International Journal of Engineering Research (ISSN:2319-6890)(online),(2347-5013)(Print)
Volume No.3, Issue No.12, pp: 807-811 1 Dec. 2014

Artificial Bee Colony-Based Design of Optimal On-Line Self-tuning PID-Controller Fed AC Drives

Essamudin A. Ebrahim

Power Electronics and Energy Conversion Department
Electronics research Institute, El-tahrir Street, 12622 Dokki, , Cairo, EGYPT
essamudin@yahoo.com

Abstract — In this paper, an optimization method based on the foraging behavior of a colony of honey bees in nature was proposed. An optimized model for Proportional Integral Derivative (PID)- speed controller of 3-phase IM drives was introduced to adapt the constant gains of this classical controller. To enhance robustness of this controller, an on-line algorithm for self-tuning of PID controller-parameters was proposed. Experimental work was introduced using DSP and test results were obtained for several speed trajectories of 2-degree of freedom IM drive and compared with conventional PID controllers.

Keywords — Artificial Bee Colony Optimization, fitness, Objective function, Scout and Onlookers, Induction Motor

I. Introduction

The proportional-integral- derivative (PID) controllers have been widely used in industry for many years. Unfortunately, robustness and optimization not be achieved with this controller due to uncertainty of plant, and system nonlinearity.

The Ziegler and Nichols tuning method is mostly used to compute optimal PID gains for the plant under certain operating point. If the parameters of the plant are changed due to uncertainty or nonlinearity, the controller becomes not robust and not in optimum state.

Particle Swarm Optimization (PSO) (E. A. Ebrahim et al, 2013), or germ of intelligent (as ant colony optimization ACO (Q. Zeng and G. Tan, 2007) and bacteria foraging BF, (A. Oshiba and E. Ali, 2014) techniques have inspired as new techniques for optimal design of PID controllers. One such a new algorithm is the Artificial Bee Colony (ABC) algorithm that proposed by (Karaboga in 2005).

Most articles suggest off-line design for PID-controller parameters and then use in on-line control. In this paper, authors have developed an optimization procedure based on the foraging behavior of an artificial bee colony. The foraging behavior is perceived as a minimization method and an optimization framework that used in designing of indirect field-oriented (IFO) vector control of induction motor controller. The proposed ABC technique is used in on-line self-tuning of PID- controller parameters to achieve optimization. This proposed intelligent controller is abbreviated and known as (ABC-PID) controller.

This paper is organized as follows: Section I introduction, Section II deals with mathematical modelling of field-oriented vector-control of IM, objective function and ABC algorithm. Section III includes test results and comparison study.

II. Material and Methodology

A. Mathematical Model of field-Oriented IM

The dynamic state equations of a 3-phase IM can be described in a fixed stator d-q frame (D. W. Novotny and T. A. Lipo,1996) as follow:

$$\begin{bmatrix} v_{ds} \\ v_{qs} \\ v'_{dr} \\ v'_{qr} \end{bmatrix} = \begin{bmatrix} [R_s+\frac{p}{\omega_{sb}}X_1] & -(\frac{\omega_b}{\omega_{sb}}X_1) & \frac{p}{\omega_{sb}}X_M & -(\frac{\omega_b}{\omega_{sb}}X_M) \\ -(\frac{\omega_b}{\omega_{sb}}X_1) & [R_s+\frac{p}{\omega_{sb}}X_1] & (\frac{\omega_b}{\omega_{sb}}X_M) & \frac{p}{\omega_{sb}}X_M \\ \frac{p}{\omega_{sb}}X_M & -(\omega_b-\omega_r).\frac{X_M}{\omega_{sb}} & [R'_r+\frac{p}{\omega_{sb}}X'_2] & -(\omega_b-\omega_r).\frac{X'_2}{\omega_{sb}} \\ (\omega_b-\omega_r).\frac{X_M}{\omega_{sb}} & \frac{p}{\omega_{sb}}X_M & (\omega_b-\omega_r).\frac{X'_2}{\omega_{sb}} & [R'_r+\frac{p}{\omega_{sb}}X'_2] \end{bmatrix} . \begin{bmatrix} i_{ds} \\ i_{qs} \\ i'_{dr} \\ i'_{qr} \end{bmatrix} \quad (1)$$

$$T_e=\frac{3}{2}\frac{P}{2}(\lambda_{ds}i_{qs}-\lambda_{qs}i_{ds})=\frac{3}{2}\frac{P}{2}L_m(i'_{dr}i_{qs}-i'_{qr}i_{ds})=\frac{3}{2}\frac{P}{2}\frac{L_m}{L'_r}(\lambda'_{dr}i_{qs}-\lambda'_{qr}i_{ds}) \quad (2)$$

Or, $T_e = J\frac{d\omega_m}{dt} + B\omega_m + T_l$ (3)

$$\omega^*_{sl} = (S\omega_s)^* = \frac{R'_r I^{e*}_{qs}}{L'_r I^{e*}_{ds}} = \frac{1}{\tau_r}\frac{I^{e*}_{qs}}{I^{e*}_{ds}} \quad (4)$$

The motor developed torque can be directly related to I^{e*}_{qs} from eqn. (2) as:

$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L'_r}\lambda'^{e*}_{dr} I^{*e}_{qs} = K_T \lambda'^{e*}_{dr} I^{e*}_{qs} \quad (5)$$

The schematic diagram of IFO-IM drive with the proposed speed controller is shown in figure 1 (E. A Ebrahim, 2001).

B. Formulation of Objective Function

Artificial bee colony (ABC) tuning algorithm is used to get the optimum values of gains for the PID-controller based on speed error. All the particles of populations are decoded for K_p, K_i, K_d. The controller design is first redrafted as an optimization problem where the objective function comprises dynamic response specifications.

International Journal of Engineering Research (ISSN:2319-6890)(online),(2347-5013)(Print)
Volume No.3, Issue No.12, pp: 807-811 1 Dec. 2014

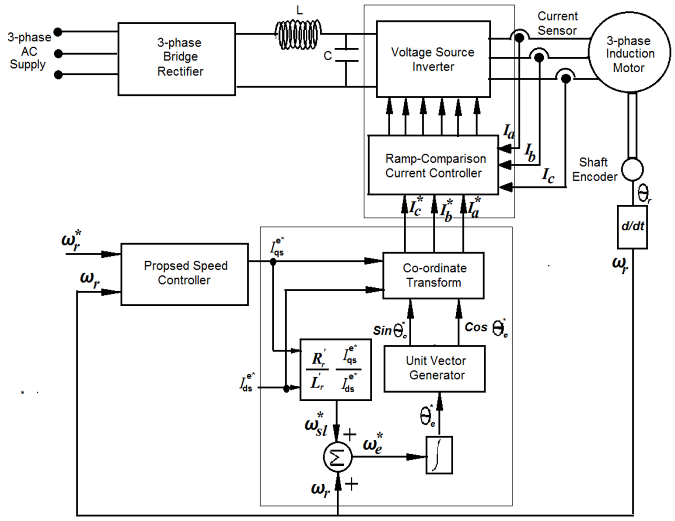


Figure. 1 Closed-loop speed control of an IM based on indirect field-oriented control

Two objective functions are proposed. The first one uses Integral Time Absolute Error (ITAE). ITAE is chosen as an objective function because of its advantage in producing smaller overshot and oscillations. Equation 6 represents the first objective function as:

$$ITAE = \int_0^{\infty} (t^* |e(t)|)\, dt \tag{6}$$

(And\Or) the second objective function can be represented as:

Minimize $F(\phi) = \Delta \omega_{rp} + t_s$ (7)

Subject to the constraint: $\phi_{\min} \le \phi \le \phi_{\max}$, where ϕ represents the set of controller K_p, K_i and K_d

Where, $e(t) = \omega_r^* - \omega_r$ (8)

In an off-line control of IM drive, equations (6-8) will be used as objective functions to optimize the initial gains of the proposed controller.

C. *Artificial Bee Colony (ABC) Optimization*

In ABC model, the colony consists of three groups of bees: employed, onlookers and scouts (A. Kaur and S. Goya, 2011)

- *Pseudo-codes of the ABC algorithm* (Teodorovic et al.,2011)

1. Load samples of controller parameters
2. Generate the initial population $x_i, i = 1,2,.... FS$
3. Evaluate the fitness (fit_i) of the population
4. set cycle to 1
5. repeat
6. For each employed bee {Produce new solution X_{new} by using eqn. (11). Calculate the value (fit_i) by using eqn. (10).Apply greedy selection process}
7. Calculate the probability values (P_i) for the solutions (X_i) by eqn. (9)
8. For each onlooker bee {Select a solution X_i depending on P_i Produce new solution X_{new} Calculate the value fit_i Apply greedy selection process}
9. If there is an abandoned solution for the scout. Then replace it with a new solution which will be randomly produced by (12)
10. Memorize the best solution so far
11. cycle= cycle+1
12. Until cycle=MCN

Where, X_i represents a solution, fit_i is the fitness value of X_i, X_{new} indicates a neighbor solution of X_i, P_i the probability value of P_i and MCN is the # of maximum cycle in the algorithm.

- *Detailed Explanation for the Algorithm*

In the algorithm, first half of the colony consists of employed artificial bees and the second half constitutes the onlookers. The number of employed bees is equal to the number of food sources (# of solutions in the population).

At first step, the algorithm starts by initializing all employed bees with randomly generated food sources (solutions), where SN denotes the size of population. Each solution $x_i, i = 1,2,....FS$ is a D-dimensional vector. Where D is the number of optimization parameters. Here, in this study, D- represents the PID-controller parameters to be optimized.

i.e., $D = \lfloor K_p \; K_i \quad K_d \rfloor$

After initialization, in each iteration of all given cycles, every employed bee finds a food source neighbourhood of its current food source and evaluates its nectar amount, i.e. fitness). In general the position of i_{th} food source is represented as: $X_i = (X_{i1}, X_{i2}, ..., X_{iD})$.

After the information is shared by the employed bees, onlooker bees go to the region of food source at X_i based on the probability P_i determined as:

$$P_i = \frac{fit_i}{\sum_{k=1}^{FS} fit_k} \tag{9}$$

Where, FS is total number of food sources. Fitness value fit_i is calculated using:

$$fit_i = \frac{1}{1 + f(X_i)} \tag{10}$$

Here, $f(X_i)$ is the objective function, in this study, an objective function is *ITAE* determined from equation 6 and /or $F(\phi)$ that determined from eqn. 7. The onlooker finds its food source in the region of X_i by using the following equation:

$$X_{new} = X_{ij} + r * (X_{ij} - X_{kj}) \tag{11}$$

Where, $k \in (1,2,3,... FS)$ such that $k \notin i$ and $j \in (1,2,3,... D)$ are randomly chosen indexes, r is a uniformly distributed random number in the range [-1,1].

The value of predetermined number of cycles is an important control parameter of the ABC algorithm, which is called "limit" for abandonment. Assume that the abandoned source is X_i and $J \in \{1,2,... D\}$, then the scout discovers a new food source to be replaced with X_i. This operation can be expressed (K. Vivekanandan et al, 2011 and M. çelik et al, 2011) in the following relation:

$$X^{j}_{new} = X^{j}_{\min} + rand(0,1)(X^{j}_{\max} - X^{j}_{\min}) \qquad (12)$$

D. On-line tuning of ABC-PID controller

The tuning process depends on the rate of change of error signal and its value. The sequence of program subroutine algorithm can introduced as a flow chart of figure 2. In the flowchart, there are three categories for error values:

$e(t) > e_{\max}$ ii. $e(t) < e_{\min}$ iii. $e_{\min} \le e(t) \le e_{\max}$

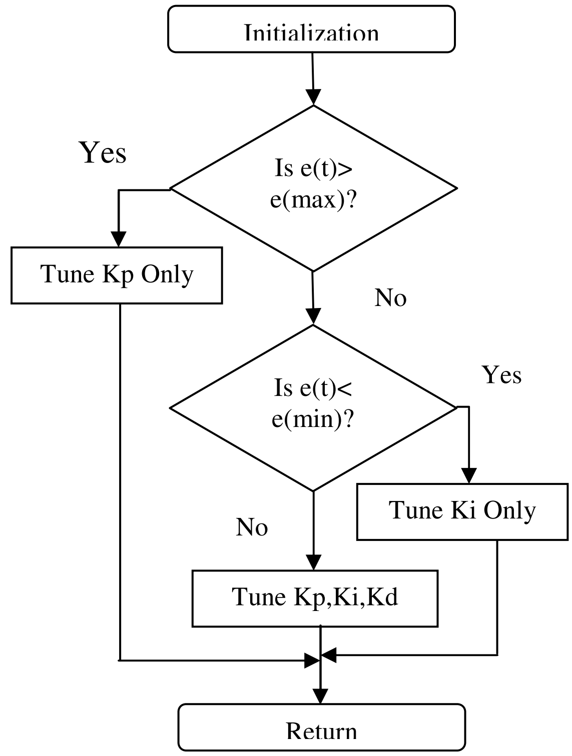


Figure 2. Flow chart of ABC-PID on-line Controller

III. Test Results and Tables

A computer program was written in MATLAB (m-file) to implement (Mathwork Co., 2012) the proposed controller design based on bees colony concept. K_p, K_i, K_d, can be defined as:

$$\theta(FS, D) = \begin{vmatrix} K_p^1 & K_i^1 & K_d^1 \\ K_p^2 & K_i^2 & K_d^2 \\ .. & .. & .. \\ K_p^S & K_i^S & K_d^S \end{vmatrix}$$

Where FS= no of food sources, and D is the parameter optimized. In this work, D=3, S=10, and the controller parameters are bounded as: {0, $K_{p\max}$ }, { 0, $K_{i\max}$ }, ABC algorithm was selected by trial and error and the following table (I) includes these parameters:

Table I. Parameters of bee foraging algorithm

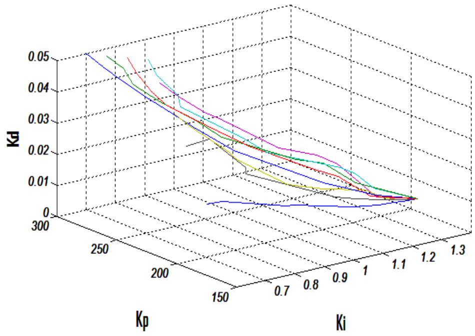
Number of Colony Size=Np	20
Number of Food Sources (S=Np/2)	10
Maximum Number of Cycles	50
Threshold Probability	0.75
Number of Optimized Parameters	3

The proposed controller was applied on an IM drive as a test machine which all parameters are shown in table (II).

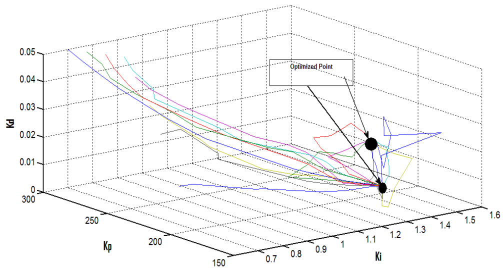
Table (II) Motor parameters

Rated line voltage (Y-connection)	380V
Rated line current	1.2 A
Rated power	205W
frequency	60 Hz
No of poles	4 poles
Stator and rotor resistances	26.5 Ω , 21.0 Ω
Stator and rotor leakage reactance's	30.75 Ω ,30.75 Ω
Magnetizing Reactance X_m	416.0 Ω
Moment of Inertia for Rotor J	0.0045 Kg. m^2

The off-line results for that optimized controller parameters are obtained in figure 3(a). As shown in this figure, the bees movement is plotted against K_p, K_i, K_d axes. From the convergence characteristics it is seen that all bees converge to a single point similar to all worker bees converge to the best flower patch in nature. The graph of figure 3 is a clear indication that the proposed algorithm truly imitates the foraging behaviour of bees. The optimality of the bees algorithm in yielding quality solutions with faster convergence rate reaches at the 39th cycle of the program iteration. At the end of these iterations, the optimized values of the ABC-PID proposed controller are obtained as follows: $K_p = 171$, $Ki = 1.33$, and finally $K_d = 0.008$



(a) Initial values



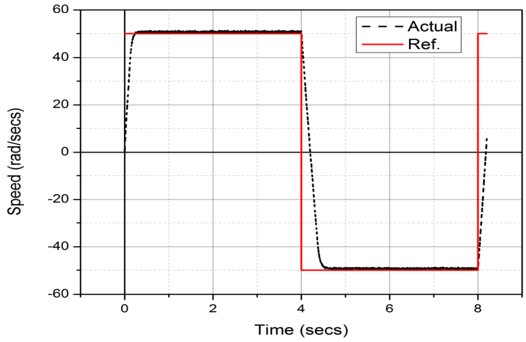
(b) with tuning

Figure. 3 *Bees movements along Kp, Ki, and Kd axes*

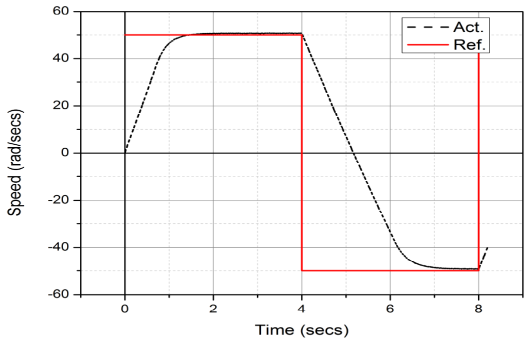
These parameters are obtained for step-input of speed-reference trajectory- for test machine - These optimized values for the controller that obtained above, are used as initial values for the proposed on-line ABC-PID controller to test the robustness of the controller in 2-DOF tracking control. Several trajectories in 2-DOF are proposed to test the controller robustness. Figure 3(b), shows the bees movement to optimize the parameters of controller at certain proposed trajectory. Point 2 in that figure, represents another nectar source that bees moved to it. This point is an operating point for the proposed controller to track the proposed trajectory. The tuned controller parameters at that operating point are:

$K_p = 234$, $Ki = 1.48$, and finally $K_d = 0.011$

Figures (4,5,6,7) show the square-wave, trapezoidal, triangle, and sinusoidal trajectories with self-tuned adaptive controller and with conventional PID- controller. Solid line represents the reference and dashed line is the actual speed trajectory. Also, on right hand, the output with ABC-PID controller (a) and on the left-hand (b), indicate the trajectory without controller. From these figures, the motor under the proposed controller, can track the proposed trajectories and matches with the reference speed values without deviation. But, the classical or traditional PID-controller with fixed values without adaptation cannot be able to track the proposed trajectories. Also, this algorithm was written in Boland C++ and coded to an executable file that drives an experimental rig includes: a test machine, and a Texas DSP-TMS320C30 Card and IGBT-intelligent power inverter module.

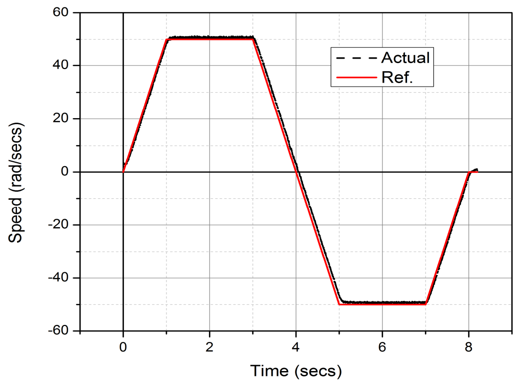


(a) With ABC-PID Controller

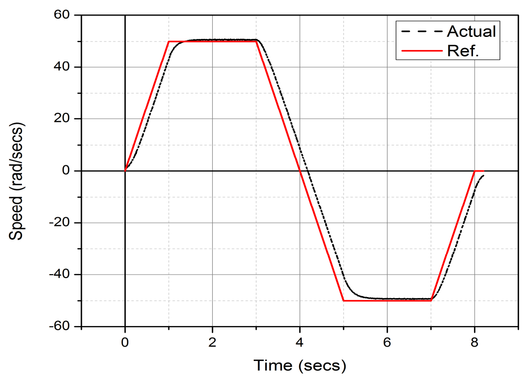


(b) With classical PID controller

Figure. 4 Square wave speed trajectory

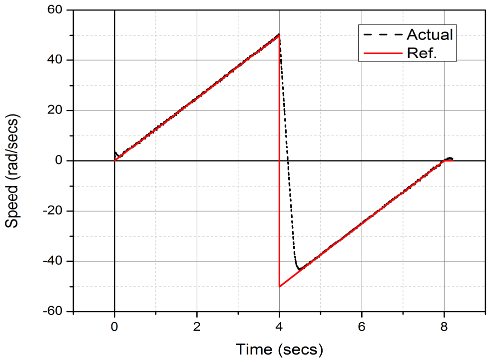


(a) With ABC-PID Controller

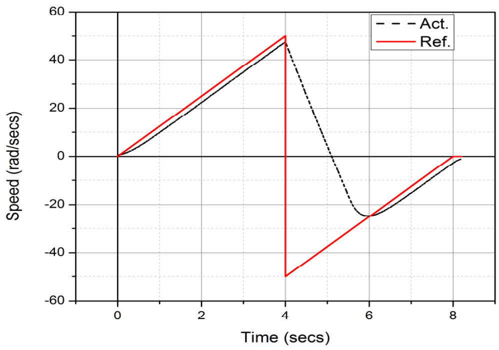


(b) With classical PID controller

Figure 5. Trapezoidal speed trajectory

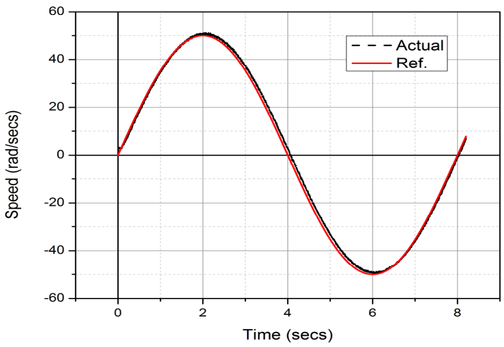


(a) With ABC-PID Controller

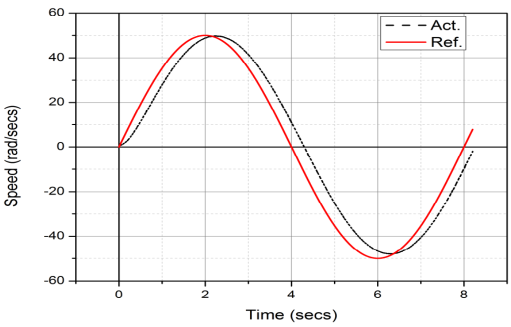


(b) With classical PID-controller

Figure 6. Triangular speed trajectory



(a) With ABC-PID controller



(b) With classical PID controller

Figure 7. Sinusoidal speed trajectory

IV. Conclusion

The foraging behavior of a colony of bees is exploited to develop an optimization algorithm and the same is used to design off-line parameters of self-tuned self-adaptive PID-controller fed induction motor drive. These optimized values are used in the proposed ABC-PID controller to make it robust through tracking control. The robustness of the controller was tested by proposing several speed reference trajectories. All proposed trajectories, the motor succeeded to track them in 2-DOF without deviation. But, the motor failed to follow the prescribed trajectories when applying conventional classical PID- controller with fixed parameters. So, by using the proposed ABC-PID controller, the controller becomes a robust with self-adaptive self-tuned parameters. Also, it provides excellent dynamic response at all operating points.

References

i. H.I. Abdul-ghaffar, E. A. Ebrahim and M. Azzam ," Design of PID Controller for Power System Stabilization Using Hybrid Particle Swarm-Bacteria Foraging ptimization, WSEAS Transaction on Power Systems, Issue 1, Volume 8, January(2013), pp. 12-23.

ii. Q. Zeng and G. Tan," Optimal design of PID controller using modified ant colony system algorithm", IEEE Third Int. Conf. on Natural Computation (ICNC'07), 24-27 Aug. (2007), China.

iii. A. S. Oshaba and E. S. Ali" Bacteria Foraging: A New Technique for Speed Control of DC Series Motor Supplied by Photovoltaic System, WSEAS Transaction on Power Systems, Volume 9, (2014), pp. 185-195.

iv. D. Karaboga, S. Okdem, and C. Ozturk, "Cluster based wireless sensor network routing using artificial bee colony algorithm", Inter. Conf. on Autonomous and Intelligent , AIS'10, June 21-23, 2010, Povoa de Varzim, Portugal.

v D. W. Novotny and T. A. Lipo, " Vector control and dynamics of AC drives", Clarendon Press, Oxford,(1996).

vi Essamudin A E, "Artificial neural network-based tracking field-oriented vector-control of 3-phase induction motor servo drives", Ph D.Thesis, Cairo University 2001.

vii. A. Kaur and S. Goyal, "A bee colony optimization algorithm for code coverage test suite prioritization", Int. Journal of Enginnering Science and Technology (IJEST), Vol. 3 No. 4, April (2011).

viii K. Vivekanandan et al., "Artificial bee colony algorithm for grid scheduling", Journal of Convergence Information Technology, Vol. 6, No. 7, July (2011).

ix. D. Teodorovic et al., "Bee colony optimization: The applications survey", ACM Transactions on Computational Logic, (2011), Pages 1-20.

x. M. çelik et al, "Artificial bee colony data miner", 2011 International Symposium on Innovations in Intelligent Systems and Applications (INISTA), 15-18 June (2011).

xi. Mathwork Corporation, "Matlab/ Simulink 2012b", User's Guide, USA (2012).