Applied Soft Computing 48 (2016) 638–649

Contents lists available at ScienceDirect

Applied Soft Computing

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

Lessons learned from using some bio-inspired optimizers for real-time controller design for a low-cost electrohydraulic system

Pranibesh Mandal[a], Rana Saha[a,*], Saikat Mookherjee[a], Amitava Chatterjee[b], Dipankar Sanyal[a]

[a] Mechanical Engineering Department, Jadavpur University, 700 032, India
[b] Electrical Engineering Department, Jadavpur University, 700 032, India

ARTICLE INFO

Article history:
Received 16 August 2015
Received in revised form 29 June 2016
Accepted 31 July 2016
Available online 3 August 2016

Keywords:
Bio-inspired optimization
Fuzzy control
Real-time study
Electrohydraulics

ABSTRACT

Well-designed searching procedures following natural processes have been developed for finding optimized solutions of complex systems. Here, a comparison of performances of some optimizers, namely differential evolution, genetic algorithm, bacterial foraging and artificial bee colony technique, have been carried out for designing a fuzzy-feedforward real-time controller of an electrohydraulic motion actuation system. The first two optimizers execute dominatingly exploratory search, while the latter two execute a combination of exploratory search with intensified exploitive search in prospective regions, thus providing faster convergence. The optimized controller has been designed by minimizing a response error integral for some standard displacement demands of the highly nonlinear system. The strong nonlinearities in the system arise from the friction of low-cost industry-grade cylinder and large deadband of rugged proportional valve. The convergences to the minimum of zero for a number of nonlinear functions have also been demonstrated for all the optimization processes. These optimizers with faster convergence rate have been shown to be robust against arbitrary demands like variable frequency sinusoidal demands and sinusoidal demands with superimposed log concave-convex variations.

1. Introduction

Many search engines developed over the years involve core algorithms emulating natural processes to optimize suitable objective functions. Notable examples of these processes are either genetic evolution or search for nutrients by a collection of agents like bacteria, insects, birds or animals. While genetic evolution progresses for improving the fitness of a population over generations, any search for food aims at the most efficient deployment of the agents for striking a balance between global exploration and local exploitation. By developing simulation algorithms through mimicking these biological processes, design optimization of many engineering systems has been achieved. The evolutionary searching inherent to any of these processes is better equipped to capture the global optimum. This is in contrast to any classical gradient-based algorithm that carries out only a directed search to a solution that is locally the best. Genetic algorithm (GA), differential evolution (DE), bacterial foraging optimization (BFO), artificial bee colony (ABC) technique and particle swarm optimization (PSO) are some of such bio-inspired optimizers. Their performances are regularly getting evaluated for newer and more challenging systems that are continuously coming up with progress in both technology and theory.

Since the early developments of GA and DE, these have been employed [1–5] to solve a wide range of problems including designing of controllers for complex nonlinear systems. These two optimizers use the genetic evolution principle to find out the fittest possible solution through a probabilistically guided search process. ABC [6,7] and BFO [8,9] are two algorithms developed later by mimicking excellent foraging behaviors, shown respectively by honey bees in a bee colony and *E. Coli* bacteria in human guts [10–14]. Compared to dominatingly exploratory search carried out in GA and DE, the other two methods exploit regions with higher prospect more aggressively. An analysis of convergence to the known minimum of any standard nonlinear function helps to understand the potential of a method. However, using it to locate the unknown minimum of a real-world nonlinear system has practical benefits.

Despite strong nonlinearities, electrohydraulic systems find abundant uses in agriculture [15], forest [16], mining [17], automotive [18–20], aerospace [21], manufacturing [22] and construction [23] industries involving large power and, in many cases, rugged environmental conditions. Orifice-like variable flow openings,

* Corresponding author.
E-mail address: rsaha@mech.jdvu.ac.in (R. Saha).

http://dx.doi.org/10.1016/j.asoc.2016.07.056
1568-4946/

leakage flow and oil compressibility [16,18,24] render nonlinear features to hydraulic systems. In low-cost electrohydraulic systems, these get compounded by the deadband [16,24–26] in proportional valves along with the complex and discontinuous variation of friction [24,27,28] opposing the motion in the industry-grade actuator or cylinder. Simple PID controllers were shown to perform inadequately beyond tracking demand [28] of 1 Hz in a servovalve-based fish-cutting machine. Fuzzy controllers resulted in significant displacement error for cycle times well over 1 s in systems with proportional valves [15] and force error in systems with servovalves [22].

The mainstream applications of high-performance electrohydraulic actuation involve sophisticated technology solutions like servovalves and servoactuators with simple controller architecture. Designing advanced controllers in a laboratory set up with low-cost components has been pursued in recent times with the objective of catching up with the performance of the sophisticated systems [29–31]. This trend of research has been extended here for comparing the roles of the optimizers in quick realization of such a design. The explorations in the laboratory set up [29–31] clearly established the effectiveness of a feedforward controller in compensating the large deadband and friction of the low-cost system. Existing variants of GA [29,30] and BFO [31] were employed to arrive at the appropriate parameter settings, focusing more on simplifying the controller design than making a comparison of the performance of the optimizers and ease of their use. The feedforward model employed by Sarkar et al. [29] was much more complex that was simplified later [30] by employing a fuzzy controller with a feedforward compensator for the external load only. Mandal et al. [31] achieved good tracking response up to 3 Hz by the BFO in comparison to that of 1.6 Hz achieved by the GA [30].

Besides pursuing the objective of arriving at even better performance and a more robust controller design, a systematic analysis of the convergence of four bio-inspired optimizers, namely DE, GA, BFO and ABC, has been carried out here. Here, GA and DE have been chosen as representative evolutionary optimizers, while ABC and BFO have additional capability of exploiting the neighborhood of a prospective region. While the real-time system and the controller are briefly described in Section 2, the implementation of the optimizers for the controller design is described in Section 3. Section 4 begins with comparison of the convergence characteristics of these optimizers for some standard functions used in an earlier study [32]. This study has then been extended to the real-life controller design as well. In order to demonstrate the robustness of the controller, arbitrary combinations of standard demands along with superimposed log concave-convex variations over sinusoidal demand have been studied. Section 5 concludes the findings and contributions of this study.

2. Electrohydraulic actuation system with controller

A Rexroth 4WRE 10E1-50-2X/G24K4/V [33] proportional valve (PV) forms the core of a hydraulic circuit shown in Fig. 1. Oil is raised at high pressure by a Rexroth pump coupled to an electric motor (M). The valve guides the oil flow through a high-stiction double-rod cylinder so that a piston in it moves against an external compression spring. An earlier study [24] found out the static friction to vary between +910N and −600N. The circuit also has a single-stage relief valve (RV), a check valve (CV) and oil filters. There is a Gefran make linear variable differential transformer (LVDT) to provide the position feedback of the piston.The motion of the piston is controlled by the PV and a real-time system (RTS) from National Instruments interfaced through a Host PC. The RTS consists of a 16-bit NI-cRIO 9102 real-time processor together with digital-to-analog and analog-to-digital signal converters, a 0–10 V NI-cRIO 9215 input module (IM), and a ±10 V NI-cRIO 9263 output module (OM). In order to direct positive command voltage to one solenoid and negative voltage to the other that results piston extension and retraction respectively, a valve-control card (VCC) is used that is a part of the PV.

Fig. 2 describes the controller architecture used in this study. For a given input position demand y_d of the piston and the corresponding LVDT measurement y_{LVDT}, the controller computes the command voltage e after every sampling interval. The total voltage input to the PV is obtained as a combination of fuzzy voltage e_{fu}, feedforward voltage e_{ff} and the bias voltage e_b given by

$$e = e_{fu} + e_{ff} + e_b. \tag{1}$$

The fuzzy voltage maintains the piston inertia overcoming the difference between the dynamic and static frictions. The bias voltage overcomes the hard nonlinearities of the static friction discontinuity in the cylinder and the valve deadband. An additional feedforward voltage e_{ff} is necessary to provide the spring compression.

The membership function of the fuzzy controller is depicted in Fig. 3. At any time instant, it receives a crisp input

$$c_e = y_e + k_e v_e, \tag{2a}$$

which is a combination of the position error

$$y_e = y_d - y_{LVDT}, \tag{2b}$$

and its Labview derivative providing the velocity error

$$v_e = \dot{y}_e, \tag{2c}$$

with coefficient k_e having the unit of time. The controller also involves two parameter vectors

$$c^T = (c_k)^T = (c_{-q}c_{-q+1}\ldots c_0\ldots c_{q-1}c_q), \tag{3a}$$

$$\text{and } e^T = (e_k)^T = (e_{-q}e_{-q+1}\ldots e_0\ldots e_{q-1}e_q), \tag{3b}$$

comprised of $(2q+1)$ discrete values of the input error and the voltages. Of course,$q = -2, -1, 0, 1, 2$ are the linguistic representations of Negative Large (NL), Negative (N), Zero (Z), Positive (P) and Positive Large (PL) respectively. A Gaussian membership function shown in Fig. 3 is evaluated as

$$\mu_k = \mathbf{exp}(-\beta_k^2(c_e - c_k)^2), \tag{4a}$$

in terms of a vector controlling the span of the distribution defined as

$$\beta^T = (\beta_k)^T = (\beta_{-q}\ldots\beta_0\ldots\beta_q). \tag{4b}$$

Corresponding to the input pair, the controller composes the crisp output voltage as

$$e_{fu} = \sum\nolimits_k \mu_k e_k \tag{5}$$

The feedforward voltage in (1) provides a signal to negotiate a position dependent external load variation. In the system used, it arises for holding the piston position against the compression spring. For the motion against a spring in the set up of Fig. 1, this has been formulated as

$$e_{ff} = k_{ff} y_d, \tag{6}$$

where k_{ff} is a constant in V/m.

The bias controller in (1) invokes either the positive or the negative voltage by the relation

$$e_b = \max[sgn(e_{fu} + e_{ff}), 0]e_{bp} + \min[sgn(e_{fu} + e_{ff}), 0]e_{bn}, \tag{7}$$

so as to overcome the hard nonlinearities. All the controller parameters in (3a), (3b), (4b), (6) and (7) have been updated from an initial guess until a convergence has been achieved.