XVI CEEL - ISSN 2178-8308
Universidade Federal de Uberlândia
05 a 09 de novembro de 2018

PROPOSAL OF A SYSTEM FOR REAL-TIME EMG ACQUIRING, PLOTTING AND DETECTION OF CONTRACTIONS

Ítalo Gustavo Sampaio Fernandes*[1], Lucas Lemos Franco[1] and Júlia Nepomuceno Mello[1]

[1]FEELT - Universidade Federal de Uberlândia

Abstract - **Electromyographic signals (EMG) are widely used in Human Machine Interface applications, in the control of myoelectric prostheses and in the control of models in virtual reality environments. To do so, it is necessary to process the EMG signals in order to extract the necessary information for each application. One of the key steps in the processing of EMG signals is the accurate detection of the beginning and end of muscle contractions. Thus, it is necessary to use methods and algorithms that aim to accurately detect onset and offset times of muscle activity, using techniques that involve the calculation of the signal envelope, calculation of thresholds and digital filters.**

The present work aims at the development of a system capable of performing the acquisition and plotting of EMG signals, as well as onset and offset detection of muscle activity. The system is consisted of hardware, in order to acquire the signal, and software, which is based on parallel processing for real-time detection. The processing method proposed consists in the application of the Hilbert transform with a low-pass filter to calculate the envelope.

There was tested two approaches for the smoothing of the rectified signal, being the moving average algorithm the one which showed better results. The methods used in this work present satisfactory results even in a computer with lower power processing, besides it was developed in a reusable way, allowing the interaction with other software applications.

Keywords- **Acquisition, Electromyography, Human Machine Interface, Signal Processing, Software.**

I. INTRODUCTION

The control of myoelectric prostheses is based on the use of electromyographic (EMG) signals collected mainly on the surface of the skin and, generally, on the remaining musculature, being such a technique widely used in prosthesis control, as in the prostheses of the upper limb [1]. The EMG signals are formed by physiological variations in the state of muscle fiber membranes, that is, they consist of electrical potentials produced by the contraction of a given muscle or muscle group [2,3]. Thus, these signals consist of an important human-machine interface, since after being captured and digitized, it is possible to process them for the recognition of patterns and classification of movements to be executed by a physical prosthesis or in a virtual reality environment [3]. The analysis of the EMG signals represents an important tool in a variety of applications like neuromuscular and psychomotor research, prosthetics, rehabilitation or Human Machine Interface (HMI) applications [3,4]. The detection of the exact onset and offset times of a muscle contraction is one of the challenges in the processing of this signals and there are already several methods and algorithms that try to solve this problem [4]. In recent studies, a method that presented good results consists of using the Hilbert Transform to extract an envelope, which is smoothed by a low-pass filter, and the onset determined from the calculation of a simple threshold [5]. Such a technique is widely used, for example, associated with Empirical Mode Decomposition for EMG signal filtering and onset detection [5,6]. In the domain of the control of prostheses and applications in HMI, it is of fundamental importance a system that works in real time [1,3]. In this context, onset detection is one of the processes to be executed, as well as feature extraction, pattern recognition and movement classification [1,3]. For control of prosthetic limbs, the response time, to reach real-time constraints, must be less than 300 ms [3]. In this way, it is possible, from the electromyographic signals, to perform a satisfactory control of myoelectric prostheses, models in virtual reality, games and others applications [1,3,7]. In the light of the above, this article presents a system for acquiring and plotting EMG signals, as well as detecting the onset and offset of muscle contractions in real time.

II. OBJECTIVES

This work has as main objective the development of a system for acquisition and plotting of EMG signals. For this central purpose, the following objectives were defined:

- Acquire an EMG signal with at least 1 kHz as sampling frequency.
- Plot this signal during the acquisition giving the impression of real time.

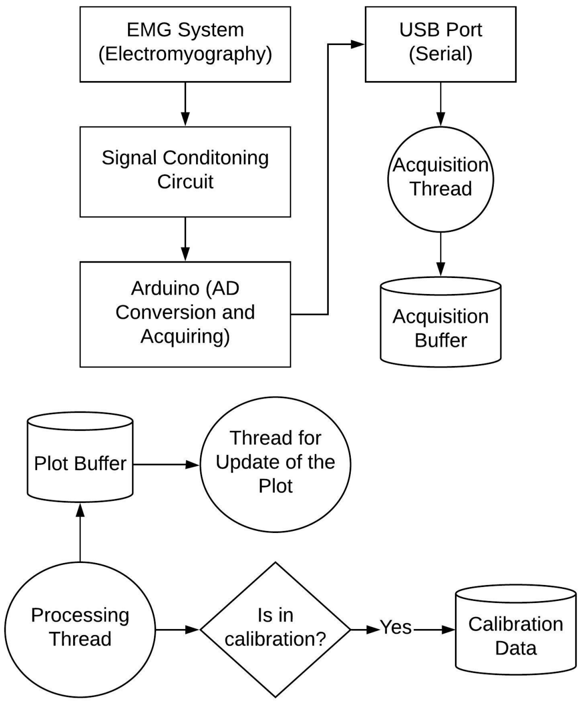
*italogsfernandes@gmail.com

- Process the raw EMG signal and subsequently filter it as well as calculate its envelope.
- Detect the contraction sites by analyzing the envelope of the signal.
- Develop the software encapsulated as classes to allow the use for other researchers and the development of more complex applications.

III. MATERIALS AND METHODS

The complete system was divided in different parts, each one being responsible for a task, this is illustrated in Fig.1.

Figura 1: Fluxogram of the data path through the hardware and software.



A. Hardware

An electromyography device was used to provide a raw EMG signal in a range from -10 V to 10 V. Besides, a circuit was projected to adjust the signal from its range to a new range, one of 0 V to 3.3 V. so it could be read by the internal analog-to-digital (AD) converter present in the Arduino Due Board. The device filters the raw EMG signal, remaining only the frequencies between 20 Hz and 500 Hz. The analog conversion was made using 12 bits of precision and at a rate of 1000 Hz. It is important to notice that, for controlling this sampling frequency, an internal timer of the Arduino Due was implemented, controlled by the library DueTimer[9]. With this, an accurate sampling frequency (1 0,008 kHz) could be obtained. After the AD conversion, each one of these 12 bits readings were sent as a packet, by serial communication, each packet with a starter flag, the data, represented as a 16 bits unsigned integer, and an ending flag. For each data, 4 bytes is sent, totalizing almost 32 kbits/s.

B. Software

The software was developed using Python 2.7. It can be separated in the following sections: acquisition, processing, calibration and plotting routines. For performance reasons, the software was developed using parallel processing. The main routines run using separated threads and exchanging data through buffers. Before explaining those routines, it is quite useful to understand how these threads and buffers are arranged. The threads are created to run a determined routine repeatedly, each thread has a worker function with the following structure: a repetition structure that calls a specified routine while the thread is alive and, after the last call, a function to conclude. The buffers are implemented as queues, each buffer was represented by a first-in-first-out (FIFO) structure with a defined maximum size.

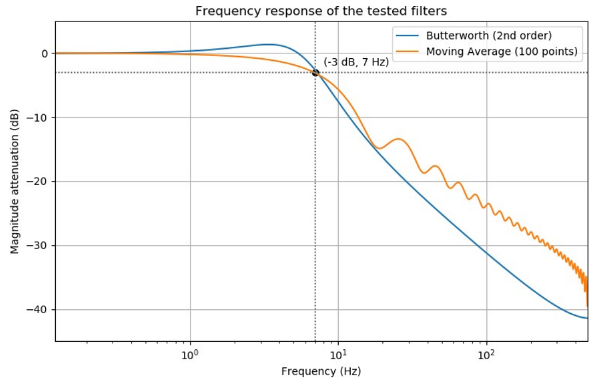
C. Signal Acquisition

A class for acquiring the data was implemented, consisting of a serial port object, a thread and a buffer. The role of this class is to continuously verify the serial interface.

D. Signal Processing

In order to process the EMG signal, a class was developed employing the process routines that were implemented in the SciPy library. Thus, the following step just after the signal acquisition consists in build the raw EMG signal envelope. For that, from the 64 points windows, which means 64 ms of data (considering the sampling frequency 1 kHz), the Hilbert Transform was calculated, since this method shows good results according to the literature about the detection of onset from muscular activity [5, 6]. Among all the vantages presented, the fact that this method can be considered as a solid option to detect fast oscillation signal stands out [6]. Therefore, to conclude the envelope's elaboration by the Hilbert Transform, a full- wave rectification was performed, removing the negative portion. Thus, to verify the efficiency from different filters in smoothing the envelope, a moving average (MVA) filter and a low-pass filter were created, as specified later. The MVA filter, which eliminates the discrepancy and points out the real variation in data set, is used in this very paper as a low-pass filter to attenuate noise inherent in varied waveforms [10]. In that way, the filter's window was stipulated with 100 points, with a sampling frequency of 5 kHz, so the space between two points is 0,2 ms. A point generated by a MVA filter corresponds to the interval of 0,2 ms multiplied by 100, the number of points, which results in a signal filtered at 50 Hz. Besides, a second order Butterworth low-pass filter with cut-off frequency of 7 Hz was used [12]. The responses in frequency of the both elaborated filters can be seen in Fig. 2. Hence, it is possible to select which filter is used to smooth the signal's envelope who was calculated through Hilbert Transform.

Figura 2: Frequency response of the proposed filters



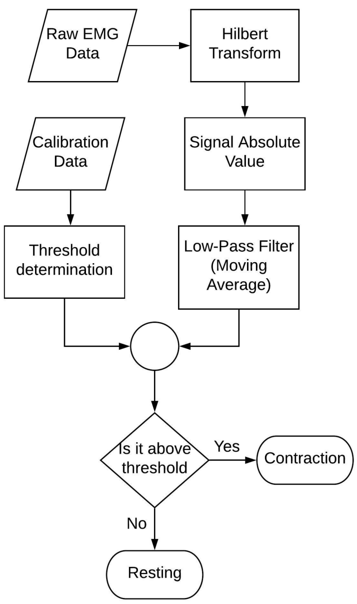
After the envelope's construction step and signal filter, a test function and an algorithm are used to determine, through a decision rule, whether a sample can be deemed as an onset or not [6]. Then, as shown in the literature, it was decided to use a simple threshold to determine the onset, being such threshold calculated from a signal compound only by noise and artefacts, i.e., without the muscular contraction [6, 11]. For that, the threshold was calculated as presented in the Equation 1 [6], considering a base signal captured during a calibration step. This method (Hilbert Transform Envelope associated with a simple threshold detection) is also described as a good method for regions of activity that may contain motor unit action potentials (MUAPS) [11]. One notes, by observing the Equation 1, that the multiplicative factor of the standard deviation chosen is 2.

$$th = y + 2 * S \tag{1}$$

Where:

th	-	Threshold
y	-	Mean of the EMG envelope.
S	-	Standard deviation of the EMG envelope.

Figura 3: Processing routine.



E. Signal Plotting

The last, but no less important step is the signal plotting. There was used the QT framework, a cross-platform framework for creating graphical user interfaces, this framework provides the widgets and tools for developing the interface where the user will interact with the software as and the library Pyqtgraph for creating charts. For an idea of continuous plotting it is necessary to update a chart frequently. Considering that the signal is acquired at a rate of 1 kHz, it is not possible to add each new point to a chart and update the chart simultaneously. To solve this, a plotting buffer was implemented, so each new point is added to this buffer and, at a determined rate, the points in the buffer are added to the chart and the older ones are removed. For not distorting the "real-time impression" that the human eyes have, it is necessary a minimum of 20 fps (frames per second).

F. Signal Testing

For testing the system, it was necessary to use some EMG signal with known characteristics. Synthetic signals were used, i.e., signals generated by computer. The same method was used by other authors[5].

In order to generate the signals, a simulator was used, which was developed by other authors[11]. This simulator is based in data EMG from real signals collected through an experimental protocol properly controlled [11]. Beside that, this strategy was tested and validate. For signal generation, it is possible to define diverse characteristics like quantity of motor units, quantity of electrodes and signal-to-noise ratio (SNR). The parameters used for signal generated are listed below on the Table 1.

Tabela 1: Parameters for the simulated EMG signals. [Adapted of Nakagawa et al [5]]

Parameter	**Value**
Amount of Contractions	10
Duration of the contractions	750 ms
Resting time during contractions	3 s
Active Motor Units	5
Signal-to-noise ratio (SNR)	10 dB

The EMG signal produced for test is represented in Fig. 4. It contains 10 contractions of 750 ms spaced in 3 seconds one from another. The SNR is 10 dB In the Fig. 5, it is shown an isolated contraction, where it is possible to notice that the beginning happens in 2 s and the end in 2.76 s. Those values are used for comparison and data validation from the beginning to the end of the contraction that were detected by the developed system. It will be calculated the error through the distance between the values and from that the mean error form the beginning and from the end of contraction will be calculated.

Figura 4: Generated signal with 10 contractions.

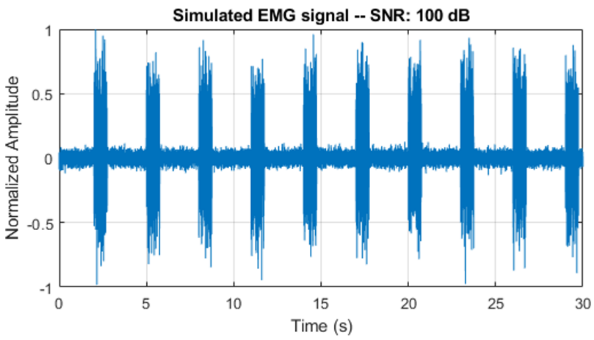
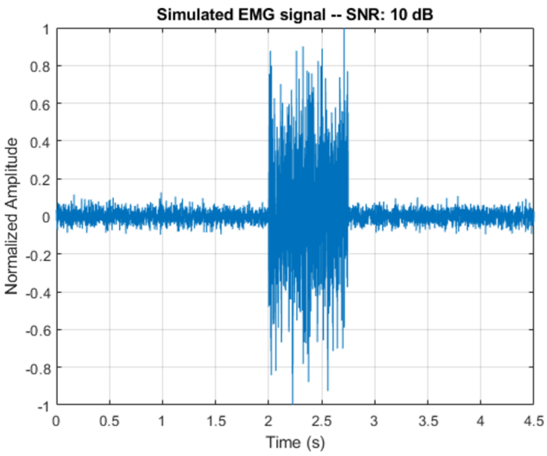


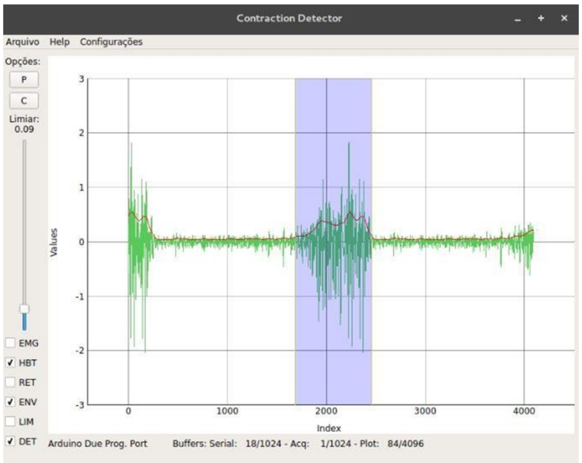
Figura 5: One single contraction beginning in 2 s and the end in 2.76s.



IV. RESULTS AND DISCUSSIONS

The developed system presented the following result:

Figura 6: Graphic User Interface developed for the system.

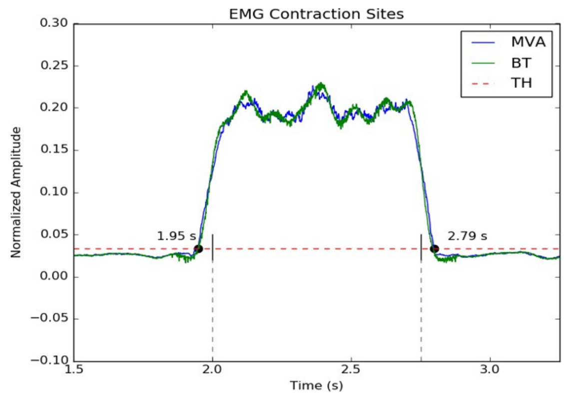


In the implemented solution, the user can select with result he wants to see. They are:

- EMG: Raw EMG values.
- HBT: Analytic Signal, after the application of the Hilbert Transform.
- RET: Rectified Signal, the absolute values of the HBT curve.
- ENV: Envelope of the signal.
- LIM: Threshold of detection, the threshold determined during the calibration.
- DET: Detection Sites, a blue marker indicating where the detected contractions sites

For having two different options of filters that smooth the signal to get the envelope, it was possible to compare them and verify whether it is possible to obtain better results, specially because it is about a real-time application. In Fig. 7, the envelope of each one can be seen. One notes that the MVA filter is more stable, since it was implemented using windows. In the other hand, the Butterworth filter has not shown satisfactory results when applied to windowed signal.

Figura 7: Comparison between the moving average filter (MVA) and the butterworth response filter (BT), it also shows the threshold value (TH).

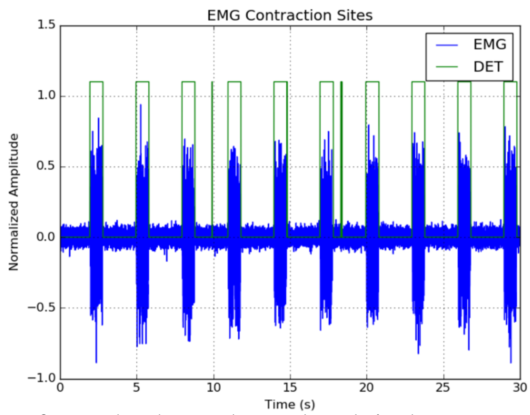


The Butterworth filter presented two false positives in the zones where the windows are equal to zero, detecting contraction in spots where that was none (Fig. 8). Besides, since from the beginning of the contractions, detection of pre-contractions occurred with a duration too small, given the fact that the signal surpasses the threshold for a quick moment and then comes below again, and after that, it repeats and being on the edge of the threshold. The filter's instability is due to the response of the signal near the borders of our windows: in the middle of the signal the response is stable, nevertheless near the border the signal becomes unstable and not reliable. Reliable publications point out the use of the moving average as algorithm (Zhang et al. and Lian et al.)[7][8].

Tabela 2: Mean errors of the detections of contractions compared to actual time of start and end.

-	Butterworth Filter	MVA filter
Start of Contraction	mean: 44 ms std: 5 ms	mean: 48 ms std: 3 ms
End of Contraction	mean: 42 ms std: 6 ms	mean: 46 ms std: 2 ms

Figura 8: System tested using the Butterworth filter.



Another factor that has to be analyzed, is the system performance. An update rate of approximately 20 fps was used, considering a computer with the following configurations:

- Processor Intel Core i3-3217U with 4 cores of 1.80 GHz;
- RAM memory of 3.8 Gb;
- Running Ubuntu as the operating system.

This update rate changes depending on the amount of information to be plotted and the strategy used to process the data. The Butterworth option has showed to be more computationally expensive than the moving average option, and an update rate of 50 fps was observed when plotting only the envelope of the signal.

V. CONCLUSION

The developed system has accomplished its objectives, being capable of acquiring the signal with a proper sampling frequency, obtaining the contraction sites by a signal processing routine and doing it in time to be quickly plotted. The software was developed to be reusable in other projects, some social coding directrices were followed, so the software is distributed as an open-source in a GitHub repository (available at `http://www.github.com/italogsfernandes/str-contraction-detector`). As referred in the introduction, the response time must be less than 300 ms [3] and the system achieved a response time of 80 ms in his worst case. This means that the software can be used together with a classifier, with more channels or with other processing steps. Besides, it is expected that the response time will remain lower than 300 ms. It is intended to improve the system performance, allowing it to be used as an acquiring a pre-processing routine for hand movements classification and also to control virtual prosthesis based on myoelectric signals.

REFERÊNCIAS

[1] Herle S, Raica P, Lazea G, Robotin R, Marcu C, Tamas, L.: Classification of surface electromyographic signals for control of upper limb virtual prosthesis using time-domain features. In: Automation, Quality and Testing, Robotics; 2008. IEEE International Conference; 2008. p. 160-5.

[2] Mattioli FE, Lamounier EA, Cardoso A, Soares AB, Andrade AO.: Classification of EMG signals using artificial neural networks for virtual hand prosthesis control. In: Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE; 2011. p. 7254-7.

[3] Nazmi, Nurhazimah et al.: A review of classification techniques of EMG signals during isotonic and isometric contractions. Sensors, v. 16, n. 8, p. 1304, 2016.

[4] Staude, Gerhard et al.: Onset detection in surface electromyographic signals: a systematic comparison of methods. EURASIP Journal on Applied Signal Processing, v. 2001, n. 1, p. 67-81, 2001.

[5] Silva, Andrei Nakagawa et al. Método para detecção do ONSET da atividade muscular para medida do limiar do reflexo de estiramento tônico. 2015.

[6] Silva, Maristella Borges et al. An improved approach for measuring the tonic stretch reflex response of spastic muscles. Computers in biology and medicine, v. 80, p. 166-174, 2017.

[7] Zhang, Xu et al.: Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors. In: Proceedings of the 14th international conference on Intelligent user interfaces. ACM, 2009.

[8] Lian, Kuang-Yow et al.: Wearable armband for real time hand gesture recognition. In: Systems, Man, and Cybernetics (SMC), 2017 IEEE International Conference.

[9] Seidel, Ivan. Due Timer, 2013. Retrieved from https://github.com/ivanseidel/DueTimer.

[10] Barros, Kety Rosa de et al. Metodologia para classificação de sinais EMG para controle de próteses com baixo esforço computacional. 2005.

[11] Andrade, Adriano O.; Nasuto, Slawomir J.; Kyberd, Peter. Extraction of motor unit action potentials from electromyographic signals through generative topographic mapping. Journal of the Franklin Institute, v. 344, n. 3-4, p. 154-179, 2007.

[12] J. Gross, L. Timmermann, J. Kujala, M. Dirks, F. Schmitz, R. Salmelin, and A. Schnitzler. The neural basis of intermittent motor control in humans. PNAS February 19, 2002.