PARAMETER OPTIMIZATION FOR NEURAL OSCILLATORS APPLIED TO TRAJECTORY GENERATION OF AN EXOSKELETON FOR LOWER LIMBS

Marciel Alberto Gomes, gmarciel@sc.usp.br
Adriano Almeida Gonçalves Siqueira, siqueira@sc.usp.br
Rafael Gomes Gobbo, rafobbo@gmail.com
University of São Paulo at São Carlos, Mechanical Engineering Department, Mechatronics Group - Rehabilitation Robotics Lab, Av. Trabalhador São-carlense, 400 - São Carlos, SP, Brazil

Abstract. *This work presents an optimization system developed to find optimal parameters for neural oscillators used for trajectory generation of an exoskeleton for lower limbs. An exoskeleton can be considered as a biped robot, with a given force interaction between the user and the robot. Since biped robots must present cyclical joint trajectories during the walking, neural oscillators are being used as trajectory generators for these systems. The neural oscillators used in this work are based on Matsuoka oscillators, consisting of two mutual inhibitory neurons which are able to produce a cyclical output. Each neuron is modeled by two differential equations whose parameters are difficult to be set for a given desired output. In this way, it was developed an optimization system to find the better values to the neural oscillator parameters given a predefined desired joint trajectory. This optimization system works to minimizing the error between the trajectory generated by the oscillator and the desired trajectory, regarding the robot dynamics. The advantage of using oscillators is justified because the trajectories can be generated in a on-line process, with a less time-consuming with relation to analytical methods. The results show that the proposed optimization system and the trajectory generator using neural oscillators can be applied in an adaptive model that include interaction forces between the user and the robot, so changing the trajectory according to the user intention.*

Keywords: *Exoskeleton, stable gait-pattern, trajectory adaptation, neural oscillator, optimization.*

1. INTRODUCTION

The field of robotic rehabilitation has received an increasing interest over the last years, and the development of devices to assist disabled people, like exoskeletons, is one of the main focus of these researches, (Pons, 2008; Gomes *et al.*, 2009). Amongst several topics, biped locomotion has been a big motivation for researchers working with recovery of human walking. In general, when the dynamic equations of the robot are well defined, they are used to generate joint trajectories, as performed in Huang *et al.* (2001), where the dynamic model and a spline interpolation method are used to find the joint trajectories with stability guaranteed by the Zero Moment Point criterion. However, the use of dynamic equations requests a high computational cost due to nonlinearities and parameter uncertainness. Besides, the resulting solutions are generally not suitable for implementation in a real time process, due to hardware limitation and computer restrictions. Then, the trajectory generator algorithm should be sufficiently fast to provide the desired trajectory and realize required adaptations, when necessary.

Based on that limitations, in the last years, alternative methods were proposed to deal with that problems. Amongst these techniques, studies on motor pattern generators using neural oscillators, called Central Pattern Generators (CPG), are presented in several papers, as in Grillner (1985) and Golubitsky *et al.* (1999). CPGs are biological neural networks that generate basic rhythmic movements, and have been applied in locomotion like walking, swimming and flying. In an artificial intelligent framework, CPGs are composed of nonlinear neural oscillators, which are differential equations able to produce coordinated patterns of rhythmic activities with no rhythmic input patterns. In Matsuoka (1985, 1987) it is mathematically modeled an shooting rate of two mutual inhibitory oscillators of a neural oscillator. This oscillator is widely used to simulate bio-inspired and rhythmic movements for robots, mainly because the basic structure of the oscillator is simple in contrast to other oscillator models, for example, the *van der Pol* oscillator, which has quadratic state nonlinearities, (van der Pol, 1927).

Righetti and Ijspeert (2006); Liu *et al.* (2008); Yang *et al.* (2006, 2007) have based their works in Matsuoka oscillators to generate and adapt desired joint trajectories for bipedal robots. Matsuoka oscillators have several tunable parameters, and its values should be well chosen to guarantee a desired oscillation in the output signal. In general, these parameters are attained just after several testing experiments. Nevertheless, few studies have described a suitable methodology to find optimal parameters to neural oscillators. Baydin (2008) has developed a optimization system where the internal connection structure and the feedback pathways from the environment were subject to a genetic algorithm optimization. This system find optimal parameters for a network of neural oscillators applied to a five-link planar bipedal robot. Arsenio (2004) has presented studies to infer stability and convergence properties of the Matsuoka oscillator. A tuning method to find Matsuoka oscillator parameters is shown in Hattori *et al.* (2010), considering a neural oscillator with a ladder-like structure. The results are applied to generate trajectories for a nematode *Caenorhabditis Elegans*, but not deal with human movements. Most of these techniques are applied in specified cases, and it is very difficult reproducing for others examples.

Then, based on previous works and motivated by the need to new and simple method for parameterize Matsuoka neural oscillators, in this paper it is presented an optimization system based on Levenberg-Marquardt method to find optimal parameters for neural oscillators applied to generate joint trajectories of an exoskeleton for lower limbs. A network can be arranged by seven neural oscillators, regarding the seven degrees of freedom of the exoskeleton. Each neural oscillator has 10 parameters to be found. The optimization system works by minimizing the error between the trajectory generated by the oscillator and a predefined desired trajectory found using the robot dynamics and the ZMP stability criterion.

This paper is divided as follows: Section 2. presents the general aspects of a Matsuoka neural oscillator; Section 3. defines the parameterization problem of the neural oscillator; Section 4. shows the optimization system based on Levenberg-Marquardt method; Section 5. presents simulation results obtained using the optimization system applied to an exoskeleton for lower limbs; Section 6. shows the conclusions.

2. MATSUOKA OSCILLATOR

Due to its simplicity and effectiveness, Matsuoka oscillator has being widely used in much research on biped robots. Basically, the Matsuoka oscillator is a nonlinear oscillator model that generate self-sustained oscillations and it is able to sensory entrainment, (Matsuoka, 1985, 1987). It is composed by two neurons, each one consisting of two first-order coupled differential equations given by:

$$\tau_r \frac{dx_i}{dt} = -x_i + \sum_{j=1}^{n} w_{ij} y_j + s_i - b f_i + feed_i, \tag{1}$$

$$\tau_a \frac{df_i}{dt} = -f_i + y_i, \tag{2}$$

where the first state variable, x_i, is a inner state which corresponds to the membrane potential of the neuron, and the second one, f_i, is the degree of adaptation (or self inhibition) in the $i-th$ neuron. The parameter b represents the adaptation constant related to self-inhibition f_i. The output of the $i-th$ neuron, y_i, is the positive part of the firing rate, e.g., $y_i = \max\{x_i, 0\}$. The time constant τ_r specify the rise time when step input is given, where the frequency of output is roughly proportional to $\frac{1}{\tau_r}$, and τ_a specify the adaptation time lag. The inhibitory synaptic connection weight from the $j-th$ neuron to $i-th$ neuron is denoted by w_{ij}, and $w_{ij} \neq 0$ for $i \neq j$, $w_{ij} = 0$ for $i = j$. $\sum_{j=1}^{n} w_{ij} y_j$ represents the total input from neurons inside a neural network, s_i is constant drive input. It is the interaction between constant and time-varying parameters that causes the self-sustained oscillations.

The parameter $feed_i$ is added to represent the input feedback signal to the neuron and represents the interaction between the robot and the environment, (Liu *et al.*, 2008). Feedback paths provide a way to maintain an adaptive mutual coordination between the oscillator and the walking mechanism subject to the environment. This is attained by modification of oscillation characteristics and phase relations of the CPG network by the external inputs, and in turn, the commands sent by the CPG network driving the walking mechanism.

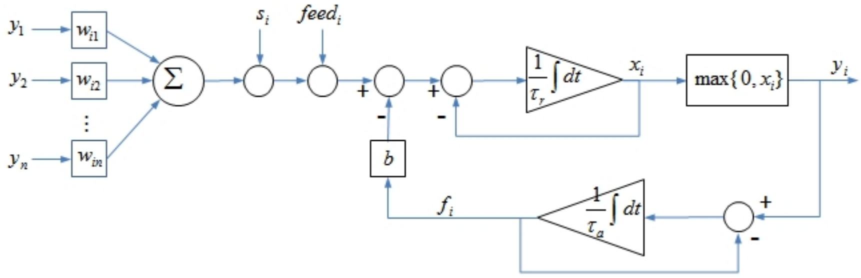


Figure 1. Neuron model (adapted from (Liu *et al.*, 2008)).

Figure 1 shows the block diagram of one neuron, representing the equations 1 and 2. In Figure 2 it is shown the basic architecture of the Matsuoka oscillator, composed by two neurons. It can be noted that the output, y, is the difference between the outputs of the neurons 1 and 2. With this generic architecture and giving appropriate values for each variable, an output signal can be generated containing a transient part and periodic stable permanent part.

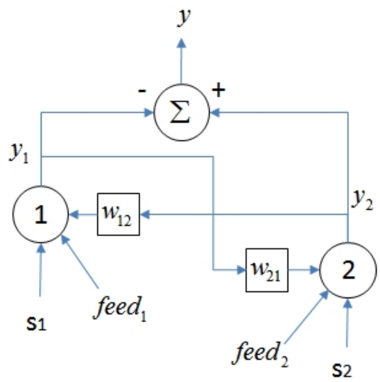
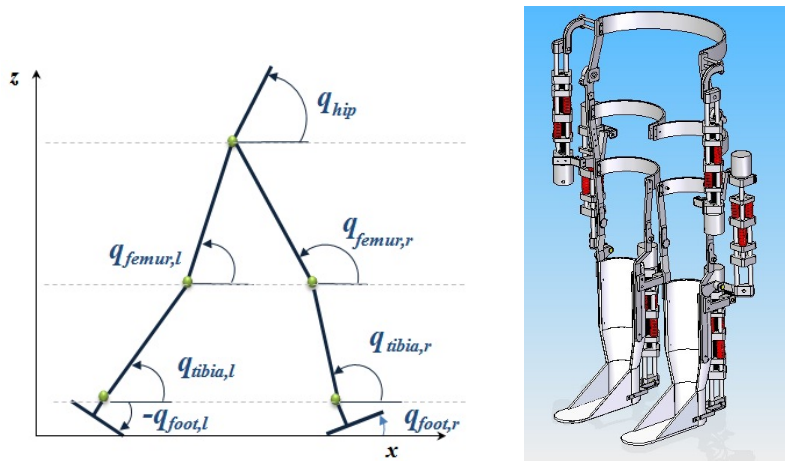


Figure 2. Matsuoka oscillator model.

3. PROBLEM DEFINITION

In this paper, it is considered a trajectory generation for the joints of an exoskeleton for lower limbs shown in Fig. 3(b). This device is being constructed in our laboratory, and is driven by series elastic actuators, (Jardim and Siqueira, 2009). The exoskeleton model used in this work has 7 joints, being 3 for each leg, referring to the foot, tibia and femur movements, plus the hip joint. The model, with the definition of the absolute angles, can be seen in Fig. 3(a), considering the sagittal plane.



(a) Kinematic model (absolute angles).

(b) Exoskeleton being constructed.

Figure 3. Kinematic and 3D model of the exoskeleton

In Gomes *et al.* (2009) it was used a trajectory generator for this exoskeleton which considers the ZMP as stability criterion. The algorithm uses the dynamic and kinematic equations of the model to create the desired trajectory. However, this procedure demands a high computational cost. In this paper, it is being used an approach with neural oscillators to overcome this problem.

The main features of neural oscillators used for joint trajectories generation are the simplicity and robustness of the resulting neural network. This fact gives the system more feasibility to the computational point of view, allowing its application in real time systems. The trajectories can be generated in a short time, and adaptations of the system during its use can occurs with efficiency, in our case, during the walking.

The mathematical condition for the network producing a stable rhythm signal is given by $|w_{12}|/(1+b) < s_1/s_2$, $|w_{21}|/(1+b) < s_2/s_1$ and $\sqrt{|w_{12}w_{21}|} > 1+\tau_r/\tau_a$, (Matsuoka, 1987). Also, it can be seen that the rhythm frequency is positively correlated to the adaptation parameter b, and is negatively correlated to the rise time constant τ_r, the adaptation time constant τ_a, and the synaptic weights of the mutual inhibition, w_{ij}.

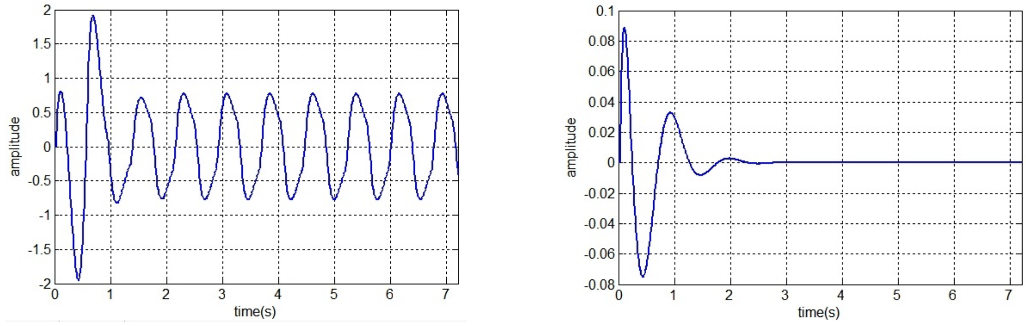
However, these conditions only guarantee the oscillation of the output signal. No conditions are imposed to the oscillator to reproduce a desired output. Figure 4(a) shows the output of a Matsuoka oscillator, considering the values to the parameters presented in Tab. 1. These parameters obey the conditions defined above for an oscillation in the output signal (here it was considered $feed_i = 0$). It can be observed that after a transient signal, the output converges to a cyclical signal. If the conditions are not obey, the output signal can turn unexpected, as shown in Fig. 4(b), and the output can not to attain a cyclical stage.

Table 1. Parameters which obey the restrictions.

τ_r	τ_a	s_1	s_2	w_{12}	w_{21}	b
0.1	5	15	15	1.5	1.5	50

Table 2. Parameters which do not obey the restrictions.

τ_r	τ_a	s_1	s_2	w_{12}	w_{21}	b
0.1	12	5	5	0.5	0.5	50



(a) Signal considering the parameters of the Table 1.

(b) Signal considering the parameters of the Table 2.

Figure 4. Neural oscillator outputs according to the parameters values.

To ensure that an oscillator will be able to reproduce with correctness a biological behavior, it is necessary an optimal parameterization of Eq. (1) and (2), taking into account the desired output signal. In general, to find the better values to these parameters, the empirical method is adopted. In this way, the parameters values are defined after a long search process and after several tests with a set of values. Also, Matsuoka (1987) has found some specific parameter values such that the oscillator is not overly sensitive. So, in general, the ranges of the parameter values that generate stable oscillations is very large, and to find the better values is a hard and time-consuming task.

In the following section it is presented an optimization system based on Levenberg-Marquardt method to find the optimal values to the parameters, taking into account the error between the output oscillator and a desired signal (joint trajectory) given by the robot's dynamic model.

4. PARAMETER OPTIMIZATION SYSTEM

In order to find the better values to the neural oscillator parameters, it was adopted a strategy that minimize the error between the desired trajectory, given by the results presented in Gomes *et al.* (2009), and the neural oscillator output. The strategy can be described as a fitting problem between nonlinear curves through the Least Mean Squares (LMS) method, given by:

$$\min_x ||F(x)||_2^2 = \min_x (f_1(x)^2 + f_2(x)^2 + \ldots + f_n(x)^2), \tag{3}$$

where $x = [x_0, x_1, x_2, ..., x_m]$ are the m parameters which need to be adjusted to optimize the function $f(x)$. The cost function related to the global error between the desired trajectory and the neural oscillator output, evaluates in a specific time interval, $[1; k]$, is given by:

$$\sum_{i=1}^{k} f_i(x)^2 = \sum_{i=1}^{k} (y(i) - y^d(i))^2, \tag{4}$$

where y is the neural oscillator output and y depends to the neural oscillator parameters, and y^d is the desired trajectory. The block diagram in Fig. 5 represents this optimization strategy, being x^* the vector with optimized parameters.

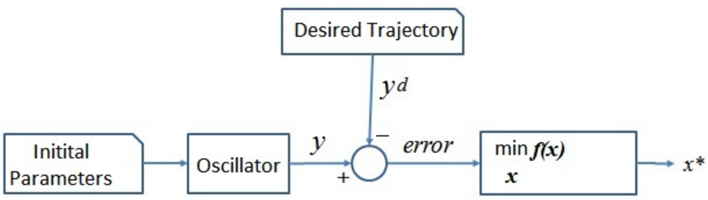


Figure 5. Optimization strategy.

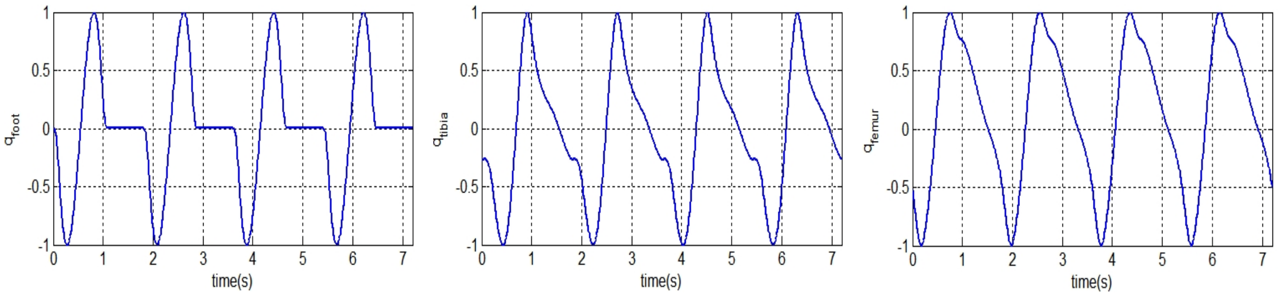
The selected optimization tool is the Levenberg-Marquardt algorithm (Marquardt, 1963), which can be considered an approximation to the Newton method, with the advantage of computing only the Jacobian matrix, instead the Hessian matrix as in that method. This method is simple to be implemented and gives a very powerful tool to realize the proposed task.

Although this problem is quite simple to be solved, in practical applications there are two important issues to be considered: the initial values for the parameters in x, to start the Levenberg-Marquardt method, should be properly selected in order to result in convergence of the optimization process; and determine a reasonable transition point, p, between the transient and the permanent output of the oscillator. The last issue is important since we want to work only with the stable output of the system, and this output should produce periodic and adaptable oscillations according to desired movements of the robot.

In this paper, we have defined the initial parameters to the optimization process by performing random searches to find values that can obey the restrictions presented in Section 3.and give a minor error between the desired trajectory and the output oscillator, i.e., this initial parameters must ensure a rhythmic signal like the desired trajectory. This procedure guarantees the convergence to the Levenberg-Marquardt method. Also, from these conditions, it can be observed that the value of b must be large to evoke the rhythm and this information is useful to give the initial parameters. The point p is defined empirically after observing the transient curve behavior, the phase and frequency of the output trajectory.

5. RESULTS

Considering the aspects presented in previous sections, the optimization system is implemented to reproduce the joint trajectories of the exoskeleton for lower limbs presented in Section 3. Figures 6(a), 6(b) and 6(c) show respectively the foot, tibia and femur trajectories, obtained from (Gomes *et al.*, 2009). The trajectory of the hip was omitted because it was considered constant (in this work, we considered hip angle with 81^o). To standardize the domain, all trajectories are considered in absolute angles and its values are normalized to $I = [-1; 1]$. This is important in the definition the initial parameters of the optimization process, since the output oscillator starts in 0 and it is able to vary the signal to positive and negative values.



(a) Desired trajectory of the foot. (b) Desired trajectory of the tibia. (c) Desired trajectory of the femur.

Figure 6. Reference trajectories for the foot, tibia and femur joints.

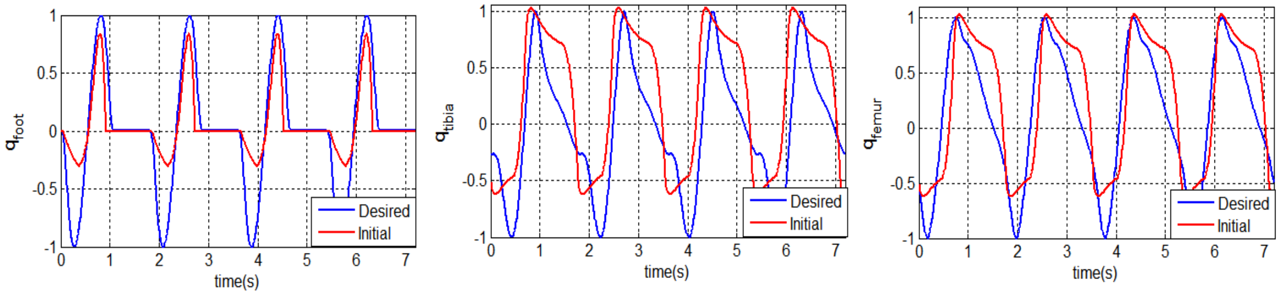
An important issue verified after some testing shows that it is necessary to use different values for the parameters τ_r, τ_a and b for different neurons in the same neural oscillator to reproduce the desired trajectory. Then, during the optimization process, these parameters are divided to each neuron, being $\tau_{r1}, \tau_{a1}, b_1$ for neuron 1, and $\tau_{r2}, \tau_{a2}, b_2$ for neuron 2, totalizing 10 parameters to be adjusted by the optimization method. For all the cases, the point p was considered as $p = 2.4$s, since it was observed that the output trajectory presents a permanent state after this time.

The Levenberg-Marquardt method was implemented considering a vector composed by: $x = [\tau_{r1}, \tau_{r2}, \tau_{a1}, \tau_{a2}, s_1, s_2, w_{12}, w_{21}, b_1, b_2]$. In order to consider the restrictions cited in Section 3. , the initial values are found by a random process with only 7 parameters, with $\tau_{r1} = \tau_{r2}, \tau_{a1} = \tau_{a2}, b_1 = b_2$. No domain restrictions are added since their values are not known.

The algorithm considers all the points of the interval separately, verifying the individuals errors for each time. Thus, as a performance index, it is defined the global error given by Eq. 4, and it is considered an optimization interval that include 4 steps in the robot walking, as shown in Fig. 6(a) and 6(b), totalizing 7.2s.

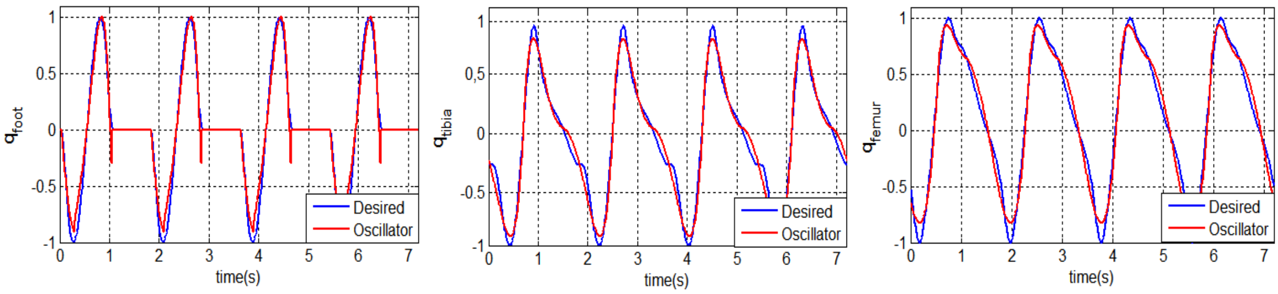
Figures 7(a), 7(b) and 7(c) show the initial neural oscillator output, given by random parameters and considering a minimum error between this trajectory and the desired one. This initial curve should be properly selected to guarantee the convergence of the optimization method.

Figures 8(a), 8(b) and 8(c) show the results considering the initial and final parameters shown in Tab. 3. It can be observed that the algorithm works satisfactorily and the neural oscillator output follows the desired trajectory.



(a) Initital foot trajectory. (b) Initial tibia trajectory. (c) Initial femur trajectory.

Figure 7. Initial curves for the optimization process.



(a) Optimal foot trajectory. (b) Optimal tibia trajectory. (c) Optimal femur trajectory.

Figure 8. Optimal curves after the optimization process.

It can be observed that there is a small error, of approximately 5.86%(global error) for the foot trajectory, and 5.49% to femur trajectory, between the desired and resulting neural oscillator trajectory. Moreover, the parameters w_{ij} for the trajectory of the foot gets negative values. These values are necessary to guarantee the flat section of the trajectory that corresponds to the stance phase of walking, when the foot is completely in contact with the ground. Again, the main issue related to the proposed optimization system is the correct choice to the initial parameters to the optimization process. Some choices are not be able to successfully found a satisfactory solution to the procedure, and the method finds a local minimum for the fitness function. For the tibia trajectory, the error after the optimization was approximately 13%. However, it was observed that this error can be decreased after a long optimization process considering a large-scale algorithm.

Table 3. Initial and optimized parameters for foot, tibia and femur joints.

		τ_{r1}	τ_{r2}	τ_{a1}	τ_{a2}	b_1	b_2	w_{12}	w_{21}	s_1	s_2
foot	initial	0.0219	0.0219	2.3349	2.3349	5.8406	5.8406	-1.6723	-1.0115	2.6731	3.3237
error: 5.86%	final	0.0485	0.1515	2.3674	0.8444	5.3569	5.6904	-2.0835	-1.5438	3.3578	4.8664
tibia	initial	0.0502	0.0502	0.8805	0.8805	1.4187	1.4187	2.2650	0.9465	1.5682	0.8772
error: 13.3%	final	0.1353	0.0154	0.1534	1.4166	2.0789	0.3733	2.1453	1.3923	1.8473	0.9365
femur	initial	0.0502	0.0502	0.8805	0.8805	1.4187	1.4187	2.2650	0.9465	1.5682	0.8772
error: 5.49%	final	0.1748	0.0424	0.2767	1.2196	1.8235	1.1211	1.8305	1.4204	1.8556	1.2293

The trajectories generated with the optimal parameters were used in a simulation, considering the dynamic for the robot shown in Fig. 3(a). The simulator is implemented in Matlab, Fig. 9, and the dynamic equations are computed according to the Eq. 5.

$$M_{ort}(q)\ddot{q} + C_{ort}(q,\dot{q}) + G_{ort}(q) = \tau, \tag{5}$$

where $q \in \Re^7$ is the position vector, $M \in \Re^{7\times 7}$ is the inertia matrix, $C \in \Re^7$ is the Coriolis and centrifugal torques, and $G \in \Re^7$ is the gravitational torques.

In these simulations, by consider just that the joints should follow the trajectories, no control it was used in these preliminary tests. Figures 10(a), 10(b) and 10(c) show the behavior of the joints of the foot, tibia and femur, respectively, subject to the generated trajectories, in absolute angles.

Note that, because no control was applied in the system, there are some unstable points during the walking, mainly when the impact with the ground occurs. However, the simulation shows that the optimized parameters can be used to

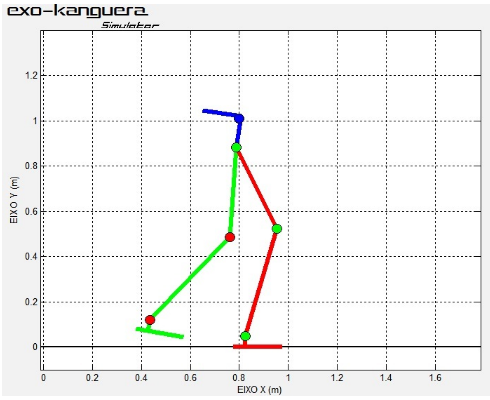
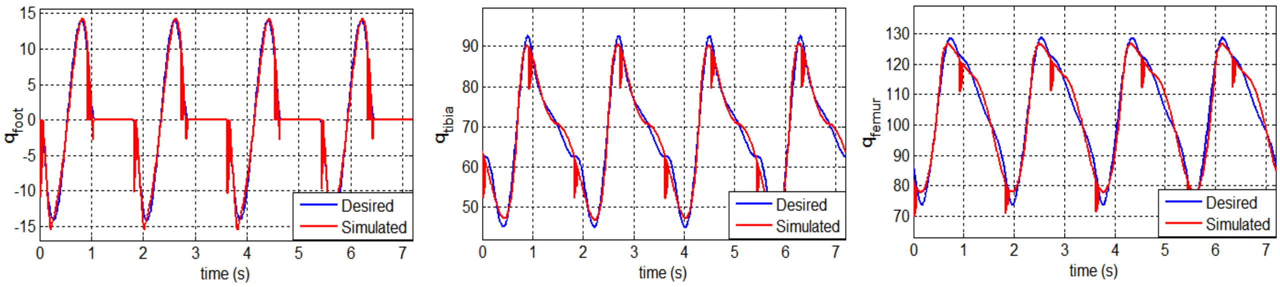


Figure 9. Simulator interface in Matlab.



(a) Foot trajectory simulation. (b) Tibia trajectory simulation. (c) Femur trajectory simulation.

Figure 10. Simulation considering the optimal parameters.

generate and adapt joint trajectories for the exoskeleton. New experiments and a new approach with a controller will be designed in future works.

6. CONCLUSION

This work deals with a procedure to optimize the parameters for neural oscillator to generate joint trajectories for an exoskeleton for lower limbs. The proposed optimization system, based on the Levenberg-Marquartd optimization method, aims to find the best values of the Matsuoka oscillators' parameters such that their output follow a given desired trajectory. The approach performed in this work consider 10 parameters for a Matsuoka oscillator model, that should be appropriately adjusted. In this case, it is observed that the optimization process works satisfactorily with a correct choice to the initial parameters. These initial values should be selected by a random process considering the range restrictions. The results show that the optimization process is feasible to find good parameters for the Matsuoka oscillators, according to the desired trajectories. It is also shown that, with these optimized parameters, the trajectory generator using neural oscillators can be applied in an adaptive model that include interaction forces between the user and the robot, so changing the trajectory according to the user intention.

7. REFERENCES

Arsenio, A.M., 2004. "On stability and tuning of neural oscillators: Application to rhythmic control of a humanoid robot". *Nural Networks 1*, pp. 99–104.

Baydin, A.G., 2008. "Evolution of central pattern generators for the control of a five-link planar bipedal walking mechanism". *CoRR*, Vol. abs/0801.0830.

Golubitsky, M., Stewart, I., Buono, P.L. and Collins, J.J., 1999. "Symmetry in locomotor central pattern generators and animal gaits". *Nature*, Vol. 401, pp. 693–695.

Gomes, M.A., Siqueira, A.A.G. and Silveira, G.L.M., 2009. "On-line trajectory adaptation for active lower limbs orthoses based on neural networks". In *Proceedings of the 20th International Congress of Mechanical Engineering, COBEM 2009*. Gramado, RS, Brazil.

Grillner, S., 1985. "Neurobiological bases of rhythmic motor acts in vertebrates". *Science*, Vol. 228(4696), pp. 143–149.

Hattori, Y., Suzuki, M., Soh, Z., Kobayashi, Y. and Tsuji, T., 2010. "A novel tuning method for neural oscillators with a ladder-like stricture based on oscillation analysis". *ICANN 2010*, pp. 401–410.

Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N. and Tanie, K., 2001. "Planning walking patterns for a biped robot". *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 3.

Jardim, B. and Siqueira, A.A.G., 2009. "Development of series elastic actuators for impedance control of an active ankle foot orthosis". In *Proceedings of the 20th International Congress of Mechanical Engineering, COBEM 2009*. Gramado, RS, Brazil.

Liu, G.L., Habib, M.K., Watanabe, K. and Izumi, K., 2008. "Central pattern generators based on matsuoka oscillators for the locomotion of biped robots". *Artificial Life and Robotics*, Vol. 12, No. 1-2, pp. 264–269.

Marquardt, D.W., 1963. "An algorithm for least-squares estimation of nonlinear parameters". *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, pp. pp. 431–441.

Matsuoka, K., 1985. "Sustained oscillations generated by mutually inhibiting neurons with adaptation". *Biological Cybernetics*, Vol. 52, pp. 367–376.

Matsuoka, K., 1987. "Mechanisms of frequency and pattern control in the neural rhythm generators". *Biological Cybernetics*, Vol. 56, pp. 345–353.

Pons, J.L., 2008. *Wearable Robots: Biomechatronic Exoskeletons*. John Wiley & Sons, Ltd., Madrid, Spain.

Righetti, L. and Ijspeert, A.J., 2006. "Programmable central pattern generators: an application to biped locomotion control". In *In Proceedings of the 2006 ieee international conference on robotics and automation*. pp. 1585–1590.

van der Pol, B., 1927. "Forced oscillations in a circuit with non-linear resistance (reception with reactive triode)". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, Vol. 3, p. 65Ű80.

Yang, M., Murai, S., Murakami, K., Seo, W. and Chong, N.Y., 2006. "Adaptation in bipedal locomotion using phase oscillator networks". In *The 3rd International Conference on Ubiquitous Robots and Ambient Intelligence (URAI 2006)*.

Yang, W., Chong, N.Y., Kim, C. and You, B.J., 2007. "Self-adapting humanoid locomotion using a neural oscillator network". In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. pp. 309 –316.

8. Responsibility notice

The authors are the only responsible for the printed material included in this paper.