DESIGNING A NEURO PD WITH GRAVITY COMPENSATION FOR SIX LEGGED ROBOT

Efrén Gorrostieta Hurtado
Instituto Tecnológico de Quéretaro, México
efren.hurtado@usa.net
Alberto Aguado Behar
ICIMAF, Cuba
aguado@icmf.inf.cu
Emilio Vargas
Centro de Ingeniería y Desarrollo Industrial
Querétaro, México
emilio@cidesiqueretaro.com

*Abstract***.** In this work it is shown the design of a neuro PD control with gravity compensation for the generation of leg trajectories of a six-legged robot. The paper is divided in three parts. In the first part the dynamic model for the robot's leg is introduced, the second part shows the control law designed by considering the gravity effect and the dynamic parameters, finally in the third part we discuss the experimental results obtained by computer simulation.

Keywords. Walking machine, control trajectory, hexapod robot.

1. INTRODUCTION

The design of the control system in a walking robot plays an important role for efficient walking manner. Several problems have to be solved to obtain an automatic locomotion behavior. Some of these problems are the stability control during the walking process, the working space restriction to avoid crashing impacts between legs or body, the force distribution in the robot and the adaptable way to walk on different terrains. At the time, several ways exist to make easy the walking locomotion with a kind of adaptability but more research is needed.

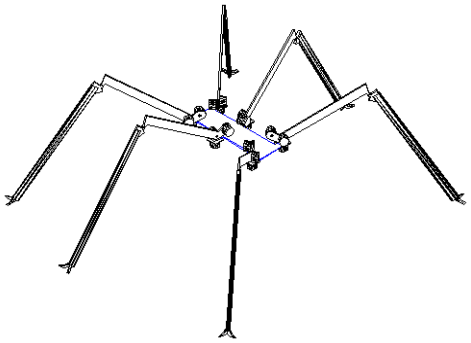

Fig. 1. Configuration of the walking robot.

We focus our attention in the validation of a PD control law considering the gravity effect of the legs as a way to increase the locomotion speed and reduce the energy consumption. A neural component is included aimed to add some adaptability capacity to the control scheme. Figure 1 shows the robot design; the morphology of the robot is similar to the given in [1].

2. DYNAMIC MODEL OF THE LEG

The design of the control law is based in the dynamic model of the robot's leg. Each leg of the robot consists of a basic configuration of three degrees of freedom as is shown in figure 2. The variables and parameters that conform the mathematical model of the robot are the following: θ_1, θ_2, θ_3 are the relative angles between the links, which are independent; l_2 and l_3 are the effective longitude for the link 2 and link 3; m_1, m_2, m_3 and J1, J2 and J3 are the mass and the inertia for link 1, link2 and link3, respectively.

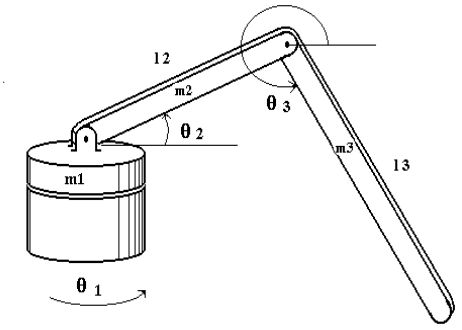

Fig. 2. Parameter and variables used in the leg modelling.

We consider each link, as a rigid body. From and energetic point of view, the Lagrange technique is used to obtain the dynamic model for the legs. Equation 1 is a fundamental relationship between internal and external energy. K represents the kinetic energy of the mechanical system and U represents the potential energy.

$$L = K - U \tag{1}$$

Equation 2 is the fundamental relationship to determine the external torque for each generalized coordinate.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_n}\right) - \frac{\partial L}{\partial \theta_n} = \tau \tag{2}$$

The mathematical dynamic model for the legs is expressed by equation 3, the system can be reasonably represented as a second order differential equation.

$$\begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & a_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ b_{21} & 0 & b_{23} \\ b_{31} & b_{32} & 0 \end{bmatrix}\begin{bmatrix} \dot{\theta}_1^2 \\ \dot{\theta}_2^2 \\ \dot{\theta}_3^2 \end{bmatrix} + ..$$

$$+ \begin{bmatrix} c_{11} & c_{12} & 0 \\ c_{21} & 0 & c_{23} \\ 0 & 0 & c_{33} \end{bmatrix}\begin{bmatrix} \dot{\theta}_1\dot{\theta}_2 \\ \dot{\theta}_1\dot{\theta}_3 \\ \dot{\theta}_2\dot{\theta}_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & f_{22} & f_{23} \\ 0 & 0 & f_{33} \end{bmatrix}\begin{bmatrix} m_1 g \\ m_2 g \\ m_3 g \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} \tag{3}$$

where:

$$\begin{aligned} a_{11} &= J_1 + m_3 l_2^2 \cos^2\theta_2 + \frac{1}{2} m_3 l_2 l_3 \cos\theta_2 \cos\theta_3 + \frac{1}{4} m_3 l_3^2 \cos^2\theta_3 \\ a_{22} &= J_2 + \frac{1}{4} l_2 m_2 + l_3 m_3 \\ a_{23} &= \frac{1}{2} m_3 l_2 l_3 (\text{sen}\theta_2 \,\text{sen}\theta_3 + \cos\theta_2 \cos\theta_3) \\ a_{32} &= \frac{1}{2} m_3 l_2 l_3 (\text{sen}\theta_2 \,\text{sen}\theta_3 + \cos\theta_2 \cos\theta_3) \\ a_{32} &= \frac{1}{4} l_3 m_3 + J_3 \end{aligned} \tag{4}$$

$$\begin{aligned} b_{21} &= -\frac{1}{2} l_2^2 (m_2 \,\text{sen}2\theta_2 + m_3 \,\text{sen}2\theta_2) - \frac{1}{4} m_3 l_2 l_3 \cos\theta_3 \,\text{sen}\theta_2 \\ b_{23} &= \frac{1}{2} m_3 l_2 l_3 (\text{sen}\theta_2 \cos\theta_3 - \cos\theta_2 \,\text{sen}\theta_3) \\ b_{31} &= \frac{1}{2} m_3 l_3 (l_2 \cos\theta_2 \,\text{sen}\theta_3 - \frac{1}{2} \text{sen}2\theta_3 \\ b_{32} &= \frac{1}{2} m_3 l_2 l_3 (\text{sen}\theta_2 \,\text{sen}\theta_3 + \cos\theta_2 \cos\theta_3) \end{aligned} \tag{5}$$

The Coriolis and centrifugal terms are defined by the conjunction of square angular velocity effect and the multiple coordination of the angular velocities for each generalized coordinate.

$$\begin{aligned} c_{11} &= -(m_3 l_2^2 \,\text{sen}2\theta_2 + \frac{1}{2} m_3 l_2 l_3 \,\text{sen}\theta_2 \cos\theta_3 + \frac{1}{4} m_2 l_2^2 \,\text{sen}2\theta_2) \\ c_{12} &= -(\frac{1}{2} m_3 l_3 l_2 \cos\theta_2 \,\text{sen}\theta_3 + \frac{1}{2} m_3 l_3^2 \,\text{sen}2\theta_3) \\ c_{21} &= \frac{1}{2} m_3 l_2 l_3 (\cos\theta_2 \,\text{sen}\theta_3 - \text{sen}\theta_3 \cos\theta_3) \\ c_{23} &= \frac{1}{2} m_3 l_2 l_3 (\cos\theta_2 \,\text{sen}\theta_3 - \text{sen}\theta_3 \cos\theta_3) \\ c_{33} &= \frac{1}{2} m_3 l_2 l_3 (\text{sen}\theta_2 \cos\theta_3 - \cos\theta_2 \,\text{sen}\theta_3) \end{aligned} \tag{6}$$

Coefficients fij are the terms for the gravity effect of the masses.

$$\begin{aligned} f_{22} &= \frac{1}{2} l_2 \cos\theta_2 \\ f_{23} &= l_2 \cos\theta_2 \\ f_{33} &= \frac{1}{2} l_3 \cos\theta_3 \end{aligned} \tag{7}$$

The numeric values used in the simulation of the model can be seen in the Table I

TABLE I

m_1	0.0326 Kg.	J_1	$1.95\text{x}10^{-3}\text{Kgm}^2$
m_2	0.0726 Kg.	J_2	$2.95\text{x}10^{-3}\ \text{Kgm}^2$
m_3	0.0653 Kg.	J_3	$1.95\text{x}10^{-3}\ \text{Kgm}^2$
l_1	0 m	ξ_1	0.03
l_2	0.12 m	ξ_2	0.04
l_3	0.24 m	ξ_3	0.02

3. Dynamic Control of the Robot's Leg

Once constructed the dynamic model of the leg, we can accomplish a simulation study that will illustrate the behavior of the system. We focus our attention to make the study considering the system torques of each joint as the inputs and the joint angles and velocities as the outputs of the model. In order to describe the position of each leg in a cartesian coordinate frame, a cinematic model to transform the robot positions to cartesian positions (X,Y,Z) was used. On the other hand, we design a control law which uses the angle joints errors to calculate the torques which must be applied to each joint actuator, and then evaluate the dynamic model to get new angles for the joints. Figure 3 shows the block diagram of the control law, this scheme is well known as control with gravity compensation which is represented by C(g) [2].

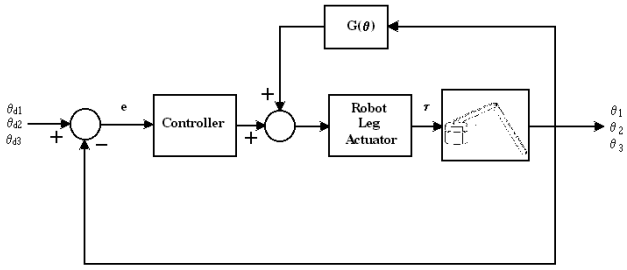

Fig. 3. Block diagram of the control law.

The control algorithm used to move a leg in the simulations was a PD control law, which requires, as it is known, to tune two constants by each degree of freedom, proportional gain K and the derivative gain Kd. In our intention to control the position and velocity of each joint, we must tune a total of 12 variables.

TABLE II
PD CONTROL CONSTANTS FOR POSITION CONTROL

Constants	K1	K2	K3	Kd1	Kd2	Kd3
Value	0.06	0.09	0.2	0.009	0.6	0.01

TABLE III
PD CONTROL CONSTANTS FOR VELOCITY CONTROL

Constant	K1	K2	K3	Kd1	Kd2	Kd3
Value	0.04	0.9	0.17	0.001	0.02	0.02

In tables II and III, values of proportional and derivative gains for the cases of leg's position and velocity control which were the best obtained by trial an error in the simulation study, are shown. Position and velocity must follow trajectories generated by the respective trajectory generators.

As will be seen in figures 7 to 12, those values of the control constants guarantee that the the angle positions move relatively smoothly until arriving at the wished trajectories even when some oscilations are observed, specially in the third joint. However, the tracking of velocity trajectories is very deficiently achieved, in fact, big deviations of desired trajectoires can be observed.

To improve the performance of the PD controller, adding an adaptive component which can cope with the external disturbances, the multivariable essence and the strong nonlinearities of the robot, we propose a neural component which can correct in real time the values of the proportional and derivative gain coefficients. In the next section, the derivation of the neural correction algorithm will be presented.

4. Neuro-PD algorithm

In figure IV, an scheme of the neuro-PD controller is presented. In the sake of clarity, only two nets are depicted, even when 6 will be requiered for every leg, one for each gain coefficiente (3 for proportional and 3 for derivative constants).

The neural nets are of the perceptron type, with a hidden layer which has only a neuron with a sigmoid activation function. The output layer is linear and has also a single neuron. The outputs of the nets are respectively ΔK_p and ΔK_v, that is the increments of proportional and derivative gains.
The inputs of all the neuronal networks are the output errors and their first differences, that is:

$$x(t) = \lfloor e_p(t) \quad \Delta e_p(t) \rfloor \tag{8}$$

where

$$e_p(t) = \theta_d(t) - \theta(t) \tag{9}$$

represents the errors in the joints angles.

The neural networks are trained by means of special backpropagation algorithm which will be

developed in detail for the case of proportional gain changes. The derivative constants modifications can be obtained in a similar way.

The objective function that must be minimized by means of the neural nets is the quadratic function:

$$E(t) = \frac{1}{2}\sum e_p^2(t) \tag{10}$$

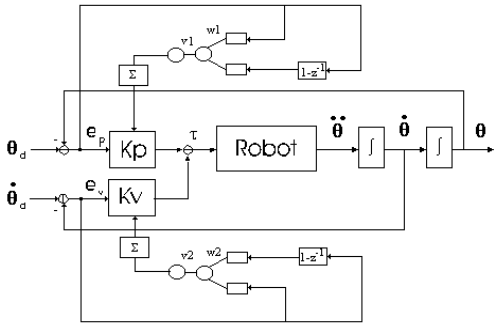

Figure 4 Neuro PD control structure.

The joint angles θ(t) are related with the input torques by means of a supposed unknown non-linear dynamic model that we denote by:

$$\theta(t) = R(\tau(t)) \tag{11}$$

The torques calculated by the controller are:

$$\tau(t) = k_p e_p(t) + k_v e_v(t) \tag{12}$$

The proportional gain will change as follows:

$$k_p = k_p + \Delta k_p \tag{13}$$

$$\Delta k_p = vh \tag{14}$$

where v is the weight coefficient which connect the hidden with the output neuron and h is the output of the hidden neuron.

The inputs to the neurons of the hidden layer are

$$s = w_1 e_p(t) + w_2 \Delta e_p(t) \tag{15}$$

The activation function of the hidden neuron is:

$$h = \frac{1}{1+e^{-s}} \tag{16}$$

To apply the back propagation algorithm it is necessary to calculate the gradient of the function E_p whit respect to the coefficient v and w, that is:

$$\nabla E(t) = \begin{bmatrix} \frac{\partial E(t)}{\partial v_1} \\ \frac{\partial E(t)}{\partial w_j} \end{bmatrix} \tag{17}$$

The partial derivatives which appear in equation (17) can be calculated by means of the chain rule of derivations, that is:

$$\frac{\partial E(t)}{\partial v_1} = \frac{\partial E(t)}{\partial e_p}\frac{\partial e_p}{\partial \theta}\frac{\partial \theta}{\partial \tau}\frac{\partial \tau}{\partial K_p}\frac{\partial K_p}{\partial \Delta K_p}\frac{\partial \Delta K_p}{\partial v_1} \tag{18}$$

$$\frac{\partial E(t)}{\partial w_j} = \frac{\partial E(t)}{\partial e_p}\frac{\partial e_p}{\partial \theta}\frac{\partial \theta}{\partial \tau}\frac{\partial \tau}{\partial K_p}\frac{\partial K_p}{\partial \Delta K_p}\frac{\partial \Delta K_p}{\partial h}\frac{\partial h}{\partial s}\frac{\partial s}{\partial w_j} \tag{19}$$

After the sustitution of the partial derivatives which appear in (18) and (19), we arrive to:

$$\frac{\partial E_p}{\partial v_1} = -e_p(t)^2 h \frac{\partial \theta}{\partial \tau} \tag{20}$$

$$\frac{\partial E_p}{\partial v_1} = -e_p(t)^2 \frac{\partial \theta}{\partial \tau} v_1 h(1-h) x_j \tag{21}$$

As can be seen, in (20) and (21) it appears the partial derivative $\frac{\partial \theta}{\partial \tau}$, which can not be evaluated under the assumption that the robot model is not precisely known. Even in the case that we dispose of a precise model, the evaluation would be very time consumming for real time realization of this algorithm. As it was shown in [7] and [8], under some not very restrictive conditions, that partial derivative, also known as the Jacobian or equivalent gain of the process under control, can be substituted by its sign. Then, equations (20) and (21) can be simplified to:

$$\frac{\partial E_p}{\partial v_1} = -e_p(t)^2 h \ \mathrm{sign}(R) \tag{22}$$

$$\frac{\partial E_p}{\partial v_1} = -e_p(t)^2 \mathrm{sign}(R)\, v_1 h(1-h) x_j \tag{23}$$

The function sign (R) is evaluated as +1 or –1 depending on the sign of the relation between the angular position and the applied torque which can be considered, for the case of the robot, as allways positive.

Equations (22) and (23) can be used to derive the adaptation equations for the weight coefficients of the neural net, w and v, using the steepest descent method as follows:

$$v_1(t+1) = v_1(t) + \eta e_p(t)^2 h \,\mathrm{sign}(R) \tag{24}$$

$$w_j(t+1) = w_j(t) + \eta e_p(t)^2 \, \text{sign}(R) \, v_1 h(1-h) x_j \quad (25)$$

The adaptive equations (24) and (25) can be calculated easily in real time and serve to adapt the values of proportional coefficients Kp, using equations (13) and (14). A similar derivation can be used to find the adaptation equations for derivative coefficients K_d .

The neural adaptive scheme can be used in a permanent way, which achieve succesive improvements of the control behavior.

In tables IV and V are shown the proportional and derivative gains obtained after several simulations of the neuro PD control algorithm, parting from the values which appear in tables II and III

TABLE IV
PD CONTROL CONSTANTS FOR POSITION CONTROL OBTAINED WITH NEURO ADAPTATION

Constant	K1	K2	K3	Kd1	Kd2	Kd3
Value	0.07	0.30	0.18	0.11	0.011	0.007

TABLE V
PD CONTROL CONSTANTS FOR VELOCITY CONTROL OBTAINED WITH NEURO ADAPTATION

Constant	K1	K2	K3	Kd1	Kd2	Kd3
Value	0.05	1.70	0.18	0.014	0.022	0.023

5. Trajectory for the Leg

The primary target that is tried for the control proposed in this paper is to follow a trajectory for a leg step. This trajectory is parametrically designed, and it can change according to the type of land or application of the robot. Figure 5 shows a parabolic type trajectory motion for the leg, and Figure 6 shows a triangle trajectory [3][4] which is tipical of the movement of some animals. In our simulation study, however, it was used the parabolic trajectory due to its simplicity. In this case the equations for the three degrees of freedom in the leg are described by th next equations [5]:

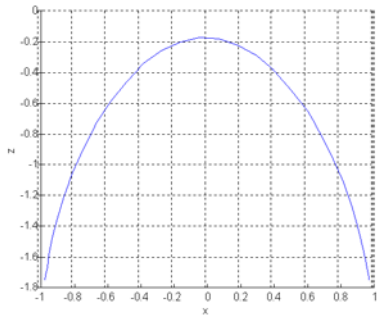

Fig. 5. Parabolic trajectory motion.

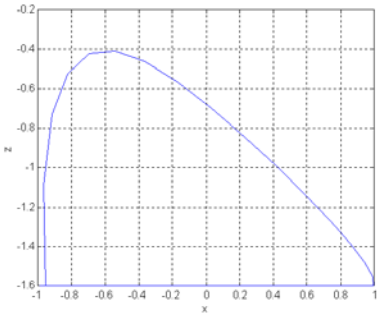

Fig. 6. Triangle trajectory motion.

$$\begin{aligned} \theta_1 &= d\gamma - A\gamma(\cos\xi - 1) \\ \theta_2 &= d\beta - A\beta(\cos\xi - 1) \\ \theta_3 &= d\chi - A\chi(\cos\xi - 1) \end{aligned} \quad (26)$$

where dγ, dβ, dχ are the initial values of angles θ1,θ2,θ3, respectively. Those values fix the initial position of the leg in the space. Variable Aγ defines the step lenght and Aβ, Aχ, define the robot leg height; ξ is an angle which takes values between 0 and π.

In equation 27, we show the values considering the dimensions that are required in the generation of steps for the walking robot of six legs.

$$\begin{aligned} d\gamma &= 70 \quad A\gamma = 20 \\ d\beta &= 4 \quad A\beta = 15 \\ d\chi &= 310 \quad A\chi = 10 \end{aligned} \quad (27)$$

6. Simulation Results

After determining the type of movement and the dynamic control law, we evaluate the leg step by analyzing the motion of each degree of freedom.

One of the considerations we did in our experiments was to generate smooth movements at the beginning and the end of the trajectory, as is shown in Figure 7 for angle θ_1.

In the figures 7 to 12, the desired position and velocity trajectories and those obtained with PD and neuro PD control are shown. As can be seen, for coordinate 2 and 3, the behavior obtained with the neuro PD is substantially improved, specially for the velocity trajectories where the oscillations are supressed and the errors are considerably diminished.

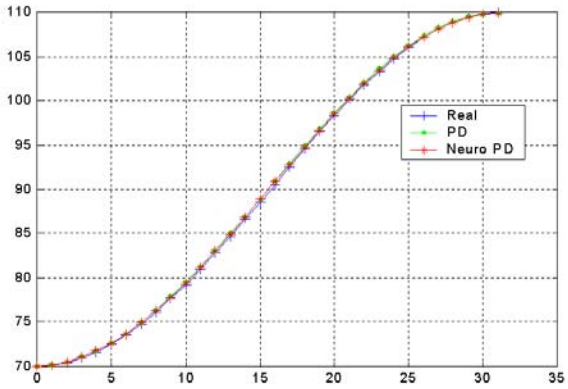

Fig. 7. Position trajectory for the first degree.

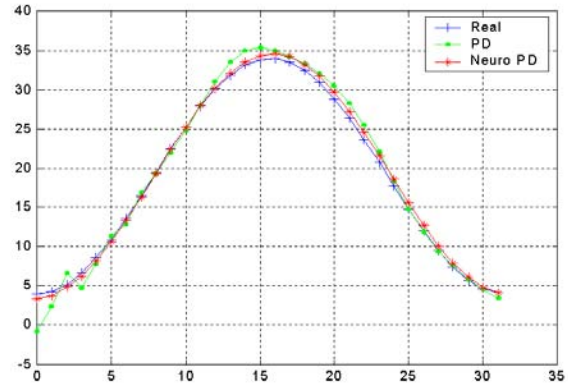

Fig. 8. Position trajectory for the second degree.

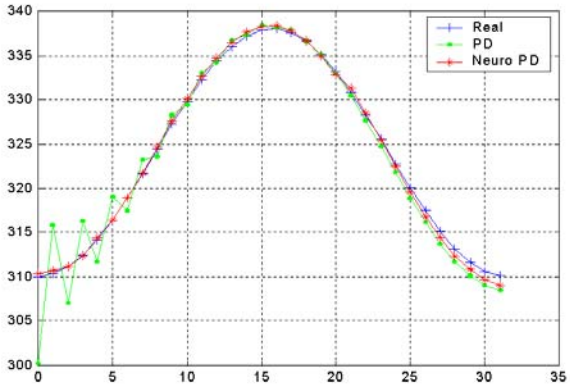

Fig. 9. Position trajectory for the third degree.

The leg step is constructed by using the simultaneous movements of θ_1 , θ_2 and θ_3. In Figure 13 are shown, projected in a plane, the desired and real step trajectories obtained with PD and neuro-PD controllers. For this step generation we simulate the altitude and step distance made for the leg, which is essential information to design algorithms oriented to adapt the step to the terrain, according with the legs positions.

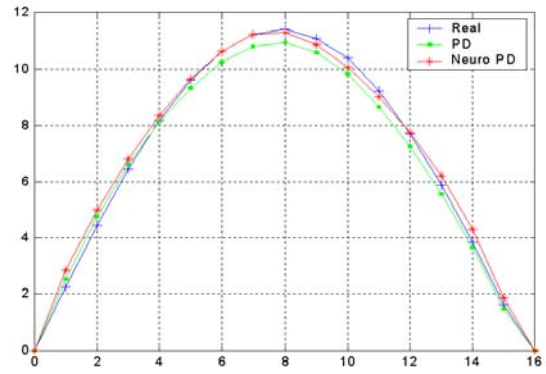

Fig. 10 Velocity trajectory for the first degree.

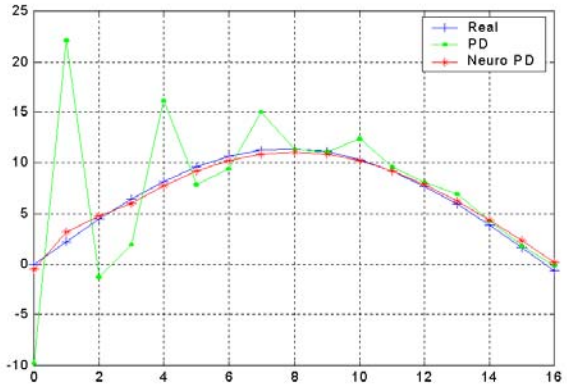

Fig. 11 Velocity trajectory for the second degree.

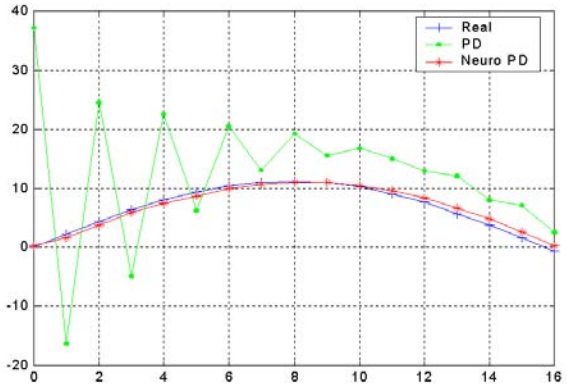

Fig. 12 Velocity trajectory for the third degree

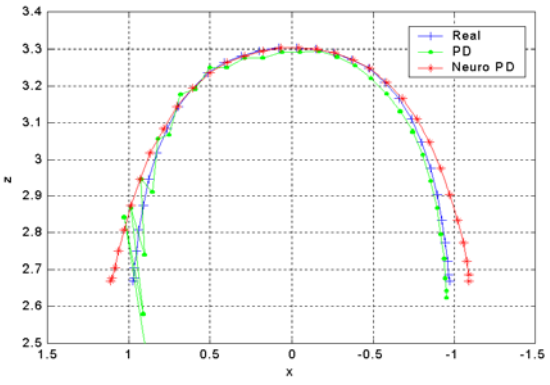

Fig. 13. Step trajectory generation.

6. Conclusions and Future Work

The dynamic model described for a leg of a six-legged robot is a highly nonlinear system, obviously this complicate the design of the control system; however a PD control with gravity compensation was designed with satisfactory results. A series of preliminary trajectories were evaluated by simulation, considering a parametric mathematical model to facilitate the walking adaptation for the leg. We could use the coefficients of the tables 2 and 3 to help us for the design of a walking generator algorithm. We continue this research by considering an intelligent algorithm which includes six simple neural nets for each leg which permit the adaptation of PD coefficients and to improve the performance of the robot. In the future, it is necessary also to research about the flexibility of the working space area for the legs which cause that the mobility of the robot can be increased substantially. The determination of the mobility for the robot and the stability evaluation is possible by using a 3D graphic simulator that is under construction.

ACKNOWLEDGMENT

The authors would like to thank to Dr. Guillermo Rodríguez for his support of this research in the Postgraduate Unit at the Industrial Development and Engineering Center, CIDESI.

REFERENCES

[1] Solano. J, Vargas E, Gorrostieta E, Morales. C. 2000 “Designing a Walking Robot of Six Legs ” In Proceedings of International Symposium on Robotics and Automation ISRA’2000 (Monterrey N. L. , Mexico, Nov 10 – 12).

[2] Tomás Francisco Calyeca Sánchez, Sergio Javier Torres Méndez, Germán Ardul Muñoz Hernández . 2000 “PD control system whit gravity compensation ” In Proceedings of International Symposium on Robotics and Automation ISRA’2000 (Monterrey N. L. , Mexico, Nov 10 – 12).

[3] Peter Nehaus and H Kazerooni. 2001. “Industrial-Strength Human-Assisted Walking Robots”, Robotics and Automation Magazine of IEEE, Vol. 8, No.4 , December 2001.

[4] V. Feliu, A Garcia, J. A. Somolinos. 2001. " Gauge-Base Tip Position Control of a New Three-Degree- of – Freedom Flexible Robot", in The international Journal of Robotics Research, Vol. 20 No. 8 August 2001.

[5] W. Ilg, T. Mühlfriedel, K. Berns, and R. Dillmanm "Hybrid Learning Concepts for a Biollogically Inspired Control of Periodic Movements for Walking Machines" In Soft Computing in Mechatronics. Germany 1999.

[6] Shaoping Bai, H Low and Weimiao Guo "Kinematographic Experiments on The Leg Movements and Body Trayectories of Cockroach Walking on Different Terrain" In Proceedings of International Conferences on Robotics and Automation (San Francisco Cal, April 2001)

[7] Aguado, A., Ordaz, A., Noriega, A. "Self-tuning neural controller". Proceedings of the IFAC Symposium on Real Time Control Architectures. Cancún, 1998.

[8] Cui, X., Chin, K.G. "Direct Control and Coordination using Neural Networks" IEEE Transactions on Systems, Man and Cybernetics", Vol. 23, No. 3, pp 686-697, 1992.