Computationally efficient methods for digital control

Duarte Antunes, João P. Hespanha, Carlos Silvestre, Nitin Kataria and Forrest Brewer

***Abstract*— The problem of designing a digital controller is considered with the novelty of explicitly taking into account the computation cost of the controller implementation. A class of controller emulation methods inspired by numerical analysis is proposed. Through various examples it is shown that these methods are capable of outperforming the conventional ones, in the sense of achieving a better performance/computation cost tradeoff. The performance indexes consider (i) the error between the continuous-time controller to be emulated and the digital controller; and (ii) the closed-loop induced norm.**

I. INTRODUCTION

With the advent of cost-effective processors, fast and reliable analog-digital (A/D) and digital-analog (D/A) converters, digital control has become the tool per excellence for controller implementation. Several approaches have been taken for designing a digital controller, essentially following two main directions: emulation and direct design. While in emulation, given a continuous-time controller the digital controller is obtained by using a numerical discretization method, in direct design, a discrete-time model of the plant is first obtained and the digital controller is then synthesized in discrete-time. We also include in this latter group the sampled-data design approach, presented in [1], [2], that directly takes into account in the design the continuous-time performance specifications. These approaches consider a fixed sampling period h for the measurements and actuation update, and recover the performance of the ideal continuous-time linear controller as $h \to 0$ [1]. However, using a very small sampling period becomes impractical for digital controllers implemented on embedded processors with limited computation resources. Moreover, the computation effort is tied in with the energy consumption of the processor, which generally needs to be kept as small as possible. This motivates the search for efficient digital controllers with minimal computation requirements, which is the subject of the present paper. More specifically, the problem addressed herein can be stated as a tradeoff performance/cost: Given a plant to be controlled with an A/D and D/A interface, find a digital controller algorithm to satisfy a given performance criteria with the least computation cost.

The work of C.Silvestre and D.Antunes was partially supported by FCT (ISR/IST pluriannual funding) through the POS Conhecimento Program that includes FEDER funds, the PTDC/MAR/64546/2006 OBSERVFLY project, and the FCT PhD Student Scholarship, SFRH/BD/24632/2005.

The work of J. Hespanha was supported by the National Science Foundation under Grant No.CNS-0720842

D. Antunes and C. Silvestre are with the Dep. of Elect. Eng. and Comp. Science, Instituto Superior Técnico, ISR, 1046-001 Lisboa, Portugal. {dantunes,cjs}@isr.ist.utl.pt

João P. Hespanha, Nitin Kataria and Forrest Brewer are with the Dept. of Elect. and Comp. Eng., University of California, Santa Barbara, CA 93106-9560, USA. {hespanha,nitin,forrest}@ece.ucsb.edu

The emulation approach is followed with the novelty of directly taking into account the computation cost of the digital controller implementation. In this setting, the main contributions are two fold (i) show that nonuniform sampling can reduce the controller computation cost; and (ii) propose a class of emulation methods inspired from numerical analysis that can significantly outperform classical implementations, in the sense of achieving a better performance/computation cost tradeoff. By nonuniform sampling we mean that the output sampling and actuation update operations might not occur at evenly spaced time instants. This is motivated by the availability of increasingly faster A/D and D/A converters, which generally allow for on demand conversions.

The performance indexes considered herein are similar to existing ones in the literature for related problems and consider (i) the error between the continuous-time controller to be emulated and the digital controller interfaced through A/D and D/A converters [3], [4]; and (ii) the closed-loop L_2 induced norm of the feedback interconnection of a continuous-time plant with the digital controller [1].

Regarding related work in the literature, see [1], [3], for a discussion of various issues associated with digital control, including the computation cost. This latter paper, outlines performance indexes to evaluate digital controller implementations, some of which, are used in the present paper. With respect to the relation between numerical analysis methods and control, see [5], where the similarities between the two areas are pointed out in terms of stability definitions. Among the many references available in the literature for non-conventional digital control algorithms, see [6] for a discussion on computation saving.

The remainder of this paper is organized as follows. Section II presents the problem statement and Section III discusses the performance indexes and costs considered herein. Section IV outlines numerical integrations methods and provides an example of their applicability to control. The proposed emulation method is presented in Section V, followed by the results in the considered performance indexes in Section VI. The conclusions are provided in Section VII.

Notation: The space of square integrable functions defined in $[0, \infty)$ will be denoted by L_2, and the space of square summable sequences $\{x_0, x_1, \ldots\}$ will be denoted by l_2. Further notation will be added when necessary.

II. PROBLEM STATEMENT

This paper addresses the digital implementation of a SISO controller to be connected to a continuous-time process. The input of the controller is a sequence y_i that results from sampling the continuous-time process output $y(t)$. The

controller output is a sequence u_l from which the continuous-time control signal $u(t)$ is generated by a hold device. Formally, the sample and hold operations are modeled by the following operators

$$\begin{aligned} S_{\sigma_i}: &\ L_2 \mapsto l_2, \quad y_i = y(\sigma_i) \\ H_{\tau_l}: &\ l_2 \mapsto L_2, \quad u(t) = u_l, \quad t \in [\tau_l, \tau_{l+1}), \end{aligned} \tag{1}$$

where the elements of the sequences of sampling times $\{\sigma_i\}_{i\geq 0}$, $0 < \sigma_i < \sigma_{i+1}$, and actuation updating times $\{\tau_l\}_{l\geq 0}$, $0 < \tau_l < \tau_{l+1}$, take values on $\mathbb{R}_{\geq 0}$. We assume that the measured output sampling and actuation update operations are available at any time instant and the sampling and updating rates might be nonuniform, that is, $\sigma_{i+1} - \sigma_i$ and $\tau_{l+1} - \tau_l$ do not need to be either constant or equal.

In broad terms, given a plant with a D/A and A/D interface (1) and a performance index related to the closed-loop behavior we want to find an efficient design method for the digital control algorithm yielding a good tradeoff between the performance index and the cost. Hence, we need to define what we mean by (i) digital control algorithm, (ii) performance index, (iii) computation cost, and (iv) efficient design method.

By *Digital Control Algorithm (DCA)* we mean a numerical algorithm implemented digitally, that causally maps y_i into u_l, that is, provides $u(\tau_l)$ based on $\{y(\sigma_i) : \sigma_i \leq \tau_l\}$.

Each DCA has associated a cost, measured for example in terms of computations per second, and an index, which measures the performance of the DCA according to some specifications. The two main ingredients to define a DCA are (i) the selection of the sampling times τ_l and σ_i and the associated *input-output map* from u_l to y_i; and (ii) implementation of this map in terms of a *computational algorithm*. The value of the performance index, depends solely on (i), whereas the computation cost depends most directly on (ii), but also indirectly on (i). To discuss computational algorithms for DCA and their associated costs, we need to provide specific pseudocode for (ii) that implements their input output map with a reduced amount of computation. For example, the implementation of a FIR system of order N working at a sampling period of h, is specified by the sampling times $\tau_k = \sigma_k = t_k$, $t_{k+1} - t_k = h$ and input-output map $Q(z) = \sum_{i=0}^{N} c_{i+1} z^{-i}$ and can be described by the following computational algorithm

DCA $\mathcal{Q}_{FIR}$
Initializations, $Y = \text{zeros}(1, N)$.
At each time t_k do,
$u_k = c_1 y(t_k) + \sum_{i=1}^{N} c_{i+1} Y[i]$,
shiftright(Y),
$Y[1] = y(t_k)$.

However, the same input-output map can be implemented with a state-space based computational algorithm which typically requires a different number of operations.

The performance index and cost are simply maps, that assign a positive number to a DCA, and are denoted by $\mathcal{I}$ and $\mathcal{C}$, respectively. By convention, we intend to minimize $\mathcal{I}$ and $\mathcal{C}$. Both the emulation methods considered herein and conventional methods depend on a design parameter η, such as the sampling period h, that is chosen to yield a good tradeoff between performance and computation cost. A reasonable way to compare the efficiency of DCAs is in terms of the tradeoff obtained while varying η. More specifically an efficiency curve for a parameterized class of DCAs $\mathcal{A}$ is defined as the set

$$(\mathcal{C}(\mathcal{A}(\eta)), \mathcal{I}(\mathcal{A}(\eta))), \eta \in \Xi. \tag{2}$$

A class of DCAs is more efficient than other if its efficiency curve lies bellow.

III. Computation cost and Performance Indexes

A. Cost

The computation cost of implementing a digital controller in a digital processor depends on the number of multiplications, additions, and memory accesses, among other factors. As a good approximation it suffices to consider the number of multiplications per second, since the computation time in a digital processor is often largely dominated by this number [7]. This is the approximation followed in the present paper. The efficient structures for implementing a digital filter discussed in [7] are the cascade, parallel and direct forms. These require a minimum of $\frac{(M+N+1)}{h}$ multiplications per second for implementing a filter at a sampling period h taking the form $\frac{\sum_{k=0}^{M} b_k z^{-k}}{1-\sum_{k=1}^{N} a_k z^{-k}}$. Notice that, in general, $M = N$ for the digital controllers resulting from zero-order hold (ZOH), first order hold (FOH), and bilinear discretizations [3], which reduces to $M = N - 1$ for the ZOH if the continuous-time controller is strictly proper.

B. Performance Indexes

The first two performance indexes considered herein are related to the error system shown in Fig. 1, where $K(s)$ is a stable continuous-time controller that we intend to approximate by the DCA $\mathcal{A}$, and $F(s)$ is a stable strictly proper system. The first performance index is the induced norm of the operator $L_2 \mapsto L_2$, $e = (K - H_{\tau_l}\mathcal{A}S_{\sigma_i})Fw$, and it will be denoted by $\mathcal{I}_E(.)$. Notice that this norm is the same as the one considered in the extensively studied model reduction and weighted model reduction problems, and therefore there is a bulk of knowledge on how this open-loop norm impacts on the closed-loop performance [3]. For example, denoting the plant transfer function by $P(s)$ and choosing $F = \frac{P}{1+KP}$, if the L_2 induced norm $w \mapsto e$ is less than 1 one can guarantee closed-loop stability. The second performance index, denoted by $\mathcal{I}_{ES}(.)$, is a sampled version of the previous one and is given by the induced norm of the operator $L_2 \mapsto L_2$, $e_s = H_{\tau_l}S_{\tau_l}(K - H_{\tau_l}\mathcal{A}S_{\sigma_i})Fw$. A similar performance index is considered in [4] for the typical case, $t_k = \tau_k = \sigma_k$, $t_{k+1} - t_k = h$.

The last performance index is given by considering directly the closed loop performance in the four-port framework depicted in Fig. 2, defined in terms of the closed-loop L_2 induced norm $w \mapsto z$. This performance index is denoted by $\mathcal{I}_{CL}(\mathcal{A}) := \|\mathcal{F}(G, H_{\tau_l}\mathcal{A}S_{\sigma_i})\|_{L_2 \mapsto L_2}$, where $\mathcal{F}$ denotes the feedback connection.

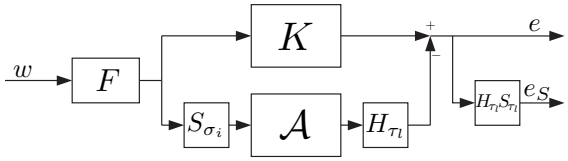

Fig. 1. Error System

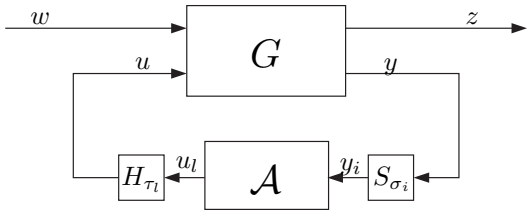

Fig. 2. Closed-loop system

Notice that to compute both induced norms associated with Fig. 1 we need to restrict $K(s)$ and $\mathcal{A}$ to be stable, and $F(s)$ must be strictly proper, otherwise the norm of the operators $w \mapsto e_s$, $w \mapsto e$ would be unbounded [1], [4]. The DCAs considered herein have an input-output map description in terms of periodically time-varying linear systems. The DCA $\mathcal{A}$ is defined to be stable if this system description is stable, which can be easily verified [8]. The induced norms considered can be computed using extensions to the periodic case of the sampled-data lifting tools provided in [1]. The derivations are omitted due to space limitations.

IV. NUMERICAL ANALYSIS AND EMULATION METHODS

The traditional emulation methods used in control are inspired in numerical techniques for ODEs and in numerical integration methods. This originates from the fact that a linear controller $K : L_2 \mapsto L_2$ can be equivalently described by the linear ODE

$$\begin{cases} \dot{x} = Ax + By, & x(t) = 0 \\ u = Cx + Dy \end{cases},$$

or by the convolution integral

$$u(t) = \int_0^t C \exp(A(t-s))By(s)ds + Dy(t)\,.$$

The Bilinear (particular case of Adams-Moulton), Runge-Kutta, Adams-Moulton and Adams-Bashforth methods [9] are numerical methods with a direct application in control by their iterative nature. The ZOH, FOH are based on approximating the convolution integral. This integral interpretation is the basis of the method to be presented shortly. Therefore we briefly outline numerical integration methods and present an example of application to controller emulation.

A. Numerical Integration and Gaussian Quadrature

The numerical integration methods approximate an integral of the form

$$I(f) = \int_a^b W(x)f(x)dx \tag{3}$$

by a quadrature formula $Q(f) = \sum_{i=0}^{I} A_i f(x_i)$, $a \le x_i < x_{i+1} \le b$. We assume the weighting function $W(x)$ verifies:

P1: $\int_a^b |x|^n W(x)dx$ is integrable and finite for all $n \ge 0$.

The integration methods are classified by the degree of precision P, which is the largest polynomial order for which the quadrature formula is exact, meaning that

$$Q(x^j) = I(x^j),\ j \in \{0, 1, \ldots, P\}. \tag{4}$$

Well-known examples, are the Trapezoidal and Simpson formulas, which are particular cases of the Newton-Cotes formulas, obtained by using evenly spaced nodes, $x_i = a + \frac{i}{I}(b-a), i = 0, \ldots, I$. For future reference we write the equations for the trapezoidal rule Q_T and for the zero-order approximation rule Q_Z, which have degree of precision $P = 1$ and $P = 0$, respectively

$$Q_T = f(a)(\int_a^b \frac{(x-b)}{a-b}W(x)dx) + f(b)(\int_a^b \frac{(x-a)}{b-a}W(x)dx)$$
$$Q_Z = f(a)\int_a^b W(x)dx. \tag{5}$$

From these class of methods, denominated simple rules, composite rules can be constructed, by dividing $[a, b]$ into smaller intervals, applying an integration method with degree of precision P on each interval, and summing the contributions. The formula is then exact for piecewise polynomials of order P. An optimal method, in the sense that attains the maximal degree of precision $P = 2I+1$ is called gaussian quadrature. This method amounts to choosing the $2(I+1)$ degrees of freedom A_i and x_i, as to make the set of equations (4) hold for $P = 2I + 1$. These are nonlinear equations, difficult to solve for large I, and therefore a different approach is typically followed. Suppose besides P1, $W(x)$ verifies:

P2: $W(x)$ does not change sign in $[a, b]$.

P3: $\int_a^b W(x)g(x)dx = 0$ for some nonnegative continuous function $g(x)$, then $g(x) \equiv 0$ on (a, b).

Without loss of generality P_2 can be replaced by assuming $W(x)$ is non-negative in $[a, b]$. Then, the gaussian quadrature nodes x_i are given by the roots of the polynomial of order $I + 1$, from the set of orthogonal polynomials according to the inner product

$$< p_n(x), p_m(x) >= \int_a^b W(x)p_n(x)p_m(x)dx, \tag{6}$$

which can be obtained by Gram-Schmidt orthogonalization [9]. The coefficients A_i can then be obtained by solving the system (4) for $P = I$, which amounts to a linear set of equations after replacing the nodes x_i.

B. Example of application to controller emulation

To gain intuition, we start by considering a first order controller, which we intend to approximate by a DCA,

$$K = \left\{ \begin{bmatrix} \dot{x} \\ u \end{bmatrix} = \begin{bmatrix} a & b \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, x(t_0) = 0 \right., \tag{7}$$

and assume the actuation updating times are fixed at a given rate. Denoting by H the actuation update period, the value

of the actuation signal $u(t)$ provided by (7) at evenly spaced sampling times τ_l, $\tau_{l+1} - \tau_l = H$, verifies

$$u(\tau_{l+1}) = a_d u(\tau_l) + I_B(\underline{y}_l),$$

where $a_d = \exp(aH)$, $\underline{y}_l(t) = y(\tau_l + t), t \in [0, H)$ and $I_B(\underline{y}_l) = \int_0^H \exp(a(H-s))\underline{y}_l(s)ds$. This formula motivates a DCA that implements the following recursion

$$u_{l+1} = a_d u_l + Q_B(\underline{y}_l), \tag{8}$$

where $Q_B(\underline{y}_l) = \sum_{i=0}^{I} R_i \underline{y}_l(\sigma_i)$ is a quadrature formula for $I_B(\underline{y}_l)$. This DCA can be implemented as follows

DCA $\mathcal{E}$
Initializations, $U = 0$.
At each time $t = lH + \sigma_i$,
save $y(t)$ in memory.
At each time $t_l = lH$ do,
$U = a_d U + Q_B(\underline{y}_{l-1})$,
$u_l = U$.

We consider five integration methods for the quadrature formula Q_B. The first two use uniform sampling and are composite rules considering at each of L intervals of length $\frac{H}{L}$ (i) the zero-order approximation simple rule; and (ii) the trapezoidal simple rule (5). The associated DCAs are denoted (i) ZOH; and (ii) FOH, because they are equivalent to consider a discretization at a fast sampling rate, $h = \frac{H}{L}$, by the ZOH and FOH methods, respectively, and provide the actuation only at times kH. Using this same procedure for the bilinear discretization yields another method, denoted by (iii) BIL. The remaining two use nonuniform sampling based on (iv) a composite rule considering at each interval of length $\frac{H}{L}$ a gaussian quadrature formula with 2 nodes, $I = 2L - 1$; and (v) pure gaussian quadrature with $I + 1$ nodes. These are denoted (iv) comp. gauss; and (v) gauss. The number of multiplications per second is $(I + 1)H$ for (i) and $(I + 2)H$ for the remaining methods.

A reasonable performance index is the norm of the error sequence $\|u_l - u(\tau_l)\|_{l_2}$, when the input of the controller $y(t)$ belongs to a certain class of signals. This index corresponds to $\frac{\mathcal{I}_{ES}}{H}$, where $\mathcal{I}_{ES}$ is the index presented in Section III-B, when the class of signals is $\{y = Fw, w \in L_2\}$, where $F(s)$ is the filter shown in Fig.1. It is possible to write the DCA as a linear periodic input-output map and therefore $\mathcal{I}_{SE}$ can be computed as indicated in Section III-B.

Example 1: Consider $K(s) = \frac{1}{s+1}$, $H = 0.25$, and

$$F(s) = \left(\frac{w_c^2}{s^2 + \sqrt{2}w_c s + w_c^2}\right)^2, \tag{9}$$

for $w_c = 2\pi$. The efficiency curves (2) in the index $\frac{\mathcal{I}_{ES}}{H}$ are shown in Fig. 3 in a logarithmic scale, for different values of the parameter $\eta = I$. The results are impressive showing that for this performance index, choosing the sampling times coincident with gaussian quadrature nodes can drastically outperform the conventional methods. It is interesting to notice that, in this example, the DCAs inherits the characteristics of the numerical method used for Q_B [9]. For example, in the case of the trapezoidal rule the error decreases by four when the number of integration nodes for Q_B doubles. We will see, however, that in other performance indexes the improvement is not so drastic.

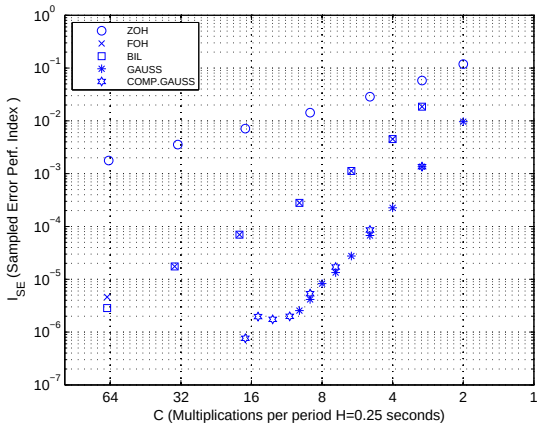

Fig. 3. Efficiency curves in index $\mathcal{I}_{ES}$ for DCAs of the illustrative example

V. NEW EMULATION METHOD

In this section, we present an emulation method based on numerical integration, and particularize it for gaussian quadrature.

A. Method Description

The continuous-time controller to be emulated by a DCA is denoted by

$$\begin{bmatrix} \dot{x} \\ u \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}, x(t_0) = 0. \tag{10}$$

Consider a time-interval H, which determines the periodicity of the DCA, and the time sequence $\rho_r = rH$. Between times ρ_r and ρ_{r+1}, the actuation is updated at times $t = \rho_r + \tau_l$, with $\tau_l \in (0, H)$ $\forall l \in \{1, \ldots, L-1\}$ and $\tau_0 = 0$. Using these definitions, the continuous-time controller (10) can be written in the form

$$\begin{cases} x_{r+1} = A_d x_r + \underline{B}(\underline{y}_r) \\ \underline{u}_r(\tau_l) = C^l x_r + D\underline{y}_r(\tau_l) + \underline{D}^l(\underline{y}_r) \end{cases}$$

where the output is considered at τ_l and $A_d = \exp(AH)$, $C^l = C\exp(A\tau_l)$, $\underline{y}_r(t) = y(\rho_r + t)$, $\underline{u}_r(t) = u(\rho_r + t)$, $t \in [0, H)$. The integral operators $\underline{B}$ and $\underline{D}^l$ can be written in the form

$$\begin{aligned} \underline{B}(\underline{y}_r) &= I_B(\underline{y}_r) = [I_B^1(\underline{y}_r) \ldots I_B^N(\underline{y}_r)]^T \\ I_B^n(\underline{y}_r) &= \int_0^H W_B^n(s)\underline{y}_r(s)ds, \\ \underline{D}^l(\underline{y}_r) &= I_D^l(\underline{y}_r) = \int_0^{\tau_l} W_D^l(s)\underline{y}_r(s)ds, \end{aligned} \tag{11}$$

where, denoting by $e_n \in \mathbb{R}^{1\times N}$ the nth standard basis vector,

$$\begin{aligned} W_B^n(s) &= e_n \exp(A(H-s))B, \quad n \in \{1, \ldots, N\} \\ W_D^l(s) &= C\exp(A(\tau_l - s))B, \quad \tau_l \in [0, H), \\ & \qquad l \in \{1, \ldots, L-1\}. \end{aligned} \tag{12}$$

To obtain a DCA we approximate these operators by the quadrature formulas,

$$Q_B^n(\underline{y}_r)=\sum_{i=0}^{I^n} R_i^n \underline{y}_r(\sigma_i^n),\ Q_D^l(\underline{y}_r)=\sum_{j=0}^{J^l} S_j^l \underline{y}_r(\nu_j^l). \quad (13)$$

The proposed DCA can then be described by

DCA $\mathcal{H}$
Initializations, X=zeros(N,1).
At each time $t = rH + \sigma_i^n$ and $t = rH + \nu_j^l$,
save $y(t)$ in memory.
At each time $t = rH$ do,
$X = A_d X + Q_B(\underline{y}_{r-1})$.
At each time $t = rH + \tau_l$ do,
$u_l = C^l X + Dy(t) + Q_D^l(\underline{y}_r)$.

For simplicity, we consider the same number of nodes for the formulas Q_B^n, that is, $I^n = I$, and evenly spaced actuation update times $\tau_{l+1} - \tau_l = \frac{H}{L}, l \in \{0, \dots, L-1\}$. Then, the DCA can be parameterized by

$$(H, L, J^l, \text{Meth}_l, I, \text{Meth}_I), \quad l \in \{1, \dots, L-1\}, \quad (14)$$

where H is the time period of the DCA, L is the number of actuation updating times τ_l within a period H. The parameters (J^l, Meth_l) and (I, Meth_I) provide the number of integration nodes and integration method used in Q_D^l, and Q_B^n, $\forall_n$, respectively, and determine σ_i^n and v_j^l.

We determine next the computation cost (number of multiplications) of $\mathcal{H}$, when an efficient block-diagonal realization for (10) is considered. Assuming A has N_r real and N_c pairs of complex conjugate simple eigenvalues, $N = N_r + 2N_c$, there exist a block-diagonal realization taking the form

$$\begin{aligned} A &= \text{diag}(a_1, \dots, a_{N_r}, A_1, \dots, A_{N_c}), \quad a_i \in \mathbb{R}, A_i \in \mathbb{R}^{2\times 2} \\ B &= \begin{bmatrix} b_1 & \dots & b_{N_r} & b_{c1}^T & \dots b_{cN_c}^T \end{bmatrix}^T, b_i \in \mathbb{R} \quad b_{ci}^T \in \mathbb{R}^{2\times 1} \\ C &= \begin{bmatrix} 1 & \dots & 1 & c_{c1} & \dots c_{cN_c} \end{bmatrix}, \quad c_{ci} = [1 \quad c_{di}] \in \mathbb{R}^{1\times 2}. \end{aligned} \quad (15)$$

The number of multiplications of the DCA $\mathcal{H}$ along a period H is given by $M = M_r + M_c + M_H$, where $M_r = N_r + N_r(I+1) + N_r(L-1)$, $M_c = 4N_c + 2N_c(I+1) + N_c(2L-1)$, and $M_H = \sum_{l=1}^{L-1}(J^l + 1)$. The terms in M_r and M_c account for the multiplications of A_d, Q_B and C^l, and M_H for the multiplications of Q_D. Adding up the terms yields,

$$M = N(I + 1 + L) + \sum_{l=1}^{L-1}(J^l + 1) + 3N_c. \quad (16)$$

The total number of multiplications M should be compared to $M_C = (2N+1)\frac{H}{h} = (2N+1)L$, which is the number of multiplications in a time period H needed by a conventional implementation of a digital controller with the same actuation updating period $h = \frac{H}{L}$. Notice that M can be smaller than M_C for large N. This means we can apply numerical integration methods with high degree of precision even for small I and J^l, use less number of multiplications and potentially obtain better results, due to the high efficiency of the numerical methods illustrated in Example 1.

It is possible to show that this DCA $\mathcal{H}$ is a linear periodic input-output map, and therefore the performance indexes considered in Section III can be computed as indicated therein.

B. Gaussian Sampling

Due to the optimality of gaussian quadrature in the sense already mentioned, we show how to particularize the proposed method to the case where the nodes and weights of the quadrature formulas Q_B^n and Q_D^l, given by (13), are chosen according to gaussian quadrature. The caveat is that the orthogonal polynomial approach, addressed in Section IV-A, requires the technical conditions P2 and P3, besides P1. In the next lemma we give conditions on the time period H under which these conditions hold.

Lemma 5.1: There exists a block-diagonal realization (15) of (10) such that the weighting functions $W_D^l(s)$, $W_B^n(s)$, given by (12), verify conditions P1-P3 in the interval $[a, b] = [0, H]$ where $H \in [0, \min\{\epsilon_1, \epsilon_2\})$ for
$\epsilon_1 = \sup\{\epsilon : C\exp(At)B \text{ does not change sign in } [0, \epsilon)\}$
$\epsilon_2 = \begin{cases} \frac{\pi}{\max(a_{cn})} & \text{if } \exists_n : a_{cn} \neq 0 \\ \infty & \text{otherwise} \end{cases}$,
where a_{cn}, $n \in \{1, \dots, N_c\}$ are the imaginary parts of the eigenvalues of A.

VI. Results

In this section we test the proposed emulation method in the three performance indexes $\mathcal{I}_{ES}$, $\mathcal{I}_S$ and $\mathcal{I}_{CL}$.

A. Error System Norms $\mathcal{I}_E$, $\mathcal{I}_{ES}$

Consider the continuous-time controller taken from [10](pp.565)

$$K(s) = -\frac{94.5(s+7.98)(s+2.52)}{(s+4.28 \pm 6.42j)(s+10.6)},$$

and the filter $F(s)$ of Fig.1 taking the form (9) for $w_c = 10\pi$. A first DCA $\mathcal{M}_1$ is obtained for the parameters (14) $L = 4$, $I = 1$, $J^l = 1, \forall_l$, using the pure gaussian quadrature for Meth_l and Meth_I. A second DCA $\mathcal{M}_2$ is obtained for the same parameters except for $I = 0$, $J^l = 0, \forall_l$. For an actuation update period of $h = \frac{H}{L}$, the number of multiplications per second (16) is $\frac{27}{4h}$ for $\mathcal{M}_1$, $\frac{21}{4h}$ for $\mathcal{M}_2$, $\frac{6}{h}$ for the ZOH, and $\frac{7}{h}$ for the FOH and bilinear methods. The efficiency curves (2) are shown in Fig. 4 in the index $\mathcal{I}_{ES}$ for $\eta = h \in \{0.04/2^{i-1}, i \in \{1, \dots, 8\}\}$, and in Fig. 5 in the index $\mathcal{I}_E$ for $\eta = h \in \{0.04/2^{i-1}, i \in \{1, \dots, 5\}\}$. In the index $\mathcal{I}_{ES}$ very significant amount of computation savings are obtained, whereas for $\mathcal{I}_E$ the results are more moderate. However, by properly choosing the degrees of freedom, as in the choice of $\mathcal{M}_2$, there is still a significant amount of computation savings.

B. Closed Loop

We consider here a standard L_2 induced norm problem, depicted in Fig. 7, which can be formulated in the framework of Fig. 2.

The plant P is a simple double integrator $P = \frac{1}{s^2}$. The frequency weights $W_1(s) = 1$, $W_2(s) = 0.2$, $Z_1(s) = 0.5$,

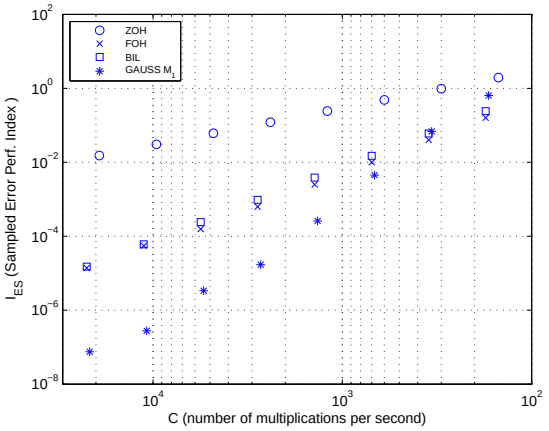

Fig. 4. Efficiency curves in index $\mathcal{I}_{ES}$ for various DCAs

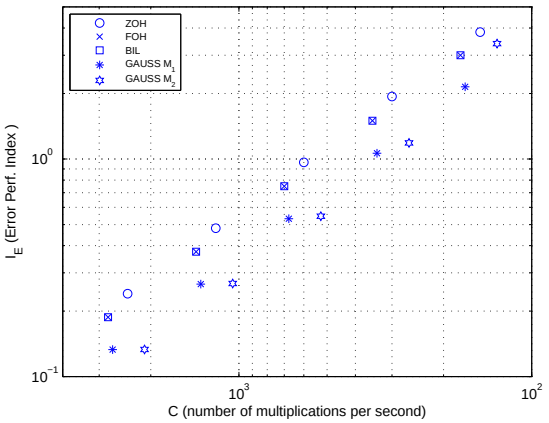

Fig. 5. Efficiency curves in index $\mathcal{I}_E$ for various DCAs

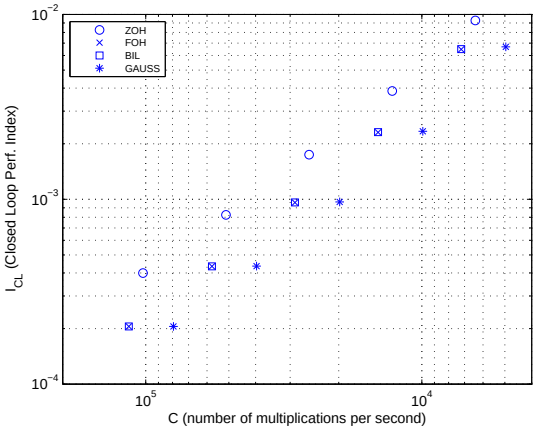

Fig. 6. Efficiency curves in index $\mathcal{I}_{CL}$ for various DCAs

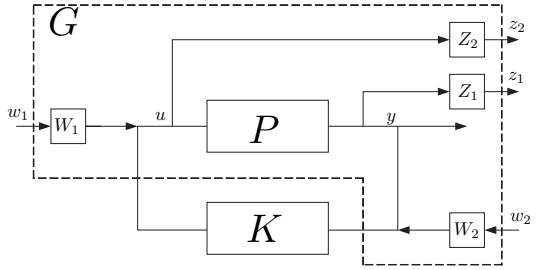

Fig. 7. H_∞ problem set-up

and $Z_2(s) = 0.5(\frac{s+10}{s+100})^2$ are chosen in order to make of the L_2 induced norm of $G : [w_1\, w_2] \mapsto [z_1, z_2]$, a useful index of the closed-loop behavior of the system, which we desire to minimize. The continuous-time optimal controller that minimizes the closed-loop induced norm is given by

$$K_{opt}(s) = -\frac{51.41(s^3 + 201.12s^2 + 1.02.10^4 s + 1.12.10^4)}{s^4 + 68.16s^3 + 1636s^2 + 1.98.10^4 s + 1.28.10^4}$$

For evaluating the performance of the digital implementations we limited the bandwidth of the disturbances w_2 by setting the frequency weight $W_2(s)$ to $W_2(s) = \frac{p_2^2}{s^2+\sqrt{2}p_2 s+p_2^2}$ with $p_2 = 100$. This is in fact needed, to make the norm-computation well-posed[1]. The controller K_{opt} in this performance index yields $\gamma = \|\mathcal{F}(G, K_{\text{opt}})\|_{L_2 \mapsto L_2} = 0.7283$. Figure 6 shows the results in the performance index $I_{CL}(\mathcal{M}) := \|\mathcal{F}(G, H_{\tau_l}\mathcal{M}S_{\sigma_i})\|_{L_2 \mapsto L_2} - \gamma$ for the conventional methods and for a DCA $\mathcal{M}$ with parameters (14), $L = 5$, $I = 0$, and $J^l = 0$, $\forall_l$ and using the pure gaussian quadrature for Meth_l and Meth_I. The set of values considered for the actuation update rate $\eta = h$, which is the parameter of the efficiency curves (2), is $h \in \{0.00125/2^{i-1}, i \in \{1, \ldots, 5\}\}$. Once again, the proposed method outperforms the conventional ones.

VII. CONCLUSIONS

The design of computationally efficient digital controllers was tackled with a direct concern with the computation cost in the continuous-time controller emulation procedure. A technique was provided that takes advantage of highly efficient numerical integration methods such as the gaussian quadrature. The results show that moving away from the paradigm of using a uniform sampling rate can lead to significantly more efficient methods.

REFERENCES

[1] T. Chen and B. Francis, *Optimal Sampled-Data Control Systems*. Springer-Verlag, 1995.

[2] B. Bamieh and J. Pearson, "A general framework for linear periodic systems with applicationsto h_∞ sampled-data control," *IEEE Transactions on Automatic Control*, vol. 37, no. 4, pp. 418–435, Apr. 1992.

[3] B. Anderson, "Controller design: moving from theory to practice," *IEEE Transactions on Automatic Control*, vol. 13, no. 4, pp. 16–25, Aug. 1993.

[4] H. T. Toivonen, "Discretization of analog filters via h_∞ model matching theory," *International Journal of Adaptive Control and Signal Processing*, vol. 6, no. 5, pp. 499–514, 1992.

[5] G. O. Beale and T. T. Hartley, "Stability considerations: Numerical methods and control theory equivalences," *System & Control Letters*, vol. 34, no. 2, pp. 180–187, May 1987.

[6] J. Ding, F. Marcassa, S.-C. Wu, and M. Tomizuka, "Multirate control for computation saving," *IEEE transactions on control systems technology*, vol. 14, no. 1, pp. 165–169, Jan. 2006.

[7] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Upper Saddle River, NJ, USA: Prentice Hall, Inc., 1989.

[8] R. Meyer and C. Burrus, "A unified analysis of multirate and periodically time-varying digital filters," *IEEE Transactions on Automatic Control*, vol. 22, no. 3, pp. 162–168, Mar. 1975.

[9] K.E.Atkinson, *An Introduction to Numerical Analysis*. John Wiley and Sons, 1989.

[10] G. F. Franklin, D. J. Powell, and A. E.-Naeini, *Feedback Control of Dynamic Systems*. Upper Saddle River, NJ, USA: Prentice Hall, 2001.