Nonlinear Analysis: Hybrid Systems 22 (2016) 116–136

Contents lists available at ScienceDirect

Nonlinear Analysis: Hybrid Systems

journal homepage: www.elsevier.com/locate/nahs

Performance regulation of event-driven dynamical systems using infinitesimal perturbation analysis ☆

Y. Wardi [a,*], C. Seatzu [b], X. Chen [a], S. Yalamanchili [a]

[a] *School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA*

[b] *Department of Electrical and Electronic Engineering, University of Cagliari, Italy*

ARTICLE INFO

Article history:
Received 17 March 2015
Accepted 23 March 2016

Keywords:
Infinitesimal perturbation analysis
Timed DEDS
Stochastic hybrid systems
Performance regulation

ABSTRACT

This paper presents a performance-regulation method for a class of stochastic timed event-driven systems aimed at output tracking of a given reference setpoint. The systems are either Discrete Event Dynamic Systems (DEDS) such as queueing networks or Petri nets, or Hybrid Systems (HS) with time-driven dynamics and event-driven dynamics, like fluid queues and hybrid Petri nets. The regulator, designed for simplicity and speed of computation, is comprised of a single integrator having a variable gain to ensure effective tracking under time-varying plants. The gain's computation is based on the Infinitesimal Perturbation Analysis (IPA) gradient of the plant function with respect to the control variable, and the resultant tracking can be quite robust with respect to modeling inaccuracies and gradient-estimation errors. The proposed technique is tested on examples taken from various application areas and modeled with different formalisms, including queueing models, Petri-net model of a production-inventory control system, and a stochastic DEDS model of a multicore chip control. Simulation results are presented in support of the proposed approach.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

This paper describes a regulation technique for a class of dynamical systems, designed for output tracking of a given setpoint reference. The regulator consists of an integral control with a variable gain, computed on-line so as to enhance the closed-loop system's stability margins and yield effective tracking. The gain-adjustment algorithm is based on the derivative of the plant's output with respect to its input control, and therefore the regulation technique is suitable for systems where such derivatives are readily computable in real time. This includes a class of stochastic timed Discrete Event Dynamic Systems (DEDS) and Hybrid Systems (HS) where the derivative is computable by the Infinitesimal Perturbation Analysis (IPA) sample-gradient technique. Our motivation is derived from the problem of regulating instructions' throughput in multicore computer processors, and following an initial study of that problem in Ref. [1] we extend the technique to a general class of DEDS and HS.

The need for regulating instruction throughput at the hardware level in modern computer processors stems from real-time applications where constant throughput facilitates effective real-time task and thread (subprogram) scheduling, as well

☆ Research supported in part by the NSF under Grant CNS-1239225.

* Corresponding author.
E-mail addresses: ywardi@ece.gatech.edu (Y. Wardi), seatzu@diee.unica.it (C. Seatzu), xchen318@gatech.edu (X. Chen), sudha@ece.gatech.edu (S. Yalamanchili).

http://dx.doi.org/10.1016/j.nahs.2016.03.007
1751-570X/© 2016 Elsevier Ltd. All rights reserved.

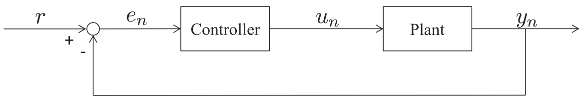

Fig. 1. Basic regulation system.

as from multimedia applications where a fixed frame rate must be maintained to avoid choppy video or audio. The design of effective regulators is challenging because of the lack of predictive analytical or prescriptive models, and unpredictable high-rate fluctuations of instructions-related switching activity factors at the cores. For this reason, we believe, most of the published control techniques are ad hoc (see the survey in Ref. [2]). A systematic control-theoretic approach has been pursued in Refs. [3,4,2] which applied a PID controller and analyzed the effects of proportional controls with fixed gains. Concerned with the unpredictability and rapid changes in the thread-related activity factors, Ref. [1] sought a controller with adaptive gain. Furthermore, it considered scenarios where measurements and computations in the control loop must be performed as quickly as possible, even at the expense of accuracy. To this end it considered controlling the instruction throughput by a core's clock rate, and applied an integral controller whose real-time gain-adaptation algorithm is designed for stabilizing the closed-loop system and yielding effective tracking convergence. The gain-adaptation algorithm is based on IPA as described in the sequel.

An abstract, discrete-time configuration of the closed-loop system is shown in Fig. 1, where n denotes the time-counter, r is the setpoint reference, u_n is the control input to the plant, y_n is the resulting output, and $e_n := r - y_n$ is the error signal. The system is single-input–single-output so that all the quantities u_n, y_n, e_n and r are scalar.

Let $J : R \rightarrow R$ represent a performance function of the plant with respect to its input u, and assume that the function $J(u)$ is differentiable. Given the *nth* input variable u_n, suppose that the plant's output y_n provides an estimation of $J(u_n)$. The controller that we consider has the form

$$u_n = u_{n-1} + A_n e_{n-1}, \tag{1}$$

and we recognize this as the discrete-time version of an integrator (summer) with a variable gain. As mentioned earlier, the gain sequence $\{A_n\}$ is designed to enhance the stability margins of the closed-loop system and reduce oscillations of the tracking algorithm while speeding up its convergence. As we shall see, one way to achieve that is to have A_n be defined as

$$A_n = \left(J'(u_{n-1})\right)^{-1}, \tag{2}$$

with "prime" denoting derivative with respect to u. However, it may not be possible to compute the derivative term $J'(u_{n-1})$, and approximations have to be used. Denoting the approximation error by ϕ_{n-1}, the computed gain A_n is defined as

$$A_n = \left(J'(u_{n-1}) + \phi_{n-1}\right)^{-1}. \tag{3}$$

In the systems considered in this paper the plant represents average measurements taken from a physical system or a cyber system over contiguous time-intervals called *control cycles*. For example, suppose that the physical system is a continuous-time dynamical system with input $\upsilon(t)$ and output $\zeta(t)$, $t \geq 0$; its state variable is immaterial for the purpose of this discussion. Divide the time axis into contiguous control cycles C_n, $n = 1, 2, \ldots$, suppose that the control input is fixed during C_n to a value $u_n := \upsilon(t)$ $\forall t \in C_n$, and define y_n by

$$y_n := \frac{1}{|C_n|} \int_{C_n} \zeta(t) dt,$$

where $|C_n|$ is the duration of C_n. Alternatively, y_n can represent average measurements taken from the output of a discrete-time or discrete-event system. Generally we impose no restriction on the way the control cycles are defined, they can be fixed a priori or determined by counting events in a DEDS; we only require that the input u_n remains unchanged during C_n and can be modified only when the next control cycle begins.

Observe that Eq. (3) suggests that the computation of A_n takes place during the control cycle C_{n-1}. In fact, we assume that the implementation of the control law takes place in the following temporal framework. Suppose that the quantities u_{n-1}, and y_{n-1}, e_{n-1}, and A_n have been computed or measured by the starting time of C_n. Then u_n is computed from Eq. (1) at the start of C_n and we assume that this computation is immediate. During C_n, the plant produces y_n from the applied input u_n while A_{n+1} is computed from Eq. (3), with the index $n + 1$ instead of n. Finally, e_n is computed at the end of C_n from the equation

$$e_n = r - y_n, \tag{4}$$

and we assume that this computation is immediate.

The plant's actions yielding y_n from u_n during C_n may represent a physical or cyber process or measurements thereof, and the computation of A_{n+1} is assumed to be carried out concurrently. Of a particular interest to us is the case where $J(u_n)$ is an expected-value performance function of a DEDS or HS, y_n is an approximation thereof computed from a sample path of the

Download English Version:

https://daneshyari.com/en/article/8055355

Download Persian Version:

https://daneshyari.com/article/8055355

Daneshyari.com