Journal of Process Control 61 (2018) 23–35

Contents lists available at ScienceDirect

Journal of Process Control

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

Research paper

A new robust controller for non-linear periodic single-input/single-output systems using genetic algorithms

André Bégin-Drolet*, Simon Collin, Jérôme Gosselin, Jean Ruel

Département de Génie Mécanique, Université Laval, Québec, QC G1 V 0A6, Canada

ARTICLE INFO

Article history:
Received 25 January 2017
Received in revised form 2 November 2017
Accepted 10 November 2017

Keywords:
Genetic algorithm
Optimization
Robust control
Cardiac bioreactor
Indirect encoding
Multi-parent crossover

ABSTRACT

In this paper, a robust control approach designed for periodic systems and based on the well-known genetic algorithms is presented. This control scheme introduces the innovative concept of indirect encoding through finite Fourier series, which greatly contributes to the efficiency of the algorithm. Moreover, emerging concepts such as multi-parent crossover and local mutation are employed. These features bring undeniable exploitation and exploration capabilities to the algorithm, which are essential for controlling ever-changing environments such as our case of application: a cardiac bioreactor. For about a decade, our research team has been working on the development of this system for the growth of tissue-engineered heart valves. The work presented in this article address the major challenge of optimal and robust control of the flowrate in the bioreactor. Several *in silico* and physical simulations based on a simplified model of this complex system allowed for a quick development process and are presented in this paper. Finally, experimental validation of the simulations were conducted on the cardiac bioreactor. These confirmed that the proposed approach can effectively be used for the optimal and adaptive control of this non-linear periodic single-input/single-output system.

1. Introduction

Repetitive operations with performance requirements are found in many scientific and engineering applications such as robotic tracking [1], chemical engineering processes [2,3], electrical systems [4] and biomedical devices [5]. This paper focuses on solving the important challenge of controlling the flow in a cardiac bioreactor. The method proposed can potentially be adapted to the control of any single-input/single-output (SISO) non-linear periodic system. Many control methods such as *iterative learning control (ILC)*, *repetitive control (RC)* and *run-to-run control (R2R)* [6] were proposed and used in the past to address control problems where a system is asked to perform the same task time after time. The common denominator to all these control methods, highlighted by Wang et al. [7], is to "*use previous information to design a new control signal*", much as someone would do using past experiences to guide future decisions using his own intelligence. Up to recently, control of repetitive systems was considered as an emerging research field, which has mostly used an internal explicit model within the controller to close the loop [4]. Generally, bioreactors need to exhibit repetitive behavior and must rely on adequate controllers. Nowadays, these devices are becoming more and more popular, particularly due to the emergence of tissue engineering applications. Even though simple periodic conditions, such as pulsatile flows, are to be reproduced, the control of such devices can become challenging. On one hand, this can be explained by the complexity of the bioreactors leading to non-linear behavior. On the other hand, control challenges can be explained by the fact that bioreactors are designed to grow cells and/or tissues, which leads to an ever changing environment. Therefore, an adequate controller should rely on two characteristics: the ability to converge rapidly towards a solution and to adapt efficiently to any variation in the system. In recent work, Beelen et al. [8] used repetitive controllers to oversee the flow regulation in a cardiac bioreactor. They were able to converge efficiently to the desired outputs; however, their control method lacked adaptiveness and did not prove to be adequate in a changing environment. Similarly, the artificial neural network approach used by Laterreur [9] for the control of another cardiac bioreactor was shown to converge to an adequate solution, but also lacked the ability to adapt in a changing environment. Some research papers claim to have achieved adaptiveness in repetitive controllers. For example, Cao and Ledwich [10] designed an adaptive repetitive controller which was able to track variable periodic signals with fixed sampling rate. However, this controller

---

* Corresponding author.
E-mail address: andre.begin-drolet@gmc.ulaval.ca (A. Bégin-Drolet).

https://doi.org/10.1016/j.jprocont.2017.11.007
0959-1524/© 2017 Elsevier Ltd. All rights reserved.

was designed for discrete systems only and is not of interest for more complex situations. In addition, Dracopoulos and Piccoli [11] claimed of having achieved adaptive control over a numerical simulation of a chemical bioreactor. However, their controller was not tested online on a chemical bioreactor, which limits the reach of this claim. Thus, to our knowledge, reaching adaptiveness in repetitive controllers used on physical non-linear complex processes is still a major challenge to the scientific community. The novel control approach presented in this paper, which uses evolutionary computing techniques, addresses this specific issue.

In any repetitive SISO process of known finite time, each cycle can be seen as a batch where a finite length input signal leads to a finite length output signal. In our control strategy, an arbitrary input signal is seen as a potential solution and is conceptualized as a creature. Several potential solutions, or creatures, form a population that is being evolved according to an adaptive plan. Much as nature would close the loop with natural selection in an ecological system, the adaptive plan is engineered to implicitly close the loop, wiping out weaker creatures and improving stronger ones using different operators. This can be achieved using a computer science technique known as genetic algorithms (GA), a subset of evolutionary computing. We hypothesize that evolutionary computing, and therefore GAs, can be used as a universal adaptive control method when dealing with SISO repetitive and/or cyclic processes. We also found that indirect encoding, *i.e.* the way the different creatures are represented, can be used advantageously to reduce the search space and improve convergence. New genetic operators designed specifically for SISO periodic control systems are proposed and tested to show the performance on both a linear and a non-linear system. Our approach, using GAs and emerging genetic operators for SISO cyclic processes, is supported by the application of the method to the control of the flowrate in a cardiac bioreactor.

2. Genetic algorithms: a short overview

The concepts defining GAs are generic so that they can be applied to totally different fields and/or applications other than bioreactors. Over the course of this paper, the target or desired waveform is referred as the reference trajectory $y_r(t)$. The input to the system, also referred as the control signal, is represented as $u(t)$ and the output of the system, also referred as the system response, is represented as $y(t)$. The results obtained show the versatility and the simplicity of the proposed approach when dealing with SISO repetitive control tasks. These results, inspired by nature's survival of the fittest, bridge an important gap between evolutionary computing, a branch of artificial intelligence, and control theory.

GAs were introduced by John Holland in 1975 [12] as search strategies adapted from nature. Simply put, these biologically inspired mathematical algorithms try to emulate a simplified representation of the natural selection process: survival of the fittest. Solving optimisation problems with GAs is done using a four-step approach [13–16]. The basic idea of GAs is to bring creatures together in a virtual world, make them fight for their survival and interact together to create offsprings that are fitter generation after generation.

GAs are composed of four modules that needs to be defined in order to obtain fully functional algorithms. The population module is the first step of a GA and this module is used to encode potential solutions as creatures. The genotype of each solution (*i.e.* each creature) is mathematically encoded in a single chromosome [14]. As it can be observed in nature, the phenotype of each creature emerges from the interaction of its genotype with the environment. In the context of the control problem outlined above, the phenotype is the response signal of the system to a control input. The evaluation module is the second step and is used to estimate the fitness of each creature within the population. The fitness of a creature is directly related to the evaluation of its phenotype and it is obtained by comparing the actual response to the desired response (*i.e.* the reference trajectory). The next step is the selection module, which is used to select the creatures that should survive. Fitter creatures see their chances of survival increased similarly to what would occur in an ecological system. The reproduction module is the last step and consists in using genetic operators that act upon selected creatures to create new ones. After reproduction, a predetermined number of mutated (new) creatures are created, which completes a cycle that is known as a generation. This new population replaces the old one and the cycle starts over until a termination criterion is reached (*e.g.* fitness threshold, fitness gradient threshold, maximum number of epoch, maximum number). The maximum number of creatures evaluated has shown to be an effective termination criterion in order to compare different adaptive plans with one another. An adaptive plan is a strategy engineered to navigate in the fitness landscape. Particular adaptive plans are obtained when specific values are attributed to the various parameters of each module. The key to a properly designed adaptive plan is to adequately balance the "*exploration of new possibilities with the exploitation of old certainties*", as stressed by March [17]. Therefore, an appropriate adaptive plan is one that is designed to explore until satisfactory solutions are found, without getting trapped into local maxima, and exploit until changes in the environment occur, hence helping the population adapt to changes. The adaptability and versatility exhibited by GAs is a major advantage over traditional control approaches.

2.1. Population module: encoding the solution

In the population module, the size of the population (P), the length (W) of the chromosome and the type of encoding are decided. For this specific control problem, we were confronted to several dilemmas during this important setting step. The first decision was to use real values instead of binary values for chromosome (creature) encoding. This decision was based on our own experience with both encoding methods as well as GA practitioners' insights. Several researchers [13–16] recommend to use the encoding that best fits the needs of the problem since "*there are no rigorous guidelines for predicting which encoding will work best*" [16]. We therefore postulated that real encoding would not affect the performances of our GA. We also postulated that a periodic response would be obtained using a time-varying function whose waveform $u(t)$ exactly repeats itself at regular intervals. One peculiar feature of those waveforms is that they can be expanded into an infinite Fourier series as presented in Eq. (1).

$$u(t) = Y_0 + \sum_{n=1}^{\infty}(Y_n \cos(2\pi n f_1 t - \theta_n)) \approx Y_0 + \sum_{n=1}^{K}(Y_n \cos(2\pi n f_1 t - \theta_n)) \tag{1}$$

where f_1 is the fundamental frequency, Y_0 is the static component of the waveform, Y_n is the magnitude of the n^{th} harmonic, θ_n is the phase of the n^{th} harmonic and K is the last harmonic considered.

To make it practical, only a finite number of coefficients of the Fourier series were used to encode the different solutions $u(t)$. This encoding strategy (finite Fourier series coefficients instead of the time domain waveform) reduced the search space greatly, a major advantage of using indirect encoding in the frequency domain. In such encoding, the size of the chromosome is independent of the size of the waveform it generates. Since the search space was reduced, we also postulated that this type of encoding would improve convergence speed. In our approach, the Fourier coefficients (*i.e.* Y_0, Y_n and θ_n) were used as the genes defining the genotype of our creatures. To our knowledge, in the field of artificial