A Second Order Approximation for DEVS Simulation of Continuous Systems

Ernesto Kofman
Laboratorio de Sistemas Dinámicos - FCEIA - Universidad Nacional de Rosario
Riobamba 245 bis - (2000) Rosario - Argentina
Email: ekofman@eie.fceia.unr.edu.ar

Abstract

In this paper, based on the methodology of discrete event simulation of continuous systems via Quantized State Systems (QSS), a new second order approximation, called *Second-Order Quantized State Systems* (QSS2), is proposed. This new approximation, which satisfies the same stability and convergence properties that were deduced for QSS in previous works, also allows to reduce the number of calculations with respect to the former method. It is shown that in the particular case of Linear Time-Invariant (LTI) Systems, the QSS2 can be exactly represented by a DEVS model and in nonlinear systems an approximated DEVS model can be also obtained. For the LTI case, a closed formula giving the necessary quantization that allows achieving a bound in the error during the whole simulation is deduced. This formula, which stands for both QSS and QSS2 approaches, also proves that for any quantization the error is always bounded.

Finally, all the mentioned results are verified and illustrated through the simulation of linear and nonlinear examples, where some advantages over the classic discrete time methods can be observed.

Keywords: Simulation of ODE's, Discrete Event Simulation, DEVS, Quantized State Systems.

1 Introduction

The complexity that carries the analitical resolution of differential equations and the impossibility of obtaining closed expresions for their solutions, except in Linear Time Invariant (LTI) systems and a very small class of nonlinear systems, has motivated the development of several numerical methods to obtain approximate solutions.

The basic tool used by the classic numerical methods is the discretization of the time variable. This discretization can be either constant or variable and according to this feature, the algorithms are called *fixed step* or *variable step* methods respectively (Press et al., 1986). In both cases, the resulting simulation model constitutes a *Discrete Time System.*

Recently, different methods have been being developed where the time discretization is avoided. As a result, instead of arriving to *Discrete Time* they arrive to *Discrete Event* simulation models within the DEVS formalism framework (Zeigler, 1976), (Zeigler et al., 2000).

One possible way to obtain discrete event simulation models is replacing the time discretization by the quantization of the continuous system state variables. This is the idea followed by Zeigler (Zeigler and Lee, 1998) and the definition of *Quantized Systems.* Although this method implies a formal transformation of a continuous model into a discrete event system, it has a problem related to the presence of an infinite number of events in a finite interval of time. This difficulty was solved with the addition of hysteresis in the quantization and the definition of Quantized State Systems in (Kofman and Junco, 2000).

A QSS can be obtaining from a continuous system after adding it *quantizers* with hysteresis which transform the state trajectories into piecewise constant functions so that they can be represented by a sequence of events carrying their successive values. In the mentioned work it is shown that QSS can be exactly simulated by DEVS models. As it is also proven there, the QSS also conserve some stability properties of the original systems. The simulation with QSS shows some advantages with respect to classical discrete time algorithms related to the reduction of the computational costs and the possibilities of dealing with hybrid systems under a unique formalism. Another important feature of QSS is the capability of decoupling certain classes of structural singularities that are very common in systems of equations derived from object oriented modelling paradigms such as Bond Graphs (Kofman and Junco, 2001).

Despite these advantages, the QSS simulation cannot achieve a good accuracy without a significant increase in the number of calculations. This is due to the fact that the method performs only a first order approximation.

In this work, following the line opened with the definition of QSS and the proof of their stability properties, a second order approach is presented and the mentioned properties are studied in LTI systems in order to obtain closed formulas relating the quantization and the bounds of the error committed during the simulation.

The second order approach is based on the introduction of *First Order Quantizers.* These quantizers give piecewise linear output trajectories, which are represented by sequences of events carrying their successive values and slopes. At this point, Giambiasi's work must be mentioned, since his definition of GDEVS (Giambiasi et al., 2000) gives a way of representing piecewise polynomial trajectories by segments of events.

The paper is organized as follows. In Section 2 the main definitions and properties of QSS are recalled. Then, the second order approximation (QSS2) is introduced in Section 3 and its main properties are deduced. After that, the DEVS model related to a QSS2 is built in Section 4. This DEVS model is presented as the coupling of submodels corresponding to integrators and static functions.

In a previous general study of stability an algorithm to obtain the quantization that guarantees a desired bound in the final error was deduced. In Section 5, working with LTI system, that algorithm is replaced by a closed formula giving the quantization that assures achieving a desired bound in the error not only at the end but also during the whole simulation.

Finally, in Section 6, the results are applied to the simulation of linear and nonlinear examples

2 Quantized State Systems

Quantized State Systems (QSS) are continuous time systems where each state variable is affected by a quantization function equipped with hysteresis.

Before recalling the formal definition of QSS, the concept of *quantization function with hysteresis* will be introduced.

2.1 Quantization Functions

Let $D = \{d_0, d_1, ..., d_r\}$ be a set of real numbers where $d_{i-1} < d_i$ with $1 \le i \le r$ and let $x \in \Omega$ be a continuous trajectory, where $x : \mathbb{R} \to \mathbb{R}$. Let $b : \Omega \times t_0 \to \Omega$ be a mapping and let $q = b(x, t_0)$ where the trajectory q for $t \ge t_0$ satisfies

$$q(t) = \begin{cases} d_m & \text{if } t = t_0 \\ d_{i+1} & \text{if } x(t) = d_{i+1} \wedge q(t^-) = d_i \wedge i < r \\ d_{i-1} & \text{if } x(t) = d_i - \varepsilon \wedge q(t^-) = d_i \wedge i > 0 \\ q(t^-) & \text{otherwise} \end{cases} \tag{1}$$

and

$$m = \begin{cases} 0 & \text{if } x(t_0) < d_0 \\ r & \text{if } x(t_0) \ge d_r \\ j & \text{if } d_j \le x(t_0) < d_{j+1} \end{cases}$$

Then, the map b is a *Quantization Function with Hysteresis*. The width of the hysteresis window is ε. The values d_0 and d_r are the lower and upper saturation values. Figure 1 shows a typical quantization function with uniform quantization intervals.

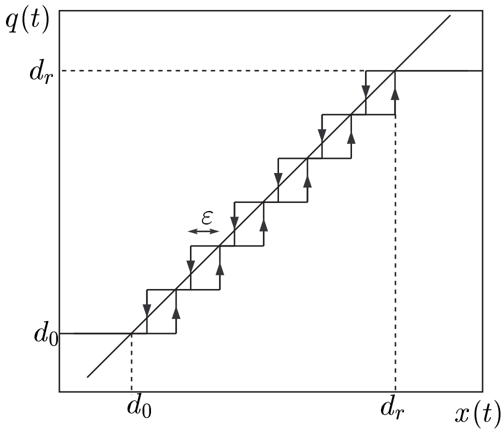

Figure 1: Quantization Function with hysteresis

A fundamental property of a Quantized Function with hysteresis is given by the following inequality.

$$d_0 \leq x(t) \leq d_r \Rightarrow |q(t) - x(t)| = |b(x(t)) - x(t)| \leq \max_{1 \leq i \leq r} (d_i - d_{i-1}, \varepsilon) \tag{2}$$

2.2 QSS related to a State Equation System

Consider the State Equation System given by:

$$\begin{cases} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= g(x(t), u(t)) \end{cases} \tag{3}$$

This is a typical representation of Differential Equation Systems, where the components of the vectors x, u and y are called state, input and output variables, respectively. This kind of equations allows the representation of most continuous systems.

Related to this system, an associated QSS is defined as follows:

$$\begin{cases} \dot{x}(t) &= f(q(t), u(t)) \\ y(t) &= g(q(t), u(t)) \end{cases} \tag{4}$$

where $q(t)$ and $x(t)$ are related (componentwise) by quantization functions with hysteresis. The components of the vector $q(t)$ are called *quantized variables*. Figure 2 shows a block diagram of a QSS.

2.3 Properties of QSS

The most significant properties of the QSS are related to the form of the trajectories. Provided that the inputs have piecewise constant trajectories and the function f is continuous and bounded in any bounded domain, the following properties are satisfied:

- The quantized variables have piecewise constant trajectories
- The state variable derivatives have also piecewise constant trajectories

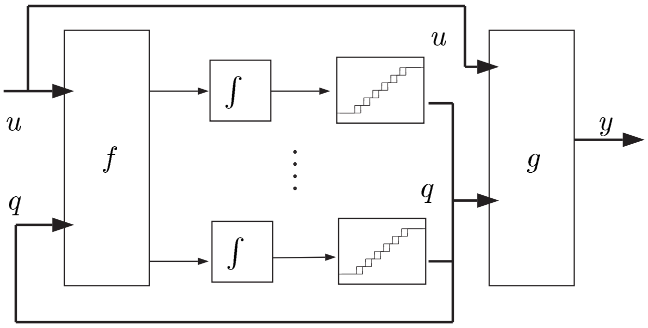

Figure 2: Block Diagram of a generic QSS

- The state variables have continuous piecewise linear trajectories

Figure 3 shows typical trajectories in a QSS.

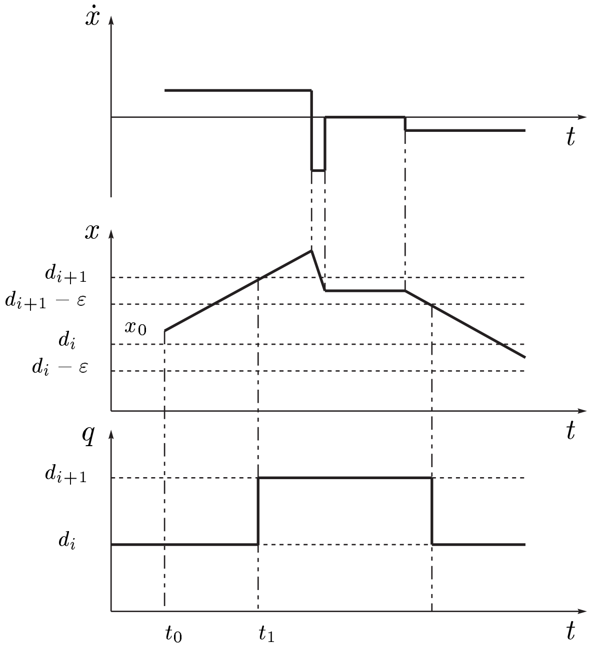

Figure 3: Typical trajectories in a QSS

As a consequence of these properties the QSS can be exactly simulated by a discrete event model within the DEVS formalism framework.

QSS have other important convergence and stability properties that will be recalled later. The DEVS model related to a QSS and the proof of the mentioned properties can be found in (Kofman and Junco, 2000). That DEVS model also shows formaly the way of exactly simulating a QSS.

3 The Second Order Approximation

As it was mentioned above, the approach of QSS constitutes a first order simulation method. Thus, it does not allow to achieve a good accuracy without increasing considerably the number of calculations.

This section introduces a new method that results in a second order approximation and conserves most of the properties that where shown for QSS.

3.1 The First Order Quantizer

In most applications of QSS, the distance between quantization levels is taken equal to the hysteresis width. The reason for this choice is that both terms must be taken as big as possible in order to reduce the computational costs but the error introduced by the quantization is bounded by the maximum between them, as it can be deduced from equation (2). Thus, the best solution is taking them equal to each other.

When a quantizer has the same hysteresis and quantum size, it can be seen as a function that produces a piecewise constant output trajectory that changes when the difference with the input value reaches some threshold. This threshold is given by the quantum size or the hysteresis width.

Using this idea, a new *first order* quantization function can be defined as a function that gives a piecewise linear output trajectory, whose value and slope change when the difference between this output and the input becomes bigger than certain threshold. Figure 4 illustrates this idea. An output line starts from the input value

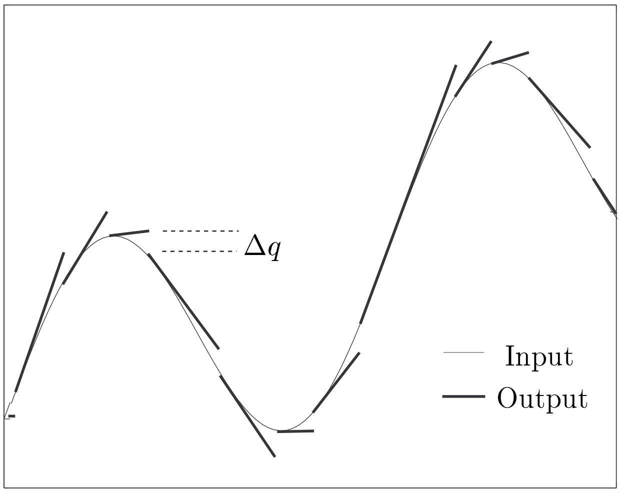

Figure 4: Input and Output trajectories in a *First Order* quantizer

with some slope and then, when that line and the input trajectory differ from each other in a quantity Δq, the output is represented by a new line starting from the new input value.

There are several possibilities to choose the slope of the output lines, but only two of them will be considered here. The first one is to take the derivative of the input trajectory as the slope of the output line (Figure 5a). The second possibility is to use a previous output value to *extrapolate* the line (Figure 5b).

Although the first choice constitutes a better approximation and it is easier to implement, the second one also have some advantages because in this case the information about the slope can be obtained looking at the successive values that define the trajectory while in the first case, the value of the slope has to be transmitted to the rest of the system.

Then, for both approaches, the relationship between the input $x(t)$ and the output $q(t)$ of a first order quantizer can be writen as follows:

$$q(t) = \begin{cases} x(t) & \text{if} \quad t = t_0 \vee |q(t^-) - x(t^-)| = \Delta q \\ q(t_j) + m_j(t - t_j) & \quad \text{otherwise} \end{cases} \tag{5}$$

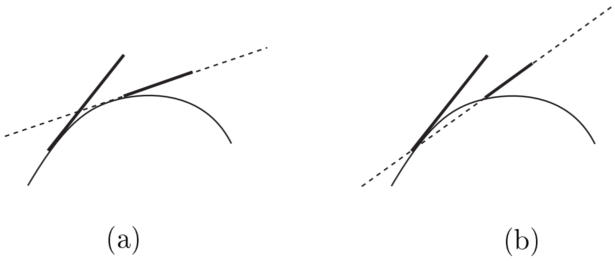

Figure 5: Different possibilities for the choice of the slope

with the sequence $t_0, \ldots, t_j, \ldots$ defined as

$$t_{j+1} = \min(t | t > t_j \wedge |x(t_j) + m_j(t - t_j) - x(t)| = \Delta q) \tag{6}$$

The slopes, considering the first possibility (i.e. taking the derivative of the input variable), can be calculated according to

$$m_0 = 0,\ m_j = \dot{x}(t_j^-) \quad j = 1, \ldots, k, \ldots \tag{7}$$

while for the second choice we define

$$m_0 = 0,\ m_j = \frac{x(t_j) - x(t_{j-1})}{t_j - t_{j-1}} \quad j = 1, \ldots, k, \ldots \tag{8}$$

When two trajectories $x(t)$ and $q(t)$ satisfy the relationships given by the previous equations we will say that those variables are related by a *first order quantization function*. A fundamental property of a first order quantization function is given by the following inequality:

$$|x(t) - q(t)| \leq \Delta q \qquad \forall t \geq t_0 \tag{9}$$

3.2 Second-Order Quantized State Systems (QSS2)

The addition of quantizers at the output of the continuous system integrators transforms it into a QSS. When first order quantizers are used instead of zero order quantizers, the resulting systems will be called *Second Order Quantized State Systems* or QSS2.

These new systems can be still represented by equation (4), but now the components of $x(t)$ and $q(t)$ are related by first-order quantization functions. Figure 6 shows a block diagram representation of QSS2.

The QSS2 properties are quite similar to the properties of QSS. In fact, the stability and convergence proofs in (Kofman and Junco, 2000) are still satisfied by QSS2 since they are based on the representation (4) (which holds for QSS2) and the inequality (2), which is verified by the QSS2 with the form of the inequality (9).

However, there is an important difference between QSS and QSS2 related to the trajectory forms. In general QSS2, all we can claim is that the quantized variables $q(t)$ will have piecewise linear trajectories. But in QSS2 related to linear systems the state trajectories will be piecewise parabolic with piecewise linear derivatives.

Due to this reason, it will be possible to find DEVS models that exactly represent the behaviour of QSS2 related to generic linear time invariant systems. Unfortunately, QSS2 related to general nonlinear systems do not have exact representation in terms of DEVS. However, as it will be explained later, the method can also be applied in the simulation of these nonlinear cases.

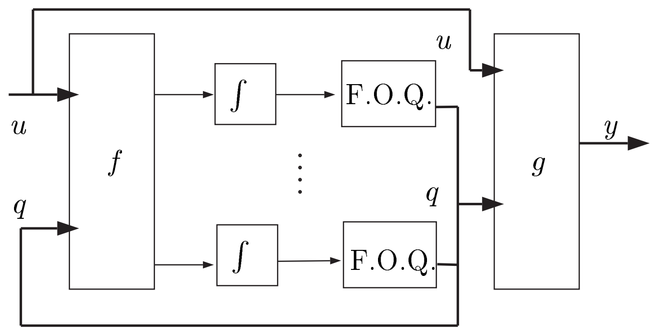

Figure 6: Block Diagram of a generic QSS2

3.3 Trajectories in a QSS2

It was mentioned in the previous paragraphs that the trajectories of the quantized variables of a QSS2 are piecewise linear. It was also said that in linear time invariant systems the state derivatives have piecewise linear trajectories while the state derivatives have piecewise parabolic trajectories. The following theorems give sufficient conditions and prove these properties. Although we will deal with the first choice of the slope (7) all the properties deduced here can be easily extended for the other case.

Theorem 1. *Given the QSS2 defined in* (4) *with f continuous and bounded in any bounded domain and $u(t)$ being bounded, the trajectories of $q(t)$ are piecewise linear while they remain inside a bounded region.*

Proof. Let $q_i(t)$ and $x_i(t)$ denote the trajectory of the i-th component of $q(t)$ and $x(t)$ respectively. Since $x_i(t)$ and $q_i(t)$ are related by a first order quantization function, the last trajectory can be written as

$$q_i(t) = \begin{cases} x_i(t) & \text{if} \quad t = t_0 \vee |q(t^-) - x(t^-)| = \Delta q \\ q_i(t_j) + m_j(t - t_j) & \quad \text{otherwise} \end{cases} \tag{10}$$

with the sequence $t_0, \ldots, t_j, \ldots$ defined according to

$$t_{j+1} = \min(t | t > t_j \wedge |x_i(t_j) + m_j(t - t_j) - x_i(t)| = \Delta q_i) \tag{11}$$

and where, according to (7), we have

$$m_0 = 0, \; m_j = \dot{x}_i(t_j^-) \quad j = 1, \ldots, k, \ldots \tag{12}$$

Although equation (10) implies that $q_i(t)$ is formed by sections of lines, in order to assure that it is piecewise linear it is also necessary to prove that the sequence $t_0, \ldots, t_j, \ldots,$ does not contain an infinite number of components in a finite interval of time.

Since it was assumed that $q(t)$ is bounded over some interval of time $[t_0, t_f]$, and taking into account the hypothesis made about f and the fact that $u(t)$ is bounded, we have

$$|f_i(q(t), u(t))| \leq F_i \qquad t_0 \leq t \leq t_f \tag{13}$$

From (12) we have

$$|m_j| \leq |f_i(q(t), u(t))| \leq F_i \tag{14}$$

Thus, the slope results always bounded by a constant F_i. From (11) we have

$$\begin{aligned}\Delta q_i &= |x_i(t_j) + m_j(t_{j+1} - t_j) - x_i(t_{j+1})| \\ &\leq \frac{|x_i(t_{j+1}) - x_i(t_j)|}{t_{j+1} - t_j}(t_{j+1} - t_j) + |m_j|(t_{j+1} - t_j) \\ &\leq |m_{j+1}|(t_{j+1} - t_j) + |m_j|(t_{j+1} - t_j) \\ &\leq 2F_i(t_{j+1} - t_j)\end{aligned}$$

Finally, it results that

$$t_{j+1} - t_j \geq \frac{\Delta q_i}{2F_i} \tag{15}$$

This inequality implies that the sequence $t_0, \ldots, t_j, \ldots,$ has a minimum time between successive components and it is impossible to have an infinite number of components in a finite interval of time. As a consequence of this, the trajectories of $q(t)$ are piecewise linear. □

Theorem 2. *In a QSS2 related to a LTI system with bounded piecewise linear inputs, the trajectories of the derivatives of the state variables are piecewise linear when the quantized variables remain in a bounded region.*

Proof. In LTI systems we have $f = Ax + Bu$ and then, the hypothesis of continuity and boundedness formulated in Theorem 1 are satisfied. Taking into account the assumptions on the input and quantized variable trajectories it can be easily seen that the mentioned theorem holds and then, the quantized variables have piecewise linear trajectories.

A LTI system can be expressed as follows

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{16}$$

where A is the evolution matrix of the original and B is called the input matrix. Then, the associated QSS2 satisfies

$$\dot{x}(t) = Aq(t) + Bu(t) \tag{17}$$

where $x(t)$ and $q(t)$ are related componentwise by first order quantization functions.

Since $q(t)$ and $u(t)$ have piecewise linear trajectories, it is clear that the trajectories of $\dot{x}(t)$ are also piecewise linear □

A corollary of this theorem is that in the case of QSS2 related to LTI systems, the state variables have piecewise parabolic trajectories, as it was mentioned before.

The piecewise constant and piecewise linear trajectories of a QSS allow its DEVS representation, as it is shown in (Kofman and Junco, 2000). In a similar way, the piecewise linear and piecewise parabolic trajectories will allow to find a DEVS model that exactly represents the behaviour of a QSS2 associated to a LTI system.

4 DEVS Representation of QSS2

The DEVS model of QSS built in (Kofman and Junco, 2000) was based on the use of events representing the changes in the values of the variables. Here, in a similar way, the changes in the slopes and values of the trajectories will be represented by events carrying the new values and slopes of the corresponding variables.

In this section we will deal with the case in which the slopes are chosen equal to the derivative of the input variable of the quantizers (see Figure 5a). The other case is quite similar, but there the events does not need to represent the slopes. However, in that case, the partition into submodels becomes more complicated.

The DEVS model will be defined as a coupling of first order quantized integrators (integrators with a first order quantizer at the output) and two systems that calculate the evolution function (f) and the output function (g) (see Figure 7).

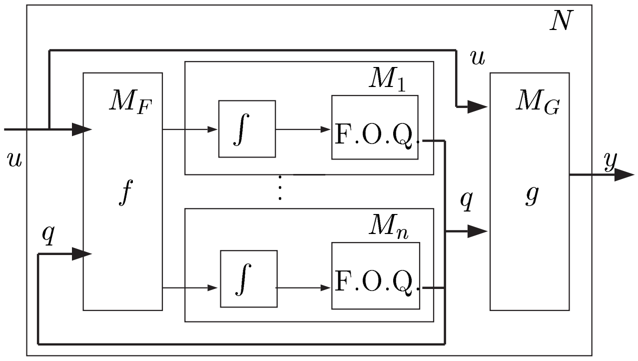

Figure 7: Partition into sub-models of the QSS2

4.1 DEVS model of a First Order Quantized Integrator

The following DEVS model represents the behaviour of a First Order Quantized Integrator with a piecewise linear input trajectory.

$$
\begin{aligned}
&M_j = < X, S, Y, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, ta >, \text{ where:}\\
&X = \mathbb{R} \times \mathbb{R} \times \{inports\} = \mathbb{R} \times \mathbb{R} \times \{1\}\\
&S = \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}_0^+ \infty\\
&Y = \mathbb{R} \times \mathbb{R} \times \{outports\} = \mathbb{R} \times \mathbb{R} \times \{1\}\\
&\delta_{\text{int}}(u, m_u, x, q, m_q, \sigma) = (u + m_u \cdot \sigma, m_u, x + u \cdot \sigma + \frac{m_u}{2}\sigma^2, x + u \cdot \sigma + \frac{m_u}{2}\sigma^2, u + m_u \cdot \sigma, \sigma')\\
&\delta_{\text{ext}}(u, m_u, x, q, m_q, \sigma, e, v, m_v, port) = (v, m_v, x + u \cdot e + \frac{m_u}{2}e^2, q + m_q \cdot e, m_q, \hat{\sigma})\\
&\lambda(u, m_u, x, q, m_q, \sigma) = (x + u \cdot \sigma + \frac{m_u}{2}\sigma^2, u + m_u \cdot \sigma, 1)\\
&ta(u, m_u, x, q, m_q, \sigma) = \sigma
\end{aligned}
$$

where

$$
\sigma' = \begin{cases} \sqrt{\frac{2\Delta q}{m_u}} & \text{if} \quad m_u \neq 0 \\ \infty & \quad\quad \text{otherwise} \end{cases} \tag{18}
$$

and $\hat{\sigma}$ can be calculated as the least positive solution of

$$
|x + u \cdot e + \frac{m_u}{2}e^2 + v \cdot \hat{\sigma} + \frac{m_v}{2}\hat{\sigma}^2 - (q + m_q \cdot e + m_q\hat{\sigma})| = \Delta q \tag{19}
$$

The input and output events are constituted by numbers representing the value, slope and port of the corresponding trajectory. The state $s \in S$ saves the last input value and slope (u and m_u), the current value of the state variable (x), the last output value and slope (q and m_q) and the time to the next output event (σ).

The internal transition function calculates the new state after an output event and the external transition function does the same but after an input event. Here, the equations (18) and (19) calculates the time to the next event, which is equal to the time when the difference between the trajectories of $x(t)$ and $q(t)$ becomes equal to Δq. Finally, the output function (λ) gives the output value and slope.

4.2 DEVS model of the static functions

The functions $f(x, u)$ and $g(x, u)$ are static functions. In order to simplify their representation, they will be considered as the coupling of their scalar components.

The DEVS model associated to a generic scalar function $f_i(z_1, ..., z_q)$, will be the following:

$$
\begin{aligned}
&F_i =< X, S, Y, \delta_{\text{int}}, \delta_{\text{ext}}, \lambda, ta >, \text{ where:}\\
&X = (\mathbb{R} \times \mathbb{R}) \times \{inports\} = (\mathbb{R} \times \mathbb{R}) \times \{1, ..., q\}\\
&S = (\mathbb{R} \times \mathbb{R} \times \mathbb{R})^q \times \mathbb{R}_0^+ \infty\\
&Y = (\mathbb{R} \times \mathbb{R}) \times \{outports\} = (\mathbb{R} \times \mathbb{R}) \times \{1\}\\
&\delta_{\text{int}}((z_1, m_{z_1}, c_1), ..., (z_q, m_{z_q}, c_q), \sigma) = ((z_1, m_{z_1}, c_1), ..., (z_q, m_{z_q}, c_q), \infty)\\
&\delta_{\text{ext}}((z_1, m_{z_1}, c_1), ..., (z_q, m_{z_q}, c_q), \sigma, e, v, mv, port) = ((z_1', m_{z_1}', c_1'), ..., (z_q', m_{z_q}', c_q'), 0)\\
&\lambda((z_1, m_{z_1}, c_1), ..., (z_q, m_{z_q}, c_q), \sigma) = (f_i(z_1', ..., z_q'), c_1 m_{z_1} + ... + c_q m_{z_q}, 1)\\
&ta((z_1, m_{z_1}, c_1), ..., (z_q, m_{z_q}, c_q), \sigma) = \sigma
\end{aligned}
$$

with

$$
z_j' = \begin{cases} v & \text{if} \quad j = port \\ z_j + m_{z_j} e & \text{otherwise} \end{cases}
$$

$$
m_{z_j}' = \begin{cases} m_v & \text{if} \quad j = port \\ m_{z_j} & \text{otherwise} \end{cases}
$$

$$
c_j' = \begin{cases} \dfrac{f_i(z + m_z e) - f_i(z')}{z_j + m_{z_j} e - z_j'} & \text{if} \quad j = port \wedge z_j + m_{z_j} e - z_j' \neq 0 \\ c_j & \text{otherwise} \end{cases} \tag{20}
$$

If the function f_i is linear, this DEVS model exactly represents its behavior. The equation (20) calculates the coefficients that multiply the input trajectory slopes. In the linear case they will coincide with the coefficients $A_{i,j}$ and $B_{i,j}$ corresponding to the evolution and input matrices respectively.

In the nonlinear case, the output trajectory of the function f_i will not be piecewise linear. However, the trajectory given by the DEVS model, which is interpreted as piecewise linear, constitutes a good approximation to the true output. The reason of this is that the coefficients c_j, calculated with (20), are closed to the corresponding partial derivatives of f_i evaluated at the points given by the input trajectories. Thus, we can affirm that the DEVS model of a static function can be applied to general nonlinear functions and general nonlinear systems can be simulated under the QSS2 approach.

The complete DEVS model related to a QSS2 will be formed by the coupling of submodels corresponding to first order quantized integrators and static functions.

5 Convergence and stability properties

As it was mentioned before, the convergence and stability properties that were proven for QSS are based on the fact that these systems can be writen according to (4) satisfying the inequality (2). QSS2 can be also represented by (4), where the last inequality takes the particular form of (9).Thus, the QSS2 approximation will satisfy the same convergence and stability properties.

The problem is that in the nonlinear cases the DEVS model does not represent exactly the behavior of the QSS2. As a consequence, it cannot be affirmed that in general nonlinear systems the second order approximating DEVS model shows that properties.

The mentioned properties can be synthesized as follows.

- Under certain conditions, the solutions of a QSS (or QSS2) associated to a continuous system converge to the solutions of the last one when the quantization goes to zero (Convergence).

- It is always possible to find a quantization so that the solutions of the QSS (QSS2) associated to an asymptotically stable continuous system finish inside an arbitrary small region around the equilibrium points of the originally continuous system (Stability).

The first property guarantees that an arbitrarily small error can be achieved during the whole simulation. The second one, not only assures that the final error can be bounded by some given value but it also allows finding the quantization that achieves that purpose by following some algorithm.

It is interesting to join both properties so that we are able to find a quantization that assures having a minimum error during the whole simulation. Unfortunately, this cannot be done for general nonlinear cases, but it can be solved for LTI systems as it will be shown.

The particularization of the stability properties to in simulation methods is an important task since it allows giving a measure of the relationship between the error and the discretization to be applied. There is a well known result in discrete time methods which says that the application of Euler's method to a stable LTI systems has a bounded error when the step size h satisfies:

$$h < \frac{2}{\lambda_{max}} \tag{21}$$

being λ_{max} the maximum eigenvalue of the evolution matrix (A).

Although the inequality (21) only holds for LTI systems, it can be also approximately applied to general nonlinear systems using the linearized systems (through the Jacobian matrix of the function f). The validity of the linearized models in simulation can be justified taking into account that there are only small changes in the variables during successive steps. Thus, after each step, the trajectories stay inside the region where the linearized model constitutes a good approximation.

In this section an analogous formula to (21) will be deduced for QSS and QSS2 related to LTI systems. It will be shown that independently of the quantization, the error never diverges and the mentioned formula will give a bound of the error during the whole simulation.

5.1 Error in LTI systems

Consider the following linear time invariant system

$$\dot{\tilde{x}}(t) = A\tilde{x}(t) + Bu(t) \tag{22}$$

Then, the associated QSS (QSS2) will have the following expression:

$$\dot{x}(t) = A(x(t) + \Delta x) + Bu(t) \tag{23}$$

where $\Delta x(t) = q(t) - x(t)$, being $q(t)$ the quantized version of $x(t)$. Thus, the error $e(t) = x(t) - \tilde{x}(t)$ satisfies:

$$\dot{e}(t) = A(e + \Delta x) \tag{24}$$

Since the initial condition $\tilde{x}(t_0) = x_0$ is suppossed to be known, we can take $x(t_0) = \tilde{x}(t_0) = x_0$ and our goal will be studying the evolution of $e(t)$ starting from the initial condition $e(t_0) = 0$.

Taking into account (2) and (9), the election of the quantum size (and the hysteresis width in QSS) for each variable is equivalent to the choice of a bound for the corresponding component of Δx. Hence, in the following theorems, the necessary conditions for the choice of that bounds will be given in order to guarantee that each component $e_i(t)$ in $e(t)$ remains bounded by some maximum desired error in the corresponding variable $x_i(t)$.

Formally, the bounds $\Delta q_1, \cdots, \Delta q_n$ will be obtained so that the condition

$$|\Delta x_i(t)| \leq \Delta q_i \qquad i = 1, \cdots, n$$

assures that

$$|e_j(t)| \leq \Delta e_j \;\forall t \geq t_0, \qquad j = 0, \cdots, n$$

or equivalently, that the vectorial inequality[1]

$$|\Delta x(t)| \leq \Delta q$$

implies that

$$|e(t)| \leq \Delta e \tag{25}$$

with $e(t_0) = 0$.

Here, Δq is the vector of quantum sizes and Δe is the vector of maximum desired errors.

Theorem 3. *Consider the system* (24) *being A a scalar with its real component being negative. When $e(t_0) = 0$, the condition given by* (25) *can be guaranteed by taking*

$$\Delta q = \frac{|\mathbb{R}\mathrm{e}(A)|}{|A|}\Delta e \tag{26}$$

Proof. If A is a real number the proof is straightforward since when $e(t) = \Delta e = \Delta q \geq \Delta x$, from (24) we see that $\dot{e}(t) \leq 0$. Similarly, when $e(t) = -\Delta e$, it results $\dot{e}(t) \geq 0$. Thus, $e(t)$ cannot abandon the interval $[-\Delta e, \Delta e]$.

For the complex case, we define

$$e(t) = r(t) \cdot e^{j\theta(t)}$$

Then (24) can be rewritten as:

$$\begin{aligned} \frac{d}{dt}(r \cdot e^{j\theta}) &= A(r \cdot e^{j\theta} + \Delta x) \\ \dot{r}e^{j\theta} + jre^{j\theta}\dot{\theta} &= A(r \cdot e^{j\theta} + \Delta x) \end{aligned}$$

and then,

$$\begin{aligned} \dot{r} + jr\dot{\theta} &= A(r + \Delta x \cdot e^{-j\theta}) \\ &= A \cdot r + A\Delta x \cdot e^{-j\theta} \end{aligned}$$

Taking only the real components of the last equation we have

$$\begin{aligned} \dot{r} &= \mathbb{R}\mathrm{e}(A) \cdot r + \mathbb{R}\mathrm{e}(A\Delta x \cdot e^{-j\theta}) \\ &\leq \mathbb{R}\mathrm{e}(A) \cdot r + |A||\Delta x| \\ &\leq \mathbb{R}\mathrm{e}(A) \cdot r + |A||\Delta q| \end{aligned}$$

Considering that $r(t) = |e(t)|$ and the using the equation (26), it results that

$$\begin{aligned} \frac{d}{dt}|e(t)| &\leq \mathbb{R}\mathrm{e}(A) \cdot |e(t)| + |A|\frac{|\mathbb{R}\mathrm{e}(A)|}{|A|}\Delta e \\ &= \mathbb{R}\mathrm{e}(A)(|e(t)| - \Delta e) \end{aligned}$$

Since $\mathbb{R}\mathrm{e}(A)$ is negative and $|e(t_0)| = 0$, $|e(t)|$ cannot become greater than Δe. $\square$

In the real case, this theorem expresses that the error in a stable first order system is bounded by the quantization size. In the complex case, it does not have any meaning. However, it will be useful for later results.

[1] We say that $a \leq b$ when $a_i \leq b_i$ for all the components of the vectors a and b

Theorem 4. *Consider the system of* (24) *being A a diagonal matrix with $\mathbb{R}e(A_{i,i}) < 0$. When $e(t_0) = 0$, the condition given by* (25) *can be guaranteed taking*[2]

$$\Delta q = \mathrm{diag}(\frac{|\mathbb{R}e(A_{i,i})|}{|A_{i,i}|})\Delta e$$

Proof. The proof is straightforward applying the theorem 3 for each component of $e(t)$. □

Theorem 5. *Consider the system of* (24) *being A a diagonalizable Hurwitz*[3] *matrix. When $e(t_0) = 0$, the condition given by* (25) *can be guaranteed taking Δq so that*

$$|V| \cdot \mathrm{diag}(\frac{|\lambda_i|}{|\mathbb{R}\mathrm{e}(\lambda_i)|}) \cdot |V^{-1}|\Delta q = T\Delta q \leq \Delta e \tag{27}$$

where V is the eigenvector matrix and λ_i are the eigenvalues of the matrix A, that is [4]

$$V^{-1} \cdot A \cdot V = \mathrm{diag}(\lambda_i) \tag{28}$$

Proof. Let be $e(t) = Vw(t)$. Thus, from (24) it results

$$V\dot{w}(t) = A(Vw + \Delta x)$$

and then

$$\dot{w}(t) = V^{-1}AV(w + V^{-1}\Delta x) = V^{-1}AV(w + \Delta x_w) \tag{29}$$

and

$$|\Delta x_w(t)| = |V^{-1}\Delta x| \leq |V^{-1}| \cdot |\Delta x| \leq |V^{-1}|\Delta q$$

From (28) $V^{-1}AV$ is a diagonal matrix. Thus the system (29) satisfies the theorem 4. From this theorem and the last inequality we have

$$|w(t)| \leq \mathrm{diag}(\frac{|\lambda_i|}{|\mathbb{R}\mathrm{e}(\lambda_i)|}) \cdot |V^{-1}|\Delta q$$

It results from the definition of $w(t)$ that

$$|e(t)| = |Vw(t)| \leq |V| \cdot \mathrm{diag}(\frac{|\lambda_i|}{|\mathbb{R}\mathrm{e}(\lambda_i)|}) \cdot |V^{-1}|\Delta q = T\Delta q \leq \Delta e$$

which completes the proof. □

The inequality (27) can be satisfied using different sets of bounds Δq. A formula that allows satisfying that condition is the following

$$\Delta q_i = \frac{1}{n} \min_j (\frac{\Delta e_j}{T_{ji}}) \tag{30}$$

[2]The expression $\mathrm{diag}(A_{i,i})$ refers to the diagonal matrix formed with the components $A_{i,i}$ in its diagonal.

[3]An $n \times n$ matrix is said to be Hurwitz when all its eigenvalues have negative real component (i.e. the system is asymptotically stable)

[4]$|\cdot|$ denotes the componentwise moduli of the corresponding matrix or vector.

5.2 Some remarks about the results

The theorem 5 and the equation (30) give a *practical* formula to calculate the quantization that assures that the error in each variable never becomes bigger than certain bound Δe_j. However, that formula requires the calculation of the eigenvectors of the matrix A. If that eigenvectors are calculated, the exact solution of (22) can be obtained (at least when the input is not very complicated) and then the simulation might become unnecessary (except for the case in which the goal is simulating a system for different and complex input trajectories).

In spite of this, the analysis shows important theoretical properties with practical and conceptual consequences. In one hand, it can be seen that the error is bounded by a linear function of the quantization. Thus, practical rules to modify the quantization in order to obtain more *adequate* results can be easily deduced. The identification of the relationship between the error and the step size in discrete time simulation methods constitutes the basis for the development of variable step size algorithms. In our case, the mentioned analysis can constitute the basis for the future formulation of adaptive quantization methods.

Another property that can be deduced from the analysis done which was already mentioned is that for any arbitrary quantization, the error will be bounded during the whole simulation. In classic discrete time algorithms, this is distinctive property of implicit methods. However, in QSS and QSS2 there is no implicit formula, which constitutes an important advantage from the point of view of the computational costs. This is probably the most important feature of the QSS and QSS2 methods.

It is also interesting to mention that the formula (27) only involves *geometrical* parameters of the system (the eigenvectors and the ratio of the real part to the absolute value of the eigenvalues). The relationship between the error and the quantization does not depend on the speed of the system as it occurs in discrete time methods (consider inequality (21) in Euler's method, for instance). This is coherent with the fact that in quantized state systems the time discretization is replaced by the discretization of the state variables.

Although all the analysis in this section was made over linear systems, it was already mentioned that simulation involves a succession of local problems. This is, there are not important variations between successive steps or events and then, the linearized models of nonlinear systems consitute good approximations in most cases. Thus, the results obtained here can be approximately extended to general nonlinear systems.

6 Examples and Simulation Results

6.1 A simple linear example

In this first example, we will verify the validity of Theorem 5 using the results of the QSS simulation example in (Kofman and Junco, 2000). There, the following system corresponding to a RLC serial circuit was introduced.

$$\begin{cases} \dot{x_1} &= \frac{1}{L}x_2 \\ \dot{x_2} &= U - \frac{1}{C}x_1 - \frac{R}{L}x_2 \\ y &= \frac{1}{L}x_2 \end{cases} \tag{31}$$

Using the parameters $R = 100.01, L = 0.01, C = 0.01$, a constant input $U = 100$ and a uniform quantization of 10^{-2} and 10^{-4} in x_1 and x_2 respectively, the use of QSS gave the output trajectory of Figure 8.

The evolution matrix A is

$$A = \begin{bmatrix} 0 & 100 \\ -100 & -10001 \end{bmatrix}$$

and its corresponding eigenvector and eigenvalues matrices result:

$$V = \begin{bmatrix} 1 & -0.01 \\ -0.01 & 1 \end{bmatrix}; \ \mathrm{diag}(\lambda_i) = \begin{bmatrix} -1 & 0 \\ 0 & -10000 \end{bmatrix}$$

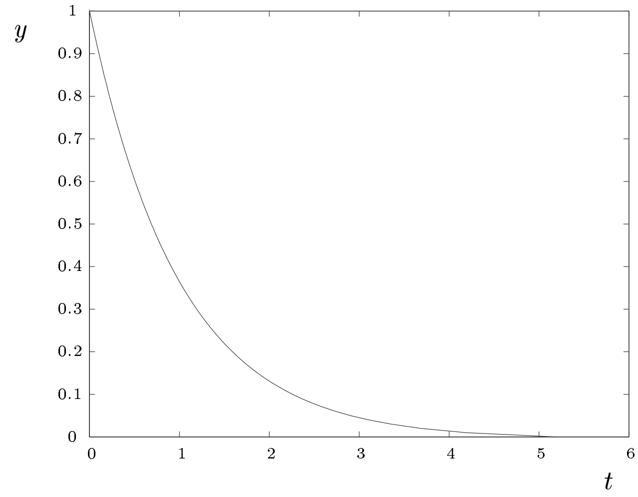

Figure 8: Output trajectory $y(t)$ in the QSS simulation of (31)

Applying the equation (27) we have:

$$T\Delta q = |V| \cdot \mathrm{diag}(\frac{|\lambda_i|}{|\mathbb{R}\mathrm{e}(\lambda_i)|}) \cdot |V^{-1}|\Delta q \simeq \begin{bmatrix} 1.0002 & 0.02 \\ 0.02 & 1.0002 \end{bmatrix} \begin{bmatrix} 0.01 \\ 0.0001 \end{bmatrix} \simeq \begin{bmatrix} 0.01 \\ 0.0003 \end{bmatrix}$$

Thus, it can be assured that the error in $x_2(t)$ cannot be greater than 0.0003. Then, the error in the variable $y(t) = 100 \cdot x_2(t)$, cannot become greater than 0.03. In fact,as it is shown in Figure 16, the error calculated using the exact solution of the system equations is always less than $6 \cdot 10^{-3}$.

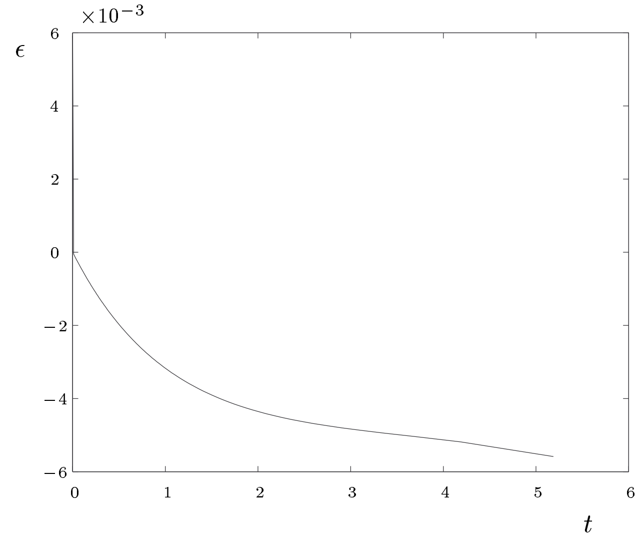

Figure 9: Error in the QSS simulation of (31)

In this example it can be seen that the bound obtained using the equation (27) is satisfied and it is not very conservative.

6.2 The second order method in a nonlinear example

The famous second order Lotka-Volterra's model is a nonlinear system of differential equations that tries to represent the evolution of the population of two species, pray and predator, in a common habitat. Due to the nonlinearity of the system, this is a case in which an analitic solution for the general case cannot be obtained.

The following set of state equations constitutes one possible representation of the mentioned system

$$\begin{cases} \dot{x_1} &= \epsilon x_1 + \alpha x_1 x_2 - \sigma x_1^2 \\ \dot{x_2} &= -m x_2 + \beta x_1 x_2 \end{cases}$$

where the variables x_1 and x_2 represent the population of prays and predators respectively.

The system was simulated using QSS2 with a quantization of $\Delta q_1 = \Delta q_2 = 0.001$. The parameters taken were $\epsilon = 0.1$, $\alpha = 0.01$, $\sigma = 0.01$, $m = 0.4$, $\beta = 0.5$ and the initial condition adopted was $x_1(0) = x_2(0) = 10$. The results are shown in Figures 10 and 11.

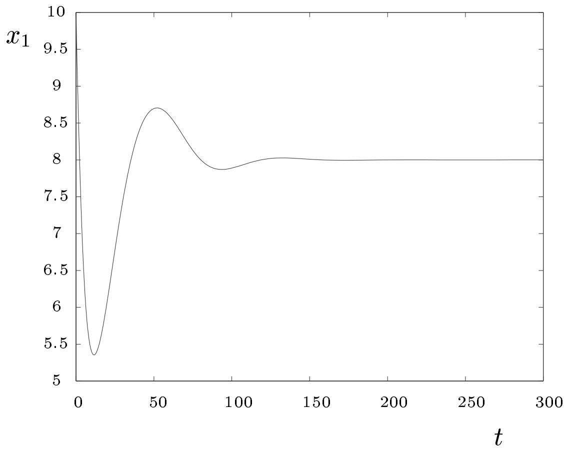

Figure 10: Number of prays in Lotka Volterra's model

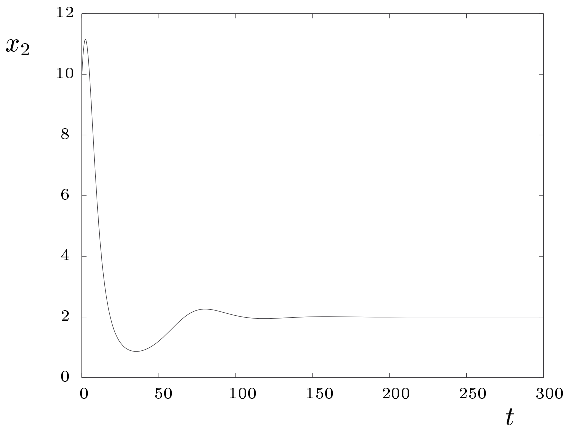

Figure 11: Number of predators in Lotka-Volterra's model

The simulation was performed with 211 internal transitions in the first integrator and 264 in the second, which gives a total of 475 steps. These results were compared with the trajectories obtained using Heun's method, the classic second order fixed step method. The step size used in Heun's method was 0.63 so that it performed the same number of steps. Figure 12 shows a part of those trajectories and the "true" trajectory. The "true" trajectory was obtained using the fifth order Dormand-Prince method with a step size of 0.1.

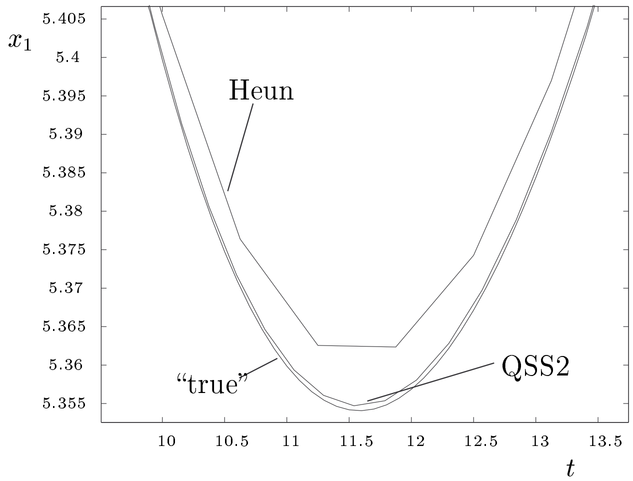

Figure 12: Number of prays according to Heun's method, QSS2 and the "true" trajectory.

The absolute error of the QSS2 and Heun's method during the simulation is shown in Figure 13. The use of QSS2 achieves a considerable reduction of the maximum error. However, the error does not converge to zero as in Heun's method.

In spite of the lack of convergence, the results obtained with QSS2 simulation are much more reliable than the results with Heun's method since the error is bounded during the whole simulation. Here we can see that this property deduced for LTI systems in Section 5 is still verified in nonlinear examples.

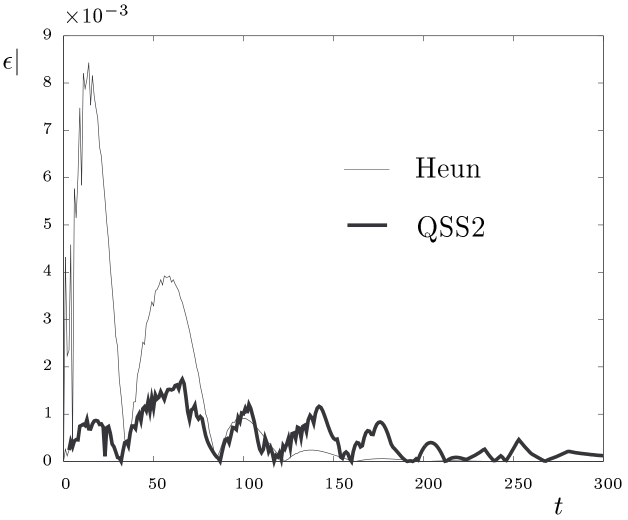

Figure 13: Error in Heun and QSS2 simulation of Lotka-Volterra's model

6.3 QSS2 in a high order linear model

The circuit of Figure 14 represents an RLC transmission line. This model is useful to study the performance of integrated circuits transmitting data at a very fast rate. Although the length of the wires is only of a few centimetters, the high frequency of the signal produces that the delays introduced by the wires cannot be ignored and the transmission line theory must be applied.

The transmission line models are usually described as systems of partial differencial equations. However, they can be approximatted by *lumped models* where the distributed effects of capacity, inductance and resistance are represented by a cascade of single capacitors, inductors and resistors, as Figure 14 shows.

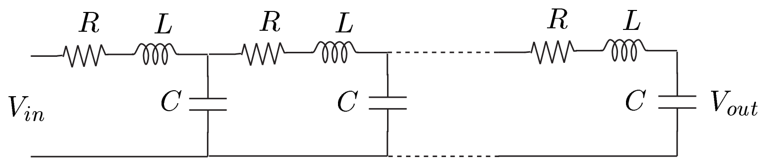

Figure 14: RLC Transmission Line

In order to constitute a good approximation, the RLC model must be formed by several sections. As a consequence of this, it results in a high order ordinary differential equation.

In (Ismail et al., 1999) an example composed by five sections of RLC circuits is introduced. The resistance, inductance and capacitance values used there corresponds to real parameters. The model obtained is a tenth order linear system, where the evolution matrix (A) can be written as follows:

$$A = \begin{bmatrix} -R/L & -1/L & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/C & 0 & -1/C & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/L & -R/L & -1/L & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/C & 0 & -1/C & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/C & 0 \end{bmatrix} \qquad (32)$$

The mentioned system was simulated applying a trapezoidal input wave. The input and output trajectories are shown in Figure 15.

The parameters used were $R = 80\Omega$, $C = 0.2pF$ and $L = 20nH$. These parameters correspond to a transmission line of one centimetter divided in five sections, where the resistance, capacitance and inductance are $400\Omega/cm$, $1pF/cm$ and $100nH/cm$ respectively.

The trapezoidal input has a rising time $t_r = 10ps$. The quantization adopted was $\Delta v = 4mV$ in the state variables representing voltages and $\Delta i = 10\mu A$ in the state variables representing currents. That quantization, according to (27), assures that the maximum error is smaller than $250mV$ in the variable V_{out}.

The simulation took a total of 2536 steps (between 198 and 319 internal transitions at each integrator) to obtain the first $3.2ns$ of the system trajectories.

The experiment was repeated using a quantization 100 times smaller, which assures having a maximum error in V_{out} of $2.5mV$. This new simulation was performed with a total of 26883 internal transitions.

The comparision of the results of both experiments (Figure 16) shows that the difference between the trajectories never becomes greater than $14.5mV$ which implies that the error in the first simulation was not greater than $17mV$. In this case, the theoretical prediction of the error (the bound was $250mV$) was quite conservative (this can be easily understood taking into account that the theoretical bound of the error holds for any input trajectory and for any initial condition).

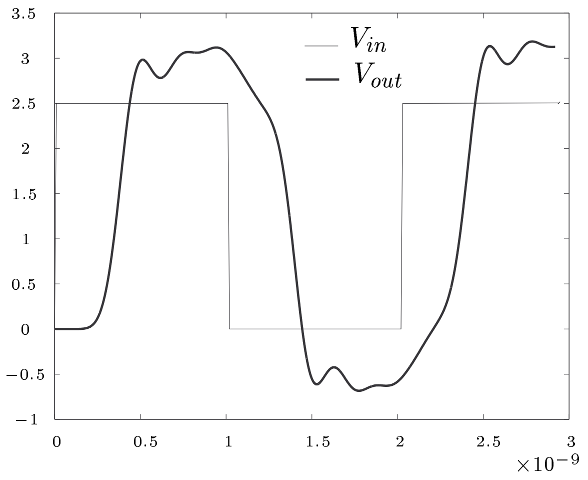

Figure 15: QSS2 simulation results in a RLC transmission line

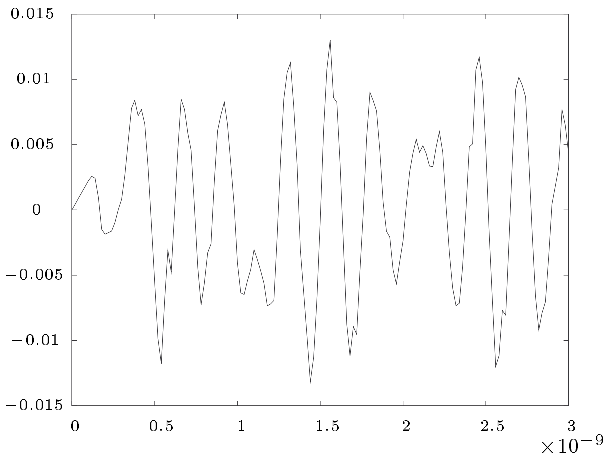

Figure 16: Difference in V_{out} using two different quantization sizes

Although the number of steps (2536) seems to be quite big, it is important to take into account that each step only involves calculations at three integrators (the integrator that performs the internal transition and the two integrators directly connected to its output). This is due to the particular form of the matrix A (which is sparse).

It is important to mention that any fixed step discrete time method should use a step size of about $1ps$ in order to obtain a good representation of the input signal since its rising time is very short (mainly if we are interesting in the behaviour of the system during the rising time). As a result, the number of steps would be 3200 or greater with each step involving calculations over all the state variables.

7 Final remarks

7.1 QSS2 vs. QSS

It was shown that QSS and QSS2 satisfy the same stability and convergence properties. Moreover, using the same quantization, they have exactly the same bound for the error during the whole simulation of a LTI system.

However, the simulation using QSS2 requires much less steps than a simulation using QSS for the same quantization (i.e. for the same error bound).

The cause of the reduction of the number of steps can be found in the comparision of the figures 4 and 17. The number of steps used by a QSS to represent the same trajectory with the same quantization size (Δq) is much greater than the used by the QSS2. The key is that using the information of the slope in QSS2 the time required to obtain a difference equal to Δq with respecto to the input trajectory becomes much greater.

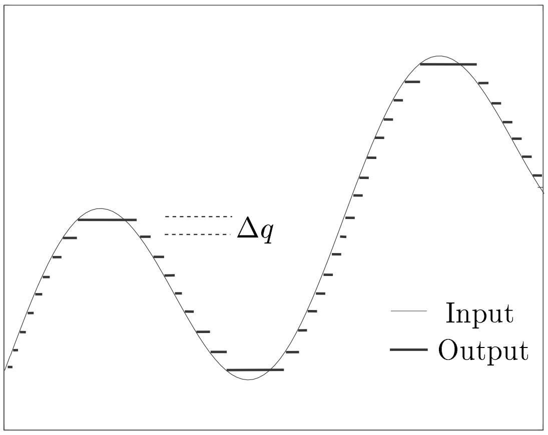

Figure 17: Input and Output trajectories in a *Zero Order* quantizer

In Lotka-Volterra's example, the simulation via QSS using the same quantization would have taken more than 10000 internal transitions at each integrator[5]. This number, compared against the 264 and 211 steps performed by the QSS2 simulation shows clearly the advantage of the new method.

The difference between the performance of QSS and QSS2 becomes greater as well as the quantization is taken smaller. In fact the number of internal transitions in QSS is approximately proportional to the inverse of the quantization size while in QSS2 it is approximately proportional to the square root of that number (see equation (18)). This relationship is verified in the example of the transmission line, where the use of a quantization 100 times smaller resulted in an increment of about 10 times in the number of steps.

In order to quantify the advantages of QSS2 with respect to QSS a simple linear system was simulated using both methods and different quantum sizes. The mentioned system is

$$\begin{cases} \dot{x_1} &= x_2 \\ \dot{x_2} &= 1 - x_1 - x_2 \\ y &= x_2 \end{cases} \tag{33}$$

Figure 18 compares the CPU time in the simulation for different quantum in both methods. QSS shows a better performance for bigger quantization but when the quantization becomes smaller QSS2 reduces considerably the computational costs. In fact, for $\Delta q = 0.0001$ the second order method is 40 times faster than QSS.

7.2 Hysteresis in QSS2

The addition of hysteresis to the zero order quantizer was necessary in order to avoid arriving to illegitimate DEVS models (Kofman and Junco, 2000) but in QSS2 it seems to be absent. However, the hysteresis is also present since it is implicit in the equations (5) and (6) which define the first order quantizer and in the equation (19) corresponding to the DEVS model of the first order quantized integrator.

[5]This number can be obtained comparing the amplitude of the trajectories and the quantization size 0.001

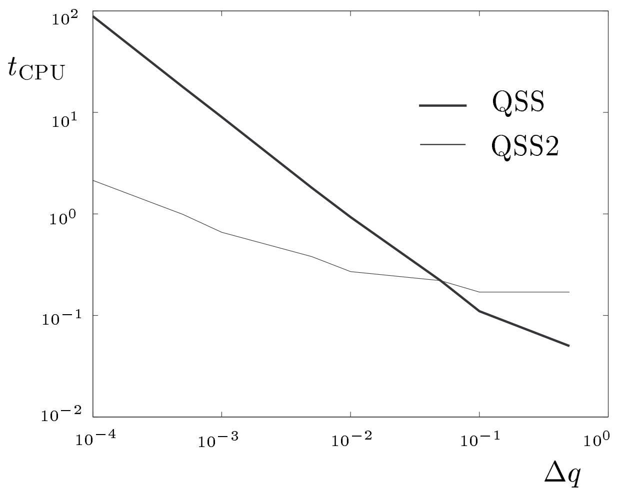

Figure 18: CPU time vs. quantum size in QSS and QSS2 simulation

In QSS2, the quantization Δq can be interpreted as both, the quantization size and the hysteresis width. The hysteresis can be seen in the following example.

Consider a first order quantizer that produces an output event (i.e. its input and output values become the same) with a slope equal to zero. Assume that after that, the input value starts rising and then falling again. If there were not hysteresis, when the input value reaches the output value the quantizer would produce a new event (that is what would happen in a quantizer without hysteresis). However, according to (6) that new event will not be produced until the difference between the input and output variables reaches the value Δq. This is equivalent to what happens in a zero order quantizer with hysteresis window $\varepsilon = \Delta q$.

8 Conclusions

The second order method solves one of the most important problems of the previous discrete event based integration algorithms since it can achieve a good accuracy without an excesive increment in the number of steps.

The theoretical analysis showed that the approximation via QSS2 satisfies the same stability and convergence properties that were proven for QSS. In addition, a new property was deduced for both methods when they are applied to the simulation of LTI systems. That property assures that the use of QSS or QSS2 produces a bounded error during the whole simulation which is proportional to the quantization. In fact, the inequality (27) gives a closed formula to obtain the mentioned bound as a function of the system parameters. It is also important to mention that the bound does not depend on the input trajectories or the initial conditions.

This result implies that the quantization constitutes a measure of the error at any time of the simulation. Since the forms of the trajectories in QSS and QSS2 are well known (they are piecewise linear or piecewise parabolic), that information can be used to obtain the exact value of the quantized trajectories not only at the event times but also at any intermediate time. Generally, in classic discrete time methods the bound of the error is deduced only for one step (typically using Taylor's series) and there is no information about the error committed when intermediate values are interpolated.

Although the mentioned result was obtained for LTI systems, it can be approximately extended to nonlinear systems as it was discussed in section 5 and verified through the Lotka-Volterra's example.

With respect to the future work, the theorem 5 should be completed for general Hurwitz matrices (i.e. including the non diagonalizable cases) and it would be desirable to find a formal proof of what was mentioned in the previous paragraph related to nonlinear systems.

The methods QSS and QSS2 can be generalized (QSSn) using higher order polynomials for the representation of the trajectories. The use of such higher order methods should result in an important reduction of the number of steps. However, a new problem might appear here since the equation (19) will turn into an equation of order n. When n is bigger than 3 the resolution becomes quite difficult.

Another important future task is the introduction of adaptive quantization. The idea is similar to what is done by the variable step algorithms, that is, varying the step size according to some estimation of the error. In that way, we can obtain a method in which the user does not have to choose the quantization. It would be automatically chosen by the algorithm according to the desired error. The adaptive quantization might be also used to avoid the final oscillations in the quantized solutions (using fixed quantization the solutions does not converge to the equilibrium point but they remain in a region around that point as it is was mentioned before). A result that can be useful for this goal can be found in (Brockett and Liberzon, 2000), where the authors show that a system with quantizers can reach the equilibrium point if the quantization size is reduced as well as the trajectories get closer to that point.

Finally, the use of the methodology in the example of the transmission line showed some advantages that might be applied to more general distributed parameters systems. Taking into account that here the resulting DEVS model corresponding to the QSS (QSS2) can be built as the coupling of several identical submodels (associated to each section of the system), the combination of the method with the work made on Cell-DEVS (Wainer, 2000) can result in an interesting application.

References

Brockett, R. and Liberzon, D. (2000). Quantized feedback stabilization of linear systems. *IEEE Trans. Automat. Contr.*, 45:1279–1289.

Giambiasi, N., Escude, B., and Ghosh, S. (2000). GDEVS: A generalized Discrete Event specification for accurate modeling of dynamic systems. *Transactions of SCS*, 17(3):120–134.

Ismail, Y., Friedman, E., and Neves, J. (1999). Figures of merit to characterize the importance of On-Chip inductance. *IEEE Trans. on VLSI Systems*, 7(4):442–449.

Kofman, E. and Junco, S. (2000). Quantized State Systems. A DEVS approach for continuous system simulation. Technical Report LSD0004, LSD, UNR. To appear in *Transactions of SCS*. Available at www.eie.fceia.unr.edu.ar/∼ekofman.

Kofman, E. and Junco, S. (2001). Quantized Bond Graphs: An Approach for Discrete Event Simulation of Physical Systems. In *Proceedings of ICBGM'01*, pages 369–374, Phoenix.

Press, W., Flannery, B., Teukolsky, S., and Vetterling, W. (1986). *Numerical Recipes*. Cambridge: Cambridge University Press.

Wainer, G. (2000). Improved cellular models with Parallel Cell DEVS. *Transactions of SCS*, 17(2):73–88.

Zeigler, B. (1976). *Theory of Modeling and Simulation*. John Wiley & Sons, New York.

Zeigler, B., Kim, T., and Praehofer, H. (2000). *Theory of Modeling and Simulation. Second edition*. Academic Press, New York.

Zeigler, B. and Lee, J. (1998). Theory of quantized systems: formal basis for DEVS/HLA distributed simulation environment. In *SPIE Proceedings*, pages 49–58.