XII Reunión de Trabajo en Procesamiento de la Información y Control, 16 al 18 de octubre de 2007

Linearly Implicit Discrete Event Methods for Stiff ODEs. Part I: Theory

Gustavo Migoni† y Ernesto Kofman‡

†*Laboratorio de Sistemas Dinámicos FCEIA - UNR - CONICET. Riobamba 245 bis - (2000) Rosario*

***Abstract*— This paper introduces two new numerical methods for integration of stiff ordinary differential equations. Following the idea of quantization based integration, i.e., replacing the time discretization by state quantization, the new methods perform first and second order backward approximations allowing to simulate stiff systems. It is shown that the new algorithms satisfy the same theoretical properties of the previous methods. A companion paper with applications illustrate the practical advantages of the methodology.**

***Keywords*— Stiff System Simulation, Quantization Based Integration, DEVS**

I INTRODUCTION

The use of traditional methods [2, 3, 1] based on time discretization to integrate stiff systems require the use of implicit algorithms since the required step size used by explicit methods is limited by the stability region and the resulting step size becomes inadmissibly small [1].

In fact, numerical integration methods that include in their numerically stable region the entire left half ($\lambda \cdot h$) plane (or at least a large portion of it) are necessary [1]. Only some implicit methods have this type of stability region. Explicit algorithms showing that feature do not exist.

The problem with implicit methods is that they are computationally expensive because in each step they need to use iterative algorithms to determine the next value (usually with the Newton Iteration). The problem becomes critical in applications related to real time simulation, where in many cases performing iterations becomes unacceptable.

An alternative approach to classic time slicing started to develop since the end of the 90's, where time discretization is replaced by state variables quantization. As a result, the simulation models are not discrete time but discrete event systems. The origin of this idea can be found in the definition of Quantized Systems [12, 11].

This idea was then reformulated with the addition of hysteresis –to avoid the appearance of infinitely fast oscillations– and formalized as the Quantized State Systems (QSS) method for ODE integration in [7]. This was followed by the definition of the second order QSS2 method [4], the third order QSS3 method [6].

The QSS–methods showed some important advantages with respect to classic discrete time methods in the integration of discontinuous ODEs [5], sparsity exploitation [4], the property of absolute stability, and the existence of a global error bound [1].

In spite of these properties, QSS, QSS2 and QSS3 fail when applied to stiff systems due to the appearence of fast oscillations. To solve this problem, a first order backward QSS method (called BQSS) was proposed in [8, 10]. The basic idea of the BQSS method is to use a *future* value of the states to obtain the quantized value. Since the future quantized magnitude can only take two values (one from below and one from above the current quantized variable), BQSS method results does not call for iterations. In other words BQSS was the first explicit method for stiff ODEs.

The main drawback of BQSS is that it performs only a first order approximation and accurate results cannot be obtained. Another problem is that BQSS introduced an extra perturbation term that increases the error bound and, in some nonlinear systems, might provoke the appeareance of spurious equilibrium points.

This paper presents first a new method that combines the idea of BQSS and linearly implicit integration. The new method, called LIQSS (after Linearly Implicit QSS), solves the problem of the extra perturbation term and the appearence of spurious equilibrium points.

Then, combining the idea of LIQSS with QSS2, we obtained a second order accurate method called LIQSS2. This new method solves, up to certain limits, the problem of accuracy.

The work is organized in the following way: Section II recalls the principles of Quantization–Based Integration and introduces the problems of QSS methods to deal with stiff systems. Then, Section III introduces the new methods, namely, LIQSS and LIQSS2. Section IV studies the properties of the new methods (legitimacy and error bound) and finally, Section V presents some conclusions.

Implementation issues and application examples are presented in the companion paper [9].

II QUANTIZATION-BASED INTEGRATION

This section introduces the principles of QSS methods and shows the problems they have in the simulation of stiff systems

A QSS Method

Consider a time invariant ODE in its State Equation System (SES) representation:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^m$ is an input vector, which is a known piecewise constant function.

The QSS-method simulates an approximate system, which is called Quantized State System:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{q}(t), \mathbf{u}(t)) \tag{2}$$

where $\mathbf{q}(t)$ is a vector of quantized variables which are quantized versions of the state variables $x(t)$. Each component of $\mathbf{q}(t)$ follows a piecewise constant trajectory, related with the corresponding component of $\mathbf{x}(t)$ by a hysteric quantization function so that:

$$q_j(t) = \begin{cases} x_j(t) & \text{if } |q_j(t^-) - x_j(t)| = \Delta Q_j \\ q_j(t^-) & \text{otherwise} \end{cases} \tag{3}$$

and $q_j(t_0) = x_j(t_0)$. Thus, $q_j(t)$ only changes when it differs from $x_j(t)$ in $\pm\Delta Q_j$. The magnitude ΔQ_j is called quantum.

Using the fact that $\mathbf{q}(t)$ is piecewise constant and provided that the input $\mathbf{u}(t)$ is piecewise constant, it can be seen that $\mathbf{x}(t)$ is piecewise linear [7]. Consequently, the integration of Eq.(2) is straightforward. Moreover, the system is equivalent to a discrete event model in terms of the DEVS formalism [11]. A summarized explanation the DEVS formalism can be found in [9].

B QSS and Stiff Systems

The system

$$\begin{aligned} \dot{x}_1(t) &= 0.01\, x_2(t) \\ \dot{x}_2(t) &= -100\, x_1(t) - 100\, x_2(t) + 2020 \end{aligned} \tag{4}$$

has eigenvalues $\lambda_1 \approx -0.01$ and $\lambda_2 \approx -99.99$ which mens that the system is stiff.

The QSS method approximates this system as

$$\begin{aligned} \dot{x}_1(t) &= 0.01\, q_2(t) \\ \dot{x}_2(t) &= -100\, q_1(t) - 100\, q_2(t) + 2020 \end{aligned} \tag{5}$$

Considering initial conditions $x_1(0) = 0$, $x_2(0) = 20$, and quanta $\Delta Q_1 = \Delta Q_2 = 1$, the QSS integration performs the following steps:

In $t = 0$ we set $q_1(0) = 0$ and $q_2(0) = 20$. Then, $\dot{x}_1(0) = 0.2$ and $\dot{x}_2(0) = 20$. This situation remains until $|q_i - x_i| = \Delta Q_i = 1$.

The next change in q_1 is then scheduled at $t = 1/0.2 = 5$ while the change in q_2 is scheduled at $t = 1/20 = 0.05$.

Thus, a new step is performed in $t = 0.05$. After this step it results $q_1(0.05) = 0$, $q_2(0.05) = 21$, $x_1(0.05) = 0.01$, $x_2(0.05) = 21$. The derivatives are $\dot{x}_1(0.05) = 0.21$ and $\dot{x}_2(0.05) = -80$.

The next change in q_1 is rescheduled at $0.05 + (1 - 0.01)/0.21 = 4.764$ while the next change in q_2 is scheduled at $0.05 + 1/80 = 0.0625$. Hence, the next step is performed at $t = 0.0625$.

In $t = 0.0625$ it results $q_1(0.0625) = 0$, $q_2(0.0625) = x_2(0.0625) = 20$, $x_1(0.0625) \approx 0.0126$ and the derivatives coincide with those of $t = 0$.

This is cyclically repeated until a change in q_1 occurs. That change occurs at $t \approx 4.95$, after 158 changes in q_2 (which oscillates between 20 and 21).

The simulation continues in the same way. Fig.1 shows the evolution of $q_1(t)$ and $q_2(t)$ through 500 units of simulated time.

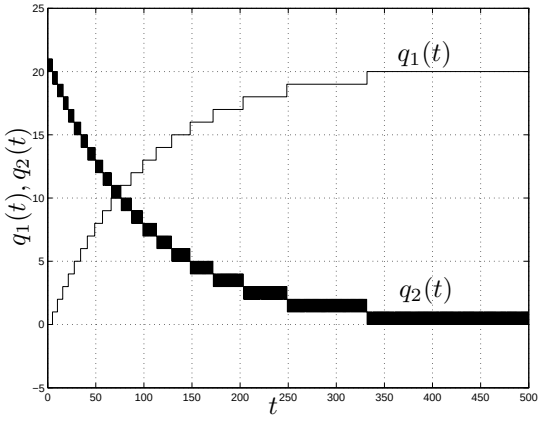

Figure 1: QSS Simulation

The fast oscillations of q_2 provoke a total of 15995 transitions in that variable, while q_1 only changes 21 times. Consequently, the total number of steps to complete the simulation is greater than 16000 (this number is of the order of the 25000 steps needed by Forward Euler method to obtain a stable result).

Evidently, the QSS method is unable to efficiently integrate System (4).

C QSS2 Method

The second order QBI method uses first order quantization. As it is shown in Figure 2, a first order quantizer produces a piecewise linear output trajectory. Each section of that trajectory starts with the value and slope of the input and finishes when it differs from the input in ΔQ_i. A formal definition of a first order quantization function can be found in [4].

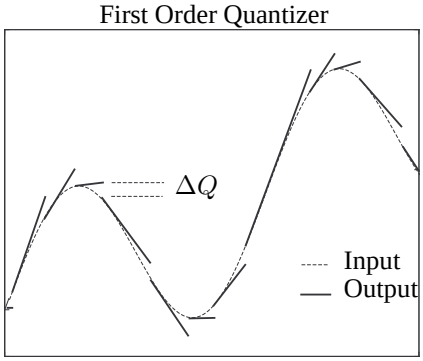

Figure 2: Trajectories of a first–order quantizer.

The QSS2 method then approximates a system like (1) by (2) but now, the quantized variables $q_i(t)$ follow piecewise linear trajectories and the state variables $x_i(t)$ are piecewise parabolic functions of the time.

Like the first order QSS, QSS2 can be represented by a DEVS model. The advantage of QSS2 is that it permits using a small quantum –i.e., setting a small error tolerance–without increasing considerably the number of calculations. In QSS, the number of steps is inversely proportional with the quantum, while in QSS2 it is only inversely proportional with its square root.

QSS2 also exhibits the problem of fast oscillations in the simulation of stiff systems. For instance, if we use the QSS2 method to simulate System (4), it performs 65467 steps (19 changes in q_1 and 65448 changes in q_2).

D BQSS Method

The BQSS method is similar to QSS, but q_i is always chosen so that $x_i(t)$ goes to $q_i(t)$.

Basically, given a state variable $x_j(t)$, BQSS uses two hysteretic quantization functions: one from below ($\underline{q}_j(t) \leq x_j(t)$) and the other from above ($\overline{q}_j(t) \geq x_j(t)$). Both quantization functions are defined so that they never differ from x_j in more than $2\Delta Q_j$.

The quantized variable q_j is chosen equal to either $\underline{q}_j$ or $\overline{q}_j$, according to the direction of $\dot{x}_j(t)$. When $\dot{x}_j > 0$ we use $q_j = \overline{q}_j$, and viceversa.

When $\dot{x}_j = f_j(\mathbf{q}, \mathbf{u})$ depends on q_j, it could happen that the sign of the derivative changes when we evaluate f_j using each possibility, i.e., $f_j|_{\underline{q}_j} > 0$ and $f_j|_{\overline{q}_j} < 0$. Thus, we cannot find a correct value for q_j.

However, in that situation, continuity in f_j ensures that a value $\hat{q}_j$ exists, with $\underline{q}_j < \hat{q}_j < \overline{q}_j$, such that $f_j|_{\hat{q}_j} = 0$.

Thus, the BQSS method sets either $q_j = \overline{q}_j$ or $q_j = \underline{q}_j$ and enforces the derivative $\dot{x}_j = 0$ adding an extra perturbation term $\Delta f_j = f_j|_{q_j}$.

Then, given the system Eq.(1), instead of simulating a system like Eq.(2), BQSS simulates a system of the form:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{q}(t), \mathbf{u}(t)) + \Delta\mathbf{f}(t) \tag{6}$$

where $\Delta(t)$ is normally zero, except when the situation described above is found (i.e., when we cannot find a correct value for q_j).

BQSS works fine with most stiff systems. Anyway, the presence of the perturbation term $\Delta\mathbf{f}(t)$ increases the error and can cause the appearence of spurious equilibrium points in some nonlinear systems.

Another limitation of BQSS is that it is only first order accurate. We could not find, based on that idea, a second order accurate method.

III LINEARLY IMPLICIT QSS METHODS

In this section, we introduce the new Linearly Implicit QSS (LIQSS) methods of first and second order.

A First Order LIQSS Method

The idea of LIQSS is very similar to BQSS. The only difference is that, when a value for q_j so that x_j goes to it cannot be found, instead of adding the perturbation term Δf_j to enforce the situation $\dot{x}_j = 0$, LIQSS tries to find the value $\hat{q}_j$ that provokes that situation in a linearly implicit way.

In order to illustrate this idea, we shall simulate System (4) from the same initial conditions and quantum than before

In $t = 0$, we can chose $q_2 = 19$ or $q_2 = 21$ according to the sign of $\dot{x}_2(t)$. In both cases, $\dot{x}_1(0) > 0$ so the quantized future value of x_1 will be $q_1(0) = 1$. On the other hand, if we choose $q_2(0) = 21$ then $\dot{x}_2(0) = -180 < 0$ and if we choose $q_2(0) = 19$, it results $\dot{x}_2(0) = 20 > 0$ so there exist a point $19 < \hat{q}_2(0) < 21$ in which $\dot{x}_2(0) = 0$. The value for $\hat{q}_2(0)$ can be calculated (exploiting the linear dependence of $\dot{x}_2$ with q_2) as

$$\hat{q}_2(0) = 21 - \frac{-180}{-100} = 19.2$$

Then, the state derivatives result: $\dot{x}_1(0) = 0.192$, $\dot{x}_2(0) = 0$.

The next change in q_1 is scheduled for $t = 1/0.192 \approx 5.2083$ while the corresponding in q_2 is scheduled for $t = \infty$

Then, the next step takes place in $t = 1/0.192 \approx 5.2083$. At this time, $x_1 = 1$ and $x_2 = 0$. After that, $q_1(5.2083) = 2$ (because $\dot{x}_1(5.2083) > 0$). If we reevaluate $\dot{x}_2$ for $q_2 = 19$ and $q_2 = 21$ it results lower than zero in both cases, so the correct value is $q_2(5.2083) = 19$ because in this way x_2 goes to q_2.

With this values of q_1 and q_2 de following derivatives states are obtained: $\dot{x}_1(5.2083) = 0.19$ and $\dot{x}_2(5.2083) = -80$. The next change in q_1 is scheduled to $t = 1/0.192 + 1/0.19 \approx 10.47149$, while the one in q_2 is scheduled to $t = 1/0.192 + 1/80 \approx 5.22083$. So, the next step is given in $t = 5.22083$, when x_2 reaches q_2.

Calculations follow in this way. Fig.3 show the evolution of $q_1(t)$ and $q_2(t)$ through 500 units of simulated time. As it can be seen, in this method the fast oscillations of q_2 are not present. In this way, q_1 changes 21 times and q_2 changes 25 times, which totalizes 46 steps (this constitutes a rather decent result for a stiff system).

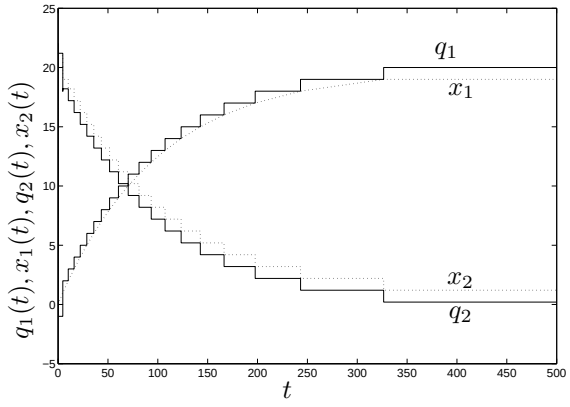

Figure 3: LIQSS Simulation

B LIQSS Definition

Given System (1), the LIQSS method approximates it by Eq.(2), where each q_j is defined by the following function

$$q_j(t) = \begin{cases} \underline{q}_j(t) \text{ if } f_j(\mathbf{q}(t),\mathbf{u}(t))(\overline{q}_j(t) - x_j(t)) \leq 0 \\ \overline{q}_j(t) \text{ if } f_j(\mathbf{q}(t),\mathbf{u}(t))(\overline{q}_j(t) - x_j(t)) \geq 0 \\ \quad \wedge f_j(\mathbf{q}(t),\mathbf{u}(t))(\overline{q}_j(t) - x_j(t)) > 0 \\ \widetilde{q}_j(t) \text{ otherwise} \end{cases} \tag{7}$$

with

$$\underline{q}_j(t) = \begin{cases} \underline{q}_j(t^-) - \Delta Q_j \\ \quad \text{if } x_j(t) - \underline{q}_j(t^-) \leq 0 \\ \underline{q}_j(t^-) + \Delta Q_j \\ \quad \text{if } x_j(t) - \underline{q}_j(t^-) \geq 2 \cdot \Delta Q_j \\ \underline{q}_j(t^-) \text{ otherwise} \end{cases} \tag{8}$$

$$\overline{q}_j(t) = \underline{q}_j(t) + 2\Delta Q_j \tag{9}$$

$$\widetilde{q}_j(t) = \begin{cases} \overline{q}_j(t) - \frac{1}{A_{jj}} \cdot f_j(\overline{\mathbf{q}}^j(t),\mathbf{u}(t)) & \text{if } A_{jj} \neq 0 \\ q_j(t^-) \quad \text{otherwise} \end{cases} \tag{10}$$

where $\overline{\mathbf{q}}^j(t)$ is equal to $\mathbf{q}(t^-)$ except for the j–th component, where it is equal to $\overline{q}_j$ and $A_{j,j}$ is the j,j component of the Jacobian matrix evaluated in $\overline{\mathbf{q}}^j$, i.e.,

$$A_{jj} = \left.\frac{\partial f_j}{\partial x_j}\right|_{\overline{\mathbf{q}}^j, u(t^-)} \tag{11}$$

As we shall see now, when $A_{j,j} \neq 0$, setting $q_j = \widetilde{q}_j$ provokes (in the linear case) the situation $\dot{x}_j = 0$.

C Calculation of $\widetilde{q}_j$

We take $\underline{\mathbf{q}}^j(t)$ equal to $\overline{\mathbf{q}}^j(t)$, except that the j–th component is $\underline{q}_j$. Notice that we use $\widetilde{q}_j$ when f_j changes the sign between $\underline{q}_j$ and $\overline{q}_j$. Thus, an intermediate point $\hat{q}_j$ exists where $f_j = 0$.

We take $\hat{\mathbf{q}}^j(t)$ equal to $\overline{\mathbf{q}}^j(t)$, except that the j–th component is $\hat{q}_j$.

Defining

$$A_j = \left.\frac{\partial f_j}{\partial x}\right|_{\overline{\mathbf{q}}^j, u(t^-)}$$

and the residual

$$g(\mathbf{x},\mathbf{u}) = f_j(\mathbf{x},\mathbf{u}) - A_j\mathbf{x},$$

calling $\hat{q}_j$ the point where $f_j = 0$, we can write

$$\begin{aligned} f_j(\overline{\mathbf{q}}^j,\mathbf{u}) &= A_j\overline{\mathbf{q}}^j - A_{j,j}\overline{q}_j + A_{j,j}\overline{q}_j + g(\overline{\mathbf{q}}^j,\mathbf{u}) \\ f_j(\underline{\mathbf{q}}^j,\mathbf{u}) &= A_j\underline{\mathbf{q}}^j - A_{j,j}\underline{q}_j + A_{j,j}\underline{q}_j + g(\underline{\mathbf{q}}^j,\mathbf{u}) \\ f_j(\hat{\mathbf{q}}^j,\mathbf{u}) &= A_j\hat{\mathbf{q}}^j - A_{j,j}\hat{q}_j + A_{j,j}\hat{q}_j + g(\hat{\mathbf{q}}^j,\mathbf{u}) \end{aligned}$$

Taking into account that $A_j\overline{\mathbf{q}}^j - A_{j,j}\overline{q}_j = A_j\hat{\mathbf{q}}^j - A_{j,j}\hat{q}_j$ (because $\overline{\mathbf{q}}^j$ and $\hat{\mathbf{q}}^j$ only differ in the j–th component) and considering that $f_j(\hat{\mathbf{q}}^j,\mathbf{u}) = 0$, we can solve the previous equations for $\hat{\mathbf{q}}^j$, obtaining

$$\hat{q}_j = \overline{q}_j - \frac{f_j(\overline{\mathbf{q}}^j,\mathbf{u})}{A_{j,j}} + \frac{g(\overline{\mathbf{q}}^j,\mathbf{u}) - g(\hat{\mathbf{q}}^j,\mathbf{u})}{A_{j,j}} \tag{12}$$

In order to estimate $A_{j,j}$ we can use the expresion

$$A_{j,j} \approx \frac{f_j(\overline{\mathbf{q}}^j,\mathbf{u}) - f_j(\underline{\mathbf{q}}^j,\mathbf{u})}{\overline{q}_j - \underline{q}_j} \tag{13}$$

If f_j depends linearly on q_j, Eq.(13) gives the exact value of the Jacobian. Moreover, in that case the last term in Eq.(12) results zero and $\widetilde{q}_j = \hat{q}_j$, i.e., the value of $\widetilde{q}_j$ provokes $\dot{x}_j = 0$.

In a nonlinear case, we will have $\widetilde{q}_j \approx \hat{q}_j$, and $\dot{x}_j \approx 0$. Although the oscillation will not disappear in this case, they will have a low frequency.

This is analogous to what linearly implicit discrete time methods do by solving the implicit equation only for the linear part of the problem, and this is the reason for calling LIQSS to this method.

D Second Order LIQSS: Basic Idea

The second order linearly implicit method, called LIQSS2, was developed combining the ideas of QSS2 and LIQSS.

The quantized variables of this new method are piecewise linear instead of being piecewise constant and they are chosen in such a way that $\ddot{x}_j \cdot (q_j - x_j) > 0$, i.e., so that x_j goes toward q_j.

As an example, in Figure 4 a general trajectory is shown using this quantization idea.

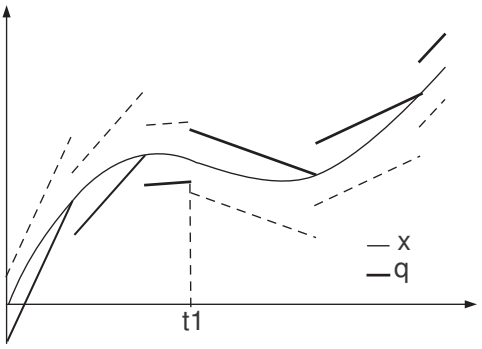

Figure 4: LIQSS2 Trajectories

Analogously to the case of LIQSS and BQSS, it can happen that the sign of $\ddot{x}_j$ changes when we start a new segment of $q_j(t)$. Then, an intermediate slope m_j exists that makes $\ddot{x}_j = 0$. In this case, we can also select the initial value q_j of the new segment so that $\dot{x}_j = m_j$, i.e., we can make the state trajectory to run parallel to the quantized trajectory so that no events are generated. Both values, q_j and m_j, can be easily obtained when $\dot{x}_j$ depends linearly on q_j.

If we simulate System (4) from the same initial conditions but whit quantums $\Delta q_1 = 0.1$ and $\Delta q_2 = 0.1$ (times smallers), the LIQSS2 method only performs 59 steps(20 changes in q_1 and 49 in q_2). The simulation results can be seen in Figure 5

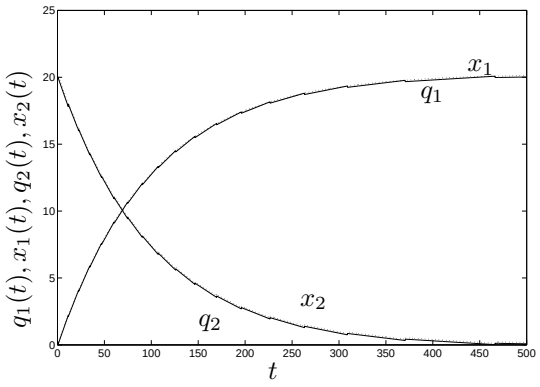

Figure 5: LIQSS2 Simulation

E LIQSS2 Definition

Given the system (1), the LIQSS2 method approximates it by (2), where each component q_j is defined as:

$$q_j = \begin{cases} \overline{q}_j(t) & \text{if } \ddot{x}_j(t) > 0 \vee \\ & (\ddot{x}_j(t) = 0 \wedge \dot{x}_j(t) > m_j) \\ \underline{q}_j(t) & \text{if } \ddot{x}_j(t) < 0 \vee \\ & (\ddot{x}_j(t) = 0 \wedge \dot{x}_j(t) <= m_j) \\ \widetilde{q}_j(t) & \text{otherwise} \end{cases} \tag{14}$$

with

$$\underline{q}_j(t) = \begin{cases} x_j(t_0) - \Delta Q_j & \text{if } t = t_0 \\ \underline{q}_i(t^-) + \Delta Q_j & \\ \quad \text{if } (x_j(t) = \underline{q}_j(t^-) + 2\Delta Q_j & \\ \underline{q}_i(t^-) - \Delta Q_j & \\ \quad \text{if } (x_j(t) = \underline{q}_j(t^-) & \\ \underline{q}_j(t_j) + m_j \cdot (t - t_j) & \text{otherwise} \end{cases} \tag{15}$$

$$\overline{q}_j(t) = \underline{q}_j(t) + 2 \cdot \Delta Q_j \tag{16}$$

$$\widetilde{q}_j(t) = \begin{cases} \dfrac{m_j(t) - \ddot{x}(t^-)}{A_{j,j}} + q_j(t^-) & \\ \quad \text{if } A_{j,j} \neq 0 & \\ q_j(t^-) \quad \text{otherwise} & \end{cases} \tag{17}$$

and

$$m_j = \begin{cases} \dot{x}_j(t^-) \text{ if } \ddot{x}_j(t) \cdot \ddot{x}_j(t^-) > 0 \vee A_{j,j} = 0 \\ m_j(t^-) - \ddot{x}_j(t^-)/A_{j,j} \quad \text{otherwise} \end{cases} \tag{18}$$

Note that in Eq.(18), the condition $\ddot{x}_j(t) \cdot \ddot{x}_j(t^-) < 0$ means that an intermediate value m_j exists so that $\ddot{x}_j = 0$. In a linear system this value can be calculated analogously to LIQSS with the expression $m_j(t) = m_j(t^-) - \ddot{x}_j(t^-)/A_{j,j}$. In a nonlinear case, we shall also obtain an approximate value.

IV THEORETICAL PROPERTIES

We shall treat here the most important properties of the LIBQSS methods. We shall show first that the methods perform a finite number of steps in a finite interval of time (this guarantees that the simulation time will always advance). Then we shall analyze the stability and accuracy properties.

A Trajectories and Legitimacy of LIBQSS

A crucial requirement of QSS methods is the legitimacy condition, which ensures that a finite number of events occurs in any finite interval of time. The following theorem proves this property for the first order LQISS method.

Theorem 1. *Suppose that function* $\mathbf{f}$ *in* (1) *is bounded in a domain* $D \times D_u$*, where* $D \subset \Re^n$*,* $D_u \subset \Re^m$ *and assume that the trajectory* $\mathbf{u}(t) \in D_u$ *is piecewise constant. Then,*

1. *Any solution* $\mathbf{x}(t)$ *of* (2) *is continuous while* $\mathbf{q}(t)$ *remains in* D*.*
2. *The trajectory* $\mathbf{q}(t)$ *is piecewise constant while it remains in* D*.*

Proof. The proof of (1) is straightforward since, according to (2), the derivative of $\mathbf{x}$ is bounded.

For the item (2), in order to prove that $\mathbf{q}$ is piecewise constant it is necessary to ensure that it only experiences a finite number of changes in any finite interval of time.

Let (t_1, t_2) be an arbitrary interval of time in which $\mathbf{q}(t)$ remains in D. We shall prove that, within this interval, $\mathbf{q}(t)$ has a finite number of changes.

The assumptions of the theorem ensure that $\mathbf{f}(\mathbf{q}, \mathbf{u})$ is bounded and, taking into account the relation between x_j and q_j, positive constants $\bar{f}_j$ exist so that, for $t \in (t_1, t_2)$

$$|\dot{x}_j(t)| \leq \bar{f}_j; \text{ for } j = 1, \ldots, n.$$

Let $t_c \in (t_1, t_2)$ and suppose that $\overline{q}_j(t_c^-) \neq \overline{q}_j(t_c^+)$. According to (9), this situation cannot be repeated until $|x_j(t) - x_j(t_c)| \geq \Delta Q_j$. Thus, the minimum time interval between two discontinuities in $\overline{q}_j(t)$ is

$$t_j = \frac{\Delta Q_j}{\bar{f}_j}$$

Then, calling $\overline{n}_j$ the number of changes of $\overline{q}_j(t)$ in the interval (t_1, t_2), it results that

$$\overline{n}_j \leq (t_2 - t_1)\frac{\bar{f}_j}{\Delta Q_j}$$

Since $\mathbf{u}(t)$ is piecewise constant, it will perform a finite number of changes n_u in the interval (t_1, t_2).

The definition of q_j ensures that it can only change when $\overline{q}_j(t)$ changes or when there is a change in some other quantized or input variable ($q_i(t)$ or $u_i(t)$) that inverts the sign of $\dot{x}_j$.

In conclusion, changes in $q_j(t)$ are linked to changes in some $\overline{q}_i(t)$ or $u_i(t)$. Thus, the total number of changes will be equal or less than the sum of all the changes in those variables, i.e.,

$$n_j \leq n_u + (t_2 - t_1)\sum_{i=1}^{n} \frac{\bar{f}_i}{\Delta Q_i}$$

which is a finite number. □

B Perturbed representation

The theorical properties of the QSS methods are based in a perturbed representation of the original system (1) that is equivalent to the approximation Eq.(2). Defining $\Delta\mathbf{x}(t) = \mathbf{q}(t) - \mathbf{x}(t)$ each row of System (2) can be rewritten as:

$$\dot{x}_i = f_i(\mathbf{x}(t) + \mathbf{\Delta x}(t), \mathbf{u}(t)) \tag{19}$$

From (7), (8), (9) and (10) in the definition, it can be ensured that each component $\Delta x_i(t)$ of $\mathbf{\Delta x}(t)$ is bounded by

$$|\Delta x_i(t)| \leq 2 \cdot \Delta Q_i \tag{20}$$

where ΔQ_i is the quantization adopted for $x_j(t)$. Thus, the LIQSS methods simulate an approximate system which only differ from the original SES(1) due to the presence of the bounded state perturbation.

C Global Error Bound and Stability

Given the LTI system

$$\dot{\mathbf{x}}_a(t) = A\mathbf{x}_a(t) + B\mathbf{u}(t) \tag{21}$$

were A is a Hurwitz matrix with Jordan canonical form $\Lambda = V^{-1}AV$, the LIQSS(2) approximation simulates the system

$$\dot{\mathbf{x}} = A(\mathbf{x}(t) + \Delta\mathbf{x}(t)) + B\mathbf{u}(t) \tag{22}$$

Defining the error as $e(t) = x(t) - x_a(t)$, and following the procedure of [4] and [1], it results that

$$|e(t)| \leq |V||\mathbb{R}\mathrm{e}(\Lambda)^{-1}\Lambda||V^{-1}|2\Delta Q \tag{23}$$

where ΔQ is the vector of quantum adopted.

The error bound is twice the error bound of QSS, QSS2 and QSS3.

D Equilibrium Points

One of the drawbacks of BQSS was the appearence of spurious equilibrium points. Due to the term $\Delta\mathbf{f}$, Eq.(6) admits equilibrium points also when $\mathbf{f}(\mathbf{q}, \mathbf{u}) \neq 0$.

However, the only possibility in which LIQSS or LIQSS2 arrive to an equilibrium point is when $\mathbf{f}(\mathbf{q}, \mathbf{u}) = 0$, i.e., when the quantized variables reach an equilibrium point.

V CONCLUSIONS

We presented two new QSS methods that are based on linearly implicit principles. We showed that these methods, called LIQSS, satisfy the global error bound property of QSS method, but they can also simulate stiff systems.

The methods improve the performance of BQSS by solving the problem of the spurious equilibrium points and by increasing to 2 the order of accuracy.

The applications and examples are presented in a companion paper [9]

References

[1] F.E. Cellier and E. Kofman. *Continuous System Simulation*. Springer, New York, 2006.

[2] Ernst Hairer, Syvert Norsett, and Gerhard Wanner. *Solving Ordinary Differential Equations I. Nonstiff Problems*. Springer, 2nd edition, 1993.

[3] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II. Stiff and Differential–Algebraic Problems*. Springer, 1st edition, 1991.

[4] E. Kofman. A Second Order Approximation for DEVS Simulation of Continuous Systems. *Simulation*, 78(2):76–89, 2002.

[5] E. Kofman. Discrete Event Simulation of Hybrid Systems. *SIAM Journal on Scientific Computing*, 25(5):1771–1797, 2004.

[6] E. Kofman. A Third Order Discrete Event Simulation Method for Continuous System Simulation. *Latin American Applied Research*, 36(2):101–108, 2006.

[7] E. Kofman and S. Junco. Quantized State Systems. A DEVS Approach for Continuous System Simulation. *Transactions of SCS*, 18(3):123–132, 2001.

[8] E. Kofman, G. Migoni, and F.E. Cellier. Integración por Cuantificación de Sistemas Stiff. Parte I: Teoría. In *Proceedings of AADECA 2006*, Buenos Aires, Argentina, 2006.

[9] G. Migoni and E. Kofman. Linearly Implicit Discrete Event Methods for Stiff ODEs. Part II: Applications. Technical Report LSD0207, LSD–UNR, 2007. Enviado a RPIC 2007.

[10] G. Migoni, E. Kofman, and F.E. Cellier. Integración por Cuantificación de Sistemas Stiff. Parte II: Aplicaciones. In *Proceedings of AADECA 2006*, Buenos Aires, Argentina, 2006.

[11] B. Zeigler, T.G. Kim, and H. Praehofer. *Theory of Modeling and Simulation. Second edition*. Academic Press, New York, 2000.

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