in Appendix A with expressions that read the relative current and voltage from an experimentally determined curve, and use interpolation to determine the degree of current change for the calculated voltage error. This correction is then directly applied to the raw current. The current voltage curves can be simple (10 or fewer points) or complex (1000s of points), although the best results are achieved with the minimum number of points needed to describe the curve. The current voltage curve must be entered into the array starting with negative holding potentials. The capacitative correction is identical to that described above.

```
extern int i_numdatapts; // number of data points
extern float f_rawdata[]; // recorded current data in amperes
extern float f_adinterval; // seconds
extern float f_holdingpotential; // volts
extern float f_reversalpotential; // volts
extern float f_Rs; // ohms
extern float f_Cm; // farads
extern int i_num_IV_pts; // number of points in correction IV array
extern float f_IV_data[2][]; // two dimensional array of IV curve with
// voltage in f_IV_data[0][] and normalized
// amplitude in f_IV_data[1][]
// f_correction_Fc in Hz
void NonLinearRsCorrection(float f_frac_V_compensation, float f_frac_C_compensation,
float f_correction_Fc)
{
int i=0, j=0;
float f_Icap=0.0, f_volt_this_pt=0.0, f_volt_last_pt=0.0, f_V_correct=0.0;
float f_V_amp1=0.0, f_V_amp2=0.0;
f_volt_last_pt=f_holdingpotential-f_rawdata[0]*f_Rs;
f_V_amp1=f_V_amp2=0.0;
for (j=1; j<i_num_IV_pts; j++) {
  if (f_volt_this_pt<f_IV_data[0][j] && f_V_amp1==0.0)
    f_V_amp1=f_IV_data[1][j-1]+(f_volt_last_pt-f_IV_data[0][j-1]) /
(f_IV_data[0][j]-f_IV_data[0][j-1])*(f_IV_data[1][j]-f_IV_data[1][j-1]);
  if (f_holdingpotential<f_IV_data[0][j] && f_V_amp2==0.0)
    f_V_amp2=f_IV_data[1][j-1]+(f_holdingpotential-f_IV_data[0][j-1])/
(f_IV_data[0][j]-f_IV_data[0][j-1])*(f_IV_data[1][j]-f_IV_data[1][j-1]);
}
f_V_correct=f_frac_V_compensation*f_V_amp2 / f_V_amp1;
f_rawdata[0]*=f_V_correct;
for (i=1; i<i_numdatapts; i++) {
  f_volt_this_pt=f_holdingpotential-f_rawdata[i]*f_Rs;
  f_V_amp1=f_V_amp2=0.0;
  for (j=1; j<i_num_IV_pts; j++) {
    if (f_volt_this_pt<f_IV_data[0][j] && f_V_amp1==0.0)
      f_V_amp1=f_IV_data[1][j-1]+(f_volt_this_pt-f_IV_data[0][j-1]) /
(f_IV_data[0][j]-f_IV_data[0][j-1])* (f_IV_data[1][j]-f_IV_data[1][j-1]);
    if (f_holdingpotential<f_IV_data[0][j] && f_V_amp2==0.0)
      f_V_amp2=f_IV_data[1][j-1]+(f_holdingpotential-f_IV_data[0][j-1])/
(f_IV_data[0][j]-f_IV_data[0][j-1])* (f_IV_data[1][j]-f_IV_data[1][j-1]);
  }
  f_V_correct=f_frac_V_compensation*f_V_amp2 / f_V_amp1;
  f_Icap=f_Cm*(f_volt_this_pt-f_volt_last_pt) / f_adinterval;
  f_Icap*=(1-exp(-2.0*3.141592654*f_adinterval*f_correction_Fc));
  f_rawdata[i-1]-=(f_frac_C_compensation*f_Icap);
  f_rawdata[i-1]*=f_V_correct;
  f_volt_last_pt=f_volt_this_pt;
}
}
```

References

Marty A, Neher E. Tight-seal whole-cell recording. In: Sakmann B, Neher E, editors. Single-channel Recording, 2nd edn. New York: Plenum, 1995.

Silver RA, Traynelis SF, Cull-Candy SG. Rapid time-course miniature and evoked excitatory currents at cerebellar synapses in situ. Nature 1992;355:163–6.

Traynelis SF, Wahl P. Control of rat GluR6 glutamate receptor open probability by protein kinase A and calcineurin. J Physiol 1997;503:513–31.