Mechatronics 36 (2016) 54–62

Contents lists available at ScienceDirect

Mechatronics

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

Embedded explicit model predictive vibration control

Gergely Takács*, Gabriel Batista, Martin Gulan, Boris Rohal'-Ilkiv

Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Nám. slobody 17; 812 31 Bratislava, Slovakia

ARTICLE INFO

Article history:
Received 17 June 2015
Revised 18 February 2016
Accepted 21 April 2016
Available online 4 May 2016

2010 MSC:
74H45
70Q05
49J15
93C83

Keywords:
Vibration control
Model predictive control
Explicit solution
Parametric programming
Embedded systems
Microcontroller unit

ABSTRACT

This article presents an embedded active vibration suppression system featuring real-time explicit model predictive control (EMPC) that is implemented on a microcontroller unit (MCU). The EMPC controller minimizes the tip deflection of an aluminum cantilever beam driven by piezoceramic actuators, gaining its feedback from direct position measurements. The output and input performance of the EMPC method is compared to an analogously tuned positive position feedback (PPF) controller. An extensive analysis is provided on the cycle timing and memory needs of the explicit predictive vibration control scheme. The results demonstrate that the EMPC controller may achieve the same vibration suppression results compared to PPF with less input effort, while inherently respecting process constraints. Furthermore, we show that EMPC task execution timing is comparable in the random access memory (RAM) and read only memory (ROM) alternatives, suggesting that numerous current microcontrollers are suitable for EMPC-based active vibration control, in case the prediction model is kept simple.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

A couple of decades have passed after the first academic studies on active vibration control (AVC) appeared in the literature [17]. By now, the technology has been embraced by a spectrum of real-life applications, ranging from active rotor blades in helicopters [5], through advanced automotive suspensions [18] and even medical devices for those suffering from Parkinson's disease [35]. This exciting technology transfer can be partly attributed to the ever developing range of actuators available in the market and, more importantly, to the constantly falling price of microcontroller units (MCU) as well [8]. Machines are getting faster to satisfy consumer needs and becoming thinner to maximize profits by cutting back material use. This trend often leads to structures that are vibration prone and may benefit from the use of active vibration control. This time, instead of actuators or mechanisms, our attention will be focused on the algorithms that drive active vibration suppression systems and their implementation on inexpensive embedded computing platforms.

There are several well-known control algorithms routinely used in active vibration control. The de-facto standard amongst AVC algorithms is the positive position feedback (PPF) controller that is related to the proportional-integral-derivative (PID) controller very closely [10]. Common methods known from control theory are often featured, such as pole-placement (eigenvalue assignment) [27], linear quadratic (LQ) control [18] and others. As for the more exotic approaches, soft computing methods like genetic algorithms or neural networks may hold a great promise for active vibration control in the future, however, their use is at the moment mostly limited to simulation studies or off-line parameter tuning [6].

The technique known as model predictive control (MPC) or receding-horizon control (RHC) has been actively sought out by industry for the past thirty years and its formulation is regarded to be one of the fundamental developments in control engineering [12,21,30]. Its advantages compared to traditional control engineering approaches are widely acclaimed and are twofold: it may provide an increased control performance in most applications and it inherently handles process constraints [25]. In fact, MPC is often referred to as the only control approach that may explicitly handle the constraints that are present in every real-life system [23]. These constraints represent the natural limitations of actuators at the input side, but may also express economic, safety or

* Corresponding author. Tel.: +421907165720; fax: +421252495315.
E-mail address: gergely.takacs@stuba.sk (G. Takács).

http://dx.doi.org/10.1016/j.mechatronics.2016.04.008
0957-4158/© 2016 Elsevier Ltd. All rights reserved.

other bounding factors [33]. Thus, the typical contemporary model predictive control algorithm uses a linear quadratic cost function to quantify controller performance over and beyond the horizon using a state-space model, which is then minimized subject to the aforementioned constraints in real time by quadratic programming (QP). Consequently, the improvement in performance and the constraint handling feature are not without a price: model predictive control is computationally intensive and its application has been limited to systems with slow dynamics, such as the ones encountered in the petrochemical industry.

In light of the computational complexity of MPC, it is no wonder that active vibration control with its typically fast sampling times is not a common field of application. Most academic studies focus on legacy formulations, like generalized matrix control (GMC) or dynamic matrix control (DMC) that do not include or handle constraints explicitly [34], while other studies feature bulky and expensive computing platforms to implement model predictive control for AVC [26]. Thus, in order to use model predictive vibration control in real-life applications, we shall concentrate on up-to-date formulations used in combination with embedded computing solutions that are suitable for mass production, miniaturization and product integration. One of the few examples of using MPC in AVC on a microprocessor is a digital signal processing (DSP) chip implementation that is described by Wills et al. in [39,40]. The authors achieved high sampling speeds with the original optimal MPC formulation, albeit with a work intensive manual transcription of the algorithm into a customized code-efficient machine code formulation.

Besides the question of using the right computing hardware, researchers have recently invested a great deal of effort into modifying the formulation of model predictive control, in order to make it computationally more efficient. There are efficient MPC approaches that sacrifice the controller performance to limit its computational needs, creating so-called sub-optimal predictive controller formulations [19,38]. The other main class of efficient MPC formulations preserves the optimality of the problem, while still removing some of the burden of solving the task in real time [11]. The best known representative formulation of the latter class rests on the idea of transferring the computational complexity from on-line control into off-line, and is called multi-parametric MPC (MPMPC) or explicit MPC (EMPC) [2,3].

In EMPC, the optimization problem is essentially solved using parametric programming beforehand, formulating the MPC control problem as a set of regions in state space to which linear control laws are assigned. Instead of solving a quadratic programming problem in each sample, the microprocessor needs to find the region to which the currently observed state belongs to, then evaluate a linear control law that is not much more complicated than LQ control. This, of course, means that the computational burden is now transferred from processor speed into memory requirements. Although modern microprocessors—also known as microcontrollers or system on a chip (SoC) systems—have increased their power for a unit price, a traditional quadratic optimization-based MPC (QPMPC) is still thought to be prohibitive for AVC because of direct execution speed requirements, while EMPC because of its memory needs. Niederberger used a clever idea to turn an EMPC controller computed for a vibration suppression problem into a completely electronic system, instead of using the real-time algorithm of the EMPC formulation [28]. Although this is an interesting concept, it cannot be considered as a true real-time application of explicit MPC in vibration control, as it only emulates the behavior of the EMPC control law approximating its structure using electronic components. Previously an EMPC active vibration suppression scheme was demonstrated experimentally and in real time, however, only using a personal computer-based prototyping system [36,37]. Up to now—according to the knowledge of the authors—explicit model predictive control has not been utilized for active vibration suppression using a cost-efficient miniaturized embedded microcontroller that would be suitable for mass production and close system integration.

This article presents a real-time application of explicit model predictive control for active vibration suppression, using a 32-bit embedded microcontroller unit. The purpose of the AVC system is to minimize the tip deflections of an aluminum cantilever beam, by supplying the input decisions of the EMPC algorithm to the piezoceramic actuators in the form of a driving voltage. The EMPC algorithm is running stand-alone and real-time on the microcontroller, gaining its feedback from position measurements and supplying input to the AVC system via an operational amplifier. The proposed control scheme is evaluated by release tests and its performance and timing properties are compared to the open-loop case without control as well as PPF control. An extensive computation speed and memory requirement study is performed on the microcontroller, to evaluate the possibilities of increasing sampling speeds for stiffer structures or to use even lower-priced and smaller microcontrollers for the same class of flexible mechanical systems.

It is also important to note what is not in the ambition and scope of this article. In Section 2.1 we begin with the assumption that the dynamics of the beam may be represented by a single degree of freedom (SDOF) model. This is an essential premise to this work, as it is unlikely that complex prediction models are feasible to implement on relatively simple embedded hardware, like the one used here [23,38]. Although by using embedded computing devices with large memory footprints one may be able to utilize EMPC for the vibration control of up to 2–3 resonant modes or multiple-input multiple-output systems, it is unreasonable to expect a complex electro-mechanical model derived from a finite element model (FEM) to be viable on current hardware.

Even though we present a performance comparison between EMPC and PPF to provide a baseline for the reader, it is not our aim to prove the superiority of constrained model predictive algorithms against other methods employing saturation only. The performance advantage of constrained MPC compared to saturated control has been extensively studied by others in the past using fundamental mathematical, numerical and experimental comparisons [23,24,31,36].

2. Controller design

2.1. Modeling

Let us assume that the dynamic behavior of the beam is dominated by its first resonant frequency [7]. Moreover, let us represent the beam by a single degree of freedom point mass-spring-damper with an outside driving force. This elementary assumption may under-represent the beam dynamics from a mechanical viewpoint, however, one cannot utilize elaborate mechanical models in on-line optimization procedures such as MPC, since the controller will be infeasible to implement in real time [36,37]. Keeping this in perspective, the simple SDOF system assumed in this work models the dynamics of thin flexible cantilevers adequately. The SDOF mass-spring-damper with a linear driving force may be described by $m\ddot{q}(t) + b\dot{q}(t) + kq(t) = F(t)$, where m (kg) is the mass, b (Nsm^{-1}) the viscous damping and k (Nm^{-1}) is the stiffness of the equivalent model. The position output is denoted by $q(t)$ (m), while the force input can be expressed by the force exerted by the piezoceramic actuators $F(t) = c_m u(t)$ (N), where $u(t)$ (V) is the driving voltage, and c_m (NV^{-1}) is the mass-specific force conversion constant.

Download English Version:

https://daneshyari.com/en/article/731048

Download Persian Version:

https://daneshyari.com/article/731048

Daneshyari.com