Contents lists Available at Science Direct

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Validating an Electromyography Simulation Tool using a Genetic Algorithm to Match Simulation Parameters with Recorded Data

S Tallam Puranam Raghu[a], DT MacIsaac [a], PA Parker [a], Y Shi [a]

[a] *Department of Electrical and Computer Engineering, University of New Brunswick, New Brunswick, Canada*

A R T I C L E I N F O

Article History:
Received dd MONTH YYYY
Revised dd MONTH YYYY
Accepted dd MONTH YYYY
Available online dd MONTH YYYY

Keywords:
Surface Electromyography (SEMG), validation, Genetic Algorithm (GA)

A B S T R A C T

A Surface Electromyography (SEMG) simulation tool is introduced, which generates signals based on the user input. The tool uses a finite fibre length model to generate Single Fibre Action Potentials (SFAPs) which are subsequently then used to generate the SEMG signal. In order to validate the tool, a matching process involving a genetic algorithm (GA) was devised.

©2017 Elsevier Ltd. All rights reserved.

1. Introduction

Surface electromyography (SEMG) is a technique used to measure the electrical activity of muscles from the skin surface. Signals recorded through SEMG have been studied extensively for their use with powered prosthetics (Parker, Englehart & Hudgins, 2006; Scheme & Englehart, 2011) and muscle assessment (Türker & Sözen, 2013; Bandpei et al., 2014). More recently studies have expanded to include a wide range of SEMG-based muscle computer interfaces (MCIs) as a means for hands-free human-computer interaction (Wei, Hu & Zhang, 2011; Choi et al, 2013).

While the potential for SEMG-based applications is high, complex techniques must be used to measure, process and interpret SEMG data because of its inherently random nature and susceptibility to noise during data capture (Grugic, Skelin & Cic, 2014). For instance, Scheme & Englehart (2011) suggest pattern recognition techniques based on linear discriminant analysis, support vector machines, and hidden Markov models for classification of SEMG signals, Fraser et al (2014) propose a technique for SEMG quality assessment relying on a one-class support vector machine, and artificial neural networks have been proposed as a means for quantifying fatigue from SEMG data (Rogers & MacIsaac, 2010).

SEMG simulations can be a valuable tool to assist in the development of SEMG-based applications. Simulations offer researchers a way to test their techniques on a controlled data set, and to explore the influence of signal generation parameters on features extracted from observable signals. This is particularly useful for SEMG since signal generation is a complex process affected by parameters associated with muscle anatomy and neuromuscular physiology which are subject and activity dependent and cannot be directly measured. However, for a simulation to be useful, it must render a reasonable reflection of the process it is simulating.

MacIsaac, Rogers and Bhandarkar (2006) presented a MATLAB tool called Myosim, which allows users simulate SEMG signals. In this paper, we propose an improved version of the tool (herein called Myosim 2.0) which better accounts for the amplitude of the SEMG signal, has better signal reproduction capabilities, and provides options to incorporate the effects of data capture (such as instrumentation noise). To test the validity of the upgraded tool, we used a genetic algorithm to optimise generative parameter values in such a way to match signals which were captured *in vivo* (i.e. recorded). We then compared features of the recorded signal to the features of the simulated signal.

2. Background

2.1. SEMG Model

The fundamental unit of skeletal muscle is the motor unit (MU) which encompasses a single motor neuron and all the muscle fibres it innervates. A muscle contraction is the result of simultaneous electrical activation of one or more motor units. Whenever a motor neuron is activated, all the fibres it innervates are also activated. The electrical activity activating a single muscle fibre is called the single fibre action potential (SFAP).

The Motor Unit Action Potential (MUAP) is the spatiotemporal sum of activity from all the fibres belonging to a particular MU. A steady contraction is achieved by rapidly and repeatedly recruiting multiple MUs. Thus, the signal recorded from a single MU over time appears as a train of MU potentials, commonly referred to as a Motor Unit Action Potential Train (MUAPT) or simply as a spike train. The time separation between spikes are random and are commonly assumed to have a Gaussian probability density function (Clamann 1969). The spike trains from several MUs produce a smooth force output.

The SEMG signal recorded at skin surface is the sum of all the MUAPTs that are produced to achieve the required force. Therefore, in order to generate an SEMG signal, the SFAPs of fibres belonging to a MU must be modelled, which can then be used to generate the MUAPs and consequently the SEMG signal. Gonzalez-Cueto & Parker (2002) proposed a finite length model of muscle to model an SFAP as observed at the surface of the skin. The model convolves a double layer differential source $s(t)$, originally proposed by Plonsey (1974), with a tissue filter $h(t)$ that considers various geometric and physiological parameters of the fibre to produce the SFAP, as follows:

$$SFAP(t) \cong s(t) * \mathrm{h}(t) \tag{1}$$

The SFAP are then summated to form the MUAPs, which are then convolved with impulse trains $p(t)$ with a pseudo-random placement of spikes to form the MUAPTs. Several spike trains with different characteristics are summed to form SEMG signals.

$$MUAP(t) = \sum_i SFAP_i(t) \tag{2}$$

$$MUAPT(t) = MUAP(t) * \mathrm{p}(t) \tag{3}$$

$$SEMG(t) = \sum_j MUAPT_j(t) \tag{4}$$

Figure 1 shows an example source signal, the tissue filter and the SFAP produced; while Figure 2 shows an example motor unit, the corresponding spike train, and an example SEMG signal obtained from the summation.

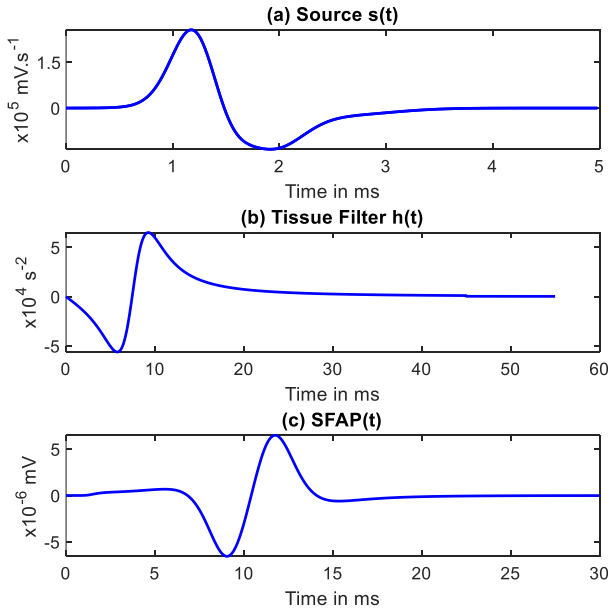

Figure 1: Example SFAP

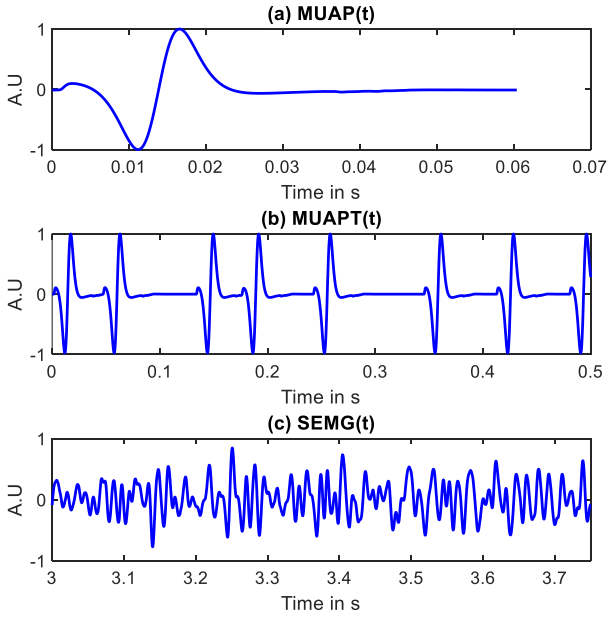

Figure 2: Example MUAP, spike train, and SEMG

2.2. Myosim

The SEMG model was implemented in a user friendly tool using MATLAB, called Myosim (MYOelectric SIMulator) and was first presented in 2006 (herein called Myosim 1.0). Since then, several improvements to the tool have been made, including: incorporating measurement instrumentation effects, and establishing a better set of internal model parameters.

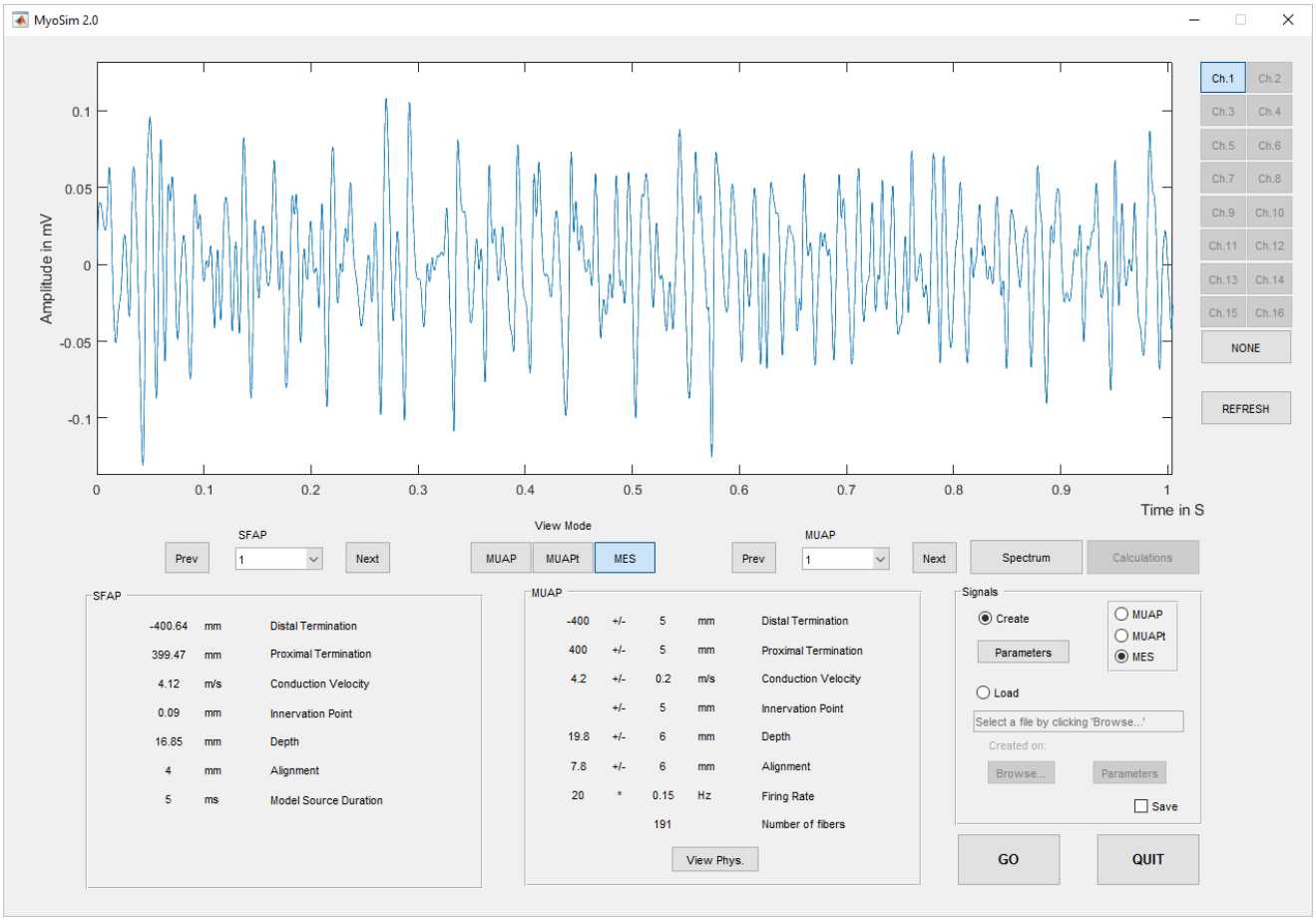

Figure 3: Myosim 2.0

In the Myosim 1.0, the amplitude of the SFAPs were normalised which allowed for some internal model parameters to be nominally set. Because of this, amplitude of the generated SEMG signals could not be easily compared to data records measured *in vivo*. Previous studies that used the tool mostly focussed on signal characteristics that did not depend on the amplitude of the signal, such as conduction velocity (CV) or mean frequency (MF). However, several other features used commonly in SEMG-based applications, rely on amplitude values, such as the mean absolute value (MAV) and waveform length (WL). To account for this, we surveyed the literature to establish values for the model parameters set nominally and for expected SFAP amplitudes. The internal model parameter values were adjusted based on values used by van Veen et al. (1992) for a similar model and finally scaled to match MUAP amplitudes reported by Buchtal, Guld and Rosenfalck (1957).

Data captured through SEMG are subject to several instrumentation effects, including noise from the instrumentation, bandpass filtering, and quantisation. Though these effects are expected to be minimal for a measurement system set up properly, including these effects as part of the simulation model may provide useful insight into the effects of improper instrumentation setup on SEMG signal features. Thus, we included options to add baseline noise modeled as a piecewise linear combination of pink and white noise as expected from instrumentation Mancini (2003), to bandpass filter the signal based on low and high cut-off frequencies set by a user, and to quantize the signal based on a bit resolution provided by a user.

Finally, while Myosim 1.0 had the option to save output signals along with the generative parameter values, and the constituent components (SFAPS, MUAPs, impulse trains etc.), it did not provide any means to load components back into Myosim for reuse. It was therefore difficult to recreate setups for comparison since the tool randomises several generative parameter values based on ranges specified by a user. Since such comparisons may be useful, for example in studies looking to ascertain the effects of generative parameters on signal features, we added the option to load signal components back into Myosim 2.0 for reuse. Now the tool provides the option to regenerate signals with a subset of components (e.g. the impulse trains, half of the MUAPS etc.) from another simulated signal.

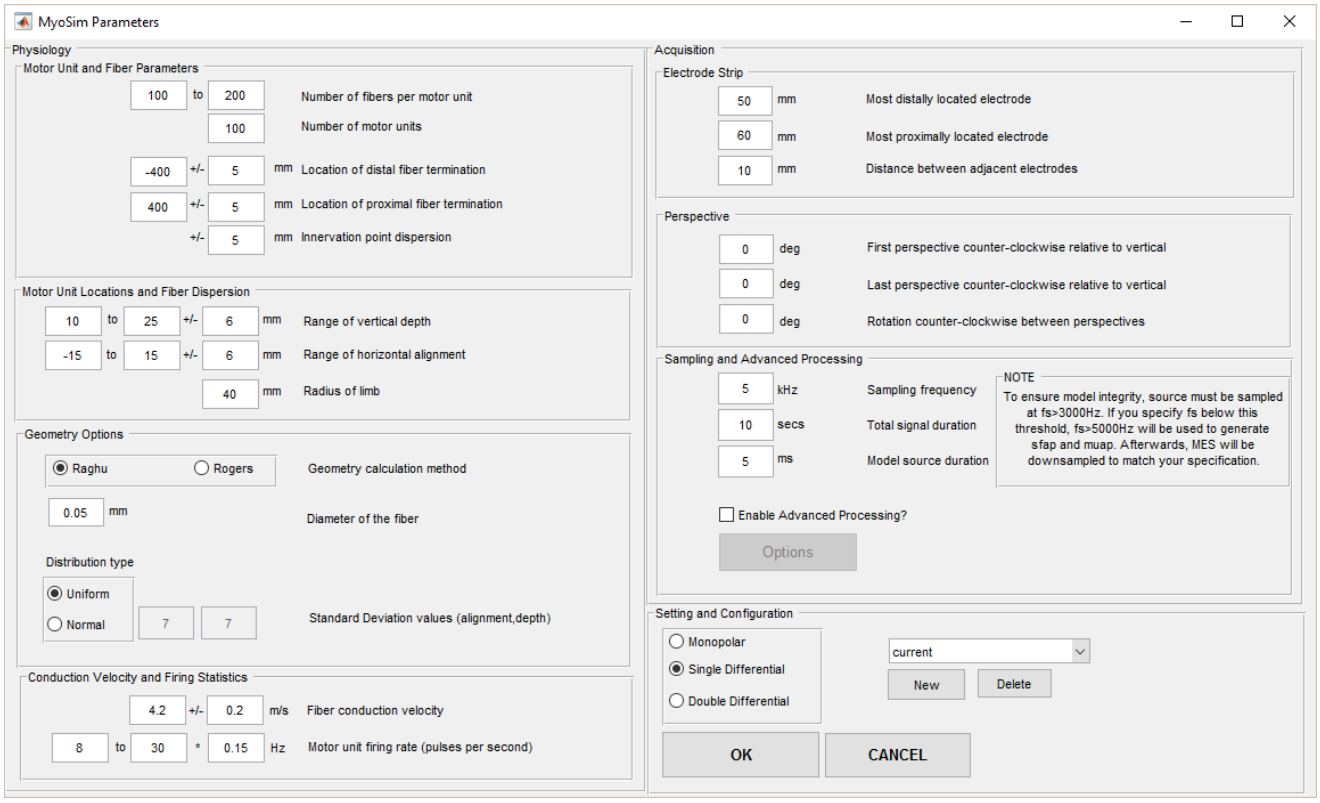

Figure 4: Myosim 2.0 parameters

3. Methods

Because of the random nature of SEMG signals, it is difficult to verify that an SEMG simulator is producing signals representative of those collected through SEMG. Since the values of the generative parameters driving a real signal are unknown, it is impossible to know what generative parameter values to set when comparing a simulated output to a real signal. To offer some support for the validity of Myosim 2.0, we used a genetic algorithm (GA) to identify a set of generative parameter values optimized to match a simulation output with a recorded signal.

3.1. Genetic Algorithm

Genetic Algorithms (GAs), first proposed by John Holland in 1975, are a subset of Evolutionary Algorithms (EAs) which were inspired by biological evolution and use the concepts of chromosome propagation and random mutation to find optimal solutions(s) for a given problem (Bodenhofer 2000). GAs are good at optimising a wide range of problems and generally find acceptable solutions in a relatively short time. They are often a preferred strategy for optimisation because they do not need gradient information and avoid getting caught searching about local minima/optima.

The canonical GA consists of three main steps or operators that are repeatedly performed in a loop: fitness proportionate selection, crossover, and mutation. The algorithm starts by arranging the parameters to be optimised in a structure commonly referred to as "chromosome". An initial collection of potential solutions is generated by creating combinations of randomly generated values for each of parameters. Each set of random values (potential solution) is called an "individual" and the collection is called the "population".

The number of individuals in a population is user-defined and depends on the problem. The function being optimised, called the "objective function", is applied to the population and a corresponding "fitness score" is calculated for every individual, based on the output of the objective function. If an individual produces an acceptable value of the objective function, the algorithm terminates and returns those individual(s), which are the solutions for the optimisation problem.

The next stage is the selection process, where the algorithm then probabilistically selects individuals from the current population to propagate to the next cycle. The selection process favours individuals with better fitness scores, and the selected individuals are used to construct an intermediate population. The intermediate population can have multiple copies of an individual. Within this intermediate population, a subset of random individuals are selected to undergo crossover. The number of individuals selected for the crossover is determined by the user-defined parameter crossover rate (p_c). The crossover

operator essentially swaps values of randomly selected parameters (determined by the crossover points) between two individuals, potentially creating better individuals from existing population.

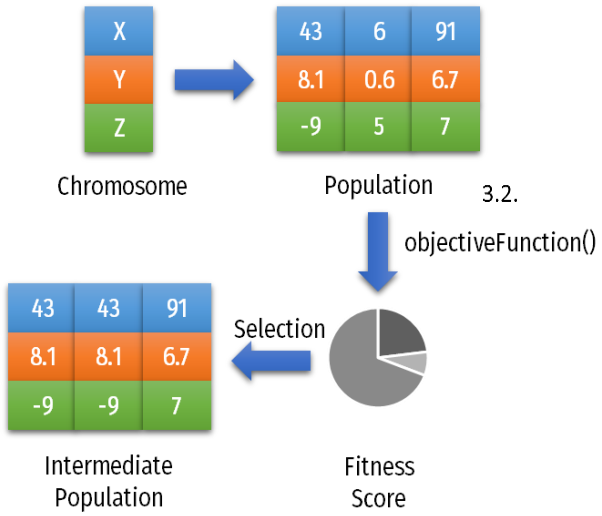

Figure 5: Population initialisation and selection in a GA.

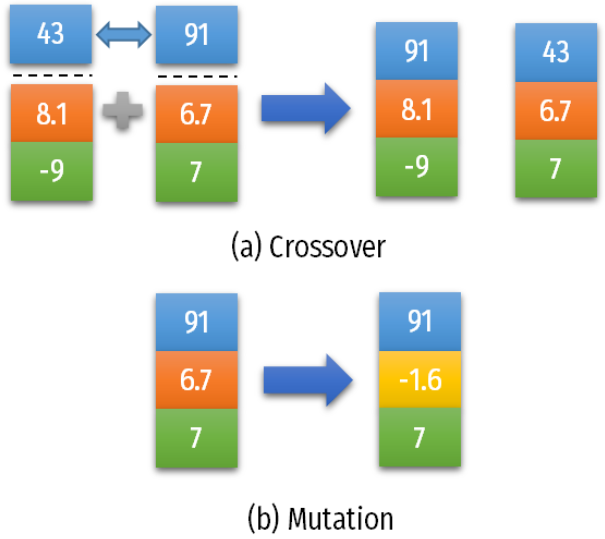

Figure 6: Crossover and mutation operations

The final step in the loop is to mutate random parameters in the population. The mutation operator effectively swaps the value of a parameter to another random value. The number of parameters that undergo mutation in a population is determined by the user-defined parameter mutation rate (p_m). The algorithm then repeats the three steps (fitness proportionate selection, crossover and mutation) until a solution is found or for a pre-determined number of iterations. The control parameters of the GA such as population size, p_c, p_m, etc., influence the performance of the algorithm and it has been suggested that the GA can perform relatively well with a range of settings (Grefenstette 1986). A flow chart providing an overview of the steps is shown in Figure 7.

GA as a validation technique

In order to test if Myosim 2.0 can generate representative SEMG signals, a validation technique involving a genetic algorithm was devised. Several parameters have to set before Myosim 2.0 can produce an SEMG signal. Preliminary analysis identified four of these parameters in playing a major role in defining the generated SEMG signal: number of Motor Units (MUs) (M), number of fibres per MU (N), the depth of the MUs (D) and the average Conduction Velocity (CV).

For each of these parameters, a range of reasonable values based on physiological limits was established as listed in Table 1, and a canonical GA was setup to optimise the four parameters. The GA control parameters were determined through trial and error and are also listed in Table 1.

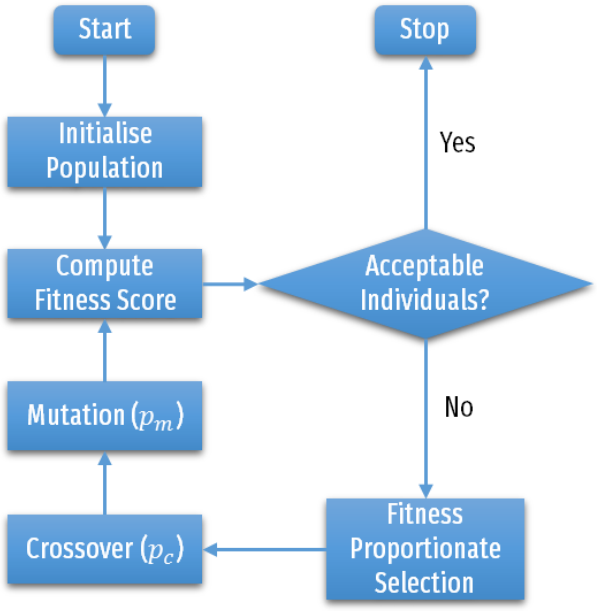

Figure 7: Genetic Algorithm process

Parameter	Value
M	50 – 200
N	100 – 400
D	1 – 15 mm
CV	3 – 6 m/s
Population size	100
Crossover rate	10%
Mutation rate	5%

Table 1: GA parameters

The goal of the optimisation was to find parameters that would generate a simulated signal that "matches" signals collected *in vivo* (records). "Match" here is defined as having similar power spectral densities (PSDs). In order to measure the similarity of two spectra, the mean absolute error (MAE) was computed between a generated signal and a particular record; The MAE was normalised by the total power of the record.

The fitness score was derived from MAE between the PSDs. Search continued until a solution was found with a fitness score which indicated that the MAE was within the variation expected among power spectra of SEMG signals assumed to come from the same contraction. To determine the expected variation, a long signal was simulated and split into segments. The first segment was used as a template and the remaining segments were used to determine the mean MAE and standard deviation. The fitness threshold was then set to 1.15 times the mean + 3 standard deviations, which yielded a threshold of about 0.0075.

The GA was used to match a total of 22 recordings taken from 11 subjects at 2 contraction levels. These signals were collected as part of another SEMG study aimed at investigating variability in SEMG signal features (Shi et al 2018). Signals were measured from the biceps brachii, with electrodes in bipolar configuration, and amplifiers set to bandpass the signal from 20-500Hz. An instrumentation amplifier with CMRR>100dB was used. Signals were sampled at 5000Hz.

In order to verify that the results of the matching process, four commonly used time domain features were computed for both the record and the signals matched (Scheme & Englehart, 2014): Mean Absolute Value (MAV), Slope Sign Changes (SSC), Waveform Length (WL), and Zero Crossings (ZC). The mean frequency (MEANFREQ or MNF) of the signals were also computed as it is commonly used in studies related to muscle fatigue (Phinyomark et al. 2012).

4. Results and discussions

Figure 8 compares the feature values calculated from a recorded signal with feature values calculated from the matched simulated signal. Each of the recorded signal data points comes from a 1 sec segment of a 5-sec record. Each of the simulated signal data points come from a 1-sec segment of the matching simulated signal. Each data point has been normalized to the mean value of the recorded features for visual clarity. This example is representative of the results across all 22 matches. In all cases, feature values extracted from the simulated signal fell within the same range as values extracted from different segments of the same recorded signal. Thus, when tool parameters are set appropriately, Myosim 2.0 will produce simulated SEMG signals with features within the same range as recorded data. This result supports the validity of Myosim 2.0 as an automated solution for simulating signals representative of SEMG recordings.

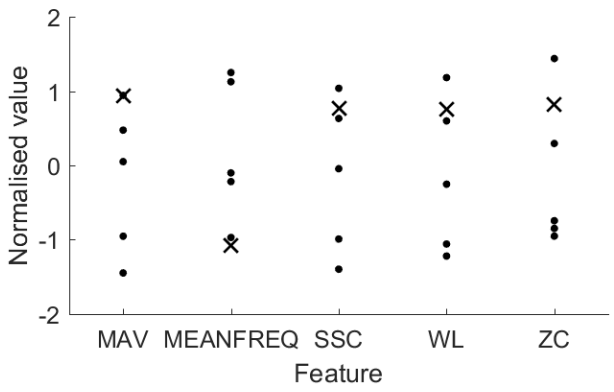

Figure 8: Feature values calculated from Recorded Signal (**.**) and matching simulated signal (**x**).

5. References

Parker, P., Englehart, K., & Hudgins, B. 2006. Myoelectric signal processing for control of powered limb prostheses. Journal of Electromyography and Kinesiology, 16(6), 541–548. https://doi.org/10.1016/j.jelekin.2006.08.006

Scheme, E., & Englehart, K. (2011). Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use. The Journal of Rehabilitation Research and Development, 48(6), 643. https://doi.org/10.1682/JRRD.2010.09.0177

Turker, H., & Sze, H. (2013). Surface Electromyography in Sports and Exercise. In D. H. Turker (Ed.), Electrodiagnosis in New Frontiers of Clinical Research. InTech. https://doi.org/10.5772/56167

Mohseni Bandpei, M. A., Rahmani, N., Majdoleslam, B., Abdollahi, I., Ali, S. S., & Ahmad, A. (2014). Reliability of Surface Electromyography in the Assessment of Paraspinal Muscle Fatigue: An Updated Systematic Review. Journal of Manipulative and Physiological Therapeutics, 37(7), 510–521. https://doi.org/10.1016/j.jmpt.2014.05.006

Choi, C., Na, Y., Rim, B., Kim, Y., Kang, S., & Kim, J. (2013). An SEMG computer interface using three myoelectric sites for proportional two-dimensional cursor motion control and clicking for individuals with spinal cord injuries. Medical Engineering & Physics, 35(6), 777–783. https://doi.org/10.1016/j.medengphy.2012.08.006

Wei, L., Hu, H., & Zhang, Y. (2011). Fusing EMG and visual data for hands-free control of an intelligent wheelchair. International Journal of Humanoid Robotics, 8, 707–724. https://doi.org/10.1142/S0219843611002629

Rogers, D. R., & MacIsaac, D. T. (2010). Training a multivariable myoelectric mapping function to estimate fatigue. Journal of Electromyography and Kinesiology, 20(5), 953–960. https://doi.org/10.1016/j.jelekin.2009.11.001

Fraser, G. D., Chan, A. D. C., Green, J. R., & MacIsaac, D. T. (2014). Automated Biosignal Quality Analysis for Electromyography Using a One-Class Support Vector Machine. IEEE Transactions on Instrumentation and Measurement, 63(12), 2919–2930. https://doi.org/10.1109/TIM.2014.2317296

Supuk, T. G., Skelin, A. K., & Cic, M. (2014). Design, Development and Testing of a Low-Cost sEMG System and Its Use in Recording Muscle Activity in Human Gait. Sensors (Basel, Switzerland), 14(5), 8235–8258. https://doi.org/10.3390/s140508235

MacIsaac D. T., Rogers D., & Bhandarkar D. (2006). Simulating myoelectric signals with a finite length model of muscle. Proceedings of the 29th Conference of the Canadian Medical and Biological Engineering Society.

Gonzalez-Cueto, J. A., & Parker, P. A. (2002). Deconvolution estimation of motor unit conduction velocity distribution. IEEE Transactions on Biomedical Engineering, 49(9), 955–962. https://doi.org/10.1109/TBME.2002.802011

Plonsey, R. (1974). The active fiber in a volume conductor. IEEE Transactions on Biomedical Engineering, BME-21(5), 371–381. https://doi.org/10.1109/TBME.1974.324406

Bodenhofer, U. (2003). Genetic algorithms: theory and applications.

Grefenstette, J. J. (1986). Optimization of Control Parameters for Genetic Algorithms. IEEE Transactions on Systems, Man, and Cybernetics, 16(1), 122–128. https://doi.org/10.1109/TSMC.1986.289288

Shi, Y., MacIsaac D.T, Parker P.A (2018), FILL ME.

Scheme, E., & Englehart, K. (2014). On the robustness of EMG features for pattern recognition based myoelectric control; A multi-dataset comparison. In 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 650–653). https://doi.org/10.1109/EMBC.2014.6943675

Phinyomark, A., Thongpanja, S., Hu, H., Phukpattaranont, P., & Limsakul, C. (2012). The Usefulness of Mean and Median Frequencies in Electromyography Analysis. In G. R. Naik (Ed.), Computational Intelligence in Electromyography Analysis - A Perspective on Current Applications and Future Challenges. InTech. https://doi.org/10.5772/50639

Veen, B. K. van, Rijkhoff, N. J. M., Rutten, W. L. C., Wallinga, W., & Boom, H. B. K. (1992). Potential distribution and single-fibre action potentials in a radially bounded muscle model. Medical and Biological Engineering and Computing, 30(3), 303–310. https://doi.org/10.1007/BF02446968

Buchthal, F., Guld, C., & Rosenfalck, P. (1957). Volume Conduction of the Spike of' the Motor Unit Potential Investigated with a New Type of Multielectrode. Acta Physiologica Scandinavica, 38(3–4), 331–354. https://doi.org/10.1111/j.1748-1716.1957.tb01396.x

MT-048: Op Amp Noise Relationships: 1/f Noise, RMS Noise and Equivalent Noise Bandwidth. (2009). Analog Devices, Inc. Retrieved from http://www.analog.com/media/en/training-seminars/tutorials/MT-048.pdf

Mancini, R. (2003). Op amps for everyone: design reference. Newnes.

Clamann, H. P. (1969). Statistical Analysis of Motor Unit Firing Patterns in a Human Skeletal Muscle. Biophysical Journal, 9(10), 1233–1251. https://doi.org/10.1016/S0006-3495(69)86448-9