Memorias del XVI Congreso Latinoamericano
de Control Automático, CLCA 2014
Octubre 14-17, 2014. Cancún, Quintana Roo, México

Model Predictive Control for a robotic arm in the aiding of rehabilitation of lateral epicondylitis

A. Arrieta*, A. I. González**

**Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Ciudad Juárez, Ciudad Juárez, México (e-mail: alan.arrieta@item.mx).*
***Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba (Guest professor ITESM Jan-May 2014 Ciudad Juárez, México) (e-mail: anita@electrica.cujae.edu.cu).*

Abstract: The work presents a Model Predictive Controller (MPC) for a robotic exoskeleton arm determined to aiding in the process of physical rehabilitation of people with lateral epicondylitis. A short introduction to the modern control systems, going through the explanation of the proposed system and finally the design process of the controller is presented. This work aims to give the reader an example of the Model Predictive Control applications in the biomedical field. The tuning of controller goes through simulation via the MPC tool included in the MATLAB® software.

Keywords: Control process, predictive control, biomedical devices, simulation, Tennis elbow rehab.

1. INTRODUCTION

One of the actual trends on industrial process, is the development of robust and flexible control systems, which sometimes are not achieved through traditional controllers like the Proportional-Integral-Derivative (PID), and this is where modern control or intelligent control techniques are applied, techniques such as Artificial Neural Networks (ANN), Fuzzy Logic controllers (FLC), etc. Another modern control technique is the Model Predictive Control (MPC), which is the one addressed in this work (Camacho and Bordons, 1999), (Qin and Badgwell, 1999), (Agachi et al, 2006).

As the technology evolves through time, technological applications in different fields are achieved, and the medical field is not the exception. Many full-tech systems such as robot surgeons, mechanical ventilators, intelligent prosthesis and physical rehabilitation systems are used day by day to enhance the living quality of the people. The last of the previous mentioned systems, the physical rehabilitation systems, have a tremendous opportunity area regarding control systems. This is because many of the traditional systems can be automated, boosting the rehabilitation process of those with the suffering.

One of the most common diseases presented in the upper limbs, explicitly affecting the elbow, is the lateral epicondylitis, also known as the *Tennis elbow*, which is progressively generated as one person lifts heavy loads constantly over large periods of time, causing a tear in a tendon and letting the affected person without the capabilities of making any effort without feeling pain.

Most of the times, the treatment for this affections, consists on complete rest, as well as anti-inflammatory medicine, however in critical cases, surgery is needed. With the incapability of making any effort to lift up things, people is limited in their daily activities.

Fig. 1. Graphical explanation of the Lateral Epicondylitis.

2. Model Predictive Control Strategy

Basically, the MPC control strategy, consists on an optimizer and a very accurate knowledge model of the system to be controlled. It is important to highlight that this control strategy involves computerized processing of the model variables of the system in question. In other words the predictive control is an advanced control strategy that involves an optimization problem, where optimal control signals (U) are to be calculated (Bordons, 2000).

In the traditional closed loop control systems, the controller signal is determined based on the error signal (E), which is the difference between the set point reference (R) to be reached and the actual output of the system (*Y*), as it is shown in figure 2.

On the other hand, the MPC, involves a closer relation with the process, this is, because a model of the same system must be integrated as part of the controller. The strategy of the MPC consists on determining the optimal future control actions (*U (t | t)*) within a horizon of time called prediction horizon (Np), this every instant of time *t* (Camacho and Bordons, 1999).

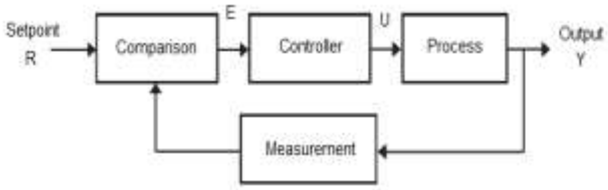

Fig. 2. Block diagram of a traditional closed loop control system.

To obtain the optimal future control actions, previous inputs and outputs of the system are fed into the process model, so predictive outputs (*y (t | t)*) are determined and compared to a reference trajectory (*w (t | t)*), taking in count the physical and operational constraints and the importance of each variable.

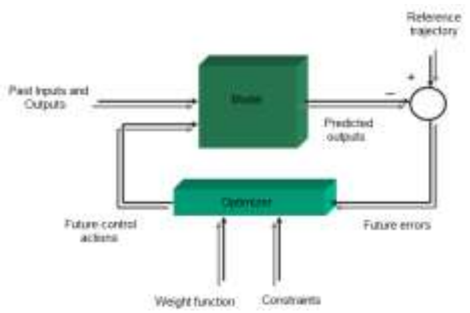

Fig. 3. MPC block diagram.

The MPC controller, being an optimization solver as mentioned before, chooses the optimal future control actions from all the options available at each instant of time; after sending the first optimal control signal to the process, everything starts over and new optimal future control signals are needed to be calculated.

One of the most considerable challenges of the MPC consists on the timing, this is because the optimal control actions must be computed based on the predicted outputs, and sent to the process before the sampling time cycles again, another critical challenge is to achieve an accurate model of the system, this could be taken as the own spine of the MPC. If a bad model is integrated in the controller, bad predicted outputs will be generated and with that bad future control actions. This last point contrasts the designing process of a conventional controller such as the PID, where most of the effort was focused on tuning the controller and setting the right parameters, however in the MPC, most of the effort is coming from determining the most accurate model of the system to be controlled.

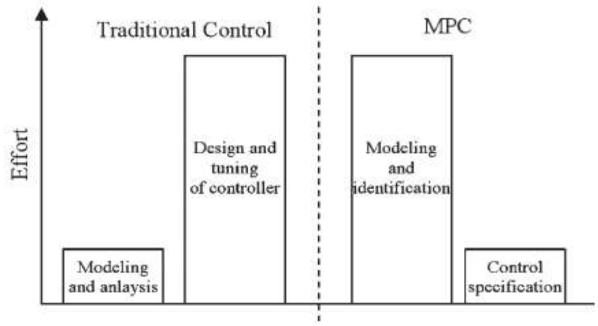

Fig. 4. Comparison between the effort in the traditional control and the MPC *(Camacho and Bordons, 1999)*.

The few needed parameters to tune the MPC consists mainly on horizons of time, the first parameters was already previously mentioned, it is *N*, however, it can be expressed as

$$Np = N_2 - N_1 \tag{1}$$

Where N1 and N2 are the minimum and maximum prediction horizons.

Another critical tuning variable for the controller is the Control Horizon (*Nu*), which consists on the number of movements available for the control action. The smaller the value of the parameter is better.

The target function to be accomplished by the optimizer can be expressed as (2) where δ and λ correspond to a weight function.

$$J(N1, N2, Nu) = E\left\{ \sum_{j=N1}^{N2} \delta(j)[y(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{Nu} \lambda(j)[\Delta u(t+j-1)]^2 + \right\} \tag{2}$$

3. PROPOSED SYSTEM

The exoskeleton system proposed to accomplish as main objective the compensation of the effort that, a person with tennis elbow can't do, will consist on 2 mechanical links with 2 rotatory joints as kinematic chain, each joint granting a degree of freedom (DOF) via electric actuators, using in this case electric geared DC motors. To grant connection to the person, an interface based on the electrical activity of the arm muscles, or electromyography (EMG) will be developed through superficial electrodes and a microcontroller; the EMG signals will be processed to act as references to the system and all the computations will be held by the microcontroller, that by this point an Arduino Uno board is proposed due its friendly programming interface.

One motor will be assembled recreating the rotation of the shoulder, and a worm-spindle mechanism will do the same for the elbow rotation.

The physical constraints impacting the system consists on the parameters shown in the table 1.

Fig. 5. CAD representation of the proposed system.

Parameter	Magnitude	Units
Nominal motors voltage	12	VDC
Maximum input voltage	13	VDC
µcontroller sampling time	.001	Seconds
µcontroller input resolution	256	bits
Voltage Up/Down rate	0.0464	VDC/period

Table 1. Physical constraints of the proposed system.

The resulting system consists on a MISO configuration, having as inputs the excitation voltages of the two motors (*Ei1* and *Ei2*) and as the output the total torque arround the shoulder. It is important to highlight that the model includes all the loads corresponding to the weight of the mechanical links, actuators, average of the human arm and a fixed load of 2 kg.

4. MPC DESIGN PROCESS

The design process followed, consisted on different scenarios and simulations of controllers, where different parameters were applied, until granting a good level of controllability. All the simulations were made through the MPC Tool of MATLAB® (Mathworks, 2014) and included an initial value of the total torque equivalent to -43.37N.m, corresponding to the loads incorporated in the system.

4.1 Process model

As commented previously, to achieve a good MPC, it is needed an accurate knowledge model of the system to be controlled, for the proposed system a model was obtained in a previous work, represented as transfer functions in blocks (Arrieta and Gonzalez, 2014).

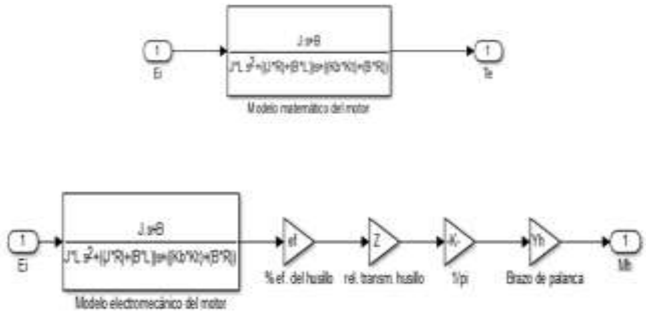

Fig. 6. Block diagrams of the mathematical model corresponding to the proposed system implemented on SIMULINK.

For the process modelling, the laws that rule the mechanical an electrical involved systems were the basis, as well as the mathematical tool of the Laplace transform and the fabricant technical information of each components.

The obtained model was traduced to a MATLAB® script, to later work on it via the MPCTool.

```
                  %Modelo de brazo robótico
E1=12;
E2=12;
J=0.1;
B=0.003;
L=2;
R=3.5;                                   %Parámetros Físicos IDEALES
Kb=0.00388;                              % de los motores eléctricos
Kt=19.63;                              % a utilizar en el sistema propuesto
rel=30;
ef=0.95;
Z=1;
k=0.3083098862;
Yh=0.10;

m1=tf([J B],[J*L ((J*R)+(B*L)) ((Kb*Kt)+(B*R))]) %Modelo del motor 1
m2=tf([J B],[J*L ((J*R)+(B*L)) ((Kb*Kt)+(B*R))]) %Modelo del motor 2

c1=E1*m1;            %Motor 1 con alimentación
c2=E2*ef*Z*k*Yh*m2;  %Motor 2 con mecanismo del husillo y alimentación

T1=rel*c1;      %Torque del motor 1 incluyendo relación de transmisión
T2=rel*c2;      %Torque del motor 2 incluyendo relación de transmisión

Tt=[T1 T2];     %Vector de las TF de los motores

brazodiscreto=c2d(Tt,0.001);   %Modelo discretizado tomando en cuenta
                               %Ts del uc a utilizar
brazodiscreto.InputName={'Ei1','Ei2'};   %Declaración de las entradas
brazodiscreto.OutputName={'Torque_total'}; % Declaración de la salida
brazodiscreto
```

Fig. 7. MATLAB script of the mathematical model for the proposed system.

4.2 *Scenario 1: simulation without physical constraints*

For a first simulation, the model was loaded including only the minimum and maximum input voltages, but leaving in default values *Np*=10 and *Nu*=2; the reference trajectory consisted on a step of amplitude 10 N.m applied at *t*=2 seconds.

The results of this tuning scenario are showed in figure 8.

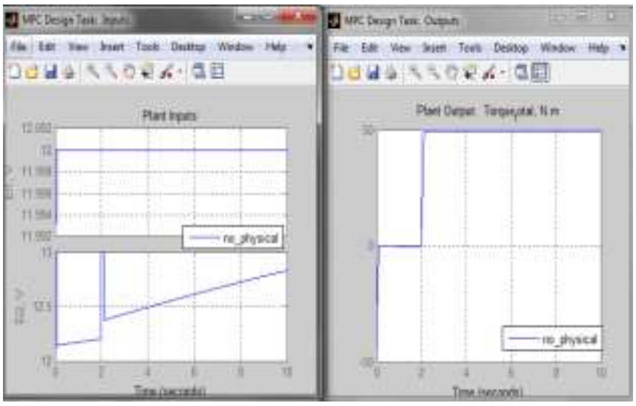

Fig. 8. Simulation of the first scenario: no physical constraints.

4.3 *Scenario 2: physical constraints applied*

On the second simulation, the rest of the physical constraints shown in table 2 were applied to the model; again *Np, Nu* and the simulation time were the default values.

The results of this tuning scenario are showed in figure 9.

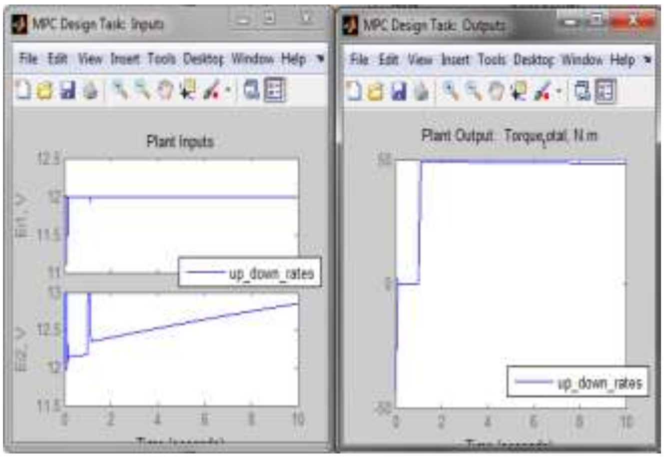

Fig. 9. Simulation of the second scenario: physical constraints applied.

4.4 *Scenario 3: Following another reference trajectory*

In this case, another reference trajectory was applied. A pulse with a period of 3 seconds and an amplitude of 50N.m was applied at *t*=1 second. The main purpose of changing the reference, was to give to the system a more logical trajectory to follow.

The results of this tuning scenario are showed in figure 10.

4.5 *Scenario 4*: *Changing the Prediction Horizon*

Np, as general tuning rule, must be refered to the steady state of the system, taking this on count, *Np* was changed to a value of 2 intervals of time.

The results of this tuning scenario are showed in figure 11.

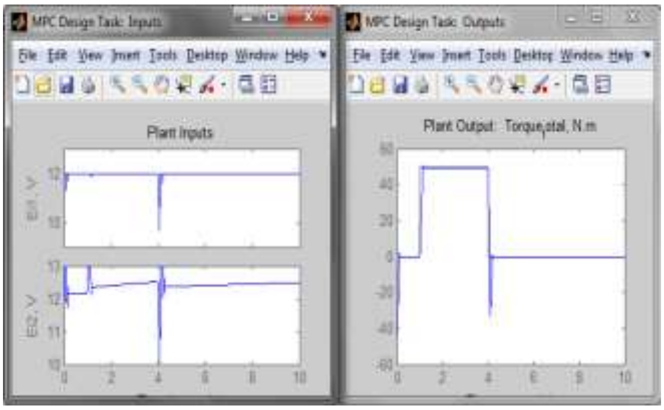

Fig. 10. Simulation of the third scenario: change in the reference trajectory.

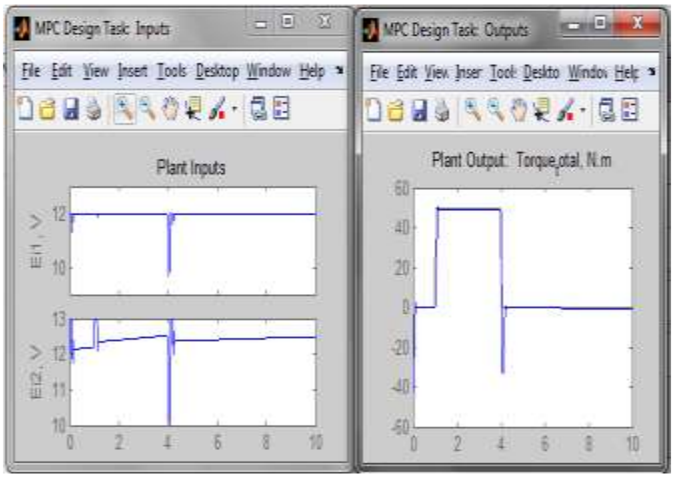

Fig. 11. Simulation of the fourth scenario: change in the Np.

4.6 *Scenario 5: Changing the Control Horizon*

Another parameter to change in the tuning process is the *Nu*, the general rule for this parameter is making it as small as possible. Before this point, *Nu=2*, so in this simulations Nu was changed to 1.

The results of this tuning scenario are showed in figure 12.

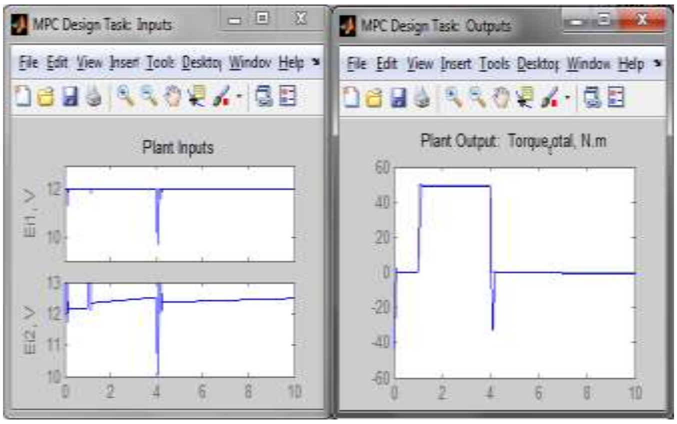

Fig. 12. Simulation of the fifth scenario: change in the *Nu*.

4.7 *Scenario 6: Weights and control schemes modification*

The weights presented in equation (2), were modified. Before this scenario, both input variables had the same weight of 1, but now they were changed to the values shown in figure 13. The values were obtained after the use of the Tuning advisor for MPCTool of MATLAB® and choosing the performance scheme to ITAE.

Another important thing in this scenario, was that the reference trajectory was also modified, eliminating the possibility of receiving a negative total torque, in other words the total torque around the elbow should be 0 N.m as minimum.

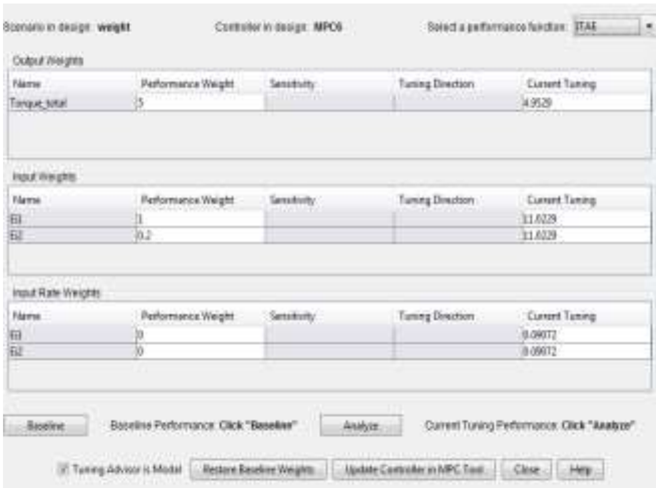

Fig. 13. Tuning advisor for MPC of MATLAB.

5. RESULTS

After making the modifications presented previously, a MPC for the proposed system was developed, having the simulation shown in figure 14. Making an analysis about the simulation, it can be seen that deep valleys are present when the pulse falls down, this could be a complication for the system, meaning that the arm of the person will fall for a very short period of time, this noise on the control signal can be filtered, however the main goal was to develop the MPC. In the inputs graphs it can be seen that the motor 1 works saturated at 12 VDC, meanwhile the motor 2 has a wider operation regime. In figures 10 to 12, overshoots can be seen in the response graphics, this due to the voltage fall and rise in the actuator, generated because of the response time of themselves.

6. CONCLUSIONS

After viewing the results and analysing the purpose of the proposed system, it can be concluded that the MPC is a suitable solution as controller to the rehabilitation system, this due to the mounting obtained in the simulations. The reader must take in count that the system must be inclined to the predictive control because it must anticipate the movement that the person wants to make before it actually does, and that this is one option and it can be modified to achieve better results. It also can be concluded that the system model used its quite accurate, because the controller could track down the reference trajectory with so little error on it.

This work leaves the reader an alternate solution to the control schemes for the fast-growing biomedical devices developed for different meanings including the one studied here: physical rehabilitation.

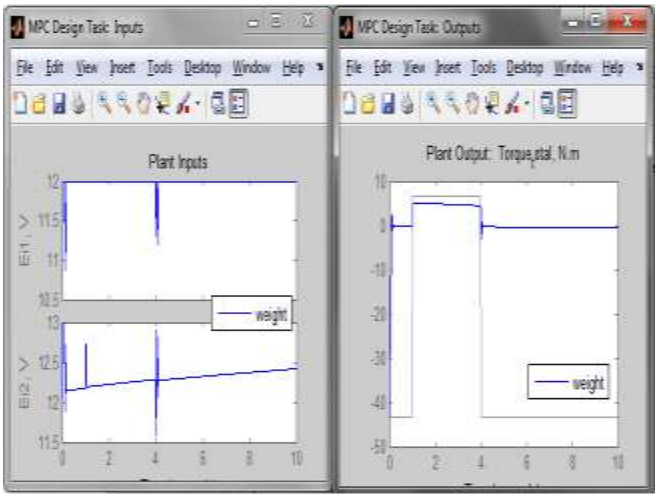

Fig. 14. Simulation of the sixth scenario: change in weights.

REFERENCES

Agachi, P., Nagy, Z., Cistrea, M., Imre-Lucaci A. (2006). *Model Based Control*. Wiley-Vch, Germany.

Arrieta A., A. I. González (2014). *Robotic arm modelling for aiding in elbow rehabilitation*. IEEE Conference in Mechatronics. Not published yet.

Camacho, E., Bordons, C. (1999). *Model Predictive Control, with 90 figures*. Springer, Glasgow U.K.

Bordons, C. (2000). Control Predictivo: Pasado, presente y futuro. Jornadas de Almería. CEA-IFAC, España.

Lehnert, C., Wyeth, G. (2013). *Locally weighted learning model predictive control for nonlinear and time varying dynamics*. IEEE 2013 ICRA, Karlsruhe, Germany.

Mathworks Inc (2014). MPCTool. MATLAB software.

Menezes, V., Fetter, W., (2007). *Comparison between two actuation schemes for underactuated brachiation*. IEEE, U.S.

Qin, S., Badgwell, T. (1999). *An overview of industrial model predictive control technology*. Springer, U.K.