Mathematical Techniques for Neuromuscular Analysis

JF Williams[*] *Geertje Hek*[†] *Alistair Vardy*[‡]
Vivi Rottschäfer[§] *Jan Bouwe van den Berg*[‡] *Joost Hulshof*[‡]

Abstract

In the central nervous system, *alpha*-motor neurons play a key role in the chain that results in muscles producing force. A new non-invasive technique to measure the electrical activity involved with force production called High Density Surface Electromyography (HDsEMG) has been proven to be effective in providing novel clinical information on the way *alpha*-motor neurons control the muscles. This is important for the monitoring of the progression of certain neuromuscular disorders such as polio. The result of HDsEMG is, however, very difficult to interpret. In this paper we augment the usefulness of HDsEMG with automated mathematical techniques to aid the Motor Unit Number Estimation (MUNE) problem. Also, we create a stochastic model for the firing behavior of an *alpha*-motor neuron.

1 Introduction

The movement of parts of the body is an area studied by many disciplines. Combining the knowledge and techniques of multiple disciplines can help solve problems related to movement in a more fruitful way. Here, we will combine medical science, neuroscience and mathematics. First, we address two questions from medical science. Next we describe the techniques used in neuroscience with which we collect the relevant data. Finally, we describe the mathematics needed to process the data and reflect on the questions posed in this introduction.

Movement requires force produced by muscles. Before the muscles contract a chain of events take place. These events form the basis of the questions we will pose later. For descriptive purposes we assume that the origin of movement is activity in the brain (this is not the only starting point, eg. reflexes do not need intervention from the brain). The brain sends out signals which consist of 'spikes' called *action potentials* (see Figure 1); signals are built up of trains of action potentials. The brain is part of the central nervous system which embodies all neural tissue in the body including the spinal cord. From the brain, signals travel to the spinal cord. The spinal cord can be divided into sections, called vertebrate discs, which are responsible for

[*] Simon Fraser University
[†] Universiteit van Amsterdam
[‡] Vrije Universiteit Amsterdam
[§] Universiteit Leiden

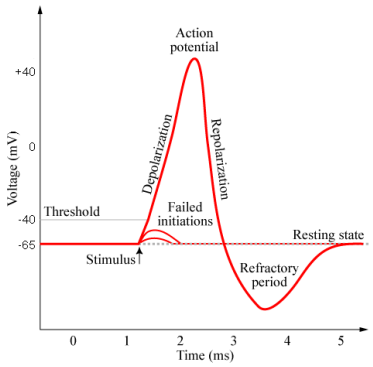

Figure 1: An action potential.

the control of a certain part of the body. The higher levels control the head and arms, the lower levels control the torso and legs. Neurons branch off at every vertebrae. At such a vertebrate disc in the spinal cord, the signal passes a number of intermediate points called interneurons which act as a switchboard redirecting the signals to all the further tissue that requires the information. The signals for movement reach their final (neural) destination at neurons which control the muscles (of which the cell body also lies in the vertebrate discs) called *alpha*-motor neurons (α-mn) (see Figure 2 (Left) for the anatomy of this neuron). These neurons will be the focus of this report.

An α-mn controls a set of muscle fibers. The collection of the α-mn and the fibers it innervates is called a *motor unit*. The number of fibers per motor unit varies from 10 to 300 in different muscles. It is not known a priori how many motor units a muscle has. To complicate matters, the fibers of different motor units are not neatly bundled, but intermingle with fibers of other motor units (see Figure 2 (Right)). When an α-mn fires, it produces a 'twitch' in the muscle fibers which is the result of electric current moving across the muscle fiber membrane. This current can be measured as

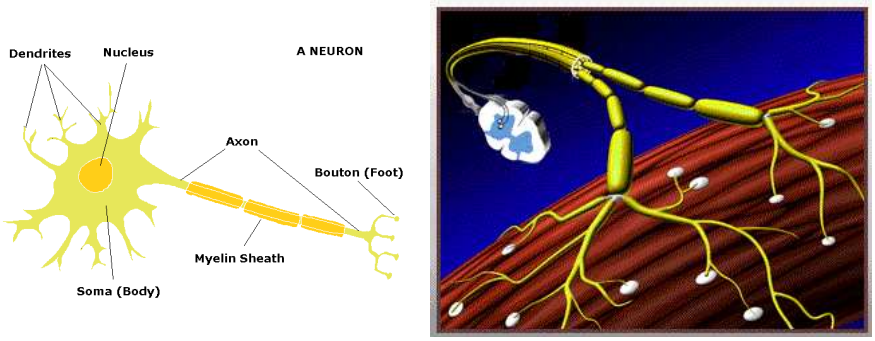

Figure 2: Left: A neuron and its components. Right: Connections between the Spinal cord, α-mns and muscle fibers.
© Post-Polio International.

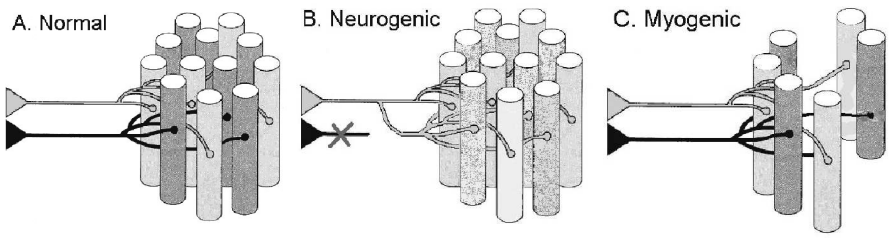

Figure 3: (A) Normal situation: Two motor units each controlled by separate α-mns. The fibers of the two motor units are intertwined. (B) Characteristics of a neurogenic disorder: reduced number of MUs and increased MU size. (C) Characteristics of a myogenic disorder: Decreased MU size. Eventually effective loss of MUs. Adapted from [3]

will be discussed later. The α-mn fires with a certain frequency resulting in repeated twitches of the fibers. The combined twitching of all fibers is what produces force and movement. Normally, the individual twitches are not perceived, only when one exerts a large amount of force can we see the effect of the underlying twitching mechanism.

We study two types of pathology which affect the distribution of fibers over the motor neurons. In the first pathology, *neurogenic disorders*, certain α-mns die off. Other neurons will take over the innervation of the 'disconnected' fibers. This causes the motor units to grow in size (i.e. there are more fibers innervated by one α-mn). The increase in size of motor units results in loss of ability to perform fine tasks. In the second pathology, *myogenic disorders*, the muscle fibers die off. These fibers are not regenerated. The result of the disorder is an increasingly weaker muscle. The activity of the motor units decreases, resulting in an effective loss of motor units (see Figure 3). Diagnosis of these disorders can be done by muscle biopsy. Monitoring the progression and severity of the disorder is more difficult. Clinicians would benefit greatly from having an accurate estimate of the number of motor units in a muscle. This problem is know as Motor Unit Number Estimation (MUNE) and is the first question of medical science we will address.

Question 1: *How many motor units are active during a particular contraction?*

The MUNE problem requires information generated by the muscle fibers. This can be done by measuring the current that travels along the muscle fiber membrane. An obvious problem in neuroscience is performing measurements on humans. This problem is more pronounced when measuring force production and movement, simply because things move around. When one wants to measure signals produced by the central nervous system and muscles there are many possibilities which can be divided into two categories. The first category consists of invasive techniques. For example, the recording of electrical signals produced by neural tissue with an electrode on the tip of a needle. The second category, non-invasive measurement techniques, uses sensors on the outside of the body, mainly on the skin. For many purposes invasive techniques give more information as the sensor is exactly where the experimenter wants it to be. However, invasive measurement is significantly more distressing to the

Figure 4: High-density sEMG stetup. This setup records up to 130 channels at once.

participant of the experiment than non-invasive techniques. Also, because ethics prohibit many invasive techniques many experimenters choose non-invasive techniques to measure signals produced by the central nervous system and muscles. The data we examine was obtained non-invasively.

A non-invasive technique which records muscle activity from the surface of the skin is called surface electromyography (sEMG). A number of electrodes are placed on the skin and the voltage difference between the electrodes and a reference electrode is measured. This particular high-density setup can record up to 130 channels of sEMG at once, as described in [11] (see also Figure 4). Muscle activity can be generated in two ways: voluntary or stimulated (the participant is given a small electric shock which triggers the α-mns) which differ in a fundamental way. As mentioned before, motor units differ in size, and not all motor units are active at each level of force. The recruitment of the motor units during voluntary contractions is such that the small motor units are recruited first and the larger motor units are added as more force is required (i.e. small to large). This is called the *size principle* [6], and allows one to control movement at different force levels. Also, during voluntary contractions, the motor units are not triggered in unison; there are time delays between the individual firings. This property makes analysis of the data very hard as the time differences are not known. This effect is not present when the muscle is stimulated. During stimulated contractions, the motor units fire at the same time. However, they do not obey the size principle. During stimulated contractions the motor units are recruited large to small. This is not an issue as the larger motor units will generate higher levels of sEMG and can be detected more easily.

The MUNE problem now comes down to extracting the number of components in the total of 130 channels of sEMG. The mathematics for this extraction is based on *Principal Component Analysis (PCA)*. This technique is very common in signal analysis, but will have to be modified to suit our purpose. To get reliable estimates, we need to record muscle activity with similar content (i.e. the same motor units firing) many times. These recordings have delays which are unknown. To compensate,

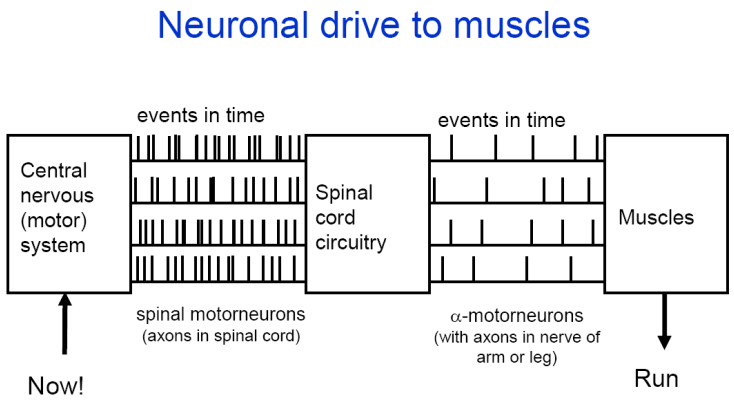

Figure 5: Neural drive from the central nervous system to the α-mn.

these delays, or shifts, will have to be estimated before we can use PCA. For the analysis we have say N recordings of 130 channels each. The high-density sEMG setup gives topological information of the muscle activity. This topological aspect has to be removed for the PCA. This is done by concatenating the information in the 130 channels of sEMG yielding N time series of which we still have to estimate the individual time delays. The N recordings are made according to a protocol which starts the first recording at a low stimulation level and continues the recordings with an increasing stimulation level. This ensures the experimenter that the largest motor units are present in almost all recordings and the smaller motor units become present in recordings with increased stimulation. The mathematics of the estimation of the shifts and the PCA are described in Section 2.

Question 2: *How is the output frequency distribution of a motor unit determined by the input frequency distribution?*

The second question we want to address concerns the response of an α-mn (see Figure 5). For this we derive a stochastic model based on certain assumptions of the nature of the signal received by the α-mn. We assume the input of an α-mn to be a Poisson process. Also, we will assume the time between action potentials arriving at the α-mn to have a Poisson(λ) distribution. It is often observed that the input to the muscle, and thus the output of the α-mn, has a characteristic frequency. Also, one can observe activity in the brain having specific frequencies. Our goal is to gain insight into the response of the α-mn to input of a known frequency.

2 Determining principle components for data with unknown shifts

Given a collection of data $X \in \mathbf{R}^{m,n}$ it is a common task to determine the principle components. Typically we model measured data as having the form

$$X_{ij} = \sum_{k=1}^{N} C_{ik} v_k(t_j) + \eta_{ij} \tag{1}$$

where $X, \eta \in \mathbf{R}^{m,n}$, $C \in \mathbf{R}^{m,N}$ and there are m times $\{t_j\}$ at which the signal was sampled. That is, the n signals can be expressed as a linear combination of a small collection of N ($N \ll n$) basis vectors and noise. In this problem we are interested in determining the dimension N of the spanning set $\{v_k\}$.

In the absence of noise this problem can be solved by decomposing the data onto it's singular values [5]; we find $U \in \mathbf{R}^{m,m}, V \in \mathbf{R}^{n,n}, \sigma \in \mathbf{R}^m$ such that

$$U^T X V = [\text{diag}(\sigma) 0_{n,m-n}]. \tag{2}$$

Additionally, U and V are orthogonal matrices and $\sigma_i \geq \sigma_{i+1} \geq 0$. Of immediate relevance is that if a matrix has rank $N < n$ then $\sigma_{N+1} = \ldots = \sigma_n = 0$. In this case there are precisely N principle components. Typically however real data is not so clearly delineated with fuzzy measurements ensuring that $\sigma_i > 0 \ \forall i$. However, the relative sizes of the principle values σ_i may still provide us with a great deal of information. In particular if we define

$$X_k = \sum_{i=1}^{k} \sigma_i U_i V_i^T$$

then σ_{k+1} is the 2-norm of the distance of X to all matrices of rank k:

$$\min_{\text{rank}(Y)=k} \|X - Y\|_2 = \|X - X_k\|_2 = \sigma_{k+1}. \tag{3}$$

Given a threshold ε this property can be used to define the ε-rank of a matrix, r_ε by requiring that

$$\sigma_{r_\varepsilon} > \varepsilon \geq \sigma_{r_\varepsilon + 1}.$$

If we have a known order of magnitude for errors ε we can define a suitable r_ε. For instance, if the entries of the matrix X_{ij} are known relatively to within $\pm 10^{-3}$ we could determine the ε-rank of the matrix by finding the minimal r_ε such that $\sigma_{r_\varepsilon+1} < 10^{-3}\|X\|_2$.

In the problem at hand we cannot directly apply these ideas as there are *two* distinct sources of errors:

1. Experimental errors of unknown type and magnitude occurring on each channel.
2. Shifts of unknown magnitude occurring *between* all pairs of channels.

Errors of the first type are not dramatically troublesome and in the next section we will discuss one algorithm for estimating the rank of a matrix in the presence of systematic noise. The second type of error needs to be examined more carefully. Denoting

$\hat{X}_{ij}$: measured data in the ith channel at time t_j,

s_i: time shift in the ith channel,

X_{ij}: data in the ith channel at time $t_j + s_i$,

η_{ij}: noise in the ith channel measured at time t_j.

Let us assume that

$$\hat{X}_{ij} = X_{ij} + \eta_{ij}$$

and that the unshifted data is spanned by N basis vectors

$$X_{ij} = \sum_{k=1}^{N} C_{ik} v_k(t_j)$$

then

$$\hat{X}_{ij} = \sum_{k=1}^{N} C_{ik} \left(v_k(t_j) + s_i v_k'(t_j) + \frac{s_i^2}{2} v_k''(t_j) + \dots \right) \tag{4}$$

$$\simeq \sum_{k=1}^{N} C_{ik} \left(v_k(t_j) + s_i v_k'(t_j) \right) \qquad \text{(assuming small shifts)}$$

$$= \sum_{k=1}^{N} C_{ik} v_k(t_j) + \sum_{k=1}^{N} \tilde{C}_{ik} \tilde{v}_k(t_j) \tag{5}$$

Generically the functions $\tilde{v}_k$ cannot be expressed as linear combinations of v_k and hence even without noise we would have doubled the rank of the matrix and the apparent dimension of the spanning set. We will know consider a small experiment to demonstrate these ideas.

Example 1 Consider four matrices of the form

$$\begin{aligned}
X_{ij} &= \alpha_i \sin(t_j) + \beta_i \sin(2t_j), \qquad\qquad i = 1, \dots, 10, t_j \in [0, 5\pi] \\
\tilde{X}_{ij} &= X_{ij} + \varepsilon \eta_{ij} \\
\hat{X}_{ij} &= \alpha_i \sin(t_j + s_i) + \beta_i \sin(2t_j + s_i) \\
&= \alpha_i (\sin(t_j)\cos(s_i) + \cos(t_j)\sin(s_i)) + \beta_i (\sin(2t_j)\cos(s_i) + \cos(2t_j)\sin(s_i)) \\
&= \tilde{\alpha}_i \sin(t_j) + \tilde{\beta}_i \cos(t_j) + \tilde{\delta}_i \sin(2t_j) + \tilde{\gamma}_i \cos(2t_j) \\
\bar{X}_{ij} &= \hat{X}_{ij} + \varepsilon \eta_{ij}
\end{aligned}$$

where α_i, β_i, s_i and η_{ij} are randomly chosen in $[-1, 1]$ and $\varepsilon = 10^{-3}$. In this example we have that $N = 2$ and $n = 10$. In Figure 6 we present a sample signal with and without shifts and noise and also the singular values.

From this example we find that we cannot simply use the singular value decomposition – even with a known threshold – to estimate the dimension of the spanning set for the shifted noisy data. To determine the number of principle components for the given data we now proceed in two steps: first we estimate upper and lower bounds for N, N_u and N_l; then we search for the shifts assuming that we know the correct dimension.

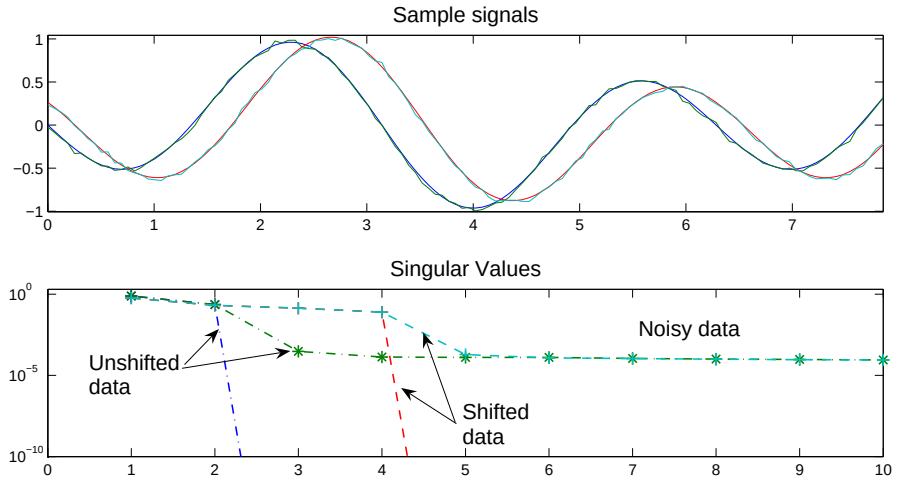

Figure 6: Example 1. Top: Sample signals from one channel showing a typical shift and small noise. Bottom: The singular values for the data with no noise fall off to $\sigma_i = 10^{-40}$ after $N = 2$ and $N = 4$ for the un-shifted and shifted data respectively. In this example the addition of the noise determines a clear ε-rank, but the shifts double the estimate for the number of principle components.

2.1 The number of principle components

We begin by modelling our data by (1). To estimate the dimension of the spanning set we follow the philosophy of [2] and the algorithm of [8] who advocate a Bayesian approach. While the full details of the method are beyond the scope of this report, we will briefly describe the methodology.

The model is that in each channel the signal is generated from a small N-dimensional vector w via a linear transformation and an error term:

$$X_i = \sum_{k=1}^{N} C_{ik} w_k + \bar{m}_i + \eta_i \tag{6}$$

with $\langle C_i \rangle = 0$, i.e. $\langle X_i \rangle = \bar{m}_i$. Critically it is assumed that both the noise vectors η_i and the principle component vectors C_i are sampled from spherical Gaussians (i.e. are normally distributed). Note that here we are implicitly assuming that the noise in each channel is of the same magnitude and is thus additive. By construction, each sample X_i is also taken to be Gaussian. For given model parameters $C, \bar{m}$ and v this distribution takes the explicit form

$$P(X|C, \bar{m}, v) = \frac{1}{(2\pi)^{Nm/2}} \left|CC^T + vI\right|^{-N/2} \exp\left(-\frac{1}{2}\mathrm{tr}(CC^T + vI)^{-1}S\right) \tag{7}$$

where S is the co-variance matrix $S_{ij} = (X_i - \bar{m}_i) \cdot (X_j - \bar{m}_j)$, and v is the variance of the noise assumed to be constant over all channels.

The evidence that the data fits the model with specific parameters can now be

determined by integrating over the parameter space

$$P(X|M) = \int_{\theta} P(X|\theta)P(\theta|M)\, d\theta \tag{8}$$

The model parameters which best fit the data with the highest probability are those which maximize this integral. The first difficult part is to find a parametrization of the family of matrices C in (7) such that the integral in (8) may be evaluated. In practice, it turns out that once this has been done the integral cannot be computed exactly but is instead approximated with Laplace's method [8]. Limiting the dimension of the spanning set to k we have that the evidence that our data X is of the form (6) with $C \in \mathbf{R}^{k,n}$ is [8]

$$P(X|k) = c_k \int \left|CC^T + vI\right|^{-N/2} \exp\left(-\frac{1}{2}\text{tr}(CC^T + vI)^{-1}S\right)\, dU\, dL\, dv \tag{9}$$

where $C = U(L - vI)^{1/2}R$, $U^TU = I$, $R^TR = I$ and c_k is a known function of n, k and m. The most probable dimension, $\tilde{N}$, is such that

$$P(X|\tilde{N}) = \max_k P(X|k). \tag{10}$$

The recent paper [8] compares several approximations to (9) and algorithms for determining the number of principle components. Based on multiple numerical tests the author asserts that Laplaces' approximation to (9) is both the least computationally intensive and the most reliable. An algorithm to determine the most likely dimension is to find $\tilde{N}$ satisfying (10) with $P(X|\tilde{N})$ approximated by Laplace's method (see [8] for details).

To determine the upper bound N_u we simply apply this algorithm and, based on the observation in (5), take $N_u = N/2$. Unfortunately this estimate is not sharp as the effects of real noise (i.e. with different magnitudes and variances) tend to lead to an over estimation for N. Also, it is not clear which terms in the expansion will be recognized as signal rather than noise.

To compute the lower bound N_l we recall (4). We will assume that the shifts are small and hence that $C_{ik} \gg \tilde{C}_{ik}$. To find N_l we mask the effect of the shifts by amplifying the noise. We find a threshold ε such that $Y_i = X_i' + \varepsilon\eta_i$ is indistinguishable from noise. In terms of the above algorithm this means $N(\varepsilon) = 1\ \forall\delta \geq \varepsilon$. We now have a way to *artificially add noise to hide the effects of the shifts*, i.e. compute

$$N_l = N(\hat{X} + \varepsilon\eta) \tag{11}$$

(In practice, this threshold may be too large. We will leave it's proper determination for later work.) One drawback of this approach is that it can *under-predict* N_l if some signals are much weaker than the others and as such can easily be drowned out by the added noise. One possible correction for this is to rescale all the signals before starting. While this rescaling does affect the particular singular values it in no way changes the shifts or the correct dimension of the spanning set.

Given the interval $[N_l, N_u]$ we now choose $N \in [N_l, N_u]$ and try to find the shifts *assuming* N.

2.2 Functional optimization

Given the geometry of the singular value decomposition we would like to choose the shifts to minimize σ_{N+1}. This maximizes the ε-rank of the matrix.

However, in practice this problem seems to be highly degenerate and difficult to solve. Instead we have chosen to consider

$$V = -\frac{\sum_{i=1}^{N} \sigma_i}{\sum_{i=1}^{n} \sigma_i} = -\sum_{i=1}^{N} \sigma_i \qquad \text{(by normalization)}. \tag{12}$$

If the singular values are decaying quickly for $i > N$ then

$$-\sum_{i=1}^{N} \sigma_i \simeq -\sum_{i=1}^{n} \sigma_i + \sigma_{N+1} \simeq -1 + \sigma_{N+1}$$

and in many cases the minimization of (12) is a good approximation of minimizing σ_{N+1}.

We have tested three approaches to minimize the functional (12) with respect to the shifts.

A1. A pseudo-Newton method to find a zero of ∇V using `minunc` in the Optimization Toolbox for Matlab.

A2. A gradient flow of V with respect to an artificial time with computation of

$$\frac{ds_i}{dt} = -\frac{\partial V}{\partial s_i}$$

using the code `ode113` in Matlab.

A3. Direct search using the Matlab routine `fminsearch`.

The results of algorithms A1 and A2 are almost identical but with A2 taking approximately ten times longer to terminate. The results of A3 are less satisfactory and take approximately ten times longer than A2.

Each step in all three algorithms requires both interpolation and the computation of singular values. Because the interpolation is being done onto a uniform grid cubic interpolation takes the form

$$X_i(t_j + s) = \alpha_{-1}(s)X_i(t_{j-1}) + \alpha_0(s)X_i(t_j) + \alpha_{+1}(s)X_i(t_{j+1}) + \alpha_{+2}(s)X_i(t_{j+2}),$$

for $s \in [0, 1]$. Here the α_i are cubic polynomials in s. This is a cheap operation taking only $\mathcal{O}(mn)$ operations.

In each algorithm, we approximate the gradient as

$$\frac{\partial V}{\partial s_i} \simeq \frac{V(s + \Delta s e_i) - V(s)}{\Delta s}$$

Here e_i is the unit vector in the $i-$th direction and $\Delta s = \sqrt{\epsilon} \simeq 10^{-8}$. Thus at each step we need to compute n singular value decompositions ($s_1 \equiv 0$) for a total

of $\mathcal{O}(mn^3)$ operations. Occasionally A1 also needs to compute the Hessian which requires $\mathcal{O}(mn^4)$ operations.

A3 is the cheapest per step but takes by far the most steps. A1 is the best as at step $n+1$

$$\left\| \frac{\partial V^{(n+1)}}{\partial s} \right\| \simeq C \left(\frac{\partial V^{(n)}}{\partial s} \right)^p, \qquad C_1 > 0, p > 1.$$

Whereas for A2 we can expect no better than

$$\left\| \frac{ds}{dt} \right\| = \left\| \frac{\partial V}{\partial s} \right\| \simeq C_2 e^{-C_3 t} \quad \text{as } t \to \infty \qquad C_2, C_3 > 0.$$

In order to reduce the number of function evaluations we use a high-order predictor corrector code but then the step sizes are limited by stability as $t \to \infty$. It is possible that hybrid method first using an explicit code and then an implicit one would make A2 competitive with A1.

2.3 A test for determining N

The minimization routine is run for the sequence of $N \in [N_l, N_u]$. Unfortunately there are still numerous problems preventing us from simply applying the evidence algorithm to determine the number of principle components. These include

1. The noise in the data does not strictly follow the model: it is not necessarily normally distributed, the variance varies between channels.
2. The shifts have not been completely eliminated.
3. There are small interpolation errors.
4. The minimization problem introduces additional numerical errors.
5. A mixture of signals of vastly different magnitudes in each channel. This makes some signals difficult to distinguish from noise and, more importantly, the effects of shifts in large signals can be on the same order of magnitude as some weak signals.

Ideally, we would like a nice obvious step from large singular values to small ones (as in Example 1) allowing for a clear ε-rank. Unfortunately, this combination of difficulties makes determining such an ε impossible. Instead, we solve for all $N \in [N_l, N_u]$ and then set

$$N_r \text{ such that } \frac{\sigma_{N_r}}{\sigma_{N_r+1}} = \max_k \frac{\sigma_k}{\sigma_{k+1}} \tag{13}$$

for each run. Then we take $\tilde{N}$ to be that which maximizes this ratio. Conceptually this approach finds the dimension such that there is a clear step from "large" components to "small" ones. But, it does not preclude the possibility that the data in each channel comprises one very large signal and several much smaller ones. In that case, for the reasons stated above, it may be very difficult to distinguish the small signals from noise as the shifts amplify the classic signal vs. noise problem.

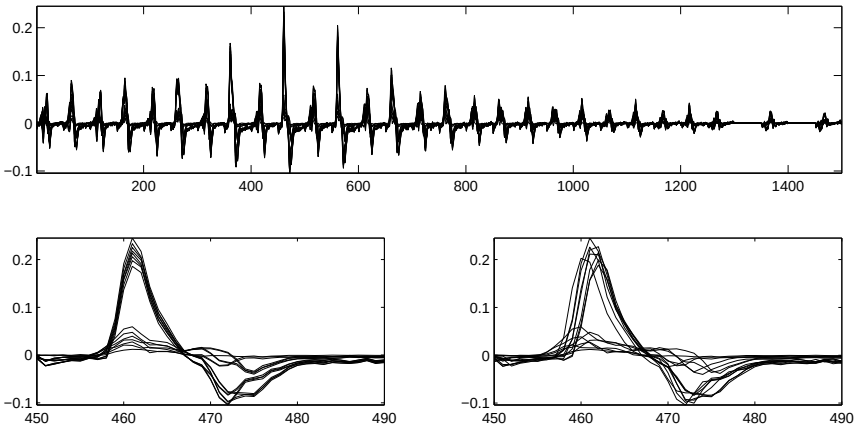

Figure 7: Artificial data. Top: Complete signals in all channels. Bottom left: Detail of unshifted data. Bottom right: Detail of shifted data. After optimization the original and output data are essentially indistinguishable.

N	N_r	$\sigma_{N_r}/\sigma_{N_r+1}$	Run time (s)
6	6	125	47
5	5	98	45
4	4	2731	33
3	4	113	39
2	4	31	31

Table 1: Results from running the algorithm with given data. From this we would correctly conclude that there are four principle components.

2.4 Example

To recap, the entire algorithm is

1. Determine N_u with (10).
2. Numerically find X' and find ε such that $X' + \varepsilon\eta$ is indistinguishable from pure noise.
3. Determine the N_l as the dimensionality of $X + \varepsilon\eta$.
4. Solve the optimization problem for $N = N_l, N_l + 1, \ldots N_u$.

We now consider a test on the artificially generated data as in Figure 7. This is "simulated" data from one of our industrial collaborators. We begin with initially unshifted data to allow us to test the reliability of the algorithm. The results are summarized in Table 1.

2.5 Conclusions

This algorithm has been shown to be effective in reducing the effects of shifts in data to give a reliable estimate of the number of principle components. However, two areas for improvement remain. Firstly, no information about the structure of the signals has been incorporated and we leave this as an avenue for further work. Secondly, when optimizing with N greater than the number of true principle components the signals falsely seperate so as to fill all available spanning dimensions. This problem is easily remedied should it be required when dealing with real data.

The question we have looked at in this Section has two components. Firstly, how can we remove the shifts? Then, how do we determine the number of principle components? We believe that we can more reliably deal with the first question. This is not surprising as the second is a long-standing research question and not completely resolved in many practical application areas.

After the preparation of this report we were furnished with "real" data with which to work. Preliminary tests suggests that this approach works but that the ODE gradient flow approach may be more appropriate. We leave this as a direction for further research.

3 Discrete integrate and fire neurons

The second question posed was to model the frequency response of a motor unit. Clinical scientists are interested in an explanation of the 40 Hz components that are measured in sEMG, both at the brain and the muscles. We did not have sufficient information to answer this specific question, but have made a simple model that combines features of other, existing models, and may help to foster insight in the frequency response.

We present a simple, phenomenological, discrete model with which the average firing time or the length of the inter-spike intervals of neurons can be estimated, given a basic exponential potential function $F_0(t)$ and a series of incoming signals $g_i(t)$ from the brain. We choose a particular $F_0(t)$, but the discrete approach in this section can be applied to any function $F_0(t)$. Our potential $F_0(t)$ varies between its minimum F_0 and its saturation state $F_0 + \alpha$, and is assumed to be exponential:

$$F_0(t) = F_0 + \alpha(1 - e^{-\beta t}) \tag{14}$$

with $\alpha, \beta > 0$. The potential is then reset to the minimum F_0 after the neuron has fired. During an inter-spike interval, i.e. as long as no firing has occurred, the potential increases monotonically.

The neuron fires if its total potential $F(t)$, that is a sum

$$F(t) = F_0(t) + \sum_i g_i(t) \tag{15}$$

of its own basic potential $F_0(t)$ and the incoming signals $g_i(t)$, reaches a certain threshold value θ.

The incoming signals $g_i(t)$ are in fact very short pulses, but they are often modeled as exponentials, as block-functions, or, since they should add up to build up the

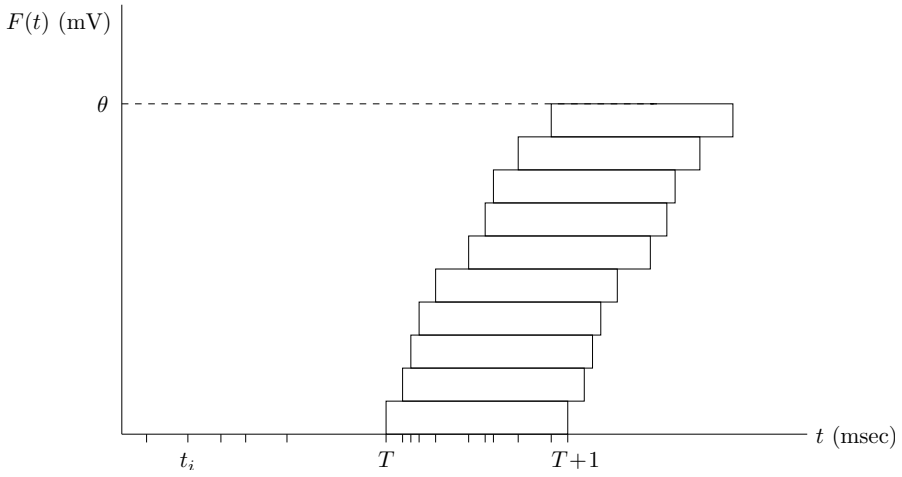

Figure 8: Blocks of length 1 arrive at arrival times t_i. Here 10 blocks have arrived within the time interval $[T, T+1]$.

potential $F(t)$, their derivatives are assumed to be delta-functions [4]. If the incoming signals $g_i(t)$ are modeled by $\frac{dg}{dt} = \delta(t - t_i)$, so as block functions that have value $g_i(t) = 0$ for all $t < t_i$ and $g_i(t) = 1$ for all $t > t_i$, they get an infinite length. There is a major drawback of this simplest assumption one can think of, since this way they all add up to build up the potential, whereas in real neurons the potential decreases as well. We therefore want our model to allow for a drop in the potential $F(t)$ if the incoming signals $g(t_i)$ are too far apart. Our approach is, to model the $g_i(t)$ by block functions, all of equal length and height. This form is of course still far from the real pulse form of the signals, but is similar to the approximation mentioned above. If a stack of these blocks reaches the threshold θ within a certain time interval (that we scale to 1), the neuron will fire. If the threshold is not reached within this time interval, the total stack will decrease, and new incoming signals will start and build up a new stack (see Figures 8 and 9).

In fact, this model combines two features of the neuron system that have been implemented separately before. As far as we know there are no existing models in which both are combined. Other models either use a constant function $F_0(t)$, which means in physiological terminology that no leaky dynamics are taken into account, or they use a function $F_0(t)$ with leaky part and use infinite-length $g_i(t)$-functions. If a constant $F_0(t)$ is chosen, then it is combined with the same infinite-length $g_i(t)$-functions or with more natural forms for $g_i(t)$, such as the above mentioned exponentials or simple pulse functions.

3.1 Stochastics

A standard assumption on the incoming signals $g_i(t)$ is that the arrival times t_i are distributed as a Poisson process. In other words, the time intervals $X_i = t_i - t_{i-1}$ are

random variables that are Poisson distributed, i.e. they satisfy

$$P(X_i > x) = e^{-\lambda x} \tag{16}$$

for a positive parameter λ. For well-known results, see for instance [10].

If the interval lengths X_i satisfy the distribution (16), then the probability that there will be n arrival times t_i within any time interval of length 1 is equal to the the probability that there will be n arrival times t_i within the interval $[0, 1]$. With N the number of arrival times within $[0, 1]$, this probability is $P(N = n) = \frac{\lambda^n}{n!}e^{-\lambda}$. If the incoming blocks have length 1 and height 1, the probability that they will together at least reach an integer height h within a time interval of length 1 will thus be $P(N \geq h) = 1 - P(N \leq h - 1)$. The expected number of arrival times (expectation value) within an interval of length 1 is λ, with variance λ.

This is illustrated in Figure 8. There 10 blocks of length 1 and equal height have arrived within the time interval $[T, T + 1]$, and the threshold $\theta = 10$ is thus reached. The probability that exactly 10 blocks would have arrived within this interval is $P(N = 10) = \frac{\lambda^{10}}{10!}e^{-\lambda}$; the probability that at least 10 blocks would have arrived within this interval is $P(N \geq 10) = \sum_{k\geq 10} \frac{\lambda^k}{k!}e^{-\lambda}$.

The next step is to let the stack fall down and start a new one, if the stack has not reached the threshold height θ within the time interval $[T, T + 1]$. In modeling the function $F(t)$ (15), we impose that the level at which the stack in an interval $[T, T+1]$ starts to build up equals $F_0(T)$, with $F_0(t)$ given by (14) as shown in Figure 9. This means that if the stack does not reach the threshold θ within $[T, T + 1]$, it will fall down to $F_0(T + 1)$. If the stack has reached the threshold value within an interval $[T, T + 1]$, the neuron fires, the stack falls down, and the time t is reset to $t = 0$, so that the potential starts to build up from $F_0(0)$ again. To simplify the calculations, we let the neuron fire at the end of the interval, at $t = T + 1$.

3.2 Expectation value and variance of firing time

The above ingredients are sufficient to calculate the expectation value and variance of the firing time in this model. We take blocks of height 1 (notice that the scaling of the horizontal and vertical axes in Figures 8 and 9 is different).

Define p_n as the probability that the neuron fires in (or by the above assumption, at the end of) the interval $[n, n + 1]$. Then

$$p_n = P(N \geq \theta - F(n)) = P(N \geq \lceil \theta - F(n) \rceil) = \sum_{k\geq \lceil \theta - F(n) \rceil} \frac{\lambda^k}{k!}e^{-\lambda} = 1 - \sum_{k=0}^{\lfloor \theta - F(n) \rfloor} \frac{\lambda^k}{k!}e^{-\lambda}. \tag{17}$$

Here we used the notations $\lceil x \rceil$, which means the first integer larger than x, and $\lfloor x \rfloor$, which means the first integer smaller than x. They play a role, since $\theta - F(n)$ need not be an integer.

Now define the stochast Y as the firing time. Define f_n as the probability that the neuron has not fired at or before $t = n$, and does fire in the interval $[n, n + 1]$, or

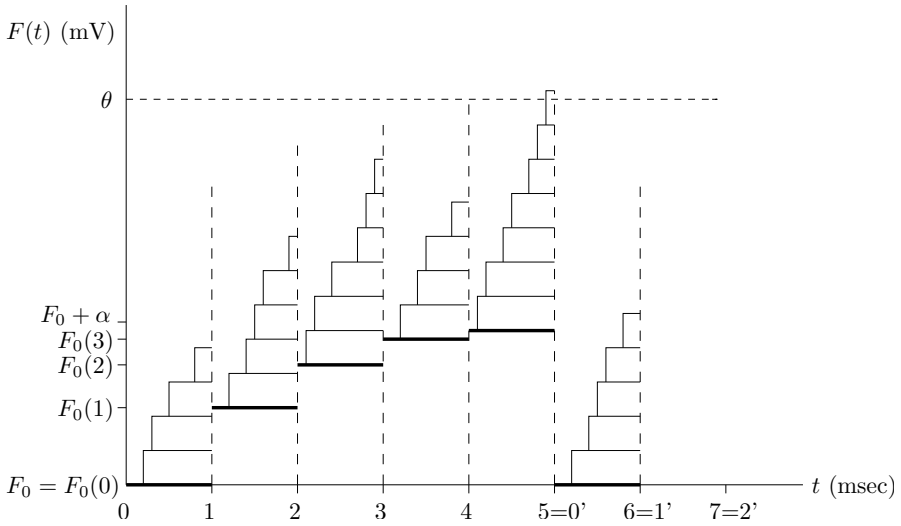

Figure 9: Stacks are being built up within intervals $[T, T+1]$. If a stack has not reached θ at $t = T+1$ it falls down at $t = T+1$, and the next block arrives at height $F_0(T+1)$. If the stack has θ at $t = T+1$, the neuron fires (not shown), the stack falls down, the time t is reset ($t = 5 = 0'$), and the next block arrives at height $F_0(0') = F_0(0)$ again.

better, at $t = n+1$. Then

$$f_n = P(Y = n+1) = (1-p_0)\dots(1-p_{n-1})p_n = p_n \prod_{j=0}^{n-1}(1-p_j). \qquad (18)$$

The expectation value for the firing time can now be calculated as

$$\begin{aligned} E(Y) &= 1.P(Y=1) + 2.P(Y=2) + 3.P(Y=3) + \dots \\ &= \sum_{l=1}^{\infty} f_{l-1} l = \sum_{l=1}^{\infty} p_{l-1} l \prod_{j=0}^{l-2}(1-p_j). \end{aligned}$$

With $e_n := \lceil \theta - F(n) \rceil$, this equals

$$E(Y) = \sum_{l=1}^{\infty} l \sum_{k \geq e_{l-1}} \frac{\lambda^k}{k!} e^{-\lambda} \prod_{j=0}^{l-2} (\sum_{k=1}^{e_{j-1}} \frac{\lambda^k}{k!} e^{-\lambda}). \qquad (19)$$

The variance in the expected firing time is $Var(Y) = E(Y^2) - E(Y)^2$, where $E(Y^2)$ is the so-called second moment

$$\begin{aligned} E(Y^2) &= 1^2.P(Y=1) + 2^2.P(Y=2) + 3^2.P(Y=3) + \dots \\ &= \sum_{l=1}^{\infty} f_{l-1} l^2 = \sum_{l=1}^{\infty} p_{l-1} l^2 \prod_{j=0}^{l-2}(1-p_j), \end{aligned}$$

which equals

$$E(Y^2) = \sum_{l=1}^{\infty} l^2 \sum_{k \geq e_{l-1}} \frac{\lambda^k}{k!} e^{-\lambda} \prod_{j=0}^{l-2} (\sum_{k=1}^{e_{j-1}} \frac{\lambda^k}{k!} e^{-\lambda}). \tag{20}$$

The n^{th} moment is likewise given by

$$E(Y^n) = \sum_{l=1}^{\infty} l^n \sum_{k \geq e_{l-1}} \frac{\lambda^k}{k!} e^{-\lambda} \prod_{j=0}^{l-2} (\sum_{k=1}^{e_{j-1}} \frac{\lambda^k}{k!} e^{-\lambda}). \tag{21}$$

3.3 Calculations

If the parameters F_0, α, θ and λ are chosen, the expected firing time and corresponding variance can be calculated from (19) and (20). The first three parameters vary for different types of neurons, and we will choose various values in our calculations. The parameter λ however is strongly related to the time-separation between the incoming pulses $g_i(t)$, since λ is the expected number of arrival times within an interval of length 1. We know that the pulses arrive with a frequency of about 10.000 Hz, so if we rescale our time t so that each interval of length 1 is an interval of 1 ms, the parameter λ should be set to $\lambda = 10$.

We calculated the expected firing time and variance for this model, with different (more or less realistic) values for F_0, α, β and θ. In [7] we find typical values for an α-motor neuron: the minimum F_0 of $F_0(t)$ is about $F_0 = -75$, the difference α between the minumum of $F(t)$ and its saturation value is about $\alpha = 5$, and the firing threshold θ is about $\theta = -55$. These values are all given in mV. In the standard book [1] we find values for other types of motor units as well; examples are $F_0 = -65$, $\alpha = 10$ or 15, and $\theta = -50$ or -45 (but of course always with $\alpha < \theta - F_0$). The parameter β is related to the saturation time of $F_0(t)$, which is typically 75 [7] to 100 ms We assume that $F_0(t)$ has reached the saturation value $F_0 + \alpha$ if it is less than 1% off this value, so if $1 - e^{-\beta t} \leq 0.01$. This means that realistic values for β are for instance $\beta = 0.05$ or $\beta = 0.06$. The results of some calculations are listed in the Table 2.

To interpret these values, it is useful to realize that an expected firing time $E(Y) = \tau$ ms corresponds to an expected firing frequency of $1000/\tau$ Hz. The last column in the table depicts the corresponding expected frequency (in Hz) in each of the calculated cases.

As realistic firing frequencies are between 8 and 20 Hz (15 Hz for an α-motor neuron) [1, 9], we conclude that for low threshold values and rather short saturation time ($\beta = 0.1$, *i.e.*, a saturation time of about 46 ms according to the 1%-rule above) the model predicts firing frequencies that are far too high. However, for higher threshold values and a somewhat longer saturation time ($\beta = 0.06$, 77 ms; or $\beta = 0.05$, 92 ms), the predicted values for $\alpha = 5$ are very reasonable. For $\alpha = 10$, the predicted frequencies are (much) higher than for $\alpha = 5$.

Note, that for the calculations only the difference $\theta - F_0$ matters, and not the separate values F_0 and θ. This explains why the first and second set contain similar rows. We only computed the first and second moments $E(Y)$ and $E(Y^2)$ for every

F_0	λ	α	β	θ	$E(Y)$	$E(Y^2)$	$Var(Y)$	$\sigma(Y)$	E(freq)
-65	10	5	0.1	-50	9.06	108.7	26.7	5.16	110
-65	10	5	0.1	-45	48.33	3693	1357	36.8	21
-65	10	5	0.06	-45	54.4	4374	1410	37.55	18
-65	10	5	0.05	-45	55.7	4504	1404	37.47	18
-65	10	10	0.1	-45	15.4	267.5	31.9	5.65	65
-65	10	10	0.06	-45	20.6	482.6	57.6	7.59	49
-65	10	10	0.05	-45	23.2	609.1	71.5	8.46	43
-75	10	5	0.1	-65	2.28	7.51	2.30	1.52	438
-75	10	5	0.1	-60	9.06	108.7	26.7	5.16	110
-75	10	5	0.06	-60	10.6	153.5	41.7	6.46	94
-75	10	5	0.1	-55	48.33	3693	1357	36.8	21
-75	10	5	0.063	-55	53.8	4297	1404	37.47	19
-75	10	5	0.06	-55	54.4	4374	1410	37.55	18

Table 2: Results of the calculations for various values of the parameters.

choice of the parameters, but if one is interested, higher moments can be computed along the same lines, using formula (21).

3.4 Conclusions

Although it is very simple and only of a phenomenological nature, the discrete model we presented is capable of producing realistic values for the expected firing time and frequency.

By its discrete nature, the model can easily be changed without destroying the global, simple framework. If one wishes, it can for instance be adapted to fit better with clinical data or to mimic other models, by changing the basic potential $F_0(t)$, the length of the "building-up period" or the value of λ.

These are two reasons why the model may serve as an additional tool to study the relation between input and output frequencies in motor units.

References

[1] M.F. Bear, B.W. Connors and M.A. Paradiso, *Neuroscience, exploring the brain*, 2nd edition, Lippincott, Williams, Wilkins (publ.)

[2] C. Bishop, *Bayesian PCA*, Neural Information Processing Systems, vol 11, 1998, pp 382-388.

[3] J.H. Blok, J.P. van Dijk, G. Drost, M.J. Zwarts and D.F Stegeman, *A high-density multichannel surface electromyography system for the characterisation of single motor units*, Review of Scientific Instruments, vol 73 (4), 2002, pp 1887-1897.

[4] A.N. Burkitt and F.M. Clark, *Neural Computation* vol 12, 2000, pp 1789-1820.

[5] G. Golub and C. van Loan. *Matrix Computations*, 3rd edition. Johns Hopkins University Press. Baltimore, 1996.

[6] E. Henneman, *Relation between size of neurons and their susceptibility do discharge*, Science vol. 126, 1957, pp 1345-1347.

[7] P. Matthews, *Relationship of firing intervals of human motor units to the trajectory of post-spike after-hyperpolarization and synaptic noise*, Journal of Physiology vol 15, 1996, pp 597-628.

[8] T.P. Minka, *Automatic choice of dimensionality for PCA*, Neural Information Processing Systems, 2002.

[9] L.J. Myers, M. Lowery, M. O'Malley, C.L. Vaughan, C. Heneghan, A. St Clair Gibson, Y.X.R. Harley and R. Sreenivasan, *Rectification and non-linear preprocessing of EMG signals for cortico-muscular analysis*, Journal of Neuroscience Methods vol 124, 2003, pp 157-165.

[10] J.R. Norris, *Markov Chains*. Cambridge series in Statistical and Probabilistic Mathematics, 1998.

[11] M.J. Zwarts and D.F. Stegeman, *Multichannel surface EMG: Basic aspects and clinical utility*, Muscle & Nerve vol 28, 2003, pp 1-17.