Electrical Power and Energy Systems 62 (2014) 364–373

Contents lists available at ScienceDirect

Electrical Power and Energy Systems

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

# Smart bacterial foraging algorithm based controller for speed control of switched reluctance motor drives

Ehsan Daryabeigi [a,*], Behzad Mirzaeian Dehkordi [b]

[a] Young Researchers and Elite Club, Najafabad Branch, Islamic Azad University, Najafabad, Isfahan, Iran
[b] Department of Electrical Engineering. Faculty of Engineering, University of Isfahan, Isfahan, Iran

ARTICLE INFO

*Article history:*
Received 2 February 2014
Received in revised form 1 April 2014
Accepted 21 April 2014
Available online 29 May 2014

*Keywords:*
Smart bacteria foraging algorithm
Speed control
Switched reluctance motor
Torque ripple

ABSTRACT

In this paper, a innovative methodology for Switched Reluctance Motor (SRM) drive control using Smart Bacterial Foraging Algorithm (SBFA) is presented. This method mimics the chemotactic behavior of the *E. Coli* bacteria for optimization. The proposed algorithm uses individual and social intelligences, so that it can search responses among local optimums of the problem adaptively. This method is used to tune the coefficients of a conventional Proportion–Integration (PI) speed controller for SRM drives with consideration of torque ripple reduction. This matter is done by applying the proposed algorithm to a multi-objective function including both speed error and torque ripple. This drive is implemented using a DSP-based (TMS320F2812) for an 8/6, 4-kW SRM. The simulation and experimental results confirm the improved performance of adjusted PI controller using SBFA in comparison with adjusted PI controller using standard BFA. Excellent dynamic performance, reduced torque ripple and current oscillation can be achieved when the coefficients of PI controller are optimized by using SBFA.


## Introduction

In recent years, there is a growing interest for using the SRM, because of low cost, simple rugged structure, and relatively high torque–volume ratio and low maintenance cost. They are the best candidate for direct-drive applications because of their ability to generate high torques at very low speed in comparison with other conventional motors [1,2]. Nevertheless, it suffers some drawbacks such as; high torque ripple and significant acoustic noise as well as speed oscillations. In order to resolve these problems, albeit various advanced control systems are proposed [3,4], PID controllers are most common controller for industrial applications [5] in spite of some disadvantages such as; tuning coefficients. Hence many researches have been presented about optimization of the controllers.

Optimization methods are used in many complex engineering problems [6]. Among various optimization methods, bio-inspired methods are developed more than classic techniques. Genetic algorithm (GA) is one of the first bio-inspired methods [7]. Based on its demonstrated ability to reach near-optimum solutions in complex problems, the GA technique is used for many applications [8]. In [9], GA is applied for optimal design of PI speed controller coefficient with considering minimum torque ripple as a cost function in simulation tests. Despite its benefits, GA requires long processing time to find optimum solution. Furthermore, sometimes a local minimum instead of global minimum may be introduced by GA. Recently, some new techniques inspired of different natural processes are proposed such as; particle swarm optimization (PSO) algorithm [10], invasive weed optimization (IWO) algorithm [11], artificial bee colony (ABC) algorithm [12], and artificial immune (AI) algorithm [13]. Moreover, some of the recently presented advanced algorithms are; Directed Searching Optimization algorithm (DSO), which is proposed to solve constrained optimization problems [14]. In [15], a control mechanism based PI controller for speed control of Switched Reluctance Motor (SRM) with torque ripple reduction using non-dominated sorting genetic algorithm has been presented. However, the proposed method has been only validated by simulation.

Another stochastic optimization method is the Bacterial Foraging Algorithm (BFA). This method was proposed in 2002, by Prof. K.M. Passino. It is based on the foraging strategy of the *E. Coli* bacterium [16] and has shown excellent results in different problems such as; adaptive control [17], harmonic estimation [18] for solving the economic dispatch (ED) problem considering valve-point effects and power losses [19], portfolio optimization problem [20], optimal power system stabilizers (PSS) design [21,22], optimal dynamical power flow [23], economic dispatching [24,25], active power filter design for load compensation [26], tun-

* Corresponding author. Tel.: +98 9132265430.
*E-mail addresses:* e.daryabeigi@gmail.com (E. Daryabeigi), mirzaeian@eng.ui.ac.ir (B. Mirzaeian Dehkordi).

http://dx.doi.org/10.1016/j.ijepes.2014.04.055
0142-0615/

ing Thyristor Controlled Series Capacitor (BFTCSC) [27], The unit commitment (UC) problem [28], and phase balancing in disturbance [29]. Although BFA is used to tune the PID controller coefficients in simulations, [8,30], it has not been experimentally evaluated yet.

The most important advantage of BFA is that area of the global answer is guaranteed. However, some parts of this method should be improved as follows; (a) in this algorithm only social intelligent and random motion are used for foraging prey and it does not include any individual intelligent in the foraging, (b) there is a probability to eliminate and disperse of each bacterium, even for the best solution of algorithm, and (c) swimming step is constant over all the stages which causes longer optimization time or less accurate. So, the convergence can be hardly achieved in some cases and moreover, it is possible the best solution is placed on the worse place in each swimming step. In this way, even the Lyapunov's stability theorem has been applied to derive the conditions of asymptotic stability of a bacterium near an isolated optimum of the fitness landscape, which indicated where the step-size is usually kept constant, the step-size violates the conditions of asymptotic stability and the bacterium starts oscillating around the optimum instead of converging to it [31].

To deal with these issues, many articles have been published and various approaches have been proposed. The swarming mechanism of the PSO algorithm, a new principle of swarming has been proposed in the context of BFA in [32], under which the bacterial swarming algorithm (BSA) has been reached. Synergetic bacterial swarming optimization (SBSO) has been proposed based on particle swarm optimization algorithm (PSO) and utilization of multi-populations in [33]. Moreover, many other papers propose hybrid versions of BFA by using some stochastic optimization algorithms (including PSO, GAs and DE), [34,29,35] which have some limitations.

In some researches fuzzy logic has been used to modify the conventional BFA. Nevertheless, it causes more complexity and expert's experience will be required [36,37].

Although the mentioned approaches improve the conventional BFA, they failed to capture the imagination.

In this paper, a new optimization algorithm named smart bacteria foraging algorithm (SBFA) is proposed to improve the conventional BFA using adaptive social and individual foraging behaviors. So, in contrast to the BFA, in proposed algorithm (SBFA); (1) a new term indicates the movement direction of bacterium smartly (as an individual intelligence), (2) elimination and dispersal process occur for all bacteria except the best bacterium, (3) swimming step is defined as a adaptive function, and (4) the process will be stopped when the Chemotaxis loop occurs.

In the proposed algorithm using new added terms, best direction for finding global minimum point is achieved based on prediction method. Also, this method is used for optimal design of the conventional Propagation–Integration (PI) controller coefficients for Switched Reluctance Motor (SRM) drive.

This work is organized as follows; bacteria foraging progress in general and proposed algorithm which is called smart bacteria foraging algorithm are explained in Section 'Bacterial foraging progress' and Section 'Smart bacterial foraging algorithm', respectively. The modeling of SRM and its drive are presented in Section 'Modeling Switched Reluctance Motor (SRM)'. Simulation and experimental results are shown in Section 'Experimental and simulation results'. Finally, the conclusion is presented in Section 'Conclusions'.

## Bacterial foraging progress

Recently, search and optimal foraging of bacteria is used to solve the optimization problem. An animal needs communication facilities for social foraging. It can take the advantages of being in group. The group can gang-up on larger prey, individuals can obtain protection from predators, and the group can forage a type of collective intelligence [38,39]. Also each animal enjoys individual intelligence to live and foraging sources of nutrition. The bacteria use this algorithm to find food substance. Finally, the bacterium that reaches to the enough and adequate food can stay alive.

In this paper, the foraging behavior of *E. Coli* is considered, which is a common type of bacterium [39]. The chemotactic action of the bacteria is modeled as follows:

- In a neutral means, if it tumbles and runs in an alternating manner, its action could be similar to search.
- If swimming up a nutrient gradient (or out of noxious substances), or swimming for a longer period of time (climb up nutrient gradient or down noxious gradient), it searches significantly desirable environments.
- If swimming down a nutrient gradient (or up noxious substance gradient), then it avoids the searching in undesirable environments

Consequently, it can climb up nutrient hills and at the same time avoid harmful substances. [38].

Natural selection tends to eliminate animals with poor foraging strategies and favor the propagation of genes of those animals that have successful foraging strategies, since they are more likely to enjoy reproductive success. After many generations, poor foraging strategies are either eliminated or shaped into better ones. The control system includes four sections: Chemotaxis, Swarming, Reproduction, Elimination and Dispersal.

- *Chemotaxis*

This process is achieved through swimming and tumbling by flagella. Depending upon the rotation of flagella in each bacterium, it decides whether it should move in a predefined direction (swimming) or change its direction to other side (tumbling). To perform a tumble, a unit length at the smart direction and random value, express $\varphi(j)$, are generated to decrease the cost function. This will be used for specifying the direction and step of movement after a tumble.

In particular:

$$\alpha^i(j+1,k,l) = \alpha^i(i,k,l) + C(i)\varphi(j) \tag{1}$$

where $\alpha^i(i,k,l)$ specifies the position of $i$th bacterium of S (total population of bacterium) at $j$th chemotactic, $k$th reproductive, and $l$th elimination and dispersal step. In this case $C(i)$ is the size of the step that is taken in the random direction only at the beginning of the algorithm and after that, it is specified smartly. Suppose that we want to find the minimum of $J(\alpha)$, as a main cost function, for a bacterium at the position $\alpha$.'$C(i)$' which is calculated based on the cost function $J(\alpha^i)$ for $i$'th bacterium

$$C(i) = \max\left(C_{\min}, \frac{|J(i) - J_0|}{K_1}\right) \tag{2}$$

where $J_0$ is the global optimum solution and $J(i)$ is cost function value for $i_{th}$ bacteria ($K_1 \gg |J|$). This variable changes adaptively and its positive constant coefficients should be tuned properly. Considering the equation, '$C(i)$' can be adapted by changing the cost function for each bacteria, so that the bacteria that is closer to the global optimum (lower cost function) can have a smaller step size to reach the global point. It can increase solution accuracy.

On the other hand, '$\varphi(j)$' can be represented as follows:

$$\varphi(i) = \beta(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{3}$$