Implementation of Numerical Non-Standard Discretization Methods on a Nonlinear Mechanical System

PATETE ANNA, VELASCO MARIA AND RODRIGUEZ-MILLAN JESUS
Departamento de Sistemas de Control
Universidad de Los Andes
Apartado 11, La Hechicera, Facultad de Ingeniería, Mérida 5101-A
VENEZUELA
apatete@ula.ve, mvelasco@ula.ve, jrmillan@ula.ve

Abstract: - In this work, we shortly review the mathematical concepts of the well known numerical standard disctretization methods: Approximate, Exact and Truncated discretization methods and, the numerical non-standard discretization methods, named: Euler, Euler-Picard and Euler-Taylor-Picard discretization methods. The standard discretization methods are applicable to continuous linear dynamics and a very limited class of nonlinear continuous dynamics; while the non-standard discretization methods are applicable to linear and nonlinear dynamics in general. The non-standard discretization methods theory was developed recently and only simulated results were presented. Our contributions in this work are to show the obtained results and analysis from the digital implementation of linear and nonlinear control laws on a nonlinear control mechanical system: the simple pendulum, using the numerical standard and non-standard discretization methods to discretize the continuous dynamics. Through the implementation we analyze the real validation of the numerical non-standard discretization methods. The results show that better approximation to the real data, obtained from the controlled real system, is given when the numerical non-standard discretization methods are used to dicretize the nonlinear dynamics. Also we validated the advantages of using digital nonlinear control laws on nonlinear control systems.

Key-Words: - Nonlinear control, nonlinear discretization, nonlinear state feedback, numerical method, mechanical system, simple pendulum.

1 Introduction

As we know, in general, real physic control systems are nonlinear. And usually the proceeding is to obtain a linear approximation of the nonlinear system model, which will represent the nonlinear real system into an operational range, then to design a linear control law that performs properly into that range. Commonly this operational range is small and, that is why, it is necessary to design nonlinear control laws that can perform properly into a much larger operational range.

Motivated by the linear control limitation explained above, were developed the extended and exact linearization methods. These methods allow to design nonlinear control laws using states feedback [1][2].

On the other hand, digital implementation of control laws has become a common way nowadays. In case of continuous linear dynamics, there are well developed methods to obtain its discrete-time dynamic representation counterpart. We called those methods the standard discretization methods: Approximate, Exact and Truncated discretization methods (see [3]-[5] among others).

For the case of continuous nonlinear dynamics the non-standard discretization methods: Euler, Euler-Picard and Euler-Taylor-Picard discretization methods were developed and presented only through simulated results (see [6]-[8]).

The non-standard discretization methods are bases on Euler polygons, Picard iteration and Taylor series expansions. Then, the methods might be described as a triplet: Euler sampling, Picard interpolation and Taylor approximation.

In this work, we present the validity of the non-standard discretization methods through implementation on a nonlinear dynamical mechanical control system: the simple pendulum, showed in Fig. 1. The nonlinear control design, using extended and exact linearization, for the simple pendulum is carried out in continuous-time and then, the obtained continuous nonlinear control laws dynamics are discretized by the numerical non-standard discretization methods. The nonlinear discrete-time control laws are implemented on a computer, so the closed-loop is obtained as show in Fig. 2.

Fig.1 Mechanical system: the simple pendulum. Control and Automation Laboratory. Universidad de Los Andes.

Fig.2 The simple pendulum in closed-loop. Control and Automation Laboratory. Universidad de Los Andes.

This work is organized as follows: section 2 gives the basic theory of the numerical standard and non-standard discretization methods; section 3 gives the design of the states feedback linear and nonlinear control laws; in section 4 we present the main results: a) discretization of the nonlinear control laws, b) implementation of the digital control laws on a nonlinear mechanical system: the simple pendulum, c) validation of the numerical non-standard discretization methods, and d) some advantage of using digital nonlinear control laws, especially for the case of the simple pendulum; finally in section 5 the conclusions and future work are stated.

Part of this work was presented in the 9th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics [9]. In this contribution we complement the work in [9] and give some new analysis about the experience and obtained results.

2 Numerical Standard and non-standard discretization methods

In this section we give a brief review of the above mentioned numerical standard and non-standard discretization methods. The standard discretization methods may be applied to linear dynamic systems and to a very limited class of nonlinear dynamic systems [3], while the non-standard discretization methods may be applied to linear or nonlinear dynamic systems in general [6]-[8].

2.1 Numerical standard discretization methods

The discretization of a linear, single input single output, dynamical control system represented in states equation as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{1}$$

where $x(0) = x_0$, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}, y(t) \in \mathbb{R}$ and always assumes the standard discrete-time form:

$$\begin{aligned} x((k+1)T_0) &= A_{dm}x(kT_0) + B_{dm}u(kT_0), \\ y(kT_0) &= Cx(kT_0) + Du(kT_0), \end{aligned} \tag{2}$$

where k is the discrete-time variable defined in the set of all integers, i.e. $k \in \mathbb{Z}$, T_0 is the sampling time period $T_0 \in \mathbb{R}$,, $x(0) = x_0$, $x(kT_0) \in \mathbb{R}^n$, $u(kT_0) \in \mathbb{R}$, $y(kT_0) \in \mathbb{R}$, and the matrices A_{dm}, B_{dm}, depend on the discretization method [3], indicated by the second suffix m. This is, for the approximate discretization method $m = a$, for the exact discretization method $m = e$, and for the truncated discretization method $m = t$.

In the approximate discretization method, we discretize the ordinary differential equation in (1) by substituting the first-order time derivative by its first-order backward difference, to obtain:

$$\begin{aligned} A_{dm} &= A_{da} = [I - T_0 A]^{-1}, \\ B_{dm} &= B_{da} = T_0[I - T_0 A]^{-1} B. \end{aligned} \tag{3}$$

In the exact discretization method we first apply zero-order hold [3] to the control signal $u(t)$, and then sample the exact trajectory of the system at $t = kT_0$, to obtain:

$$A_{dm} = A_{de} = e^{AT_0},$$
$$B_{dm} = B_{de} = \int_0^{T_0} e^{A\sigma} d\sigma B. \tag{4}$$

The jth-degree truncated discretization method is based on truncating the Taylor series expansion of e^{AT_0} at its jth-degree term to get:

$$A_{dm} = A_{dt} = \sum_{i=0}^{j} \frac{T_0^i}{i!} A^i,$$
$$B_{dm} = B_{dt} = \sum_{i=0}^{j-1} \frac{T_0^{i+1}}{(i+1)!} A^i B. \tag{5}$$

The reader may be referred to [3]-[5] for father understanding of the numerical standard discretization methods.

2.2 Numerical non-standard discretization methods

By a nth-order continuous-time nonlinear, single input single output, dynamical control system we mean a pair of equations

$$\dot{x}(t) = f(x(t), u(t)),$$
$$y(t) = h(x(t), u(t)), \tag{6}$$

where $x(0) = x_0$, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}$, $y(t) \in \mathbb{R}$, $t \in \mathbb{R}$, and the vector-field f and the output-function h are as smooth as needed. Discretize (6) means to construct a discrete-time nonlinear dynamical control system of the form:

$$\Phi((k+1)T_0) = F(\Phi(kT_0), u(kT_0)),$$
$$\Psi(kT_0) = H(\Phi(kT_0), u(kT_0)), \tag{7}$$

where k is the discrete-time variable $k \in \mathbb{Z}$, T_0 is the sampling time period $T_0 \in \mathbb{R}$, $\Phi(0) = \Phi_0$, $\Phi(kT_0) \in \mathbb{R}^n$, $u(kT_0) \in \mathbb{R}$, $\Psi(kT_0) \in \mathbb{R}$, and the functions F and H are obtained from f and h, respectively, according to a prescribed numerical non-standard discretization method.

Forward approximate discretization supports the construction of Euler polygons like approximated solutions of nonlinear dynamical control system [10], wherefore forward approximate discretization of nonlinear dynamical control system is called the Euler discretization method [6]. This is the default method used in practice to discretize nonlinear dynamical control system, and it is metaphorically described as periodic sampling plus linear interpolation [6]. The Euler-discretization of (6) is given by:

$$\Phi(kT_0) = \Phi((k-1)T_0) + T_0 f(\Phi((k-1)T_0), u((k-1)T_0)),$$
$$\Psi(kT_0) = h(\Phi(kT_0), u(kT_0)), \tag{8}$$

where $\Phi(0) = \Phi_0$, $\Phi(kT_0) \in \mathbb{R}^n$, $u(kT_0) \in \mathbb{R}$, $\Psi(kT_0) \in \mathbb{R}$ [6].

Given that nonlinear dynamical control system are in general not explicitly solvable, the exact discretization method is neither extendable to them. Yet, in [6] the Euler-Picard discretization method was proposed, and may be paraphrased as global periodic Euler-like sampling plus Picard-like interpolation. The ith-iterated discrete-time Euler-Picard trajectory Ψ_i of period T_0 [6],

$$\Psi(kT_0) = \Psi_i(kT_0) = x_0 + \sum_{j=1}^{k} \int_{(j-1)T_0}^{jT_0} f(\Phi_{i-1}^k(\sigma), u(\sigma)) d\sigma \tag{9}$$

where $\Psi_i(0) = x_0$, $\Psi(kT_0) \in \mathbb{R}$, would closely correspond to the sampling of the trajectories of (6), as the number of Picard iterations goes to infinity, and the period gets very small, i.e. $i \to \infty$ and $T_0 \to 0$.

Concerning the fitting properties of the periodic non-standard discretization methods proposed in [6], the Euler-Picard discretization method is the best. Yet, because of the computation of the integrals involved, it is also prohibitively expensive for (6) with complex nonlinearities.

To reduce the computation time and the unpredictable impact of arbitrary nonlinearities, the Euler-Taylor-Picard discretization method was proposed in [6]. It may also be paraphrased as periodic sampling plus Picard interpolation, however is not applied to (6), but to its jth-degree Taylor polynomial approximation

$$\dot{x}(t) = F(x(t), u(t), j) = \sum_{i=0}^{j} \alpha_i(x_0, u)\ (x(t) - x_0)^j \tag{10}$$

where $x(0) = x_0$. The associated Euler-Taylor-Picard discrete-time trajectories are:

$$\Psi(kT_0) = \Psi(j,i,k,T_0) = x_0 + \sum_{r=1}^{k} \int_{(r-1)T_0}^{rT_0} F(\Phi_{i-1}^{k}(\sigma), u(\sigma), j) d\sigma \quad (11)$$

where $\Psi(j,i,0) = x_0$, $\Psi(kT_0) \in \mathbb{R}$. Thus, this method depends on the sampling period T_0, the number of Picard iterations and the degree of the Taylor polynomial expansion of the vector field f.

3 Design of Linear and Nonlinear Discrete-Time States Feedback Control Laws

In this section, the linear and nonlinear discrete-time states feedback control laws are designed for the considered mechanical system, the simple pendulum.

The linear control law is designed using the linear state feedback technique and the nonlinear feedback control design is carried out by using extended and exact linearization techniques. First, the basic theory for the design techniques are explained briefly then, the linear and nonlinear control laws are given for the simple pendulum model.

3.1 Linear State Feedback Technique

This technique uses the information of all states of the system and feed it back to the control law to achieve the desired specifications. The linear control design is carried out based on the linear system model. If the system model is nonlinear, this must be linearized first. The algorithm to compute de linear state feedback control law is as follows:

Algorithm 1: Linear state feedback (linear control law)

1) Given the nonlinear system model (6), compute the operational point (or equilibrium point) of interest, this is: $OP = (x_i(t), u(t), y(t)) = (X_i, U, Y)$, where $i = 1, 2, ..., n$. Then, linearize (6) and evaluated it on OP. This gives a linear system of the form:

$$\begin{aligned} \dot{x}_\delta(t) &= Ax_\delta(t) + Bu_\delta(t), \\ y_\delta(t) &= Cx_\delta(t) + Du_\delta(t), \end{aligned} \quad (12)$$

where, $x_\delta(t) = x(t) - X$, $u_\delta(t) = u(t) - U$ and $y_\delta(t) = y(t) - Y$.

2) Check the controlability condition for (12), using the controlability matrix,

$$\ell = [B \quad AB \quad ... \quad A^{n-1}B].$$

If (12) is complete controllable, then continue with the next step, otherwise this method is not viable.

3) The linear state feedback control law is given by:

$$u_\delta(t) = -Kx_\delta(t) = -[k_1 \quad k_1 \quad ... \quad k_n]x_\delta(t). \quad (13)$$

In closed-loop, using (12) and (13), the continuous system is given by:

$$\dot{x}_\delta(t) = [A - BK]x_\delta(t) = \tilde{A}x_\delta(t), \quad (14)$$

where $\tilde{A} = [A - BK]$.

4) Compute the characterictic polinomial of (14),

$$\begin{aligned} P_c(s) &= Det[sI - \tilde{A}] = \\ &s^n + a_{n-1}s^{n-1} + ... + a_0 = 0. \end{aligned} \quad (15)$$

5) Design the desired characteristic polynomial in closed-loop,

$$P_d(s) = s^n + \alpha_{n-1}s^{n-1} + ... + \alpha_1 s + \alpha_0. \quad (16)$$

6) Find the vector gain $K = [k_1 \quad k_2 \quad ... \quad k_n]$, equating $P_c(s) = P_d(s)$.

Remark 1: In the case (12) is no complete controllable, exists a method to find the controllable sub-space of the system (6) and, if it is possible, design a linear state feedback control law of reduced order for the controllable sub-space. The uncontrollable sub-space should be stable by itself for the whole system to be controllable [11].

3.2 Extended Linearization

This technique permits to find a nonlinear control law, which is a nonlinear extension of the previously computed state feedback linear control law in Algorithm 1. The nonlinear control law must be equal to the linear control law, if the nonlinear control law is linearized and evaluated it on the operational point OP (used to design the linear control law). The algorithm for the state feedback nonlinear control law using extended linearization is given as follows:

Algorithm 2: Extended linearization (nonlinear control law)

1) Given the linear gains $K = [k_1 \quad k_2 \quad ... \quad k_n]$ obtained in Algorithm 1, find nonlinear gains $\mathrm{K}_{ext}(x_\delta(t))$,

$\mathrm{K}_{ext}(x_\delta(t)) = [k_1(x_\delta(t)) \quad k_1(x_\delta(t)) \quad ... \quad k_n(x_\delta(t))]$,

such that:

$$\left.\frac{\partial \mathrm{K}_{ext}(x_\delta(t))}{\partial x_\delta(t)}\right|_{OP} = [k_1 \quad k_2 \quad ... \quad k_n] = K. \quad (17)$$

Remark 2: There is not an unique way to find (17), however [1][2] give a way to do it, i.e.

$$\mathrm{K}_{ext}(x_\delta(t)) = \int_U^{X^{-1}(x(t))}\left[K(\sigma)\frac{dX(\sigma)}{d\sigma}\right] - X^{-1}(x(t)) + \left[\sum_{j-2}^{n} K_j\left[X^{-1}(x(t))\right]\left(x_j(t) - X_j\left(X^{-1}(x(t))\right)\right)\right], \quad (18)$$

where j is the *jth* gain, i.e. k_j.

3.3 Exact linearization

This technique allows as finding a nonlinear control law without having to pass through the linearization process for the system model. For this technique to be applied, the nonlinear system must be in the controllable canonical form [1]-[4], or may be transformed to it. The algorithm is as follows:

Algorithm 3: Exact linearization (nonlinear control law).

1) Given the nonlinear system in the controllable canonical form:

$$\begin{aligned}
\dot{x}_1(t) &= x_2(t)\\
\dot{x}_2(t) &= x_3(t)\\
&\vdots\\
\dot{x}_{n-1}(t) &= x_n(t)\\
\dot{x}_n(t) &= f_n(x_1(t),\ldots,x_n(t)) + g_n(x_1(t),\ldots,x_n(t))u(t).
\end{aligned} \quad (19)$$

Define,

$$v(t) = f_n(x_1(t),\ldots,x_n(t)) + g_n(x_1(t),\ldots,x_n(t))u(t). \quad (20)$$

2) Design a linear state feedback control law of the form:

$$v(t) = \beta_1 x_1(t) + \beta_2 x_1(t) + \ldots + \beta_n x_n(t). \quad (21)$$

3) Find the nonlinear control law $u(t)$ equaling (20) and (21), i.e.,

$$u(t) = \frac{-\beta_1 x_1(t) - \beta_2 x_1(t) - \ldots - \beta_n x_n(t)}{g_n(x_1(t),\ldots,x_n(t))} + \frac{f_n(x_1(t),\ldots,x_n(t))}{g_n(x_1(t),\ldots,x_n(t))} \quad (22)$$

3.4 Nonlinear mechanical system: the simple pendulum

The simple pendulum showed in Fig. 1 can be represented by the diagram showed in Fig. 3. Where q, l and l_c represent the angular position, the longitude and the central mass longitude, respectively.

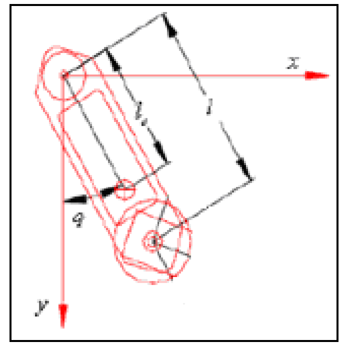

Fig.3 Simple pendulum diagram

The model is then obtained by the equation:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} + \frac{\partial D}{\partial \dot{q}} = \tau(t), \quad (23)$$

where L represent the Lagrangean, $D = \frac{1}{2}\alpha\,\dot{q}^2(t)$ is the dissipative term and $\tau(t)$ is the input. Thus the nonlinear model [12][13] is:

$$\left(ml_c^2 + I\right)\ddot{q}(t) + \alpha\dot{q}(t) + ml_c g\, sin(q(t)) = \tau(t), \quad (24)$$

where m, I, g, and α represents the pendulum mass, inertia, gravity constant and the friction coefficient, respectively.

The parameters values are: $m = 0.5400$ Kg, $l_c = 0.1070$ m, $\alpha = 0.009$ Nm.s/rad and $I = 0.011$ Nm. α is an estimated parameter obtained by identification as in [12].

In first order equation, (24) is represented by:

$$\begin{aligned}
\dot{q}_1(t) &= q_2(t),\\
\dot{q}_2(t) &= \frac{-\theta_2\, sin(q_1(t)) - \alpha q_2(t) + \tau(t)}{\theta_1},
\end{aligned} \quad (25)$$

where $\theta_1 = (ml_c^2 + I)$ and $\theta_2 = ml_c g$.

The linear model, parametrized by the position $q_1(t) = q_{op}$ is,

$$\dot{q}_\delta(t) = \begin{bmatrix} 0 & 1\\ -\frac{\theta_2}{\theta_1}cos(q_{op}) & -\frac{\alpha}{\theta_1}\end{bmatrix} q_\delta(t) + \begin{bmatrix} 0\\ \frac{1}{\theta_1}\end{bmatrix}\tau_\delta(t),$$

$$y_\delta(t) = \begin{bmatrix} 1 & 0\end{bmatrix} q_\delta(t), \quad (26)$$

The system (26) is completely controllable and observable (for further understanding on those concepts refers to [5], for example). Then, it is possible to design a state feedback control law.

First we design the linear state feedback control law for (26). Given the closed-loop desired specifications, in this particular case we use: damping coefficient $\delta = 0.59$ and natural damping $\omega_n = 13,53$, the linear control law, using Algorithm 1, is obtained as:

$$
\begin{aligned}
u(t) &= -(\omega_n^2\theta_1 - \theta_2 cos(q_{1op}))(q_1(t) - q_{1op}) \\
&+\theta_2 sin(q_{1op}) - (2\varsigma\omega_n\theta_1 - \alpha)q_2(t).
\end{aligned} \tag{27}
$$

To design the nonlinear control law by extended linearization technique, we use (27) and Algorithm 2. Then the extended nonlinear control law is:

$$
\begin{aligned}
u(t) &= \theta_2 sin(q_{1op}) + \theta_2(sin(q_1(t)) - sin(q_{1op})) \\
&-(\omega_n^2\theta_1)(q_1(t) - q_{1op}) - (2\varsigma\omega_n\theta_1 - \alpha)q_2(t).
\end{aligned} \tag{28}
$$

Using Algorithm 3, the nonlinear control law by exact linearization technique is obtained as:

$$
\begin{aligned}
u(t) &= \theta_2 sin(q_1(t)) - (\omega_n^2\theta_1)(q_1(t) - q_{1op}) \\
&+(-2\varsigma\omega_n\theta_1 + \alpha)q_2(t).
\end{aligned} \tag{29}
$$

Note that (28) and (29) are the same. This is a particular case, and it is because the simple pendulum nonlinearity isn't too complex.

4 Implementation of the Numerical Non-Standard Discretization Methods on the Simple Pendulum

The control laws are designed in continuous-time. To implement them in a digital device (as the computer shown in Fig. 2) it is then necessary to discretize them by the numerical standard (used for the linear control law) or non-standard discretization methods (used for the nonlinear control law).

The discrete-time linear control law of the continuous-time linear control law (27), using the standard discretization methods is given by:

$$
\begin{aligned}
u(kT_0) &= -(\omega_n^2\theta_1 - \theta_2 cos(q_{1op}))(q_1(kT_0) - q_{1op}) \\
&+\theta_2 sin(q_{1op}) - (2\varsigma\omega_n\theta_1 - \alpha)q_2(kT_0).
\end{aligned} \tag{30}
$$

Observe that in this case, all three standard discretization methods give the same discrete-time equation (30). This is because (27) is a linear dynamic equation of order cero. The difference of using one standard discretization method or another, in the states feedback linear control law, will be reflected in the computation of the discrete-time gains designed for the discrete-time model (2).

The extended nonlinear discrete-time control law of (29), using the Euler discretization method is:

$$
\begin{aligned}
u(kT_0) &= \theta_2 sin(q_1(kT_0)) - (\omega_n^2\theta_1)(q_1(kT_0) - q_{1op}) \\
&+(-2\varsigma\omega_n\theta_1 + \alpha)q_2(kT_0).
\end{aligned} \tag{31}
$$

This is the same discrete-time control law for (28).

Because the state feedback nonlinear control law (29) is a nonlinear dynamic of cero order, the Euler and the Euler-Picard discretization methods are the same, i.e. no need to apply Picard iteration.

Using the Euler-Taylor-Picard discretization method, the following nonlinear discrete-time control law is:

$$
\begin{aligned}
u(kT_0) &= \theta_2 F_j(kT_0) - (\omega_n^2\theta_1)(q_1(kT_0) - q_{1op}) \\
&+(-2\varsigma\omega_n\theta_1 + \alpha)q_2(kT_0),
\end{aligned} \tag{32}
$$

where, in this particular case with $j = 7$ we get a good approximation of $F_j(kT_0) = sin(q_1(kT_0))$, i.e,

$$
\begin{aligned}
F_7(kT_0) &= q_1(kT_0) - \frac{q_1^{\,3}(kT_0)}{6} + \frac{q_1^{\,5}(kT_0)}{120} - \frac{q_1^{\,7}(kT_0)}{7!} \\
&+ \frac{q_1^{\,9}(kT_0)}{9!} - \frac{q_1^{\,11}(kT_0)}{11!} + \frac{q_1^{\,13}(kT_0)}{13!} - \frac{q_1^{\,15}(kT_0)}{15!}.
\end{aligned}
$$

The discrete-time linear and nonlinear control laws are implemented on the real mechanical system, the simple pendulum, showed in Fig. 1, through a program codified in the dSPACE system. The dSPACE is a computational tool that allows us to do real time control. It is based on a digital signal processor implanted on a DS1102 card (for data acquisition) which is then attached to the PCI bus on a PC [12][13], as showed in Fig 2.

The simple pendulum has a sensor to measure position in radian, considered the output signal, $q_1(t)$. The velocity of the pendulum, $q_2(t)$, to be used in the state feedback control laws, is estimated by,

$$
q_2(kT_0) = \frac{q_1(kT_0) - q_1((k-1)T_0)}{T_0}.
$$

The sampling time T_o was obtained using the Shannon Theorem [3]-[5] and considering the hardware limitation of the closed-loop system showed in Fig 2.

4.1 Validation of the numerical non-standard discretization methods

The validation of the numerical non-standard discretization methods is analyzed by comparing the simulated (continuous-time and discrete-time) controlled system dynamics vs. the obtained data from the controlled real nonlinear system dynamic.

To show the validity of the numerical non-standard discretization methods, a suitable operational point (for the linear and nonlinear control laws to be applied) was chosen to show the results.

We first compare the simulated continuous linear system dynamic (26) controlled using the designed linear continuous state feedback control law (27), and the simulated discrete-time linear system dynamic of the form given in (2) controlled using the discretized linear state feedback control law (30) with the obtained data from the pendulum using the implemented control law (30).

Matrices A_{dm} and B_{dm} are given from (3), (4) or (5), depending on the used standard discretization method.

The results are shown in Fig. 4, 5 and 6. The figures show the dynamic pendulum position for the operational point $q_{1op} = \pi/12$ rad ($q_{1op} = 15^\circ$), initial conditions are $q_1(0) = 0$, $q_2(0) = 0$, using the sampling period as $T_o = 0.01$ s and the order for the Taylor's series is $j = 1$.

Then, we compare the simulated continuous nonlinear system dynamic (24) controlled using the designed continuous state feedback linear control law (27) with the obtained data from the pendulum using the implemented linear control law (30), as shown in Fig. 7. The linear control gains for the control law (30) ware computed based on matrices A_{de} and B_{de} in (4).

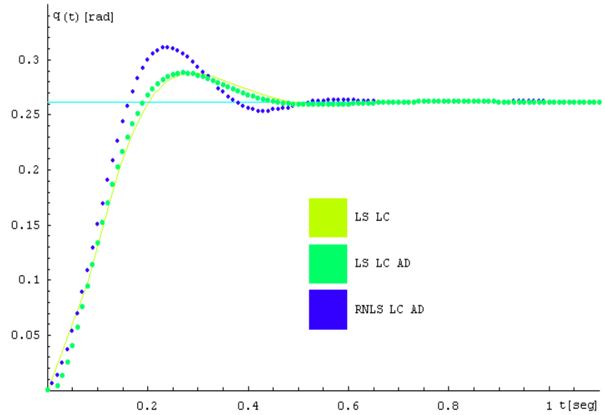

Fig.4 LS LC: simulated linear system controlled by the linear control law (in continuous). LS LC AD: simulated linear system controlled by the linear control law, using Approximate discretization. RNLS LC AD: real nonlinear system controlled by the linear control law, using Approximate discretization.

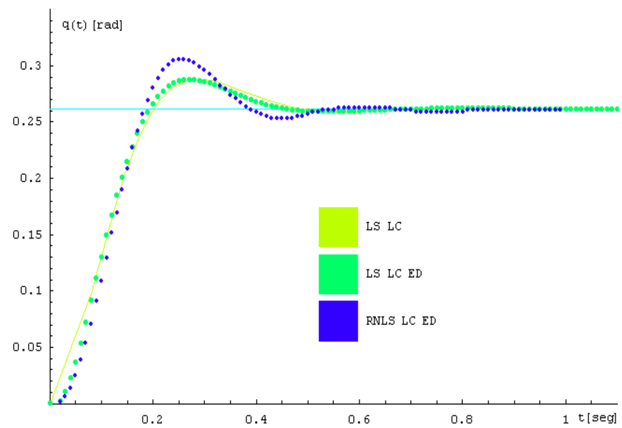

Fig.5 LS LC: simulated linear system controlled by the linear control law (in continuous). LS LC ED: simulated linear system controlled by the linear control law, using Exact discretization. RNLS LC ED: real nonlinear system controlled by the linear control law, using Exact discretization.

For the Euler and Euler-Picard cases, we compare the simulated continuous nonlinear system dynamic (24) controlled using the designed nonlinear continuous state feedback control law (29), and the simulated discrete-time nonlinear system dynamic of the form given in (8) (or (9) because, for this dynamic, no need to apply Picard iteration), controlled using the discretized nonlinear state feedback control law (31) with the obtained data from the pendulum using the implemented control law (31).

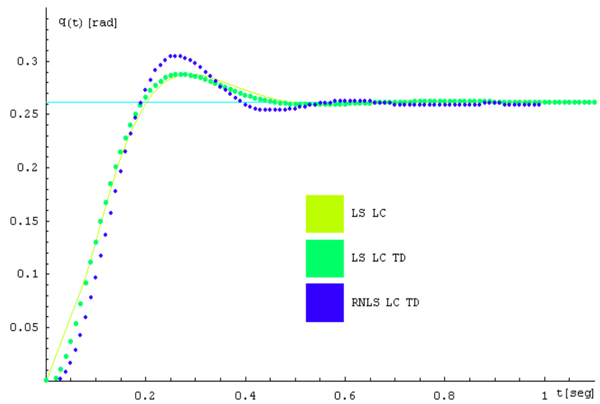

Fig.6 LS LC: simulated linear system controlled by the linear control law (in continuous). LS LC TD: simulated linear system controlled by the linear control law, using Truncated discretization. RNLS LC TD: real nonlinear system controlled by the linear control law, using Truncated discretization.

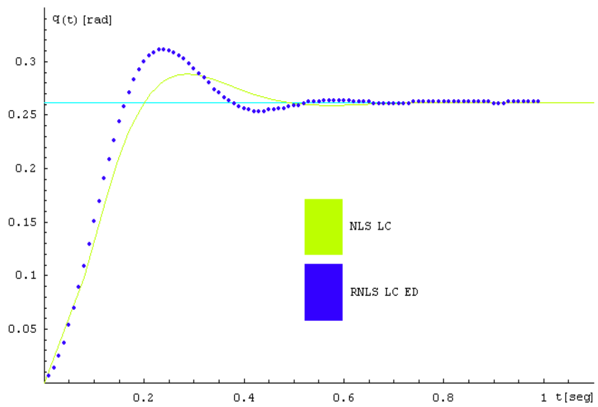

Fig.7 NLS LC: simulated nonlinear system controlled by the linear control law (in continuous). RNLS LC ED: real nonlinear system controlled by the linear control law, using Exact discretization.

Similarly, for the Euler-Taylor-Picard case, we compare the simulated continuous nonlinear system dynamic (24) controlled using the designed nonlinear continuous state feedback control law (29), and the simulated discrete-time nonlinear system dynamic of the form given in (10) controlled using the discretized nonlinear state feedback control law (32) with the obtained data from the pendulum using the implemented control law (32).

The results are shown in figures 8 and 9. Once again, the figures show the dynamic pendulum position for the operational point $q_{1op} = \pi/12$ rad ($q_{1op} = 15^\circ$), initial conditions are $q_1(0) = 0$, $q_2(0) = 0$ and using $T_o = 0.01$ s. Because the Euler and Euler-Picard methods are the same in this case, the order of the Picard iteration is 1. For the Euler-Taylor-Picard method $j = 7$ is used.

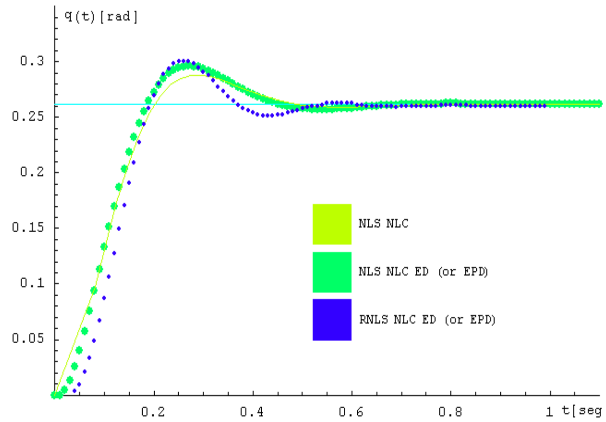

Fig.8 NLS NLC: simulated nonlinear system controlled by the nonlinear control law (in continuous). NLS NLC ED (or EPD): simulated nonlinear system controlled by the nonlinear control law, using Euler (or Euler-Picard) discretization method. RNLS NLC ED: real nonlinear system controlled by the nonlinear control law, using Euler (or Euler-Picard) discretization.

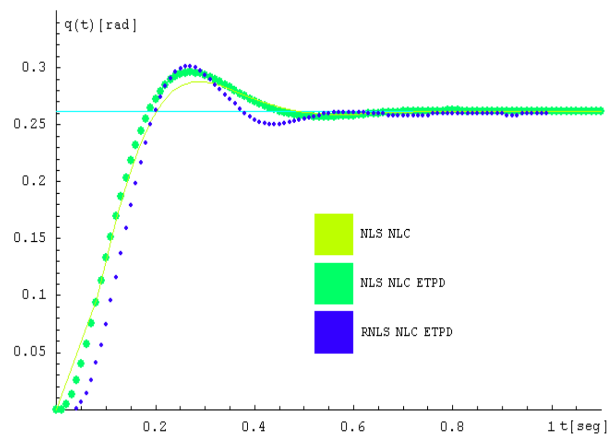

Fig.9 NLS NLC: simulated nonlinear system controlled by the no linear control law (in continuous). NLS NLC ETPD: simulated nonlinear system controlled by the nonlinear control law, using Euler-Taylor-Picard discretization method. RNLS NLC ED: real nonlinear system controlled by the nonlinear control law, using Euler-Taylor-Picard discretization.

As it is shown in the figures, from Fig. 4 to Fig. 9, all the real nonlinear system dynamics (controlled by the linear or the nonlinear control law) present quite the same performance (only in the approximate discretization, the overshot is a little bit higher). This is because the linear control law is into the linear range of operation, and the sample period is small enough.

The real data obtained from the controlled system presents a higher overshot than the continuous simulated one (linear and nonlinear cases), and this is better approximated by discrete-time simulation when the nonlinear system and the nonlinear control law are discretized using the numerical non-standard discretization methods, as it is shown in Fig. 8 (NLS NLC ED (or EPD)) and Fig. 9 (NLS NLC ETPD).

The real (obtained data) system dynamics present a short delay compared with the continuous simulated dynamics. This delay is not taken into consideration in the simple pendulum discrete-time model, and it is produced by human time reaction at the moment to start measuring the data (pressing a button to start capturing the data after the algorithm is run, the button also works as a security button). Because of this time delay, the real dynamic presents a small error during the first 0.2s in the transient state.

4.2 Digital nonlinear control laws implementation advantages

The numerical non-standard discretization methods ware also validated for different operational points (desired positions) of the simple pendulum. These operational points ware chosen to be more far away from the initial position, which is kept in the stable equilibrium point of the system $q_1(0)=0,\ q_2(0)=0$.

For the real experiment, the linear and nonlinear gains for the control law are computed. The chosen particular position to compute the gains is $q_{1op}=\pi/12=0.261$ rad, ($q_{1op}=15^\circ$). Keeping the computed gains as constant, the desired position of the pendulum, q_{1op}, is changed over a range of values.

In Table 1 the obtained results, using the discrete-time linear and nonlinear control laws, are presented. The desired position range for the pendulum is chosen from $q_{1op}=5\pi/180=0.087$ rad ($q_{1op}=5^\circ$) to $q_{1op}=5\pi/18=0.873$ rad ($q_{1op}=50^\circ$).

When the discrete-time linear control law is used, the real pendulum position is greater than the desired position, for desired positions under $q_{1op}=\pi/12=0.261$ rad. This is because the gains were computed for the desired position $q_{1op}=\pi/12=0.261$ rad, therefore for desired positions under $q_{1op}=\pi/12=0.261$ rad the gains give more energy to the controller than the necessary.

Table 1. Real pendulum position, using the discrete-time linear and nonlinear control laws, for different operational points (desired positions).

Desired Positions	Obtained Positions (Real Data)			
Radians	Approx.	Exact	Trunc.	Euler and Euler-Picard
0.087	0.174	0.172	0.173	0.086
0.174	0.218	0.215	0.215	0.173
0.261	0.260	0.260	0.261	0.260
0.349	0.305	0.305	0.306	0.349
0.436	0.351	0.353	0.352	0.435
0.523	0.398	0.396	0.399	0.524
0.611	0.442	0.441	0.441	0.612
0.698	0.487	0.489	0.489	0.710
0.785	0.536	0. 539	0.538	0.786
0.873	0.588	0.592	0.591	0.886

On the contrary, for desired positions over $q_{1op}=\pi/12=0.261$ rad, when the discrete-time linear control law is used, the real pendulum position is smaller than the desired position. This is because now the computed gains for $q_{1op}=\pi/12=0.261$ rad give less energy to the controller than the necessary.

Analyzing the obtained results in Table 1, when the simple pendulum is controlled using the nonlinear control law, discretized by the Euler and Euler – Picard methods, the pendulum is stabilized in a larger range of desired position without having error or with a minimum error (less than 1%) in the steady state. When the linear control law is used the pendulum is positioned in the desired position without having error (or minimum error) in steady state only for the desired potion $q_{1op}=\pi/12=0.261$ rad, and for all others desired positions the presented error in steady state is considerable.

It is then easy to ratify that for nonlinear systems it is better to use a nonlinear controller, which can be implemented, if the desired position may change. For the linear control law to perform well in this situation, the gain must be computed for each desired position. The numerical nonstandard discretization methods allow the nonlinear controller to be implemented digitally on a computer or a microcontroller.

In a second experiment, the initial condition is also kept in the stable equilibrium position ($q_1(0)=0,\ q_2(0)=0$), and the linear control gains are computed for each desired position of the simple pendulum. As the nonlinear gains don't depend on the desired position no gain calculation for each position is necessary. The nonlinear gains remain constant while the desired position changes.

In Table 2 the obtained results for the experience using the linear control laws are showed. In the experiment, the desired position is changed by increasing the desired position, q_{1op} ,five degrees ($\pi/36 = 0.087$ rad) for each experience. The whole experience rage is chosen from $q_{1op} = 5\pi/180 = 0.087$ rad ($q_{1op} = 5^\circ$) to $q_{1op} = 75\pi/180 = 1.309$ rad ($q_{1op} = 75^\circ$).

Table 2. Real simple pendulum position using the linear discrete-time control laws

Desired Positions	Obtained Positions (Real Data)		
Radians	Approx.	Exact	Trunc.
0.087	0.087	0.087	0.085
0.174	0.174	0.174	0.173
0.261	0.262	0.260	0.259
0.349	0.348	0.349	0.348
0.436	0.445	0.442	0.443
0.523	0.529	0.526	0.527
0.785	0.793	0.793	0.793
0. 873	0. 887	0. 888	0. 887
1.047	1.066	1.065	1.064
1.222	1.262	1.261	1.261
1.309	Unstable	Unstable	Unstable

The linear control is able to stabilizes the simple pendulum position to de desired position in a range from $\pi/36 = 0.087$ rad (5°) to $\pi/9 = 0.349$ rad (20°) with no error (or minimum error, less than 1%), showing good performance. From $25\pi/180 = 0.436$ rad (25°) to $7\pi/18 = 1.222$ rad (70°) the position error is between $1\% < error < 7\%$, and for positions over $75\pi/180 = 1.309$ rad (75°) the controlled system is unstable.

The obtained results for the experience using the nonlinear control laws are showed in Table 3. The chosen rage is from $q_{1op} = 5\pi/180 = 0.087$ rad ($q_{1op} = 5^\circ$) to $q_{1op} = 12\pi/18 = 2.094$ rad ($q_{1op} = 120^\circ$). Same as before, the desired position is changed by increasing the desired position five degrees ($\pi/36 = 0.087$ rad) for each experience.

In this case, the nonlinear control stabilizes the simple pendulum position to de desired position in a larger range. From $\pi/36 = 0.087$ rad (5°) to $45\pi/180 = 0.785$ rad (45°) with no error (or minimum error , less than 1%). From $5\pi/18 = 0.873$ rad (50°) to $75\pi/180 = 1.309$ rad (75°) the error is in the interval $1\% \le error < 5\%$. From $4\pi/9 = 1.396$ rad (80°) to $115\pi/180 = 2.007$ rad (115°) the error is $5\% \le error < 15\%$, and from $12\pi/18 = 2.094$ rad (120°) the controlled system is unstable.

Table 3. Real simple pendulum position using the nonlinear discrete-time control laws

Desired Positions	Obtained Positions (Real Data)	
Radians	Euler and Euler-Picard	Euler-Taylor-Picard
0.087	0.086	0.088
0.174	0.173	0.173
0.261	0.260	0.260
0.349	0.349	0.349
0.436	0.435	0.435
0.523	0.524	0.524
0.785	0.786	0.787
0. 873	0. 886	0. 888
0.960	0.992	0.993
1.047	1.065	1.066
1.134	1.160	1.160
1.222	1.258	1.256
1.309	1.346	1.347
1.396	1.444	1.439
1.484	1.547	1.537
1.571	1.640	1.635
1.658	1.733	1.742
1.745	1.836	1.837
1.833	1.939	1.941
1.920	2.030	2.042
2.007	2.133	2.106
2.094	Unstable	Unstable

From the results showed in Table 2 and Table 3, the controlled simple pendulum shows good performance (without or with minimum error in steady state) over a larger range of operation when the nonlinear discrete-time control laws are used (discretized by the numerical non-standard discretization methods), and also shows better performance (with small error in steady state) over a larger range of operation than the discrete-time linear control laws. The range of stabilization is much larger when the nonlinear control laws are used; this is an important advantage in unstable mechanical systems.

It is important to mention that the control analysis showed in sections 4.1 and 4.2 for the pendulum position is similar for the pendulum velocity dynamics

5 Conclusions

Through this work we have validated, by implementation on a real nonlinear mechanical system, the simple pendulum, the numerical non-standard discretization methods: Euler, Euler-Picard and Euler-Taylor-Picard.

Into a linear operational range, the implemented discrete-time controllers using the numerical non-standard discretization methods perform quite same as the implemented discrete-time controller using the standard methods. However, the non-standard discretization methods have the advantage to be applied to discretize linear and nonlinear dynamics in general.

The simulated nonlinear system, controller by the nonlinear control law, discretized using the non-standard methods has a better approximation to the obtained real data in any case.

It is important to highlight that the nonlinear gains for the nonlinear controller used in all the experience are constant, in other words, the nonlinear gains don't depend on the operational point (desired position), and this property allows the nonlinear controller to stabilize the simple pendulum in a larger operational range in steady state. This gives an important advantage and robustness over the use of linear control laws.

The operational range where the simple pendulum position shows no error (or minimal error, less than 1%) is larger when the nonlinear discrete-time control laws are used, discretized by the numerical non-standard discretization methods.

With this contribution, it is expected that the proposed numerical non-standard discretization methods may be more used for the discretization and digital implementation of nonlinear dynamics.

References:

[1] J. Rodríguez-Millán, *Sistemas Dinámicos No Lineaeles: Teoremas Fundamentales, Cuaderno de Control 97-1*, Universidad de Los Andes, Maestría en Ingeniería de Control y Automatización, Merida, 1997.

[2] H. Sira, Márquez, R., and Rivas, F., *Control de Sistemas no Lineales*, Pearson Prentice-Hall, 1997.

[3] R. Isermann, *Digital Control Systems (Vol I). Fundamentals, Deterministic Control*, Springer-Verlag, 1989.

[4] K. Ogata, *Discrete-Time Control Systems*, Prentice-Hall, 1995.

[5] G. Goodwin, S. Graebe, and M. Salgado, *Control System Design*, Prentice-Hall, 2001.

[6] J. Rodríguez-Millán, and C. González, Three Mathematica Supported Proposals for the Discretization of Nonlinear Control Systems, *Nonlinear Analysis*, Vol. 63, 2005, pp. e617-e628. (Available online at www.sciencedirect.com).

[7] J. Rodríguez-Millán, González, C., and Patete, A., Improved Non-standard Discretization Methods for Nonlinear Dynamical Control Systems, *Computer Aided Systems Theory, LNCS, 3643*, Springer-Verlag, 2005, pp. 608-613.

[8] J. Rodríguez-Millán, A. Patete, and C. González, Picard Discretization of Nonlinear Systems: Symbolic or Numeric Implementation?, *Computer Aided Systems Theory, LNCS 4739*, Springer-Verlag, 2007, pp. 121-129.

[9] Patete A, Velasco M, y Rodríguez-Millán J, 2010, Validation of the non-Standard discretization methods case of study: the simple pendulum, 9th WSEAS International Conference on Computational Intelligence, Man-Machine Systems and Cybernetics (CIMMACS '10), pp. 123-130.

[10] E. Coddington, and N. Levinson, *Theory of Ordinary Differential Equations*, Tata McGraw-Hill Publishing Co., 1977.

[11] Domínguez S, Campoy P, Sebastián J, y Jiménez A, 2006, Control en el espacio de estado, Pearson Prentice-Hall, Madrid, 2006.

[12] P. Lischinsky, C. Bravo, and G. González, Modelado y Estimación de Parámetros del Pendubot, *Memorias del Coloquio de Automatización y Control CAC'2001*, 2001.

[13] G. González, and P. Lischinsky, Desarrollo de Experiencias de Modelado y Control a Tiempo Real de un Péndulo, *IV Congreso de Automatización y Control CAC'2003*, 2003.