Electromyography haptic paddle control

Maxime Gardoni
EPFL, 271088
Lausanne, Switerland
maxime.gardoni@epfl.ch

Florian Richter
EPFL, 2576668
Lausanne, Switerland
florian.richter@epfl.ch

***Abstract*— This document presents the control of a haptic paddle via electromyography (EMG). Two electrodes were placed on a biceps and the electric signal of the muscle were recorded. Signal processing methods processing such as band pass filter, notch filter and root mean square (rms) were applied on the EMG signal to extract its envelope. A position control strategy on the haptic paddle was successfully implemented, with the angle of the paddle being proportional to the muscle activity.**

***Video* — https://youtu.be/1MMvh0Evlwo**

Keywords — EMG, muscle signal, signal processing, filters, root mean square, electromyography, human-robot interface

I. Introduction

Electromyography (EMG) is an electrodiagnostic medicine technique for evaluating and recording the electrical activity produced by skeletal muscles. [1]

The electric signal that causes the muscle to contract is recorded by electrodes places on it. This method is non-invasive and therefore is easy to use. However, there is a high signal to noise ratio since there is a non negligeable distance between the electrode on the skin and the muscle fibers.

EMG has a variety of application, from identifying neuromuscular diseases to controlling exoskeletons.

In this work, we explore the possibility of controlling a haptic paddle in position using a small pair of external electrodes placed on the biceps of the user.

II. Method

A. Hardware

The paddle is mounted on a *Faulhaber 2642W024C R* motor, that is connected to the main board (*STM32F407*). An EMG amplifier board (*Advancer Technology Myoware)* is also connected to the board. It is composed of three electrodes (two detection electrodes and one reference electrode) and an amplifier that subtract the signal of the two detection electrodes and amplifies the difference. The Myoware board can compute the envelop itself, but in this work, the signal processing step were implemented on the STM microcontroller.

B. Electrode positionning

As the pin of the two detection electrodes are fixed on the EMG amplifier board, the inter-detection distance is set and is approximately equal to 5 [cm]. Following *[2]* the detection electrodes are fixed on the biceps of the user and should not be placed on a motor point nor a tendon. Additionally, the two electrodes need to capture the signal of the same muscle fibers and thus should be aligned in the direction of muscle fibers length.

The reference electrode provides a common reference for the detection electrodes, therefore it should be placed as far as possible from the latter and on an electrically neutral tissue, such as the elbow or on the tendon of the wrist.

C. Data collection procedure

The EMG signal was recorded multiple times during muscle resting position and muscle contraction. Additionally, the transition for one state to the other (resting state or muscle contraction) was set manually by moving the paddle shaft from one end to the other. I.e. when the user contracted its muscle, he simultaneously moved the paddle from a negative angle to a positive angle. The paddle moved in the opposite direction when he stopped contracting.

This manual movement of the paddle helped separating the input signal in two signals which are the resting state signal and the muscle contraction signal. Since the movement of the paddle is not precisely correlated with the contraction or the relaxation of the muscle, we removed datapoints around the transition from one state to the other in order to avoid unwanted behaviour of the signal.

D. Signal processing

Since the recorded signal is very noisy, multiple signal processing steps are necessary to analyse it properly.

The frequency spectrum of the signal in both contraction mode and at resting state are computed in MATLAB using built-in function of the FFT. This helped identifying the main frequency response between the activated and the resting signal. Those happen between 10 Hz and 200Hz. Moreover, we noticed that a strong noise was present at 50Hz due to the AC/DC converter used in the paddle. We computed the power of the signal in the Fourier domain with a range of frequency $\in [10, 48] \cup [52, 200]$ (Hz)

$$Power = \sum_i freq_i^2 \tag{1}$$

The power was 4.5 time higher for the contraction signal than the resting state signal. Interestingly, the power of the signal was highly correlated with the moving standard deviation of the signal *using a sliding window of 1000 datapoints*. However, the power in the Fourier domain is close to 0 when the muscle is relaxing whereas the moving standard deviation has an offset.

E. Implementation on the board

Computing the FFT was computationally fast on a computer but took too much time on the embedded board.

Another attempted approach consisted of computing the moving standard deviation on the board, but it appeared to be too slow as well (even though it was much faster than the FFT) and not necessarily the optimal method.

The final implemented method is the Root Mean Square (rms) of the signal, which is computationally efficient, as in *C. De Luca*'s work [2].

1) Filtering the data

The first step consisted of filtering the data using the range of frequency computed previously on MATLAB. A low pass and a high pass first order filters were applied sequentially with a cut-off frequency equal to 200 $[Hz]$ and 10 $[Hz]$ respectively, and a sampling frequency *dt* equal to 350 [µs].

The equations used for the lowpass filtering are the following:

$$\tau = \frac{1}{2 * \pi * f_{cutoff}}$$
$$\alpha = \frac{dt}{\tau + dt}$$
$$y_t = \alpha * x_t + (1 - \alpha) * y_{t-1}$$

The equations used for the high pass filtering are the following:

$$\tau = \frac{1}{2 * \pi * f_{cutoff}}$$
$$\alpha = \frac{\tau}{\tau + dt}$$
$$y_t = \alpha * y_{t-1} + \alpha * (x_t - x_{t-1})$$

A notch filter of order two was then applied to remove the AC/DC component at 50 [Hz]. The equations are the following:

$$y[t] = a_0 x[t] + a_1 x[t-1] + a_2 x[t-2] + b_1 y[t-1] + b_2 y[t-2]$$

Where y is the filtered output, x is the raw input and the index in bracket indicates the value to take with respect to the time t. The coefficients are pre-computed using the following equations:

$$a_0 = K$$
$$a_1 = -2 * K * cos(2\pi * \mathrm{f}_{notch}/f_{sampling})$$
$$a_2 = K$$
$$b_1 = 2 * R * cos(2\pi * \mathrm{f}_{notch}/f_{sampling})$$
$$b_2 = -R^2$$

Where:

$$K = \frac{1 - 2 * \mathrm{R} * \cos(2\pi \mathrm{f}_{notch}/f_{sampling}) + R^2}{2 - 2 * \cos(2\pi \mathrm{f}_{notch}/f_{sampling})}$$
$$\mathrm{R} = 1 - 3 * BW/f_{sampling}$$

Where $f_{notch} = 50$ $[Hz]$, and the bandwidth is $BW = 0.023$ *[Hz]*.

2) Root Mean Square

The size of the RMS window (N in the equation below) is a trade-off between the stability of the estimation and the induced lag. Using a big window size induced a more stable signal, but also longer delay. Experiments with a window of size of [200, 500, 1000, 2000] have been conducted and a window size of 1000 samples (*0.35s*) was selected.

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^{N} x_i^2}$$

To implement the equation above, a naïve approach would be to compute the mean over a sliding window size at each time step, which is computationally expensive.

Instead, a circular buffer was implemented. At each step, the newest filter EMG value is raised to the power 2, divided by the window size and added to the array. The oldest element is removed. The square root is then applied over the sum of the array. This implementation was proven computationally cheap.

3) Envelope of the signal

The final step was to keep only the envelope of the processed signal only and a final smooth was needed. This was done by applying a low pass filter with a cut-off frequency equal to *3Hz*.

4) Calibration

Since the envelop value can be user dependent, a calibration procedure was developed. The mean intensity at rest and during maximum muscle contraction are computed separately over a 2 second window.

F. Control

The rest intensity and maxmium contraction intensity are mapped to angles ranging between -25° and +25°, which act as a position setpoint.
A PID controller is then used to track the desired position on the paddle, using the build-in encoder of the paddle.
The gains were manually tuned to the following values:

K_p [N*m/deg]	K_i [N*m/(deg*s)]	K_d[N*m*s/deg]
0.015	0.15	0.0006

Table II.1: PID gains

III. RESULTS

Figure [1], [2] and [3] shows the raw data in the time domain and in the frequency domain.

The magnitude of the muscle signal at rest in the frequency domain has a few unwanted peaks at 0 [Hz], 50 [Hz] and 120 [Hz]. The first two peaks correspond to the constant offset of the signal and the AC/DC component, respectively. However, the last peak does not have a clear meaning, it might be the result of any device close to the electrodes. Therefore, we only get rid of the peaks at 0 [Hz] and 50[Hz] to get a clean signal.

The magnitude of the muscle signal in contraction (figure [3]) shows a range of frequency with a higher response than the resting state. This range of frequency is quite large (10 – 200Hz) which means that there is not a single or a few specific frequencies that drives the muscle activation but rather a range even though the signal is very noisy and the magnitude varies a lot from two neighboring frequencies. However, the two peaks at 0 [Hz] and at 50 [Hz] are still distinguishable.

The filtered signal of the muscle contraction in the frequency domain is shown in figure [5], where a pass band filter [10-200 Hz] and a notch filter [50 Hz] had been applied. The two peaks at 0 [Hz] and 50 [Hz] have been removed and the envelope of the signal is smoother even though there are still some jitters.

Figure [4] shows the signal in the time domain after filtering. Compared to figure [1], the signal is much cleaner. The magnitude is approximately equal to zero when the muscle is at rest and it oscillates with a big amplitude when the muscle is in contraction.

The envelope of the signal in the time domain is shown in figure [6] where the RMS of the signal was computed.
Indeed, the signal had positive and negative values and the RMS brings back the negative value on the positive axis which is needed to compute the envelope. However, the small jitters of the RMS are not desired, and a low pass filter (orange curve) helps smoothing it at the cost of a little delay on the signal.

Finally, the smoothed signal of the RMS (figure [6]) shows clearly the resting states (0 – 0.9 [s], 3.1 – 4.5 [s] and 7.2 – 9[s]), the weak muscle contraction (0.9 – 3,1 [s]) and the strong muscle contraction (4.5 – 7.2 [s]) that can be used to control the position of the haptic paddle.

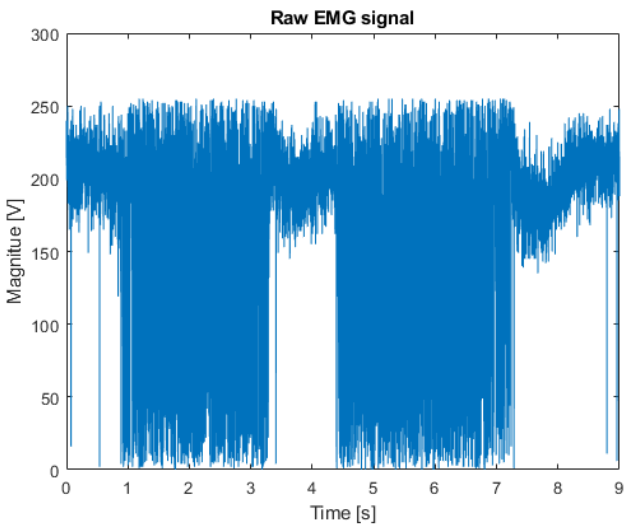

Figure III.1 Raw EMG signal in the time domain with two muscles contractions, a weak and a strong one, respectively at [0.9, 3.1]s and [4.5, 7.2]s. The rest of time the muscle is relaxing.

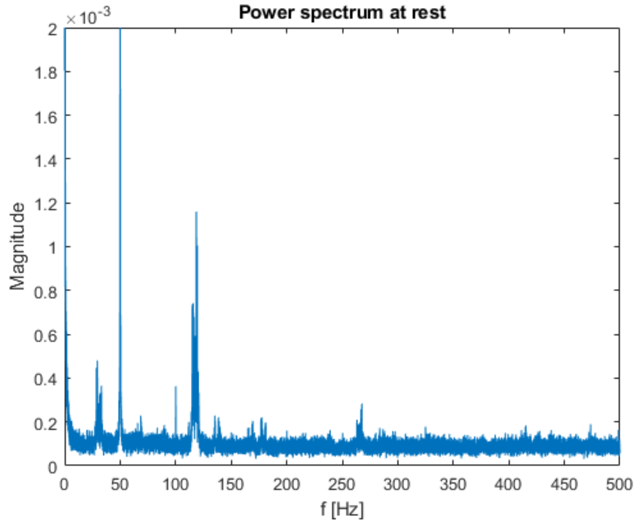

Figure III.2 Signal at rest, represented in the Fourier domain. A 50 [Hz] peak is present due to the AC/DC converter. Also, a strong undesired component is present at 0 Hz

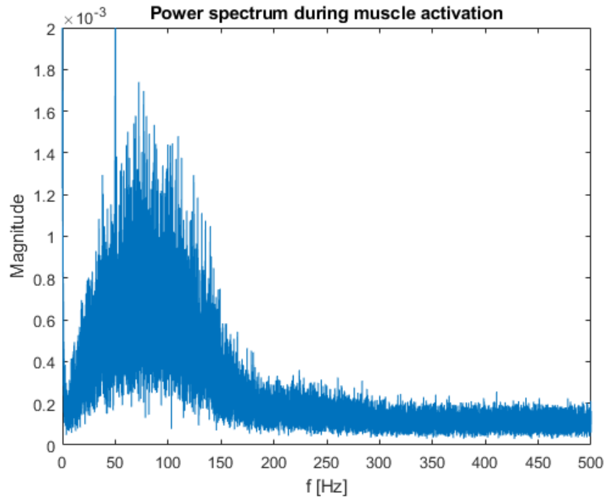

Figure III.3 EMG signal during muscle contraction, represented in the Fourier domain. The main part of the muscle activation lies in the range [10, 200]Hz.

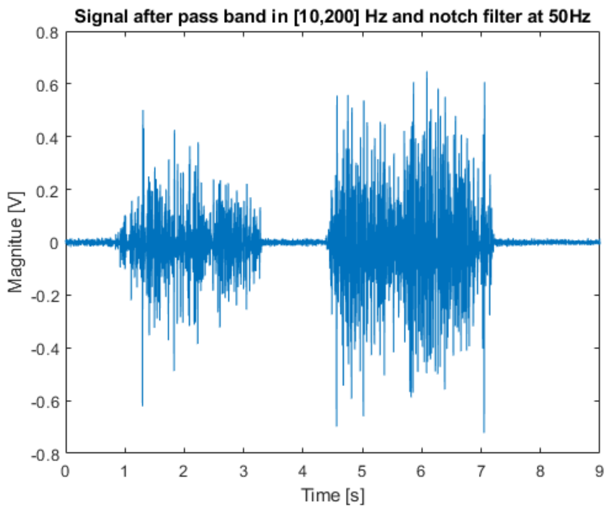

Figure III.4 Filtered EMG signal in time domain. A band pass filter [10 – 200] Hz and a notch filter at 50 [Hz] are applied.

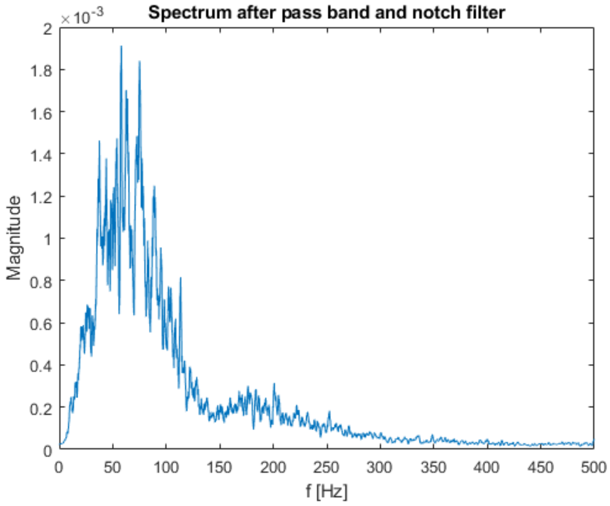

Figure III.5 Filtered EMG signal in Fourier domain. A band pass filter [10 – 200] Hz and a notch filter at 50 [Hz] are applied.

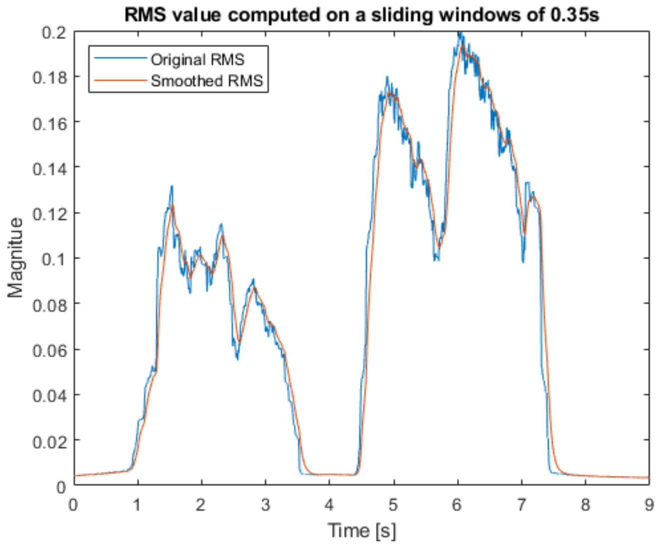

Figure III.6 : Final RMS value (orange). The first weak and second strong contraction are easily identifiable. In Blue the RMS signal and in orange the RMS signal smoothed with a low pass filter and a cutoff frequency equal to 3[Hz].

IV. Conclusion

The EMG envelop computations was successfully implemented, as well as a position controller in position. Multiple insight on digital filtering, control and efficient embed computations were gained.

The results are impressive, and the control of the paddle is quite intuitive. A future improvement would be the implementation of multiple electrodes in order to have more degrees of freedom and control more complex robots.

References

[1] Emg wikipedia page, https://en.wikipedia.org/wiki/Electromyography

[2] "Surface Electromyography: Detection and Recording", C. De Luca, DelSys Incorporated, 2002