SYNAPSE 60:32–44 (2006)

Self-Modeling Structure of Evoked Postsynaptic Potentials

KERT VIELE,[1] MARK LANCASTER,[1] AND ROBIN L. COOPER[2]*
[1]*Department of Statistics, University of Kentucky, Lexington, Kentucky, 40506-0027*
[2]*Department of Biology, University of Kentucky, Lexington, Kentucky 40506-0225*

KEY WORDS SEMOR; splines; quantum; synapse; analysis

ABSTRACT With the simplicity of the synaptic structure and physiology at neuromuscular junctions (NMJs) of crayfish and the given transmitter being released in quantal packets, a detailed assessment in the fundamental processes of chemical synaptic transmission is possible. Since the quantal event is the basic element of transmission, we consider an approach to further understand the characteristics of quantal responses. In this study, we introduce a method for combining information across excitatory postsynaptic potentials (EPSPs) that are quantal in nature. The method is called self-modeling regression, known in the statistics literature as SEMOR. This method illustrates that the differing timing and heights of EPSPs can be described with four coefficients measuring affine (shift and scale) transformations of the x and y axes. We demonstrate that this relationship allows us to provide a unified schema for the many functionals currently used in the literature, such as peak amplitude, τ, latency, area under the curve, or decay time. Computer code in R is available on the internet to perform the analysis. **Synapse 60:32–44, 2006.**

INTRODUCTION

To index synaptic efficacy, various statistical measures and derived parameters are used. Mean quantal content (m) is commonly used as an index of the average number of events (or vesicles that fuse) that occur per stimulus, with a parameter n representing the number of sites where a vesicle fuses and a third parameter, p, representing the probability of an event occurring at a fusion site (Del Castillo and Katz, 1954a,b). The values of n and p have been traditionally hard to estimate, since the structural definition of n is not clear; thus, the probability of release cannot be defined well, unless it is clear as to what is a site of vesicular fusion. Since the classical analysis of quantal responses treats all single events equal, an estimation of n and p is estimated by best fits, to known distributions, by the occurrence of single, double, or multiple quantal events (Del Castillo and Katz, 1954a,b; Dudel, 1981; Hatt and Smith, 1976; Johnson and Wernig, 1971; Wernig, 1972).

Physiological measures of quantal responses, at low frequency stimulation, at a low output neuromuscular junction in the crayfish reveal $n = 1$ by quantal counting methods, despite quite large variations in the sizes and shapes of single evoked quantal synaptic currents. The physiological and structural data would indicate that multiple sites are being utilized for vesicle fusion. In addition, the standardized approach of obtaining n and p from methods of directly counting quantal events and determining their distribution of occurrence underestimates the functional number of sites. For these reasons, we devised a means with statistical methods that incorporate differences in characteristics in the single quantal responses, to allow for a better estimate of n and p for quantal subsets in this report.

We have shown earlier that quantal responses from single events can be clustered into types of groupings, based on their characteristic shapes (Viele et al., 2003). This current study provides a means to aid in determining the types of parameters that can be readily defined from a series of quantal events. While this particular study does discuss clustering in the Results section, it's primary goal is not to introduce a new clustering technique, but to allow the current clustering methods to function better by (1) unifying the discussion of "shape" in an excitatory postsynaptic potential (EPSP) by illustrating a "self-modeling" property of the EPSPs, (2) using the self-

Contract grant sponsor: NSF; Grant number: NSF-IBN-0131459.
*Correspondence to: Robin Cooper. E-mail: RLCOOP1@uky.edu
Received 16 August 2005; Accepted 8 November 2005
DOI 10.1002/syn.20274
Published online in Wiley InterScience (www.interscience.wiley.com).

modeling structure to illustrate required relationships between functionals that were previously used independently, and (3) providing a way of producing an "average shape" for an EPSP that can reveal fine structure. We employ clustering methods previously appearing in the literature, but the methods described here reduce noise, and thus allow those methods to function better.

Using past counting methods, two currents, which both indicate one evoked event, would be recorded identically even if the two currents appear distinctly different. Since differences in current sizes or shapes may indicate multiple sites at work, this may be an useful information in determining the overall number of sites utilized. Given that many synapses have been serial-reconstructed for the crayfish opener excitatory neuromuscular junction (NMJ), the nature of synaptic size and its synaptic complexity is becoming known (Atwood and Cooper, 1995, 1996a,b; Cooper et al., 1995b, 1996a,b; Govind et al., 1994). Direct structure–function studies of discrete regions of motor nerve terminals have revealed that there can be many synapses (30–40), each with multiple active zones (AZs). It is possible that each vesicle fusion site, in respect to the postsynaptic receptor array, may provide a fingerprint of function during a basal state of transmission. Even based on detailed anatomical information of the synapse and the AZs, assigning an n value is problematic in the simplest of preparations since a single synapse is really not equivalent to $n = 1$, and a single AZ on a single synapse might not necessarily be equal to $n = 1$ since vesicles can fuse on all sides of a dense body.

In this study, we make use of the crayfish opener NMJ, since it is known to release transmitter in a quantal nature in the same manner as observed for NMJs of vertebrates, but with the exception that the probability of vesicular fusion is extremely low along the length of the entire nerve terminal and the muscle is nonspiking within a normal physiological environment. Thus, postsynaptic receptor poisons and calcium channel blockers are not needed to study quantal postsynaptic responses (Del Castillo and Katz, 1954a,b; Dudel and Kuffler, 1961). An additional nicety of the crayfish opener NMJ is that the AZs on synapses of the excitatory motor nerve terminal are not arranged in a grid nor function as a unit, as for many excitatory synapses within the vertebrate CNS, but more as discrete AZs at low frequencies of stimulation. However, as the stimulation frequency is increased, there is computational evidence that the synapses have a differential probability of vesicular events since the synaptic complexity is varied. Some synapses possess a single AZ while others might contain one or two AZs (Cooper et al., 1995a). It is presumed that the spacing among AZs influences the probability of vesicular fusion. Synapses containing AZs close together result in larger presynaptic calcium domains as compared to synapses with single AZs (Cooper et al., 1996a,b). Thus, during evoked stimulation with single or trains of stimuli, the probability is not consistent among the synapses with varying synaptic structural complexity.

These issues fall equally upon the postsynaptic target tissue, if one is using a response to measure quantal responses. In considering the postsynaptic receptor array being uniform in packing density and the issues of the point-location, the vesicle fuses with the plasma membrane, and the receptor area may only receive a portion of the transmitter if fusion occurred on the edge of the presynaptic synapse (Uteshev and Pennefather, 1997). This could result in a different size or shape of the unitary postsynaptic current as compared to the release over the center of the receptor array. In addition, if one considers the possibility of the postsynaptic arrays of receptors being saturated from the release of a single vesicle, any release within the center would saturate, whereas release on the edges might result in a nonsaturated array.

Prior studies have shown that analysis of functionals, such as peak amplitude, τ, area under the curve (AUC), and others, may be used to cluster voltage traces into groups potentially corresponding to multiple sites (Viele et al., 2003). The aim of the current research is as follows:

1. Illustrate that EPSPs follow what is called a "self-modeling" structure, meaning that if $h_i(t)$ is the voltage for the ith EPSP at time t, then there exists a single function g, and for each EPSP, a set of coefficients a_i, b_i, c_i, and d_i are given as follows:

$$h_i(t) = a_i g(c_i t + d_i) + b_i$$

   Note that the same g is used for each EPSP. The coefficients c_i and d_i stretch and shift the time axis, while the coefficients a_i and b_i stretch and shift the voltage axis. This is described in the Materials and Methods section.
2. The self-modeling structure of the data allows relationships between the previously used functionals to be determined. For example, AUC, except for noise, is proportional to the product of the peak amplitude and τ. Most of the commonly used functionals can be described solely in terms of a_i and c_i coefficients. This is discussed in the Materials and Methods section.
3. The function g allows a noise-reduced estimate of the EPSPs to be produced, alllowing fine structure of the EPSPs to be seen that is not available in the individual trace data because of the amount of

noise in the individual traces. By combining the information in all the EPSPs, we can observe this fine structure. This is discussed in the Results section.

MATERIALS AND METHODS

Materials

The preparation

All experiments were performed using the first walking leg of crayfish, *Procambarus clarkii*, measuring 4–6 cm in body length (Atchafalaya Biological Supply Co., Raceland, LA). The opener muscle of the first walking legs was prepared by standard dissection (Cooper et al., 1995a). The tissue was pinned out in a Slygard dish for viewing with a Nikon, Optiphot-2 upright fluorescent microscope using a 40× (0.55 noradrenaline) Nikon water dipping objective. Dissected preparations were maintained in crayfish saline (modified Van Harrevelds solution: 205 mM NaCl; 5.3 mM KCl; 13.5 mM $CaCl_2$; 2.45 mM $MgCl_2$; 0.5 mM Hepes/NaOH, pH 7.4) at 14°C.

Physiology: Field EPSPs

The stimulation of the excitatory nerve and the EPSP recordings on the walking leg of the crayfish opener muscle were performed by standard procedures (Logsdon et al., 2005). The nerve was stimulated at 1 Hz. Synaptic potentials were obtained with focal macropatch electrodes (Dudel, 1981) by lightly placing a 10–15 μm diameter, fire-polished glass electrode directly over a spatially isolated varicosity. The lumen of the patch electrode was filled with the same solution as the bathing medium. The seal resistance was in the range of 100 KΩ to 1 MΩ. All events were obtained with an Axoclamp 2b (Axon Instruments) 0.1× LU head stage, acquired at 10 kHz without additional filtering. The varicosities on the living terminals were visualized by the use of the vital fluorescent dye 4-Di-2-ASP (Molecular Probes) (Cooper et al., 1995a; Magrassi et al., 1987). The evoked field EPSPs were recorded online to a Dell Latitude D600 computer via a PowerLab/4s interface.

Statistical methods

After preprocessing data from many EPSPs, we arrive at a data set as in Figure 1. The preprocessing consists of (1) aligning the curves to a common baseline to remove DC drift, (2) removing the stimulation artifact in the bath, (3) determining which currents are EPSPs according to a thresholding algorithm, consistent with visual inspection, and (4) removing doublets, triplets, or other multiple EPSPs (determined by visual inspection of the EPSPs) from the analysis.

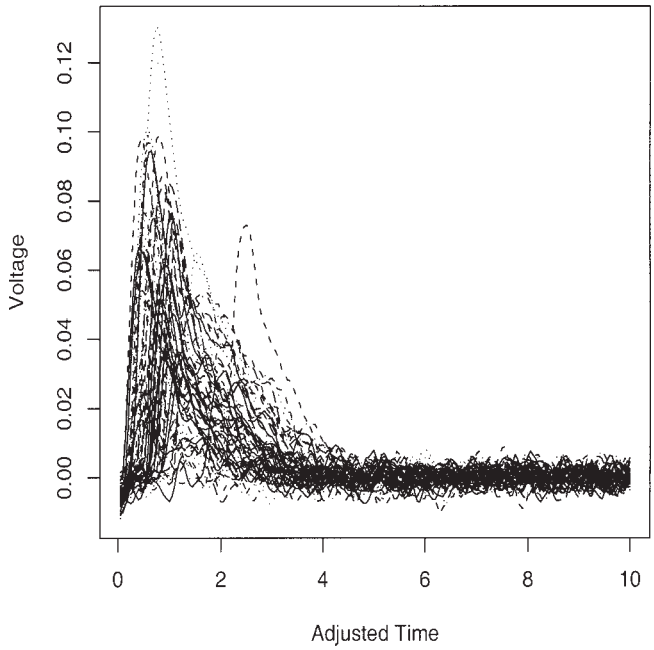

Fig. 1. A set of single quantal EPSPs obtained by focal macropatch recordings, over a defined region of the motor nerve terminal. Note that some have a delayed rise time and that there is variability to the quantal nature of he events. Voltage is relative for the analysis presented.

We performed these background steps with the computer code provided in Lancaster et al. (2005), but any method arriving at a dataset similar to Figure 1 is acceptable.

The goal of the current article is to demonstrate that although the traces in Figure 1 have obvious differences in terms of height, width, and latency, they are similar in terms of shape. They are related through what is called in the statistical literature "self-modeling" or "shape-invariant" functions (Kneip and Engel, 1995; Kneip and Gasser, 1988; Lawton et al., 1972; Lindstrom, 1995), where each EPSP can be overlaid on the others through affine (linear stretching and shifting) transformations of the time and voltage axes.

This self-modeling structure provides methodological advantages in terms of analyzing EPSPs. First and foremost, the shape-invariant structure allows regression methods based on splines to be used to estimate smoothed versions of the traces, resulting in noise reduction for subsequent analyses. Second, the functionals discussed earlier (peak amplitude, AUC, etc.) can all be placed in a unified framework, demonstrating the relationships between these functionals. Third, information from multiple curves can be combined to show "fine structure" aspects of the curves that are visually obscured by noise in the individual curves.

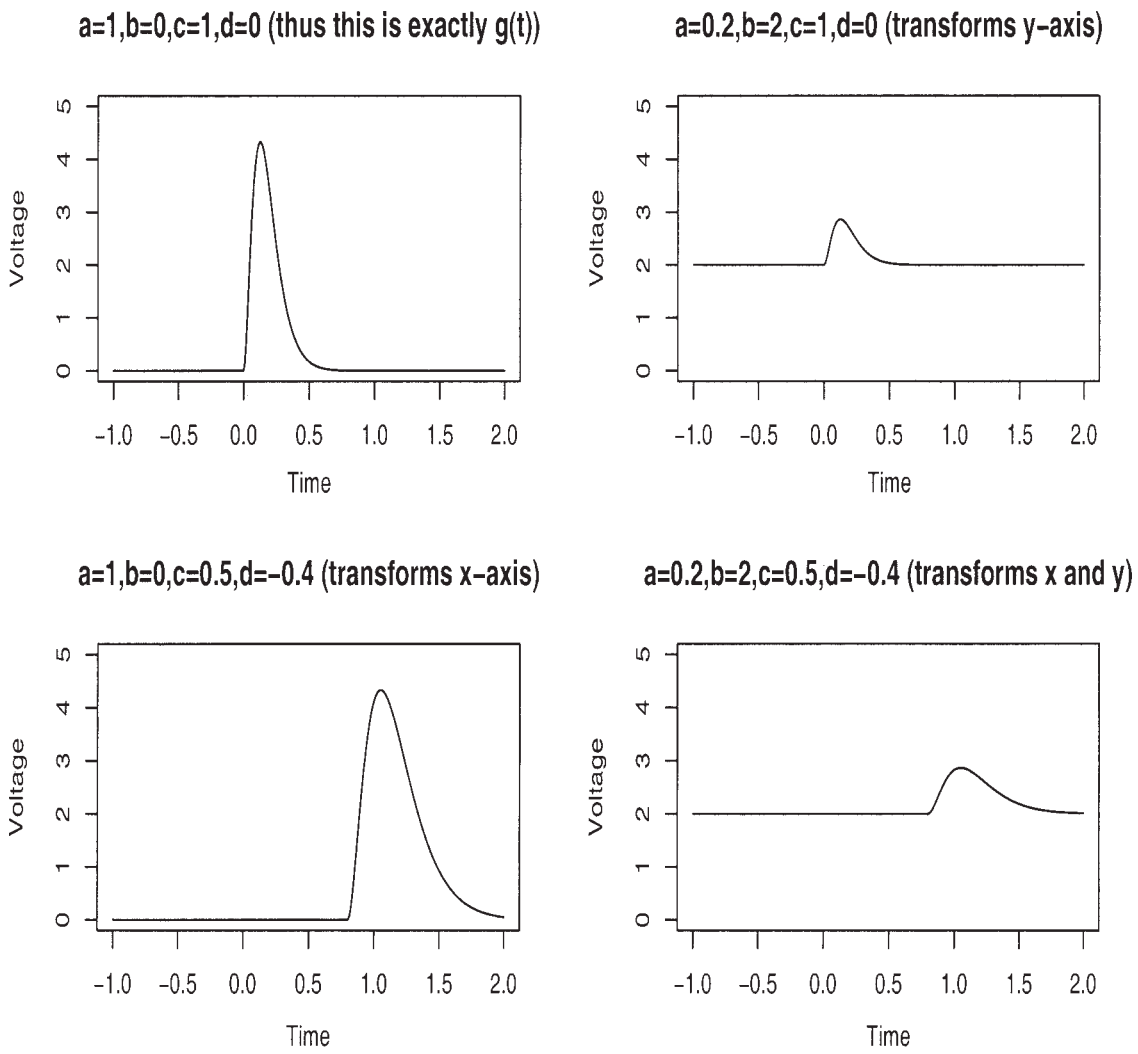

Fig. 2. Example of self-modeling functions. The time point 0 in each plot corresponds to the conclusion of the artifacts prior to the EPSP. Note that here all the functions have been shifted to the right, indicating latency. The mathematics allows for the shift to be to the left, but this does not appear in practice with EPSPs. Note that in the analysis both the voltage and time scales are relative, and their absolute scale is unimportant. The analysis works on their ratios to each other.

Self-modeling functions—main idea

In this section, we discuss self-modeling functions observed without noise. In the next section, we include the noise and discuss estimation techniques. Suppose we have I functions $h_1,\ldots, h_I$. A self-modeling assumption states that there is an underlying function g for which the equation is given as follows:

$$h_i(x) = a_i g(c_i x + d_i) + b_i \quad (1)$$

Thus, for each function of h_i there is a set of coefficients (a_i, b_i, c_i, d_i), called the *self-modeling coefficients*, which relates h_i to g. The coefficients c_i and d_i stretch and shift the x-axis, while the coefficients a_i and b_i stretch and shift the y-axis.

Figure 2 provides a graphical example. The upper-left plot shows the function g. Note that this is equivalent to using $a = 1$, $b = 0$, $c = 1$, and $d = 0$ in Eq. (1). The upper-right plot shows $a = 0.2$, $b = 2$, $c = 1$, and $d = 0$. Since $c = 1$ and $d = 0$ are the "default" values, there is no change to the x-axis, and so the features, such as the peak of the curve, appear in the same place with respect to time. However, the values of a and b stretch and shift the y-axis. The value $a = 0.2$ shrinks the function to 20% of its original value, and the value $b = 2$ increases the baseline value by 2. The bottom-left plot shows $a = 1$, $b = 0$, $c = 0.5$, and $d = -0.4$. Thus, in the bottom-left plot the y-axis is unchanged (and so the peak amplitude and baseline values are the same), but the time axis has been adjusted. The value $c = 0.5$ has stretched the x-axis

by a factor of 2 (corresponding to $1/c$) and the value $d = -0.4$ has moved the resulting curve by -0.4. Thus, the x-axis shifts change the width of the curve and the location of the peak amplitude. The bottom-right plot shows all four coefficients adjusted from their default values. Note that the values chosen for c and d shift the trace to the right. While the mathematics allow the trace to be shifted to the left or to the right, in practice with EPSPs there is only latency, resulting in a shift to the right.

Including noise in the formulation

Let $i = 1,\ldots,I$ index the EPSPs. For each trace, there is a true amount of voltage present in the nerve terminal at a particular time t. We refer to this true amount as $h_i(t)$. Voltage measurements are observed discretely (though at very close time points), and are observed with error. For each trace i, let $t_{i1},\ldots,t_{iJi}$ denote the time points where the voltage is observed (thus we observe J_i voltages for the ith trace), and let $v_{i1},\ldots,v_{iJi}$ be the observed voltages. With the measurement error included,

$$v_{ij} = h_i(t_{ij}) + \varepsilon_{ij}$$

where ε_{ij} is the measurement error and $h_i(t_{ij})$ refers to the true voltage for the ith trace at time point t_{ij}.

We assume that for the purpose of fitting the curves the errors are independent and distributed, with each $\varepsilon_{ij} \sim N(0,\sigma^2)$ for some unknown σ^2. A time series analysis of the residuals indicates that the errors are in fact correlated, with an ARMA (2,2) structure (an autoregressive, moving average process with order two on both components). Nevertheless, for our present purpose, we continue with the independent errors assumption because (1) our central goal is fitting, and as seen through the graphs and predictive correlations, the fit is quite good with this assumption, and (2) incorporating the more realistic error structure complicates the fitting process without providing a significant improvement in fit. We hope to find efficient ways to include the more realistic error structure in future work.

If the h_i functions follow a self-modeling structure as in Eq. (1), then the equation is given as follows:

$$v_{ij} = h_i(t_{ij}) + \varepsilon_{ij} + a_i g(c_i t_{ij} + d_i) + b_i + \varepsilon_{ij}$$

and thus each v_{ij} is independent and distributed.

$$v_{ij} \sim N(a_i g(c_i t_{ij} + d_i) + b_i, \sigma^2)$$

Given g, the self-modeling coefficients may be estimated for each EPSP through maximum likelihood (discussed in the appendix). Given all the self-modeling coefficients for each EPSP, the underlying function g may be estimated by weighted linear regression (discussed in the appendix). Overall, the underlying function g and the self-modeling coefficients can be estimated through an iterative procedure discussed in the appendix.

Using g to estimate functionals for each EPSP

After estimating g and the self-modeling coefficients, we have an estimate of the true voltage at each time point for each trace, with the noise smoothed out, just like a linear regression line (in fact, the spline estimate of g is a multiple regression estimate itself, with similarities to polynomial regression). This line may then be used to estimate the functionals discussed in prior papers.

Prior analyses, without the machinery to smooth, did not eliminate noise when computing functionals, such as peak amplitude, AUC, rise time, or others. With the current smoothed estimate of the underlying equation,

$$h_i(t_{ij}) = a_i g(c_i t_{ij} + d_i) + b_i$$

we may compute functionals from each h_i and acquire noise reduced estimates of the corresponding functionals. This sections relate computation of those functionals to the corresponding functionals of g and the self-modeling coefficients.

1. AUC: the area under the curve is just the integral of the function,

$$\begin{aligned}\mathrm{AUC}_i &= \int h_i(t)\,dt = \int a_i g(c_i t + d_i)\,dt \\ &= a_i c_i \int g(t)\,dt = a_i c_i \mathrm{AUC}_g\end{aligned}$$

   where AUC_g is just the area under the curve g.
2. Peak amplitude: the peak amplitude is just scaled by the constant a_i, and so

$$\max_t h_i(t) = a_i \max_t g(t)$$

3. Location of peak amplitude: Let $t_g^{\max}$ be the location of the peak of g, so $t_g^{\max} = \arg\max_t g(t)$. The location of the peak amplitude for h_i occurs at the point t, which maps to $t_g^{\max}$ through the transformation $c_i t + d_i = t_g^{\max}$, and thus the maximum for h_i occurs at

$$t_{h_i}^{\max} = \frac{t_g^{\max} - d_i}{c_i}$$

4. Latency: We define latency as the time between the end of the last artifact (removed in the preprocessing) and the beginning of the rise of the curve. The start of the base function g occurs when $t = 0$. The start of the transformed function is the point t where $ct + d = 0$, corresponding to $t = -d/c$.

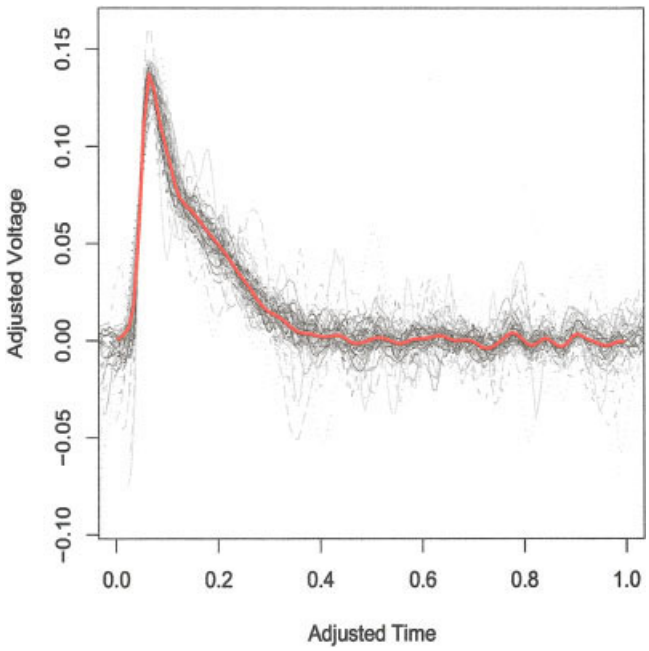

Fig. 3. Aligned data (aligned in both time and voltage), where all EPSPs are superimposed on each other. The red curve is the estimated function g, while the gray/black curves correspond to the aligned traces. Larger original EPSPs are darker, illustrating which curves are given more weight in the alignment process.

5. Decay time: The decay time is the time between when the peak amplitude occurs and when the curve has descending to p times the peak amplitude (p is often chosen to be 0.37, see Hille, 2001). The coefficient c_i determines the stretching of the time axis, and so if τ_g is the decay time for g, the corresponding decay time for h_i is

$$\tau_{h_i} = c_i \tau_g$$

6. Rise time: The rise time is similar to the decay time, in which you take the rise time for g and multiply it by c_i.

RESULTS

Fitted self-modeling coefficients and estimated *g*

Implementing the numerical procedures in the appendix, we estimated g and the self-modeling coefficients for each trace in Figure 1. For each curve, this results in estimated coefficients $\hat{a}_i$, $\hat{b}_i$, $\hat{c}_i$, and $\hat{d}_i$. The results may be viewed in two ways. First, Figure 3 shows the adjusted curves laid atop each other. This plot was constructed by finding

$$t'_{ij} = \hat{c}_i t_{ij} + \hat{d}_i \quad v'_{ij} = \frac{v_{ij} - b_i}{a_i}$$

for each i and j. This produces data governed by the equation,

$$v'_{ij} \sim N(g(t'_{ij}), \sigma^2/a_i^2$$

Thus, all the (t'_{ij}, v'_{ij}) values have expectations along the function g, though their variances are inversely propor-

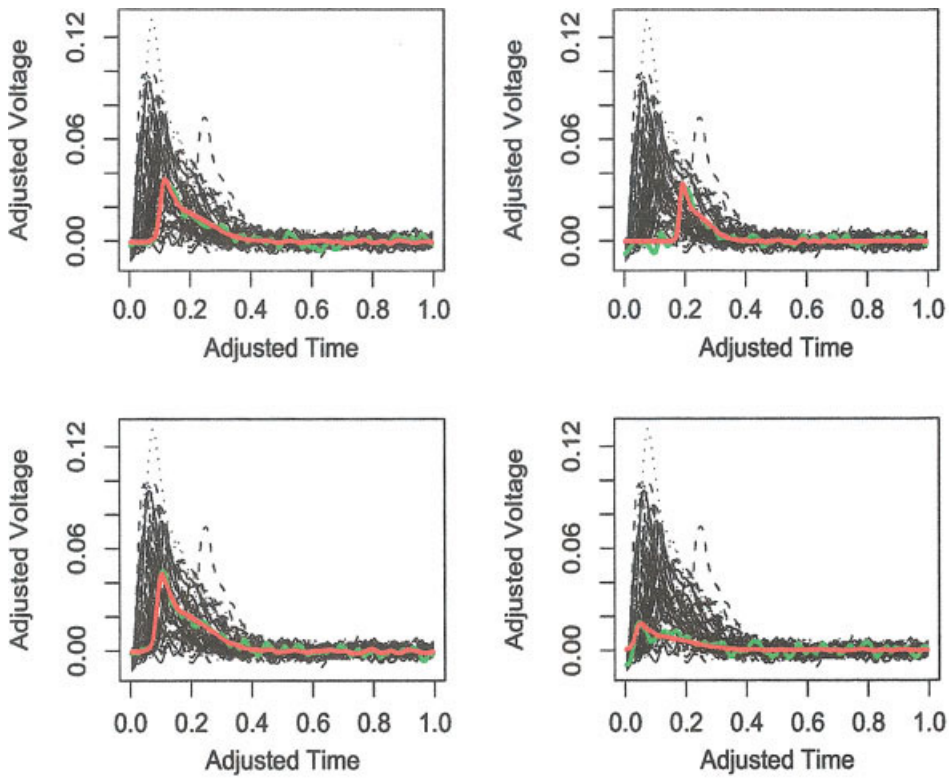

Fig. 4. Fitted data showing the fits to four of the traces. Time 0 corresponds to the conclusions of the artifacts prior to the EPSP. Except for an affine transformation of the time axis, each subplot is identical to Figure 1, with an individual trace in green and the fitted estimate of that same trace in red.

tional to a_i^2. Intuitively, this is reasonable. Smaller curves must be scaled up relative to larger curves, and thus the noise in smaller curves is scaled up. Aligned curves from smaller original curves have more noise than aligned curves from larger original curves. Figure 3 incorporates this weighting by gray-scaling the results according to their weight. Curves that were originally taller are darker than curves that were lighter. Note the darker curves have less variation. The estimated g is shown in red atop the scaled curves. As can be seen, the curves fit over each other well when scaled.

Figure 4 shows an alternative method for viewing the results. Each of the four plots shows the raw data from Figure 1 in black. In each plot, the data from a single EPSP is shown in green, and the fitted curve $h_i(t)$ is plotted in red. Thus, the four green curves represent four noisy EPSPs, and the four red curves are all self-modeling transformations of the estimated underlying g.

Note that we make no claims that this is an exact fit of the data, simply that self-modeling transformations of g account for the overwhelming majority of the variance of this dataset. The correlation between the actual data and the fitted values (i.e., between the red and green curves in Figure 4, only applied to all the traces) is 0.982.

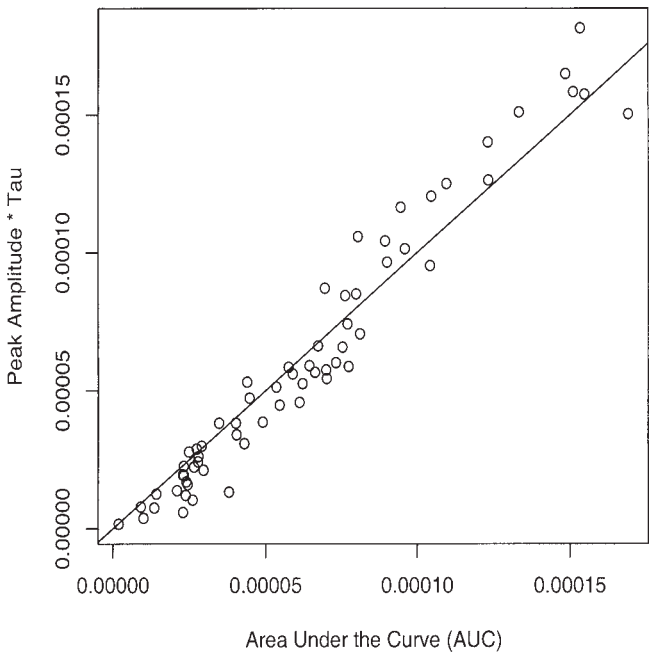

Fig. 5. Agreement between AUC and the product of peak amplitude and decay time. The self-modeling structure predicts that, except for noise, these two quantities should be equal. The values shown here were computed numerically using the raw data, without the self-modeling structure. The correlation here is 0.962, indicating that the prediction of the self-modeling structure is accurate.

Illustration of relationships between previously used functionals

The relationships in Materials and Methods section imply there is a limit to how valuable it is to look at multiple functionals of the traces. For example, the rise and decay times are both proportional to the c_i values. If one is already looking at decay time (proportional to c_i) and peak amplitude (proportional to a_i), then AUC is proportional to the product of these values (realizing that all these values have noise attached when computed directly from the data).

To illustrate this, Figure 5 shows the relationships between unsmoothed estimates of the AUC, peak amplitude, and decay time. We computed these functionals using the method described by Lancaster et al. (2005), but any method could be used to obtain these unsmoothed functionals. The point is that the data in Figure 5 have *not* been run the self-modeling algorithm, but rather were computed from the raw data. Figure 5 shows the AUC for each of the 61 EPSPs in Figure 1, plotted against the product of the peak amplitude and decay time. According to the aforementioned results, these values should be similar, which is apparent from the plot. There are differences due to noise, but the correlation between the values in the plot is 96.2%.

Combining information across several curves

Using the self-modeling structure can also allow some fine structure to be observed, which cannot be observed in individual curves. This is because the base function g is estimated using information from all the individual curves, an operation which averages out noise. A structure that may be obscured by noise in any single curve may be more clearly observed when the curves are overlaid atop each other.

Returning to Figure 3, note the bend that occurs on the downslope when the adjusted time is around 0.15 (adjusted time, would have to be rescaled to get on the time scale of the original curves). This feature is not readily observed in the original curves. In the aligned curves, we can see that it is both reflected in the "mean" curve g and that it is present in the gray aligned curves. We do not have a ready explanation for what causes this bend biologically, but having this information allows us to ask new questions and will perhaps lead to new insights.

Code available on the internet

The methods herein are implemented in a statistical language called "S," using the freeware software "R." "R" is maintained by a consortium of statisticians and others, and may be downloaded at http://www.r-project.org/. The code used here may be found at http://www.ms.uky.edu/~viele/epsps/selfmodel/selfmodel.html, which contains the following files (and a more detailed description).

1. selfmodel.pdf: contains a preprint of this paper in pdf format.
2. selfmodel.RData—an R data file: After installing the freeware program R (instructions given later), double clicking on this file will open R, with the data and code preloaded.
3. start.r: a text file of R commands that contains commands to initialize the information in the selfmodel.RData file. The purpose of this arrangement is that the initial commands generate a very large amount of calculations. The file sizes would be large (~20 MB), if these calculations were done in advance. With the current scheme, the files can be much smaller.
4. epsps.txt (optional): a text file of the data from the paper (61 traces with 200 observations each). This data is preloaded into selfmodel.RData.
5. selfmodel.r (optional): a text file of the code with documentation. The code is already preloaded in selfmodel.RData as well, but this file also contains documentation.

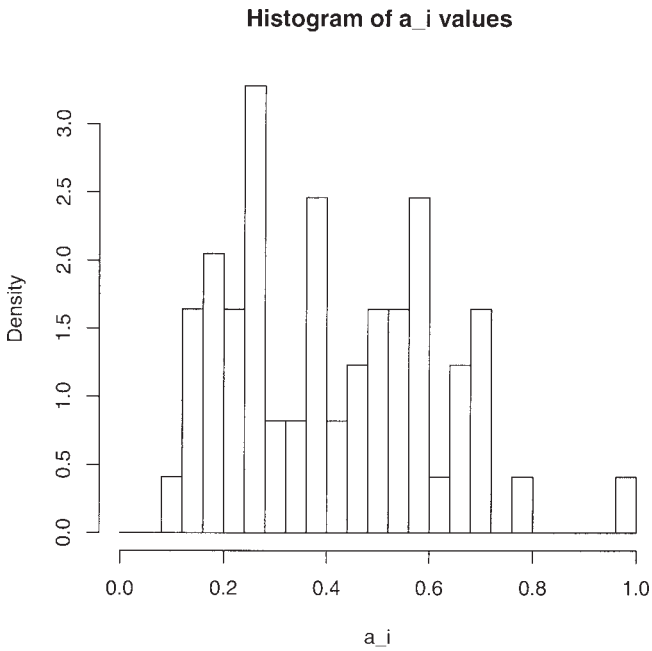

Fig. 6. Histogram of the a_i coefficients from the alignment of the data in Figure 1.

You will need to download R from the internet and install it on your computer. This is described in more detail on the web page. After you have installed R and loaded the necessary files (again instructions provided on the web page), you may replicate the results in this article or analyze your own data. We have set up the file selfmodel.RData so that R is preloaded with the results of alignment. You can either explore those results and replicate some of the figures in the manuscript, or run the algorithm on a data (either your own or the data preloaded from the article). We have implemented it this way because the code takes so long to run (hours), and we wanted to allow people to look at results immediately. More detailed instructions are available at the web site, but after reading data into a variable epsps, it may be aligned with a single command

```
myalignment = alignall (epsps, numiter = 5)
```

To see the results as in Figures 3 or 4, type either or both of

```
plot.alignment (myalignment)
plot.fitmat (myalignment, pause = TRUE)
```

The latter command shows the curves one at a time. You have to hit return to go from one curve to the next. Again, more detailed instructions are available on the web site.

S is not intended as a fast language, and depending on the number of EPSPs the code may take a moderate time (20–25 min on a 3 GHz Pentium 4) to execute. To pass the time, the code will print updates concerning which iteration of the algorithm it is performing and which trace is being aligned. There are also graphical updates of the progress of the differential evolution (DE) algorithm described in the appendix. NOTE: the code will produce several warning messages stating that ill-conditioned bases have appeared. This is due to evaluating the spline outside the interval (0,1). The warning messages may be safely ignored.

Grouping the self-modeling coefficients

The self-modeling coefficients can be clustered according to the same methods discussed by Viele et al. (2003). Figure 6 shows a histogram of the a_i coefficients (recall these are proportional to the peak amplitudes) for the alignment of the original data in Figure 1. For these data, there is no indication that the data come from distinct subgroups. Using the BIC method from Viele et al. (2003), we conclude $n = 1$ (it is important not to overestimate the importance of chance peaks in a histogram, the BIC criteria accounts for the fact that peaks may occur by chance). The BIC method thus concludes there is only type on EPSP in these data.

It is not always the case that there is no evidence of clustering. A second set of data, from a separate experiment, is shown in Figure 7. After performing

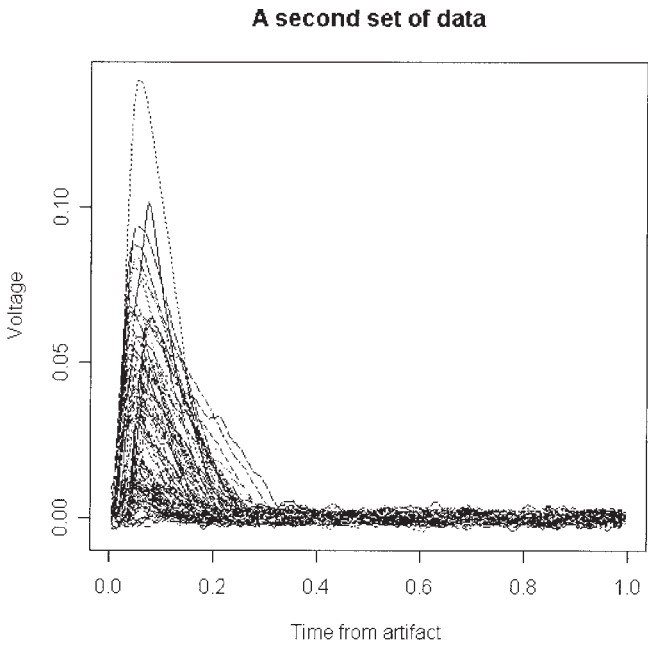

Fig. 7. A second dataset used in the Results section is used to illustrate how the self-modeling coefficients can be clustered.

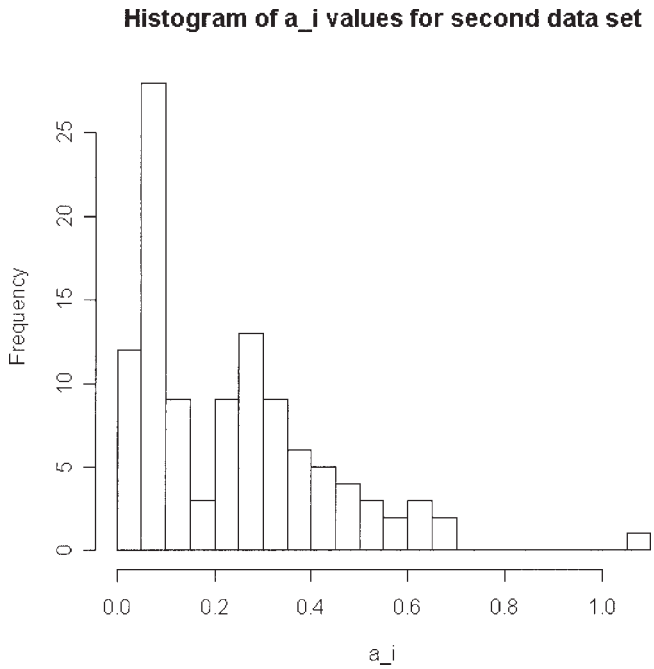

Fig. 8. Histogram of the a_i coefficients from the alignment of the data in Figure 7.

all the same manipulations discussed in the Materials and Methods section, we can align these 110 traces and acquire the self-modeling coefficients for each trace. The histogram of the a_i coefficients in Figure 8 clearly shows multiple peaks, indicating that a_i coefficients cluster into subgroups.

Applying the mixture model discussed by Viele et al. (2003), we conclude $n = 2$, with the corresponding probabilities being $p_1 = 0.39$ and $p_2 = 0.61$. The two peaks are apparent in Figure 7.

DISCUSSION

Any type of statistical analysis, including mixture modeling, benefits when the signal to noise ratio of the data is increased. The spline smoothing here decreases the noise involved in computing functionals, such as peak amplitude and decay rate, which should provide a methodological improvement to all analyses of these functionals. As an example, mixture modeling attempts to fit underlying subpopulations to the overall distribution of the functional. Suppose the peak amplitudes appeared as in Figure 9A, the underlying subpopulations are shown with dashed lines while the overall population is shown with a solid line. Here, it is difficult to discern from the solid black curve that there are two underlying subpopulations. If the noise in the data is reduced, however (corresponding to decreasing the variation in the underlying subpopulations), we arrive at Figure 9B, where the underlying subpopulations are readily apparent as multiple modes in the overall solid curve.

In this study, we have shown a relatively rapid method to characterize quantal responses with a variety of parameters. Such parameters are significant for indexing changes in receptor sensitivity, density of receptors, and examining mechanisms of pharmacological agents at synapses. In addition, the relationship of the parameters to each other offers a further ability to examine their relationships. For example, if an agonistic compound had a high affinity for the postsynaptic receptors, the peak amplitude might increase, but perhaps the compound enhanced desensitization which would then lead to a shorter decay and potentially a reduction in the area of the quantal event. The procedures presented also offer a quick ability to assess the potential changes presynaptically in vesicle filling rates, during rapid stimulation or experimental pharmacological manipulations (Logsdon et al., 2005). The analysis is not only relevant to the glutamate-ergic synapses at the crayfish NMJ but also to all chemical synapses across species. The relatively new views of why synaptic responses do not fit precisely in quantal packets due to membrane traffic and glycocalyx within the vesicle as well as the synaptic matrix can also be addressed readily by the use of computational assessments presented (Kriebel et al., 2001; Vautrin and Barker, 2003).

The "hump" on the decay of the discrete events is also depicted in the averaged fitted spline curve g.

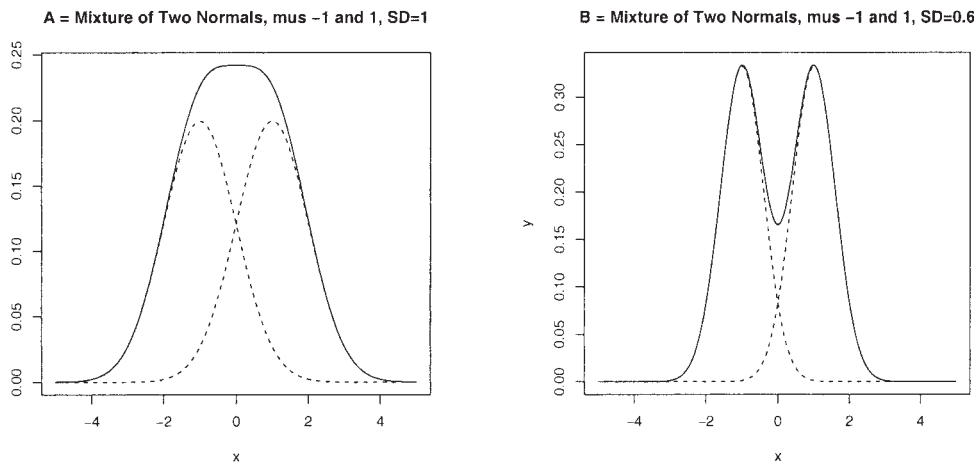

Fig. 9. Mixtures of normals with the same means but differing variances. It is easiest to identify the subgroups when the variance of the underlying subgroups is small. Thus, any method which reduces these variances will make it easier to cluster the EPSPs.

This observed phenomena have been observed before, since decay of quantal event has been shown to be fitted by two exponential curves. The initial decay is rapid followed by a slower decay, which has not only been documented to occur at the crustacean NMJ (Parnas et al., 1991) but also at glutamate-ergic NMJs in Drosophila (Pawlu et al., 2004). The glutamate receptor subtypes are slightly different in kinetic properties and pharmacology between these species (Bhatt and Cooper, 2005; Dudel et al., 1992; Shinozake and Ishida, 1981). The variation in the decay "hump" has been postulated to be due to fusion pore kinetics of the vesicle with the presynaptic membrane. The time of fusion determines the time course of glutamate discharge and the quantal decay (Pawlu et al., 2004). This would manifest to a variable concentration of glutamate at the receptors and altering the desensitization during the overall transmission process. So there does appear to be a physiological relevance, thus it is of interest to characterize the hump in the decay (Pawlu et al., 2004).

The self-modeling structure described earlier accounts for the 99.4% of the variation (R^2) of the full dataset we examined in this article, and thus simplifies the data from I separate traces into one baseline trace g and four coefficients a_i, b_i, c_i, d_i for each trace representing their alignment to g. These four coefficients have clean relationships to all the functionals commonly used in the literature, such as peak amplitude, T, or AUC. In addition, since all these functionals are related through the same set of four coefficients, the self-modeling structure allows us to see relationships between the functionals, such as AUC, being proportional to peak amplitude times T, as in Figure 5.

By performing a regression analysis (the spline fit to g is a multiple regression), we also get a smoothed, and hence less variable, fit to the data. This removes noise, and thus any subsequent analyses should be more statistically powerful. As shown in Results section, we may also observe fine structure in the curves that may motivate further research questions.

For further analyses, it would be useful to incorporate dependence among the residuals, perhaps using a method similar to Altman and Villareal (2004), and also examine any structure in the residual curves (the functions remaining after h_i has been removed). Often useful information is gleaned only after the "main structure" of the data has been removed (Viele, 2001).

REFERENCES

Altman N, Villareal J. 2004. Self-modeling regression with random effects using penalized splines. Can J Stat 32:251–268.

Atwood HL, Cooper RL. 1995. Functional and structural parallels in crustaceans and Drosophila neuromuscular systems. Am Zool 35:556–565.

Atwood HL, Cooper RL. 1996a. Assessing ultrastructure of crustacean and insect neuromuscular junctions. J Neurosci Methods 69:51–58.

Atwood HL, Cooper RL. 1996b. Synaptic diversity and differentiation: crustacean neuromuscular junctions. Invert Neurosci 1:291–307.

Bhatt D, Cooper RL. 2005. The pharmacological and physiological profile of glutamate receptors at the Drosophila larval neuromuscular junction. Physiol Entomol 30:305–310.

Cooper RL, Marin L, Atwood HL. 1995a. Synaptic differentiation of a single motor neuron: conjoint definition of transmitter release, presynaptic calcium signals, and ultrastructure. J Neurosci 15: 4209–4222.

Cooper RL, Stewart BA, Wojtowicz JM, Wang S, Atwood HL. 1995b. Quantal measurement and analysis methods compared for crayfish and Drosophila neuromuscular junctions and rat hippocampus. J Neurosci Methods 61:67–78.

Cooper RL, Harrington C, Marin L, Atwood HL. 1996a. Quantal release at visualized terminals of crayfish motor axon: intraterminal and regional differences. J Comp Neurol 375:583–600.

Cooper RL, Winslow J, Govind CK, Atwood HL. 1996b. Synaptic structural complexity as a factor enhancing probability of calcium-mediated transmitter release. J Neurophysiol 75:2451–2466.

Del Castillo J, Katz B. 1954a. Quantal components of the end-plate potential. J Physiol 124:560–573.

Del Castillo J, Katz B. 1954b. Statistical factors involved in neuromuscular facilitation and depression. J Physiol 124:574–585.

Dudel J. 1981. The effect of reduced calcium on quantal unit current and release at the crayfish neuromuscular junction. Pflugers Arch 391:35–40.

Dudel J, Kuffler SW. 1961. The quantal nature of transmission and spontaneous miniature potentials at the crayfish neuromuscular junction. J Physiol 55:514–529.
Dudel J, Franke C, Hatt H. 1992.Rapid activation and desensitization of transmitter-liganded receptor channels by pulses of agonists. In: Narahashi T, editor. Ion Channels, 3rd ed. New York: Plenum. p 207–260.
Govind CK, Pearce J, Wojtowicz JM, Atwood HL. 1994. Strong and weak synaptic differentiation in the crayfish opener muscle: structural correlates. Synapse 16:45–58.
Hatt H, Smith DO. 1976. Non-uniform probabilities of quantal release at the crayfish neuromuscular junction. J Physiol 259: 395–404.
Hille B. 2001. Ion channels of excitable membranes. Hille B., ed. 3rd ed. Sunderland, MA: Sinauer Associates. p 10–12.
Johnson EW, Wernig A. 1971. The binomial nature of transmitter release at the crayfish neuromuscular junction. J Physiol 281: 757–767.
Kneip A, Engel J. 1995. Model estimation in nonlinear regression under shape invariance. Ann Stat 25:532–551.
Kneip A, Gasser T. 1988. Convergence and consistency results for self-modeling nonlinear regression. Ann Stat 16:82–112.
Kriebel ME, Keller B, Silver RB, Fox GQ, Pappas GD. 2001. Porocytosis: a new approach to synaptic function. Brain Res Brain Res Rev 38(1/2):20–32.
Lancaster M, Viele K, Johnstone A, Cooper RL. 2005. Automated classification of evoked post-synaptic potentials. Currently in review. Available at http://www.ms.uky.edu/viele/epsps/epsps.html.
Lawton W, Sylvestre E, Maggio M. 1972. Self modeling non-linear regression. Technometrics 14:513–532.
Lindstrom M. 1995. Self-modeling with random shift and scale parameters and a free-knot spline shape function. Stat Med 14:2009–2021.
Logsdon S, Johnstone AFM, Viele K, Cooper RL. 2005. The regulation of synaptic vesicles pools within motor nerve terminals during short-term facilitation and neuromodulation. J Appl Physiol 100:662–671.
Magrassi L, Purves D, Lichtman JW. 1987. Fluorescent probes that stain living nerve terminals. J Neurosci 7:1207–1214.
Parnas I, Dudel J, Atwood HL. 1991. Synaptic transmission in decentralized axons of rock lobster. J Neurosci 11:1309–1135.
Pawlu C, DiAntonio A, Heckmann M. 2004. Postfusional control of quantal current shape. Neuron 42:607–618.
Price K, Storn R, Lampinen J. 2005. Differential evolution: a practical approach to global optimization. New York: Springer.
Shinozaki H, Ishida M. 1981. Quisqualate action on the crayfish neuromuscular junction. J Pharmacobiodyn 4:42–48.
Uteshev VV, Pennefather PS. 1997. Analytical description of the activation of multi-state receptors by continuous neurotransmitter signals at brain synapses. Biophys J 72:1127–1134.
Vautrin J, Barker JL. 2003. Presynaptic quantal plasticity: Katz's orginal hypothesis revisited. Synapse 47:184–199.
Viele K. 2001. Evaluating fit in functional data analysis using model embeddings. Can J Stat 29:51–66.
Viele K, Stromberg A, Cooper RL. 2003. Determining the number of release sites within the nerve terminal by statistical analysis of synaptic current characteristics. Synapse 47:15–25.
Wernig A. 1972. Changes in statistical parameters during facilitation at the crayfish neuromuscular junction. J Physiol 226:751–759.

APPENDIX

This appendix provides the technical details involved in aligning the curves. We first assume that a transformation of the x-axis has occurred, so that all the time values are strictly between 0 and 1. This transformation may be inverted at the end of the analysis to get back on the scale of the original data. The purpose of this transformation is to provide a unified system of starting values for the numerical algorithms. If a matrix of voltage values is provided to the algorithm available on the internet, the algorithm automatically performs this transformation by default.

Aligning a single trace to a fixed function g

Suppose we are working to align a single trace to a function g. The mathematical structure assumed states that the voltage values observed, $\boldsymbol{v} = (v_1,\ldots,v_J)$, are independent, with each $v_j \sim N(ag(ct_j + d) + b, \sigma^2)$, where t_j is the time v_j is observed. Our goal here is to estimate the coefficients (a, b, c, d). We choose to do this by maximum likelihood, thus choosing the (a, b, c, d) that maximizes the likelihood function.

$$f(\boldsymbol{v}|a,b,c,d) = \prod_{j=1}^{j} N(ag(ct_j+d)+b,\sigma^2)(v_j) = (2\pi\sigma^2)^{-J/2} \times \exp\left\{\frac{-1}{2\sigma^2}\sum_{j=1}^{j}(v_j - ag(ct_j+d)-b)^2\right\} \quad (2)$$

Note that σ^2 is irrelevant to the maximization of the likelihood with respect to (a, b, c, d), and so we may treat σ^2 as a constant. Thus, the maximization problem reduces to minimize the squared differences.

$$h(a,b,c,d) = \sum_{j=1}^{J}(v_j - ag(ct_j+d) - b)^2$$

Unfortunately, we can only get part of the way analytically. Taking derivatives and setting to 0 results in the equations,

$$\frac{\partial h(a,b,c,d)}{\partial a} = (-2)\sum_{j=1}^{J}(v_j - ag(ct_j+d) - b)\,g(ct_j+d) = 0$$

$$\frac{\partial h(a,b,c,d)}{\partial b} = (-2)\sum_{j=1}^{J}(v_j - ag(ct_j+d) - b) = 0$$

$$\frac{\partial h(a,b,c,d)}{\partial c} = (-2)\sum_{j=1}^{J}(v_j - ag(ct_j+d) - b)\,a\,g'(ct_j+d)t_j = 0$$

$$\frac{\partial h(a,b,c,d)}{\partial d} = (-2)\sum_{j=1}^{J}(v_j - ag(ct_j+d) - b)\,a\,g'(ct_j+d) = 0$$

It is not possible to simultaneously solve these four equations analytically. However, it is possible to solve for a and b in terms of c and d. For fixed c and d defining

$$\bar{v} = J^{-1}\sum_{j=1}^{J} v_j \quad \bar{g} = J^{-1}\sum_{j=1}^{J} g(ct_j+d)$$

Then,

$$\hat{b} = J^{-1}\sum_{j=1}^{J}(v_j - ag(ct_j+d)) = \bar{v} - \hat{a}\bar{g}$$

Replacing this in the equation for a,

$$\sum_{j=1}^{J}(v_j - \hat{a}g(ct_j+d) - \overline{v} + \hat{a}\overline{g})\, g(ct_j+d) = 0$$

$$\sum_{j=1}^{J}(v_j - \overline{v} - \hat{a}[g(ct_j+d) - \overline{g}])\, g(ct_j+d) = 0$$

$$\hat{a} = \frac{\sum_{j=1}^{J}(v_j - \overline{v})g(ct_j+d)}{\sum_{j=1}^{J}[g(ct_j+d) - \overline{g}]g(ct_j+d)}$$

If one assumes a common baseline value $b = 0$, then

$$\hat{a} = \frac{\sum_{j=1}^{J} v_j g(ct_j+d)}{\sum_{j=1}^{J}[g(ct_j+d)]^2}$$

If $g(ct_j + d)$ is identical for all values of t_j (this occurs, for example, if c and d are such that the transformation moves the $ct_j + d$ values completely outside (0,1), then one faces division by zero errors in the earlier calculations. However, let g_0 be the constant value of g and note the loglikelihood, in which case

$$f(\boldsymbol{v}|a,b,c,d) = \prod_{j=1}^{J} N(ag_0 + b, \sigma^2)(v_j)$$
$$= (2\pi\sigma^2)^{-J/2} \exp\left\{\frac{-1}{2\sigma^2}\sum_{j=1}^{J}(v_j - ag_0 - b)^2\right\}$$

Maximizing the loglikelihood is equivalent to minimizing

$$\sum_{j=1}^{J}(v_j - ag_0 - b)^2$$

which occurs when $ag_0 + b = \bar{v}$. Note that there are several cases. If no baseline is being set (so $b = 0$), then $a = \bar{v}/g_0$ if $g_0 \neq 0$, otherwise the value of a is irrelevant to the likelihood (so setting $a = 0$ is acceptable). If a baseline is being set, the easiest solution is to set $b = \bar{v}$ and $a = 0$. For the actual maximum of the likelihood, these special cases are typically not necessary (the maximum occurs for a value for $g(ct_j + d)$ is not constant), but they may be useful when running the numerical algorithms.

There is no way to analytically solve for the remaining coefficients, c and d. However, we may still find estimates through numerical maximization. We attempted to use Newton Raphson maximization originally, but found that the NR algorithm, while fast, is particularly sensitive to the choice of starting values. NR works best for quadratic likelihoods, something we do not observe here. For many seemingly reasonable starting values (starting values that visually lined up the trace and the function g), the NR algorithm either diverged or converged to a stationary point that is obviously inferior. This is a common problem with NR in many contexts.

Thus, we choose a more robust, but slower, numerical algorithm called DE (Price et al., 2005). DE is an example of a genetic algorithm, which begins with a set of parameter points and then iteratively improves ("evolves") those values toward the best point available. For best performance, DE requires that we begin with a rectangular region that contains the best values of c and d (recall a and b can be found from c and d, and so technically we are maximizing a profile likelihood).

We have found that maximizing a transformation of (c,d) is more stable than estimating (c,d) directly. We estimate (c,t_0), where t_0 measures the location of the transformed peak. If the peak of g occurs when $t = t^*$, then the peak of h will occur when $ct_0 + d = t^*$, resulting in $t_0 = (t^* - d)/c$. After the maximum likelihood using (c,t_0) has been found, one can transform back to (c,d) using $d = t^* - ct_0$.

The DE algorithm we specify requires an initial "rectangle" of values. We specify $t_0 \in (0, 1)$, since we assume that the transformed peak is within the range of the observed time values. Note that the coefficient c is a "stretching" coefficient that either enlarges or shrinks the time axis. Thus, an estimate of c may be constructed by examining the width of the peak for the EPSP and the width of the peak for the function g.

Recall t^* as the time point where g is maximized and t_0 as the point where h is maximized. Define w_0 as the first time point past t_0 where v_j is less than half the maximal v_j. Define w^* as the first point past t^* where g is half the maximal value of g. Thus, the distance $w^* - t^*$ is a measure of the decay time of g, while $w_0 - t_0$ is the corresponding measure of the decay time of h. The coefficient c measures the ratio of these decay rates. Thus, a crude estimate of c is

$$c^* = \frac{w^* - t^*}{w_0 - t_0}$$

In addition to our assumption $t_0 \in (0, 1)$, we also assume $c_0 \in (0, 5c^*)$. The lower bound $c > 0$ comes from assuming that there are no "mirror image" traces while the constraint $c < 5c^*$ is motivated simply by allowing a fair amount of error in the estimate c^*.

Our implementation of DE is simpler than the general version. We

1. Choose a starting set of points $p_1 = (c_1,t_1),\ldots,p_M = (c_M,t_M)$ randomly from within the rectangular region, with $c_0 \in (0, 5c^*)$ and $t_0 \in (0, 1)$, where M can be chosen by the user. We typically use $M = 40$.

2. Iterate the following until both ($\max_m c_m - \min_m c_m$)/5c^* and ($\max_m t_m - \min_m t_m$) are less than ε for some small ε (we typically chose $\varepsilon = 0.0001$. This says let the system evolve until the set of points have converged to a region 1/10,000 as wide as the original window in both dimensions (c and t). The iterations consist of
   a. for each m, generate j_{1m} and j_{2m} randomly from the integers (1,...,M). Let

$$p'_m = (1 - \text{drift})[p_m + \alpha(p_{j_{1m}} - p_{j_{2m}}] + \text{drift}\, p_{\max}$$

   where drift and α are user controlled parameters (by default, we choose drift = 0.05 and α = 0.2), and $p_{\max}$ is the point that currently has the largest function value.
   These p'_m points are "candidates." The equation given earlier represents adding random noise to each of the current points, with a slight "drag" toward the point where the current maximum has been achieved.
   b. For any point p_m such that the loglikelihood (equivalent to using Eq. (2)) is higher at p'_m than p_m, set $p_m = p'_m$ (so move the point to the candidate value). For all other points, do nothing (the candidate offers no improvement).
   c. Find the points $p_{\max}$ and $p_{\min}$ corresponding to the maximal and minimal function values. Then move the point $p_{\min}$ to (driftmin)$p_{\max}$ + (1 − driftmin) $p_{\min}$, where we set driftmin = 0.5 by default. The purpose of this step is to speed convergence, but this requires that M is sufficiently large.

3. Find the (c_m,t_m) pair among the points that has the largest value of the loglikelihood. This is the approximate maximum likelihood estimate. Transform the result back to (c,d).

With known *c* and *d* for each function, estimating the function *g*

Note that g is not a known quantity, and must be estimated from the data. Thus, this section describes how to compute an estimate of g, given that all the self-modeling coefficients are known.

Let (a_1, b_1, c_1, d_1),..., (a_n, b_n, c_n, d_n) be given (if only c and d are given, find the MLEs of a and b as in the previous section), where the subscript refers to the trace. The assumption here is that

$$v_{ij} \sim N(a_i g(c_i t_{ij} + d_i) + b_i, \sigma^2)$$

Performing a linear transformation, this implies

$$v'_{ij} = \frac{v_{ij} - b_i}{a_i} \sim N(g(C_i t_{ij} + d_i),\ \sigma^2/a_i^2)$$

Thus, the v'_{ij} values are on the same mean scale, though not each transformed trace now has a different variance σ^2/a_i^2. Defining $t'_{ij} = c_i t_{ij} + d_i$, we find

$$v'_{ij} \sim N(g(t'_{ij}),\ \sigma^2/a_i^2)$$

Thus, to fit the spline model, if you have a spline basis for g, you need to compute v'_{ij} and t'_{ij} for each i and j, and then perform a weighted regression, where the weights are the a_i^2 values.

A complete algorithm

Obviously, both g and the self-modeling coefficients must be esimated from the data. The previous sections have described how to get estimates of the self-modeling coefficients from a known g and how to get an estimate of g from the self-modeling coefficients. Our complete algorithm consists of iterating these two steps until convergence. Thus, the complete algorithm is given as follows:

1. Estimate g with a spline fit to the trace with the largest peak amplitude (this is chosen, because it has the largest signal to noise ratio). This is just used as a starting value for the algorithm.
2. Iterate the following two steps until convergence (typically five steps is sufficient, and often fewer are required for reasonable results).
   a. With g fixed, estimate (a_i, b_i, c_i, d_i) for each trace, using the method described in the Discussion section.
   b. With the self-modeling coefficients fixed, estimate g using the method described in the Discussion section.