Available online at www.sciencedirect.com
SCIENCE DIRECT®

Microprocessors and Microsystems 27 (2003) 159–169

MICROPROCESSORS AND MICROSYSTEMS

www.elsevier.com/locate/micpro

Establishing timing requirements for control loops in real-time systems

Iain Bate*, John McDermid, Peter Nightingale

Department of Computer Science, University of York, York YO10 5DD, UK

Received 23 May 2002; revised 20 November 2002; accepted 17 December 2002

Abstract

Advances in scheduling theory have given designers of control systems greater flexibility over their choice of timing requirements. This could lead to systems becoming more responsive and more maintainable. However, experience has shown that engineers find it difficult to exploit these advantages due to the difficulty in determining the 'real' timing requirements of systems and therefore the techniques have delivered less benefit than expected. Part of the reason for this is that the models used by engineers when developing systems do not allow for emergent properties such as timing. The paper presents an approach and framework for addressing the problem of identifying an appropriate and valid set of timing requirements in order that the best use can be made of the advances in scheduling theory by the use of modelling techniques that allow for emergent properties such as timing behaviour.
© 2003 Elsevier Science B.V. All rights reserved.

Keywords: Real-time systems; Control systems; Scheduling; Model based development

1. Introduction

This paper addresses the perennial problem of how to identify an appropriate and valid set of timing requirements for a hard real-time system. Our main concern is with systems where failure(s) to meet timing requirements can lead to dangerous behaviour, and may ultimately lead to loss of life. Over the years, research on real-time systems has evolved techniques which provide greater flexibility in scheduling whilst still providing a means for guaranteeing that timing requirements are met [1,2]. The increased flexibility was expected to give many benefits, including more efficient use of resources and simpler maintenance of schedules when changes to the control software are made. Maintaining schedules is often a costly and error prone manual process, therefore these techniques have the potential to offer significant economic benefits as well as engineering benefits.

Experience has shown that engineers find it difficult to exploit this increased flexibility and hence the techniques have delivered less benefit than expected. Based on our own experience and that of others in industry [3,14], the main cause of the shortfall in benefit gained is an absence of information about the *true* timing requirements which is needed to make best use of the approaches. In many cases current systems are developed with simple timing requirements, such as a timing margin to be achieved. In other cases the timing requirements are largely historic, and are simply expressed in terms of iteration rates, which have been proven effective in previous designs. Despite the changing contexts between systems, this strategy is normally successful because the requirements are over conservative, e.g. update rates specified are much faster than needed. Even where more modern control law design environments (e.g. Matlab/Simulink [4]) are used the control models are often produced assuming a particular computational model. For example a 50 ms cycle/20 Hz bandwidth is chosen because there is a regular clock tick in the system with a period of 25 ms (i.e. 40 Hz) and therefore it is easier to release tasks at a harmonic of this frequency.

An often-used way to select timing requirements is the Nyquist criterion [6]. Nyquist's criterion places a lower bound on the sampling period—equal to twice the highest input frequency. This highest frequency can be open ended, for example a square wave has frequency components up to infinity so leaving an unanswered question of where would we draw the line. Similarly while a plant model may be limited to order N, the actual plant would be much more complicated. This could result in a sampling rate that is too

* Corresponding author.
E-mail addresses: iain.bate@cs.york.ac.uk (I. Bate), john.mcdermid@cs.york.ac.uk (J. McDermid), pwn101@cs.york.ac.uk (P. Nightingale).

0141-933/03/$ - see front matter © 2003 Elsevier Science B.V. All rights reserved.
doi:10.1016/S0141-9331(03)00015-2

high. Therefore practitioners normally assume the input signal can be represented by a frequency range within which *N*% of the intensity exists. Also, Nyquist's criterion is based on the assumption that the system under control is time invariant which in the case of a computer controlled system is not valid. Despite these assumptions, the Nyquist's criterion does not lead to instability as long as the time variations in when sampling occurs does not cause the bound on the sampling rate derived using Nyquist's criterion to be exceeded.

The work contained in this paper does not purpose an alternative to the Nyquist's criterion but instead a complementary method. Even with Nyquist's criterion, there is a need to choose an actual sampling rate since the criterion only provides a lower bound on the rate.

A major contributor to the methods using for selecting timing requirements is that the research and into the practical use of control theory and scheduling theory have largely been carried out in isolation [5]. Thus, for example, work on advanced control regimes such as H∞ [6], which might benefit from more sophisticated scheduling has not been the subject of joint work between control engineers and real-time systems researchers. Perhaps as a consequence there are few examples of H∞ being used in real-time embedded control systems.

The paper presents an approach and framework for addressing the problem of identifying an appropriate and valid set of timing requirements in order that the best use can be made of the advances in scheduling theory. The essence of the approach is to use component-based models that allow for emergent properties of systems, in this case timing, so that the models are more representative of how a real system would actually behave. Then, heuristic search techniques are used to explore the design space and to identify timing requirements which enable properties such as control stability to be achieved, thus deriving and validating the requirements against more fundamental properties of the control system. The advantage of using heuristic search techniques instead of traditional model-based design approaches, such as root locus [6], include it allows many properties and effects to be considered at the same time and their demands on the system to be traded-off against one another.

The work presented here is intended for use in a range of control problems, but is illustrated with the Proportional Integral Differential (PID) control approach [6].

The rest of the paper is structured as follows. Section 2 gives further background on the control techniques to be used in the context of this work. It also provides a technical motivation (as opposed to the 'economic' motivation outlined above) for seeking a systematic approach to deriving timing requirements. Section 3 presents the approach outlining the heuristics used, and the costs of evaluating the requirements. Section 4 contains two simple case studies which have been used to evaluate the approach, as well as presenting a discussion of how the resulting timing requirements may be used. Finally, Section 5 gives a summary and suggests possible future developments for the work.

2. Background and motivation

All scheduling approaches require a minimum set of information about timing requirements so that an appropriate scheduler can be produced. For most scheduling approaches the minimum set of information for any approach is the deadline and period of tasks [7,8]. This section explains why these requirements are important in the context of PID loops and how they can be generated by considering basic control properties.

2.1. PID loop

The principal purpose of a PID loop is to ensure the controller meets its objectives. Objectives of the controller could include responsiveness to input, stability, accuracy and limits on data are maintained. Fig. 1 depicts a typical PID loop, in a control system, being used to control the operation of a plant. The Figure shows the key aspects and components of the controller. There is only one input to, and one output from, the control system. The output of the control system is the plant input. The control system input is the difference between the input demand (typically from the operator of the plant) and the plant's actual output, and it is referred to as the error (in this paper error is defined as the difference between actual and desired plant state).

In the computer-based approach, the *Input Demand* and the *Actual Plant Output* are usually analogue signals. The computer performs the rest of the processing in the digital domain. Converters are used to sample the analogue signals, e.g. to produce the *Error* input, and then converted back to analogue values at the output. Converting back to an analogue signal is often referred to as digital to analogue conversion, or de-sampling.

The controller (PID loop) works by adding scaled versions of the error, differential of the error and integral of the error to achieve the desired control response. The scale factors are K_P for the proportional term, K_I for the integral term, and K_D for the differential term.

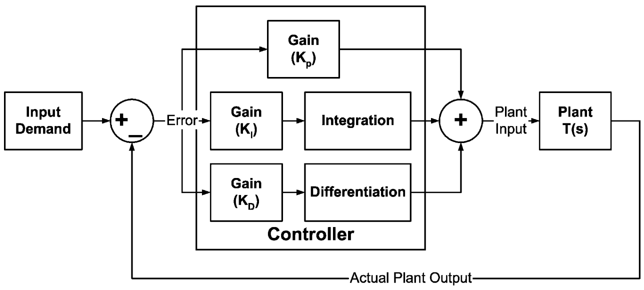

Fig. 1. Typical PID loop.

دریافت فوری ← متن کامل مقاله

ISIArticles
مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات