On Line Parameter Estimation of Electric Systems using the Bacterial Foraging Algorithm

Gabriel Noriega, José Restrepo, Víctor Guzmán, Maribel Giménez and José Aller
Universidad Simón Bolívar
Valle de Sartenejas, Baruta Edo. Miranda
Caracas, Venezuela
Tel.: +58 / (212) – 906.40.12.
Fax: +58 / (212) – 906.36.31.
E-Mail: gnoriega@cantv.net, restrepo@usb.ve, vguzman@usb.ve,
mgimenez@usb.ve, jaller@usb.ve
URL: http://siep.grupos.usb.ve/

Acknowledgements

The authors want to express their gratitude to the Dean of Research and Development Bureau (DID) of the Simón Bolívar University, for the annual financial support provided to the GSIEP (registered as GID-04 in the DID) to perform this work..

Keywords

«Estimation technique», «Modeling», «Real time simulation », «Bacterial Foraging»

Abstract

In this work the “Bacterial Foraging” strategy is presented and used to identify in real time a three-phase load. The optimization process uses the instantaneous current and voltage values as inputs, and estimates the values of the resistive and inductive parameters in the system. Simulations are used to verify the proposed strategy.

Introduction

There is a growing interest in on line or real time parameter estimation in electric systems [1, 2], but the standard methods present some problems: They are affected when the system operating conditions change, they do not guarantee a global optimal solution, they require complex control strategies having a high computational load and they are prone to errors due to simplifying assumptions in the models or to parameter variations due to changes in operating conditions [3, 4].

This work presents a method able to identify on line the parameters in an electric system, using a technique based on the bacterial foraging algorithm [5, 6]. The bacterial foraging algorithm mimics the natural selection process in which Escherichia Coli bacteria seek food and exchange information about their environment [7]. This behavior can be represented in an algorithm as a kind of group intelligence [8].

The initial validation of the proposed technique is performed using as a test circuit a three phase R-L load.

Bacterial foraging

Escherichia Coli has 6 flagella that work as motors, rotating at a speed of between 100 to 200 rev/s. When the flagella rotate counterclockwise, they pull the bacteria in one direction and it moves forward. When the flagella rotate clockwise, no coherent pull direction is established and the bacteria rotates [9].

The bacteria movements are affected by the chemical composition of the surrounding environment (chemotaxis), and it will move to avoid damaging chemical concentrations and to search for nutrients as follows:

When a negative nutrient gradient (or a positive one for damaging chemicals) is found, the bacteria will rotate until a positive nutrient gradient (or a negative one for damaging chemicals) is found.
When a positive nutrient gradient (or a negative one for damaging chemicals) is found, the bacteria will advance along the gradient.

If the medium is neutral, that is lacking in either nutrients or damaging chemicals, the flagella will alternate their rotations, producing a search pattern of alternate turns and small advances.

The bacterial foraging algorithm

The bacterial foraging algorithm mimics the Escherichia Coli behavior in order to search the complete space solution until a global maximum (or minimum) is reached.

In the algorithm a number S of search elements (bacteria) move in a search space defined by the parameters to be identified. Each bacteria represents a search element looking for the best possible solution to the problem. Initially the bacteria are randomly distributed in the search space, each bacteria position in the p-dimensions space representing a possible solution to the problem, where p is the number of unknown variables. Once the colony is set, the bacteria "forage", that is, evaluate their environment seeking the gradient line in which the cost function is reduced from its initial value. Search speed is a direct function of the step size $C(i)$, assigned to each bacteria (long steps produce fast searches), but search accuracy varies in inverse relationship with step size. After a preset number of foraging events, bacterial efficiency is evaluated, those bacteria that are "underfed" (unable to find a gradient line reducing the cost function) are eliminated and moved to new locations, and those that are progressing in promising gradient lines are reinforced with new bacteria [9]. This event is called a reproductive event, and N_c is used to represent the number of search steps that precede each reproductive event.

If the search space remains constant, eventually all bacteria will converge at optimal feeding points. To ensure that the bacteria do not lock into a local instead of a global maximum, the algorithm, with a probability P_{ed}, moves some bacteria after a number of reproductive events (generations). This is called a elimination and dispersion event, and the number of reproductive events preceding each elimination and dispersion event is N_{re}. Finally N_{ed} is the number of elimination and dispersion events considered in one program run [6].

Search and tracking efficiencies are affected by the quality of the cost function used, and by internal parameters in the foraging algorithm, such as step size, number of steps per cycle, population size and number of bacteria eliminated or created at the end of each cycle. These parameters can be optimized by trial and error while working off line in such a way that the on line optimization is done using the best performing algorithm, this is, the one with the cost function and search parameters producing the best performance: faster convergence, lower error, more stable solution, better parameter changes tracking, or any required combination of these and other possible qualities.

Problem definition

To find out if the proposed method is able to identify the values of component parameters in an electric circuit, the operation of a $R-L$ three-phase load system is considered, and the parameters to be identified are the R and L values. The example was selected since it is simple enough and the results of changes in the foraging algorithm will be easy to identify. Additionally it uses the same variables set that more interesting (and therefore complex) parameter estimation problems presented by electrical machines [10, 11].

The system is described by the following equation,

$$\frac{d}{dt}\begin{bmatrix} i_x(t) \\ i_y(t) \end{bmatrix} = A\begin{bmatrix} i_x(t) \\ i_y(t) \end{bmatrix} + B\begin{bmatrix} v_x(t) \\ v_y(t) \end{bmatrix} \tag{1}$$

Where its discrete time equivalent is:

$$\begin{bmatrix} i_x(n+1) \\ i_y(n+1) \end{bmatrix} = A_D\begin{bmatrix} i_x(n) \\ i_y(n) \end{bmatrix} + B_D\begin{bmatrix} v_x(n) \\ v_y(n) \end{bmatrix} \tag{2}$$

Defining:

$$y = \begin{bmatrix} i_x(n+1) \\ i_y(n+1) \end{bmatrix} \tag{3}$$

$$z^t = \begin{bmatrix} i_x(n) & i_y(n) & v_x(n) & v_y(n) \end{bmatrix} \tag{4}$$

$$\Theta = \begin{bmatrix} A & B \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & b_{11} & b_{12} \\ a_{21} & a_{22} & b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a & 0 & b & 0 \\ 0 & a & 0 & b \end{bmatrix} \tag{5}$$

$$y = \Theta(z) \tag{6}$$

The error is defined as the difference between the R and L values at the search point and the real R and L values. Whatever variables set is considered as a cost function, it must include the required unknown network parameters R and L. The cost function value at each search point $J(\theta)$ must increase as the error increases. In this work two cost functions are considered: the system current quadratic error, defined as,

$$J(\theta) = e_i^2 = \left(y - \hat{\Theta}z\right)^2 \tag{7}$$

and the system power quadratic error, defined by,

$$J(\theta) = e_p^2 = \left(\left|\vec{s}\right| - \left|\hat{\vec{s}}\right|\right)^2 \tag{8}$$

and, defining the instantaneous apparent power:

$$\vec{s}(t)=\vec{v}(t)\cdot\vec{i}^*(t)=p(t)+jq(t) \tag{9}$$

where,

$$\begin{aligned} p(t)&=2\pi f\left(\vec{\lambda}\times\vec{i}\right)=\lambda_x i_y-\lambda_y i_x \\ q(t)&=2\pi f\left(\vec{\lambda}\cdot\vec{i}\right)=\lambda_x i_x+\lambda_y i_y \end{aligned} \tag{10}$$

The system power quadratic error is:

$$J(\theta)=e_p^2=\left(\left|\vec{s}\right|-\left|\hat{\vec{s}}\right|\right)^2 \tag{11}$$

Whatever variables set is considered as a cost function, it must include the required unknown network parameters R and L. While the values of these parameters in a given search point are further away from the actual values in the real circuit, $J(\theta)$ will be increase, and the reverse is also true. Therefore $J(\theta)$ is a cost function related to the parameter matrix Θ, and the optimizer aims to minimize this function.

In the cost function to be minimized $J(\theta)$, $\theta\in R^p$, θ, represents the position of a given search element (a bacteria), and represents the actual value of the cost function calculated for that position. Hence, the function defined as follows:

$$P(j,k,l)=\left\{\theta^i(j,k,l)\,|\,i=1,2,...,S\right\} \tag{12}$$

represents the position of each of the bacteria in the colony during the j chemotactic search step in the k generation in the l elimination and dispersion event, and $J(i,j,k,l)$ represents the cost function for the i bacteria in position $\theta^i(j,k,l)\in R^p$.

The procedure is then to build the parameters matrix Θ, shown in (5) and use it to estimate on line the actual values of the resistance and inductance elements in the electric circuit. Those values can then be used to update the circuit model as shown in (6), improving the accuracy of the control system in use.

The search algorithm is formed by three concentric loops. The outermost, the dispersion and elimination loop initiates new searches in order to avoid local minima, and to search for new and better solutions as the real circuit parameters drift with time. This external loop is the least frequently performed. The next inner loop is the reproduction loop. In this loop the bacteria with the best cost functions are allowed to reproduce, raising the number of bacteria searching in the best areas. The innermost loop is the chemotactic search loop. In this loop the gradient for each bacteria is evaluated, and the bacteria are accordingly moved to their next position, closer to the final target. This loop is the one most frequently performed.

Simulation results

Once the system equations and the possible cost functions were defined, a program implementing the bacterial foraging algorithm was developed in C, and executed on an Analog Device DSP Prototyping System.

The real circuit parameters used in the test are $R=3\Omega$, and $L=30mH$. The first search starts with random estimates for the two parameters. In the first test the two outer loops (the reproduction and the elimination and dispersion loops) were inactive, in order to study the performance of the search loop using search steps of six different sizes. The quadratic error current cost function was selected for the initial test.

The graphs in Fig. 1 show the behavior of bacteria 1, the one with the longest search step, and bacteria 6, the one with the shortest search step. As can be seen, the bacteria with the longest search step moves faster, and within about 250 search cycles it reaches an estimate close to the real parameter values, but the overshoot is big, and afterwards it moves around the target value with a 30% ripple. The bacteria with the shortest search step reaches the target more slowly, taking about 500 search cycles, but once near the target is more precise, the overshoot is smaller and the ripple is only 1%.

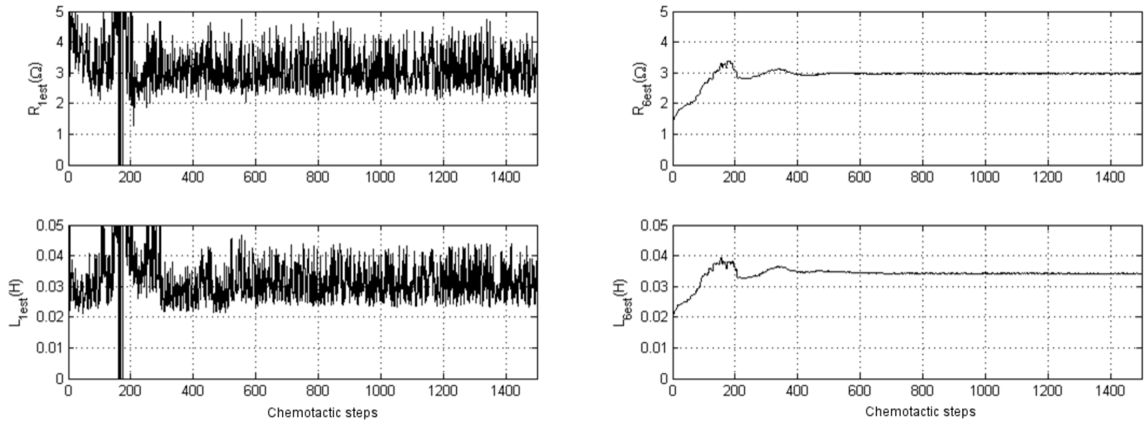

Fig. 1: R and L values estimated by bacteria 1(left) and 6 (right) using the quadratic current error as the cost function. No reproduction or elimination and dispersion events activated

Evolution can be used to combine the best characteristics of the long search step bacteria (fast convergence) and the short search step bacteria (low convergence error). To do this the reproduction loop was activated, causing one reproduction event each 100 search events. The same bacteria types and quadratic current error cost function were used. Fig. 2 shows the results obtained using the two loops in the search algorithm. As expected, less time is required to reach accurate estimates for the R and L parameters, and the overall ripple in the values when the target is reached is also smaller. There are some random initial changes, but they attenuate quickly.

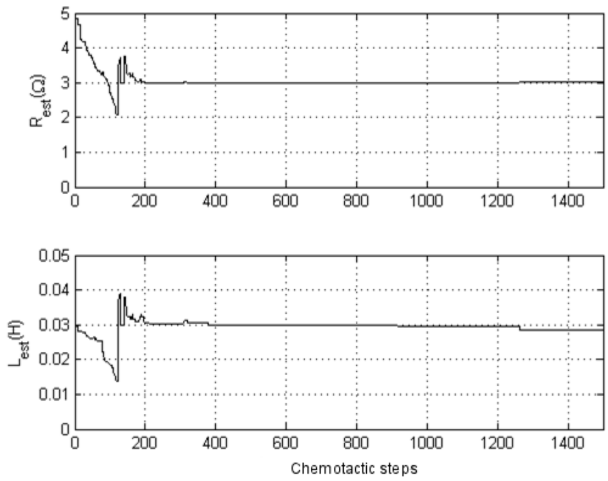

Fig. 2: R and L values estimated by bacteria 1 and 6 using the quadratic current error as the cost function. One reproduction event per each 100 chemotactic search steps. No dispersion events activated.

To test the search algorithm generality, the search was performed in the same conditions, but using the circuit power error as the cost function. Fig. 3 shows the new test results, which are very similar to those presented in Fig. 2, indicating that the method is not particularly sensitive to changes in the cost function.

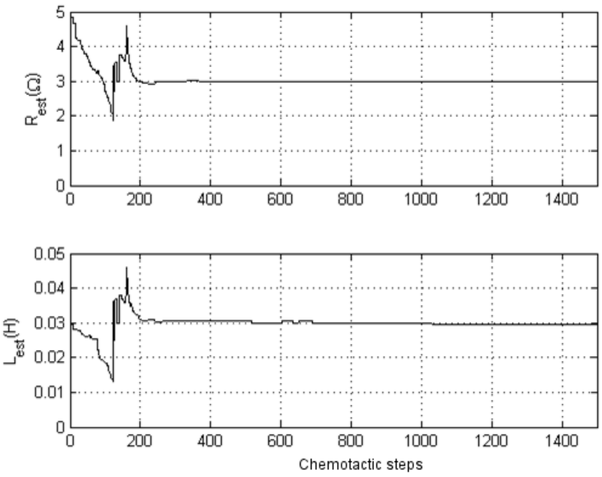

Fig. 3: R and L values estimated by bacteria 1 and 6 using the quadratic current error as the cost function. One reproduction event per each 100 chemotactic search steps. No dispersion events activated.

As can be seen comparing Fig. 2 and 3, in both cases the estimated parameter values converge with the real ones in about 200 steps with errors below 1%, giving an almost perfect match between the model and the real circuit, and the two cost functions are able to produce almost the same accuracy levels, but the best overall performance was obtained using the system current quadratic error cost function, since the estimated values were more stable and the initial oscillations were also smaller. Hence the quadratic current error cost function is the best for this particular problem.

For the next test, aiming to reduce overshoots, initial oscillations and search time, the number of reproduction events were increased to one each ten search steps. The quadratic current error cost function was used. As can be seen in Fig. 4, the objectives were achieved: When increasing the number of reproduction events, the time required to reach the target values is shorter, the overshoots are reduced and almost no oscillations are observed. Furthermore, the final estimation error is also reduced, the final values obtained are 3.000 Ω for the resistance and 29.995 mH, for the inductance.

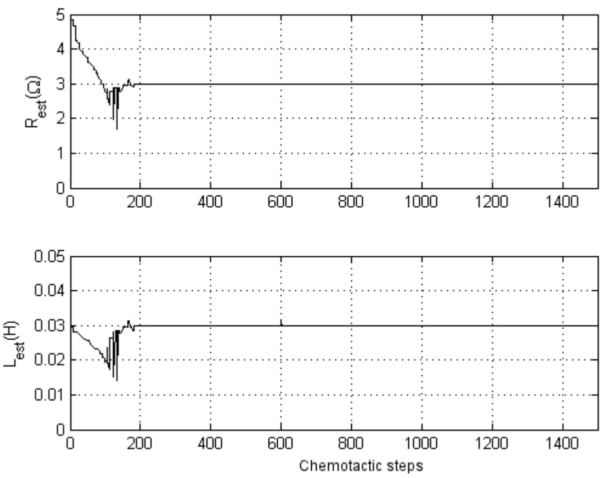

Fig. 4: R and L values estimated by bacteria 1 and 6 using the circuit power error as the cost function. One reproduction event per each ten chemotactic search steps. No dispersion events activated

Conclusion

The Bacterial Foraging is an absolute optimizer and works effectively in parameter estimation for on line models, as has been proved in the estimation of the electric model parameters. The errors in the estimated parameters are below 1% of the real circuit values, giving a very precise estimation of the system parameters.

The cost function defined using the system current quadratic error produces more stable parameter estimation than the one obtained when the system power quadratic error is used to define the cost function. This is due to the fact that system power calculations require a flux linkages evaluation. Nevertheless the results obtained using both cost functions are similar.

The method was tested using a simple electrical circuit in order to show how the Bacterial Foraging strategy can be used to find an electrical problem solution. The same method is now being used for on line parameters estimation in more complex systems, such as electrical machines.

To ensure that the Bacterial Foraging algorithm used to estimate electric circuit parameters is fast enough to be useful for on-line parameter estimation in electric motor controllers, all the calculations presented in this work were performed using the same DSP based system in which the electrical machine parameter estimator will be tested.

Considering the method performance, actually this technique is been applied to the on line parameter estimation on induction and synchronous machines.

References

[1] Ljung L.: System Identification. Theory for the User, Upper Saddle River, NJ: Prentice Hall, 1999

[2] Ichikawa T., Doki O.: Sensorless Control of Permanent-Magnet Synchronous Machine using Online Parameter Identification Based on System Identification Theory, IEEE Transactions on Industrial Electronics, 2006.

[3] Anstrom K.J. and Hagglud T.: Automatic tuning of simple regulators with specifications on phase and amplitude margins, Automatica, Vol. 20, pp. 645 – 651, 1984.

[4] Poulin E. and Pomerleau A.: PI setting for integrating processes based on ultimate cycle information, IEEE Trans. On Control Systems Technology, Vol. 7, No 4, July 1999.

[5] Passino K.: Biomimicry of Bacterial Foraging for Distributed Optimization, University Press, Princeton, New Jersey, 2001.