Analysis of the times involved in processing and communication in a lower limb simulation system controlled by SEMG

A. Profumieri, C. Bonelli, P. Catalfamo, A. Chernizi
1Laboratorio de Ingeniería en Rehabilitación e Investigaciones Neuromusculares y Sensoriales (LIRINS). Facultad de Ingeniería, Universidad Nacional de Entre Ríos, http://www.bioingenieria.edu.ar/grupos/lirins/2Centro de Investigaciones y Transferencia de Entre Ríos (CITER). Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) y Universidad Nacional de Entre Ríos (UNER). Oro Verde, Entre Ríos, Argentina.

{acherniz,cbonell}@bioingenieria.edu.ar

Abstract. Virtual reality has been proposed for different applications, including the evaluation of new control strategies and training protocols for upper limb prostheses and for the study of new rehabilitation programs. In this study, a lower limb simulation environment commanded by surface electromyography signals is evaluated. The time delays generated by the acquisition and processing stages for the signals that would command the knee joint, were measured and different acquisition windows were analysed. The subjective perception of the quality of simulation was also evaluated when extra delays were added to the process. The results showed that the acquisition window is responsible for the longest delay. Also, the basic implemented processes allowed for the acquisition of three signal channels for commanding the simulation. Finally, the communication between different applications is arguably efficient, although it depends on the amount of data to be sent.

1. Introduction

Virtual reality has been defined as the man-machine interface technology that allows real time simulation of environments, activities or situations, where the subject interacts that allows for user interaction via multiple sensory channels [1]. It was proposed in conjunction with robotic systems, for the rehabilitation of patients with cerebrovascular accidents [2] and for training in patients wearing upper limbs prosthesis [3]. Experimental work showed that interfaces had positives effects in the rehabilitation process of lower limbs and demonstrated its efficacy for recovery of gait and balance [4]. In patients with lower limb amputation, the clinical viability of the prosthesis depends on a complex adaptation process, including the ability of the patient to accept the new corporal image [5]. In these patients, virtual environments could be used to evaluate the prosthesis control, for training and psychological adaptation. Also it might be use as an alternative or complementary tool for gait analysis [6, 7].

Nowadays, it is possible to access the hardware and software technology needed to implement different types of virtual environments, and also provide the user with a real time image projection. In
our previous work, the implementation of a virtual environment that simulates specific movements of
the lower limbs was achieved. Due to its versatility, this development allows the control of the
simulation using different options and could be used in future research [8]. Surface electromyography
(SEMG) could be used to control the simulation. However, the feasibility to command the simulation
developed using this type of electrophysiological signals needs to be analysed. Also, the delay
between the limb movement and the simulation movement must be regarded as imperceptible by users
so that the response could be read as real time [9]. Hence, it was necessary to study the times involved
in the signal acquisition and processing, in order to use the simulation in the investigation of new
rehabilitation strategies.

In this work, the acquisition and processing of SEMG signals and the communication between the
different applications of the proposed system were analysed, and the delays generated by each of
stages were examined. To do that, two standard processes were used to control the simulation:
detection of muscular activity and proportional control of movements. Also alternative solutions are
proposed to optimize the simulation.

2. Materials and methods
In this section, the applications used to record the signal, build the simulation and perform the
processing of the signal to control the model are described. Also, the tests performed to analyse both
the execution times of the different processes and the subjective evaluation of the proposed system are
indicated.

2.1 SEMG signal recording
Three channels of the SEMG signal from the rectus femoral muscle from 4 men volunteers, aged
between 24 and 26 years, without lower limb pathologies, who gave their consent to participate were
recorded. Volunteers were seated, with the thigh in horizontal position and the leg flexed 90º. They
were asked to performed flexion-extension movements of knee. Disposable Ag/AgCl surface
electrodes with conductive gel were used, which were placed on the muscle body midway between the
superior iliac spine and the patella, following the SENIAM (Surface MyoGraphy for the Non-Invasive
Assessment of Muscles) recommendations [10].

The registered signal was amplified and filtered analogically between 0.1 and 300 Hz using the
Grass instrumentation amplifier, model 8-18-36, and was digitized with the A/D converter Data
Translation ECON Series board, model DT9816. Matlab R2013b was used for the acquisition and
processing of signals and the simulation command. This application is used for signal processing
because of its versatility, since it allows to work efficiently with matrices, it has specific tools for
acquisition and processing signals and it enables communication with others applications [11]. The
signal was sampled at 1000 Hz, with an acquisition window of 210 ms. The signal was filtered using a
digital fifth-order high-pass Butterworth filter with cut-off frequency of 10 Hz, to eliminate movement
artefacts, and a fifth-order notch filter at 50 Hz, to remove power line noise [12].

2.2 Lower limb simulation
A basic model of the lower limb was built, with three segments (thigh, leg and foot) and three joints
(hip, knee and ankle), using the 3D simulation software V-REP [13]. The simulation was made of
elementary figures linked through joints that allow rotation in one direction. Joints were configured in
order to be controlled from a remote application in Matlab. Figure 1 shows the aspect of this model.

Also, a simulation using the built in model “Bill” was used, as shown in figure 2, which was also
controlled remotely with Matlab. This model allows moving the right leg joints only in one dimension
and offers a better visual aspect.

Once the signals were acquired and processed with Matlab, the angles of the knee joint were
obtained, and sent to V-REP to command the simulation. In order to send the data, a RemoteApi was
used, using the function vrep.simxSetJointTargetPosition, in mode simx_opmode_oneshot. The
amount of data sent to V-REP depends on the acquisition time and processing actions to obtain the simulation command.

2.3. SEMG signal processing
Two common SEMG signal processing techniques were selected to control the simulation: a threshold-based control and a proportional control.

2.3.1. Initial processing. In order to perform either a threshold or proportional control, it is necessary to normalize and quantify the EMG signal, hence reference values are needed. The first reference value is the RMS of the EMG signal when the muscle is relaxed. To obtain this value, the SEMG signal was recorded at the beginning of the simulation with the muscle at rest (relaxed muscle) during one second, and the RMS value of this signal was computed according to equation (1), where $x_{i}$ is the discrete SEMG signal, and termed as RMS\textsubscript{rest} for future reference.

$$\text{RMS}_{\text{rest}} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} x_{i}^2}.$$ (1)

Then, the volunteers were asked to perform a contraction at maximal intensity during five seconds, only once, and the maximum voluntary contraction (MVC) was registered during one central second. Later, two processes were performed: on one hand, the signal was filtered, the envelope of the signal was calculated and $(x_{f}[i])$ and the maximum value of signal was obtained, called EMG\textsubscript{MVC}. On the other hand, the RMS value named RMS\textsubscript{MVC} was computed using (1).

$$\text{EMG}_{\text{MVC}} = \max(x_{f}[i]), i = 0, ..., N$$ (2)

2.3.2. Normalization and envelope. The envelope of the signal was calculated in order to eliminate fast variations and the normalization (with the reference value of the MVC) was computed in order to make the comparison with the threshold value, and in the future to relate information from different muscle groups [12]. To obtain the signal envelope $(x_{f}[i])$, a fifth-order Butterworth low pass filter with cutoff frequency of 40 Hz was applied. The filter was implemented with the Matlab function filtfilt, after signal rectification. The normalization of the signal was obtained by dividing $x_{f}$ by EMG\textsubscript{MVC}. The $x_{n}$ signal obtained is then used for detection of muscle activity.

2.3.3. Muscle activity detection. This process was used to determine the presence or absence of muscle activity and, a binary signal (one if muscle activity was detected or zero if not) was constructed. The
values of the envelope of the normalized SEMG signal \( x_n[i] \) were compared against a threshold value. Muscle activity was considered present if the signal exceeded the threshold. The threshold value varied between zero and one, and was obtained heuristically in preliminary tests from which it was noted that very small or big values caused an improper behaviour of the simulation either by constant moving or not responding to movements, respectively. It was determined that 0.7 was an appropriate value for the threshold. The binary signal obtained from this processing was then used to command the extension or flexion of the knee lower limb in the V-REP model.

\[
\theta[i] = \begin{cases} 
90, & x_n[i] \geq 0.7 \\
0, & x_n[i] < 0.7
\end{cases}, i = 1, \ldots, n .
\] (3)

2.3.4. Muscle activity quantization. This process is performed to obtain the angle of the leg from the SEMG signal. The RMS value for the acquisition window of the signal \( x[i] \) was obtained, as shown in (1), hereinafter RMS\(_{signal}\), and then subtracting the normalized RMS value of muscular activity at rest and dividing all by the RMS value of maximum voluntary contraction [12].

\[
v = \frac{\text{RMS}_{signal} - \text{RMS}_{rest}}{\text{RMS}_{MCV}}.
\] (4)

For the quantization of data, an interval of 0° to 90° for the knee angle was considered, varying in steps of 10°. Thus, the information \( v \) obtained with the aforementioned process was assigned one of these ten values of angle \( \theta \). These angle values were then used as input to the simulator.

2.4. Command obtained from SEMG

Based on the processes indicated before, two methods were implemented to command the simulation objects. These methods were called simulation A and simulation B and their particular characteristics are described below. Both simulations share an initial stage where the A/D converter board and the communication between V-REP and Matlab are configured.

2.4.1. Simulation A. This simulation was controlled by a binary signal (ones and zeros), converted to angle values of 0° or 90° to indicate whether the simulated limb has a full extension or flexion of the knee. This simulation was started with the configuration stage, and then the MCV values of muscle were obtained. Later, the signal was acquired, digitally filtered and its envelope and normalization was obtained. Finally, the process described in Section 2.3.3 was applied, and the angles \( \theta[i] \) that control the knee joint to V-REP (210 data per channel) were sent. This procedure was repeated from the step of acquisition while the simulation was running.

2.4.2. Simulation B. This procedure aimed at controlling the limb simulation so that it responded proportionally to the RMS value of the SEMG signal recorded. To do this, the simulation started with the configuration of the A/D converter board and applications and the initial stage where the RMS\(_{rest}\) and RMS\(_{MCV}\) were obtained (Section 2.3.1). Then, while simulation was running, the following actions were repeated: acquisition, filtering, normalization and quantization of SEMG signal (according to section 2.3.4), and sending of the value of the angle \( \theta \) (1 data per channel) for controlling the movement of the knee joint.

2.5. Tests

Given the procedures described above, two types of tests, with different objectives were considered.

2.5.1. Test 1: Individual process analysis. The delays generated by each process were calculated in order to determine which of them was regarded as more time consuming. For time measurements a Matlab tool called tic-toc was used. Each process described in section 2.3 was executed 10 times and
the average value of time accomplished was obtained. Then, some processes were selected to build the programs to get the commands to control the simulation (see section 2.4).

2.5.2. Test 2: Qualitative assessment of the simulation. In this stage, a poll was made to assess qualitatively the performance of the simulation, from the opinion of the volunteers about the time response.

Delays of 100, 150 and 200 ms were introduced to the system in order to evaluate which was the waiting limit regarded as acceptable in the simulation response.

The poll has four questions, where the last one asks about the simulation assessment when delays are added (a: 100 ms b: 150 ms and c: 200 ms):
- How do you evaluate the lower limb simulation? (Very Good-Good-Regular-Bad)
- How do you evaluate the time response of simulation A? (Acceptable-Not Acceptable)
- How do you evaluate the time response of simulation B? (Acceptable-Not Acceptable)
- Considering delays a, b and c in response simulation. How do you evaluate each delay? (Acceptable-No Acceptable)

3. Results

3.1. Analysis of Individual processes: results from test 1

Table 1 shows the time needed for configuration, storage, filtering (as described in section 2.1), normalization and envelope detection (as described in section 2.3.2) for each of the 10 trials made. All processes run using 3 acquisition channels.

| Process          | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 | Average |
|------------------|----|----|----|----|----|----|----|----|----|----|---------|
| Configuration    | 70 | 50 | 50 | 60 | 50 | 50 | 60 | 50 | 50 | 60 | 55      |
| Read/write       | 90 | 60 | 32 | 33 | 42 | 32 | 43 | 32 | 35 | 32 | 43.1    |
| Filtering        | 21 | 5  | 5  | 5  | 5  | 5  | 6  | 3  | 6  | 3  | 6.4     |
| Hilbert          | 13 | 7  | 6  | 5  | 7  | 7  | 6  | 7  | 7  | 7  | 7.2     |
| Low-pass         | 16 | 12 | 12 | 10 | 11 | 10 | 11 | 11 | 11 | 11 | 11.5    |

Table 2 shows the time demanded by the processes used for the detection of muscle activity (section 2.3.3) and quantization (section 2.3.4). In Matlab, the threshold method used for the detection of muscle activity, equation (3), can be carried out in two ways: using iterative structures (for or while) or with vector operations. The latest makes use of the advantages of working with vectors in Matlab, using relational operators to compare each signal element with the threshold. For the present paper, both implementations were used and their performances were compared. The latter method is shown in Table 2 with the name Op.relat.Vect.

Also, Table 2 shows the time needed to send the angle values obtained from the Matlab processing to the V-REP simulation. The first row shows the time required to send all the data for three channels and in the second row, the time demanded to send just 3 data to V-REP.
Table 2. Processing time demanded by muscle activity detection, quantization and data send from Matlab to VREP.

| Process                      | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | Average | Unit |
|------------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|------|
| Iteration for                | 590 | 958 | 602 | 732 | 682 | 704 | 699 | 711 | 777 | 711 | 716,6   | ms   |
| Iteration while              | 768 | 851 | 827 | 800 | 867 | 875 | 732 | 869 | 933 | 775 | 829,7   | ms   |
| Op. relat. vect.             | 50  | 10  | 11  | 19  | 18  | 10  | 9   | 18  | 15  | 9   | 16,9    | µs   |
| Quantization                 | 703 | 717 | 913 | 936 | 783 | 951 | 943 | 688 | 677 | 722 | 808,3   | µs   |
| Sending 630 data             | 108 | 74  | 75  | 79  | 81  | 78  | 84  | 75  | 68  | 77  | 79,9    | ms   |
| Sending 3 data               | 401 | 402 | 403 | 405 | 403 | 405 | 405 | 402 | 405 | 403 | 403,9   | µs   |

3.2. Qualitative assessment of the simulation: results from test 2
The results from the enquiry made to volunteers showed that 75% considered the simulation as “good” meanwhile 25% consider it as “very good”.

Seventy five percent of volunteers considered the response of simulation A as “acceptable” and 25% consider it as “not acceptable”. However, for simulation B, 100% of volunteers consider the time response as “acceptable”.

Figure 6 shows the opinions about time response with added delays of 100, 150 and 200 ms.

4. Discussion
In this section tests results are discussed. All processes were performed for three channels. However, as the work objective was the study of processing time, not the information contained in the channels, only the flexion-extension of the knee was simulated.

4.1. Analysis of the Processes
The objective of this test was to analyze the time delays generated by the different stages corresponding to signal acquisition, data processing and communication between applications, in order to evaluate if it is possible to command a VREP simulation in real time using SEMG signals. Some processes cannot be avoided, as the configuration of acquisition board, filtering and data sending to V-REP. Others, as normalization, envelope detection and muscle activity detection, depend on the application and have different ways to be implemented. The proposed simulation is useful to study others algorithms for muscle activity detection, which are now under investigation [14]. Also, some processes can be eliminated, as the write/read of data in a file.

The analysis performed in this work aimed at evaluating the impact of different system stages on simulation. Different authors reported that the response time of the system should not be more than 300 ms approximately, in order to be considered imperceptible [9].
From Table 1 it can be seen that the acquisition board configuration and processes related to communication between the V-REP and Matlab applications, and request of handlers (variables that allow communication between applications) had adequate runtimes, especially considering that they should be run only once at the start of simulation. Furthermore, the writing and reading of data in a file consumes considerable time. This is a practice that takes place to keep recorded data permanently, but, if possible, it should be avoided. In this work, the data are temporarily stored in memory, directly as an array of Matlab. The time consumed, evaluated in preliminary tests, are in the order of µs.

Filtering was performed using a Butterworth filter due to its maximally flat magnitude response. Unfortunately, its phase response is not null, but it can be cancelled applying the filter in two opposite directions. Also, a notch filter in 50 Hz was used to eliminate line noise. The processing time demanded by the filtering was low and this stage can be eliminated if the acquisition system includes the filters.

The time demanded by the envelope detection stage using a low-pass filter is acceptable. In previous tests the envelope using the Hilbert transformation was obtained. This process was more efficient (demanded 7.2 ms), but the output signal characteristics had a worse quality than the output signal from the low pass filter.

Detection of muscle activity using a threshold and based on iterative cycles for or while consumed considerable times (well above the 300 ms proposed in [9]), limiting its application in real time systems. However, the method based on relational operations using vectors shows very good results, as shown in Table 2, so this method was chosen to perform this task. From this experience, it was concluded that all processes involving long cycles in Matlab slow down the execution of the program and makes it very difficult to work in real time. The use of arrangements, however, provides good results and is one of the main advantages that Matlab provides for data processing.

The RMS value calculation, normalization and quantization demanded very low processing time and also gave an acceptable simulation response. For this work, the angular excursion was divided in 10 levels. If the number of levels increases, the simulation behaviour becomes unstable.

The time needed to send the data to VREP (Table 2) is the longest. However, for a window of 210 ms it was possible to send the total data within the schedule time. When the amount of data is small, the time to send the data is in the order of µs. But when the amount of data grows, the time needed to send the data is longer, so special care should be given to this stage. A possible solution to this problem is to make a sub-sampling of data to be sent. In addition, it is recommend to use the simx_opmode_oneshot V-REP mode (as used in this work) instead of the simx_opmode_oneshot_wait mode, which is slower given that it sends the command and waits for a response, before placing a new communication.

It can be noted from the tables that the first trials seemed to require longer times than the following. A possible explanation to this is that the first time the program is run, the variables are created and memory spaces are reserved for them, which is an extra step that slows the process. In the second execution, variables are already created in memory and only access for reading or writing is performed.

It is worthwhile to note here that all processes could be implemented within the recommended ranges. Also, all processes were carried out on three channels which is important because often it is necessary to have information of more than one muscle to perform a more physiological simulation control. Despite of having three channels of data sent to V-REP, only the command of the knee joint was accomplished. For future work, models with control for other joints or knee control using different muscles information are proposed.

4.2. Qualitative assessment of the simulation: test 2 results
Volunteers in general evaluated the simulation as acceptable. Simulation A with an ON-OFF response had an acceptance of 75% while the simulation B had 100% of acceptance. The reason of this, could be that simulation B had a shorter delay (and allowed changing the leg position in function of SEMG
The qualitative assessment of the simulation was made using the lower limb model built with elementary figures, so that volunteers focused on simulation delays, rather than the aesthetic part of the simulation. It is important to mention that the response time did not change when the Bil model was used.

The delays used for the assessment were greater than the recommended by bibliography, because some algorithms intended to be used for commanding the simulation may require higher processing times than the conventional processes. Hence, delays of 100, 150 and 200 ms were added. Figure 6 shows that the simulation acceptance decreases when delays increase. However, these results are not conclusive because the sample of volunteers is very small.

Finally, the volunteers had some problems with the simulation control. Some of the possible reasons are: MVC measurement errors, the ability of the volunteers or the lack of practice. These problems are a system weakness that shows the strong necessity of user training in devices commanded by SEMG (simulation or prosthesis). This system offers a low cost possibility to train and in case of prosthesis adaptation, it could be used even without the physical device.

5. Conclusions
It was possible to control the knee flexion and extension in a simulation, created using VREP, and to command the simulation in real time by the SEMG signal obtained from the rectus femoral.

The evaluation of the processes implemented showed that the critical processes were window acquisition size and communication between Matlab and VREP (although this depended on the type of data to send).

Volunteer’s tests indicated that simulation acceptance is good and when simulation delays are increased, acceptance decreased. This should be taken into account to propose new signal processes that require high computation cost or more channels to process.

References
[1] Adamovich SV, Fluet G G, Tunik E and Merians AS 2009 Sensorimotor training in virtual reality: a review NeuroRehabilitation 25 29
[2] Zhang X, Zhang J, Sun L and Li M 2013 Rehabilitative motion planning for upper limb rehabilitation robot based on virtual reality IEEE Int. Conf. on Robotics and Biomimetics 746
[3] Dawson MR, Carey JP and Fahimi F 2011 Myoelectric training systems Expert Rev. Med. Devices 8 581
[4] Wagner J, Solis-Escalante T, Scherer R, Neuper C and Müller-Putz G 2014 It’s how you get there: walking down a virtual alley activates premotor and parietal areas Front. Hum. Neurosci. 8 93
[5] Mayer A, Kudar K, Bretz K, Tihanyi J and Mayer Á 2008 Body schema and body awareness of amputees Prosthet. Orthot. Int. 32 363
[6] Ferrari A, Cutti AG, Garofalo P, Raggi M, Heijboer M, Cappello A and Davalli A 2010 First in vivo assessment of Outwalk: a novel protocol for clinical gait analysis based on inertial and magnetic sensors Med. Biol. Eng. Comput. 48 1-15
[7] Lambrecht J, Pulliam C and Kirsch R 2011 Virtual reality environment for simulating tasks with a myoelectric prosthesis: an assessment and training tool J. Prosthet. Orthot. 23 89
[8] Cardona N, Catalfamo P, Cherniz A and Bonell C 2015 Virtual Platform Technologies for Evaluation of Rehabilitation Progress and for Prosthetic Control VI Latin Am. Cong. on Biom. Eng. 984
[9] Hermens H, Freriks B, Desselhorst-Klug C and Rau G 2000 Development of recommendations for SEMG sensors and sensor placement procedures J. Electromyogr. Kines. 10 361
[10] MATLAB version 7.10.0 R2010a The MathWorks, Inc.
[11] DeLuca G 2003 Fundamental Concepts in EMG Signal Acquisition Copyright DelSys Inc.
[12] V-REP Virtual Robot Experimentation Platform *Coppelia Robotics*

[13] Zecca M, Micera S, Carrozza M and Dario P 2002 Control of Multifunctional Prosthetic Hands by Processing the Electromyographic Signal *Critical Reviews Biom. Eng.* **30** 459

[14] Bonell C, Tabernig C y Spaich E 2015 Evaluation of a double threshold algorithm to detect electromyographic activity in the healthy and paretic Tibialis Anterior muscle. *VI Latin Am. Cong. On Biomedical Engineering* (Springer International Publishing) p 940