Turk J Elec Eng & Comp Sci, Vol.20, No.Sup.1, 2012, © TÜBİTAK
doi:10.3906/elk-1012-956

# Identification of linear dynamic systems using the artificial bee colony algorithm

**Özden ERÇİN, Ramazan ÇOBAN***
*Department of Computer Engineering, Çukurova University,*
*01330 Balcalı, Sarıçam, Adana-TURKEY*
*e-mails: ozdenercin@yahoo.com, rcoban@cu.edu.tr*

Received: 02.01.2011

**Abstract**

*This paper presents an investigation into the development of system identification using the artificial bee colony (ABC) algorithm. A system identification task can be formulated as an optimization problem where the objective is to obtain a model and a set of parameters that minimize the prediction error between the measured plant outputs and the model outputs. The most common existing system identification approaches, such as the recursive least squares method and autoregressive exogenous method, are substantially analytical and based on a mathematical derivation of the system's model. Evolutionary computation, which seems to be a very promising approach, is an alternative to these methods because a little knowledge about the problem is sufficient in this approach and it can be easily combined with many other techniques from artificial intelligence, control engineering, machine learning, and so on. In this paper, an evolutionary approach for system identification is considered and attempted to demonstrate how the ABC algorithm can be applied in system identification tasks. Mathematical models of dynamic systems are obtained using difference equations represented in discrete time and the ABC algorithm is used to estimate the unknown parameters of the systems. Simulation results demonstrate that the proposed linear system identification method has good identification performance. Moreover, this method is applied to the identification of a direct current motor in order to show the performance of the ABC algorithm. The obtained results show that the identified and actual plant outputs successfully match each other.*

**Key Words:** *Dynamic modeling, system identification, artificial bee colony algorithm, linear time-invariant system, DC motor speed*

## 1. Introduction

System identification contains 2 tasks, such as the structural identification of the equations and estimation of the plant parameters. From the point of view of the control engineering, the aim of system identification is to find a model of the plant to control. If the structure of the model is known in advance, the only task remaining

*Corresponding author: Department of Computer Engineering, Çukurova University, 01330 Balcalı, Sarıçam, Adana-TURKEY

is to estimate the plant's parameters. Therefore, the needed knowledge depends on the numerical values of a number of parameters. The "black box" modeling approach is the technique used in system identification. This means that system identification is not interested in what the system looks like, but by applying an input signal and using the resultant output signal, an estimate of the system's transfer function can be found. There are other forms of identification that are widely used in industry, such as modeling systems by their physical properties. As system identification is only concerned with input and output signals from the system, it takes less time than physical modeling [1].

The system identification problem is based on proposing an approximated model of a real system. Control theory has many techniques to solve this problem [1-3]: Laguerre functions, the recursive least squares (RLS) method, the autoregressive exogenous (ARX) method, etc. Recently, other techniques based on computational intelligence (artificial neural networks, evolutionary programming, etc.) have been used [4-7]. The most common existing system identification approaches, such as the RLS and ARX methods, are substantially analytical and based on mathematical derivation of the system's model. Evolutionary computation, which seems to be a very promising approach, is an alternative to these methods because it does not require any derivative information, contrary to the conventional gradient-based methods. Furthermore, a little knowledge about the problem is sufficient in this approach and it can be easily combined with many other techniques from artificial intelligence, control engineering, machine learning, and so on. The artificial bee colony (ABC) algorithm is a search tool based on the idea of searching for food sources in nature. The ABC algorithm has been used successfully to solve many problems and has been applied to constrained and unconstrained single objective function optimizations [8-20]. If the task of system identification is to estimate the parameters of a system, the ABC algorithm searches for these parameters from a specified range. In this paper, a method to estimate the parameters of dynamic systems based on the ABC algorithm using input-output data is proposed. This method guides the evolution of a function toward an input-output mapping of the system. The various simulation examples of linear dynamic systems identified by the proposed method are demonstrated. As a more realistic example, the proposed method is applied to estimate DC motor parameters [21-23] in order to show the performance of the ABC algorithm.

In this paper, the parameters of different linear single input-single output (SISO) plants and a DC motor are estimated using the ABC algorithm. The paper is organized as follows. The principle of the ABC algorithm is introduced in Section 2. Linear system identification is outlined in Section 3. Application of the ABC algorithm in system identification is introduced in Section 4. In Section 5, the simulation results of numerical examples for different order linear plants and a DC motor using the ABC algorithm are reported in order to illustrate the performance of the proposed method. Finally, the conclusions are given in Section 6.

## 2. The ABC algorithm

A basic model of foraging behavior of a honeybee colony was developed based on reaction-diffusion equations by Tereshko [24]. This model mimics the behavior of the collective intelligence of honeybee swarms. It includes 3 essential components, food sources, employed foragers, and unemployed foragers [24]. A forager bee identifies the quality of a food source, such as the taste of its nectar, nearness to the hive, efficiency of the energy, and the ease or difficulty of extracting this energy, in order to choose a food source. An employed forager serves as an information carrier from a specific food source to other bees waiting in the hive. The information contains the distance between the hive and the food source, the direction, and the cost of the food source. An unemployed forager is a forager bee looking for a food source to exploit. It can be a scout who searches the environment arbitrarily or an onlooker who attempts to find a food source through the information transported by an employed bee [24].

The main steps of the ABC algorithm are given below [11-13]:

*1. Generate the population of solutions (positions of food sources) randomly* $x_i$*,* $i = 1...SN$

*2. Evaluate the generated population*

*3. Cycle = 1*

*4. Repeat*

*5. Produce new solutions* $\upsilon_i$ *for the employed bees using Eq. (2) and evaluate them*

*6. Apply the greedy selection process*

*7. Calculate the probability values* $p_i$ *for the solutions* $x_i$ *by Eq (1)*

*8. Produce the new solutions* $\upsilon_i$ *for the onlookers from the solutions* $x_i$ *selected depending on* $p_i$ *and evaluate them*

*9. Apply the greedy selection process*

*10. Determine the abandoned solution for the scout and replace it with a new randomly produced solution* $x_i$ *by Eq. (3)*

*11. Record the best solution achieved so far*

*12. Cycle = cycle + 1*

*13. Until cycle = maximum cycle number*

In the ABC algorithm, there are 3 flocks of bees: onlooker, employed, and scout bees. A colony consists of the onlooker bees plus the employed bees. If an employed bee abandons its food source, it becomes a scout bee. The number of solutions (population) to a problem is equal to the number of onlooker bees or employed bees. A possible solution to the optimization problem is presented by the position of a food source, and the quality (fitness) is measured with the amount of nectar of the associated food source. The number of food sources equals the number of employed bees. At the first step, the initial population $P(G = 0)$ of *SN* solutions (food source positions) is generated randomly by the ABC algorithm. *SN* denotes the size of the population. Each solution $x_i$ *(i = 1, 2, ..., SN)* is presented using a $D$-dimensional vector. Here, $D$ denotes the number of optimization parameters. After initialization of the population, the employed, onlooker, and scout bees repeatedly search for all of the food sources during a predetermined number of iterations denoted by the cycle *(cycle = 1, 2, ... , MCN)*. First, an employed bee starts a neighborhood search depending on the local information, and then it evaluates the nectar amount of the new food source, which corresponds to the fitness value. This fitness value is the only knowledge used to solve an optimization problem, as in other population-based optimization algorithms. If the position of the new food source is better than the previous one, it replaces it; otherwise, the position of the previous one is kept. After the search process is completely finished by all of the employed bees, they transport the nectar and position information of the food sources to the hive. The employed bees then share all of this information with the onlooker bees in the dance area of the hive. An onlooker bee evaluates the nectar information obtained from the employed bees and then it chooses a food source using a selection probability related to its nectar amount [11-13].

After an onlooker bee takes all of the information about the food sources from the employed bees, an onlooker bee selects a food source according to the selection probability value associated with that food source.

The selection probability $p_i$ is computed as follows [11-13]:

$$p_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i}, \tag{1}$$

where $fit_i$ is the fitness value of the $i$th solution, which is proportional to the nectar amount of the food source in the position $i$. $SN$ is the number of solutions (the size of the population), which is equal to the number of employed bees or the number of onlooker bees.

To create a new food position from the old one selected by the onlooker bee, depending on the probability given in Eq. (1), the ABC algorithm uses the following expression [11-13]:

$$v_{ij} = x_{ij} + \Phi_{ij}(x_{ij} - x_{kj}), \tag{2}$$

where $i$, $k$ {*1, 2,..., SN*}, and $j$ {*1, 2,..., D*}. The index $k$ is determined randomly, but it has to be different from $i$. The parameter $_{ij}$ is an independent and identically distributed (i.i.d.) uniform random number obtained by a random number generator with a different seed in each case and is chosen in the range [–1, 1]. A bee compares 2 food locations visually using this parameter. As seen from Eq. (2), as long as the change between the positions of $x_{ij}$ and $x_{kj}$ diminishes, the perturbation on the position $x_{ij}$ decreases. Therefore, the search comes close to an optimum solution in the search space.

The scout bees replace new food sources, which are produced randomly in their dynamic ranges, with the ones that the employed bees abandon. In the ABC algorithm, during a predetermined number of cycles, if a position cannot be improved further, then that food source is thought to be abandoned. Here, since only 1 source is abandoned in each cycle, 1 employed bee becomes a scout bee. The parameter's so-called "*limit*" is an important one for abandonment, which is the value of a predetermined number of cycles [25]. If the abandoned source is $x_i$ and $j$ {*1, 2,..., D*}, then a new food source is determined by the scout bee and it is replaced with $x_i$. The new food source position is given as [11-13]:

$$x_i^j = x_{\min}^j + rand(0,1)(x_{\max}^j - x_{\min}^j), \tag{3}$$

where $x_{\min}^j$ and $x_{\max}^j$ denote the lower and upper boundary values of the food source position, respectively.

After the ABC algorithm produces and then evaluates each candidate source position $v_{ij}$, its performance is compared with that of its old one. If the nectar amount of the new food source is equal to or better than the old one, it replaces the old one; otherwise, the old one is retained. This procedure is called a greedy selection. In the ABC algorithm, there are 3 control parameters: the number of food sources, which is equal to the number of employed or onlooker bees ($SN$); the value of the limit, and the maximum cycle number ($MCN$). In a robust search process, exploration and exploitation processes have to be performed together in such a manner that a trade-off between them must be considered. In the ABC algorithm, the scout bees control the exploration process, while onlooker and employed bees carry out the exploitation process in the search space [11-13]. Before applying the ABC algorithm in the identification of dynamic systems, the linear system identification process will be outlined in the next section.

# 3. Linear system identification

A linear time-invariant (LTI) discrete-time system is defined using the following linear equations in vector-matrix form:

$$\mathbf{X(k+1)} = \mathbf{AX(k)} + \mathbf{BU(k)}, \tag{4}$$

$$\mathbf{Y(k)} = \mathbf{CX(k)} + \mathbf{DU(k)}, \tag{5}$$

where the coefficients $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ are properly dimensioned matrices. The notation $k$ represents the time index. $\mathbf{U(k)}$ is the input vector, $\mathbf{Y(k)}$ is the output vector, and $\mathbf{X(k)}$ is the state vector:

$$\mathbf{U(k)} = \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_r(k) \end{bmatrix}, \mathbf{Y(k)} = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_p(k) \end{bmatrix}, \mathbf{X(k)} = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_n(k) \end{bmatrix}. \tag{6}$$

In system identification, the main task is to find a suitable model structure of a system with unknown parameters, given some prior knowledge about the system and input-output observations. Employing the ABC algorithm for identification, one can exploit its ability to learn the system behavior and its requirement of a reduced amount of knowledge, such as observed input-output data and order of the system. The proposed method approximates the system using the observed input-output data pairs and order of the system. In its estimating process, since the identification model is parallel to the system being identified, both get the same external input $\mathbf{U(k)}$. For the same input, the output of the model $\hat{\mathbf{Y}}\mathbf{(k)}$ is compared with the output of the system $\mathbf{Y(k)}$. Therefore, the error signal $\mathbf{e(k)}$ is produced by the difference between the output of the identification model and the output of the system in the following way [26,27]:

$$\mathbf{e(k)} = \mathbf{Y(k)} - \hat{\mathbf{Y}}\mathbf{(k)}. \tag{7}$$

## 3.1. Applying the ABC algorithm in systems identification

A discrete-time signal is a sequence:

$$u = \{u(0), u(1), \ldots, u(k), \ldots\} \quad and \quad y = \{y(0), y(1), \ldots, y(k), \ldots\}. \tag{8}$$

For a SISO system of order $n$, the causal LTI system's difference equation can be written as follows:

$$\begin{aligned} y[k] = &-\beta_1 y[k-1] - \beta_2 y[k-2] - \cdots - \beta_m y[k-m] \\ &+\alpha_0 u[k] + \alpha_1 u[k-1] + \alpha_2 u[k-2] + \cdots + \alpha_n u[k-n], \quad n \le m, \end{aligned} \tag{9}$$

where $\beta_1$, $\beta_2$, ... $\beta_m$, $\alpha_0$, $\alpha_1$,... $\alpha_n$ are the system parameters. Applying the Z-transform to Eq. (9), we can obtain the following equation:

$$\left(1 + \beta_1 z^{-1} + \cdots + \beta_m z^{-m}\right)_Y(z) = (\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n})_U(z), \quad n \le m. \tag{10}$$

Hence, the transfer function, $G(z)$, can be defined as:

$$G(z) = \frac{Y(z)}{U(z)}, \tag{11}$$

or:

$$G(z) = \left(\alpha_0 + \alpha_1 z^{-1} + \cdots + \alpha_n z^{-n}\right) / \left(1 + \beta_1 z^{-1} + \cdots + \beta_m z^{-m}\right), \quad n \leq m. \tag{12}$$

The concept of a transfer function can be extended to a linear system having $p$inputs and $r$ outputs, in which case a transfer function matrix, $G(z)$, is defined. It has the dimensions of $r \times p$. In the identification problem, the parameters in each element of the matrix $G(z)$need to be found. The signal $y(k)$ can be constructed by iterative computation, given the input signal and initial conditions. After a discrete system transfer function has been formed, the linear system equation is written as input and output form. The initial conditions are set to 0. The measured input-output data are used for parameter estimation. At the beginning of parameter estimation, the input and output data are known and the real system parameters are assumed as unknown. Using initial conditions and obtained real system data, system parameters are estimated with the ABC algorithm. The error between real system output and estimated system output is defined as an error function. The estimation of the system parameters is achieved as a result of minimizing the error function by the ABC algorithm. The sum-squared error function is used here as an error criterion [28-30]:

$$J = \frac{1}{2} \sum_k e(k)^2, \tag{13}$$

where $J$ is the measure of the error. For one training epoch, the root-mean-square error (*RMSE*) is represented as follows:

$$RMSE = \sqrt{\frac{\sum_{k=1}^{r} e(k)^2}{r}}, \tag{14}$$

where $r$ represents the number of data.

The ABC algorithm search is based on the well-known *Jury stability criterion* within the stability boundary for parameter estimation [31]. The roots of a transfer function in the z-plane should be located inside the unit circle $|z| \leq 1$ for stability. A zero-order hold element for discretization is used.

# 4. Simulation results and discussions

In this study, 2 main examples are utilized in order to illustrate the efficiency of the proposed algorithm: the ABC algorithm in systems identification for comparing different linear SISO plants with different orders, and the ABC algorithm in systems identification for a DC motor. Here, there are 4 control parameters in the ABC algorithm: the 1st parameter is the number of food sources, which is equal to the number of employed or onlooker bees (S$N$); the 2nd is the number of parameters of the problem to be estimated ($D$); the 3rd is the value of the limit parameter (*limit*); and the 4th is the *MCN*. The value of the limit is generally chosen as $SN \times D$ [12,32]. In the ABC algorithm, the values of the control parameters were chosen as $SN = 10$, $MCN =$ 500 for the 1st order and DC motor plant and as $SN = 25$, $MCN = 2000$ for the 3rd, 5th, and 7th order plants.

## 4.1. The ABC algorithm in systems identification for different linear SISO plants with different orders

The system identification algorithm looks for the parameters to be estimated to satisfy the desired real system parameters by the cost function computed using the *RMSE* at the each iteration. For estimation of the parameters, 4 different linear SISO processes with different order are used, as follows:

$$G_1(s) = \frac{3}{30s+1}, \tag{15}$$

$$G_2(s) = \frac{750}{s^3 + 36s^3 + 205s + 750}, \tag{16}$$

$$G_3(s) = \frac{-6.35\times10^{-6}s^4 + 4.933\times10^{-5}s^3 + 2812s^2 + 1.172\times10^4 s + 1.953\times10^4}{s^5 + 32.5s^4 + 475s^3 + 3625s^2 + 1.422\times10^4 s + 1.914\times10^4}, \tag{17}$$

$$G_4(s) = \frac{Y_4(s)}{U_4(s)}, \tag{18}$$

where:

$$\begin{aligned}Y_4(s) &= 1.435\times10^{-5}s^6 + 6.232\times10^{-6}s^5 + 8.882\times10^{-5}s^4\\ &\quad -1.699\times10^{-5}s^3 + 1.671\times10^{-4}s^2 + 17.98s - 17.98,\\ U_4(s) &= s^7 + 5.234s^6 + 19.7s^5 + 45.92s^4 + 76.52s^3\\ &\quad +84.09s^2 + 57.11s + 17.98.\end{aligned}$$

The transfer functions of the 1st (sampling time = 1.0 s), 3rd (sampling time = 0.1 s), 5th (sampling time = 0.1 s), and 7th (sampling time = 1.0 s) orders are given in the z-domain as:

$$G_1(z) = \frac{0.098352}{z - 0.967216}, \tag{19}$$

$$G_2(z) = \frac{0.057176z^2 + 0.107891z + 0.009899}{z^3 - 1.414464z^2 + 0.616755z - 0.027324}, \tag{20}$$

$$G_3(z) = \frac{0.225545z^4 + 0.071233z^3 - 0.490390z^2 + 0.197875z + 0.035971}{z^5 - 2.380647z^4 + 2.335256z^3 - 1.204551z^2 + 0.328146z - 0.038774}, \tag{21}$$

$$G_4(z) = \frac{Y_4(z)}{U_4(z)}, \tag{22}$$

where:

$$\begin{aligned}Y_4(z) &= 0.008550z^6 + 0.102751z^5 - 0.123576z^4 - 0.502080z^3\\ &\quad -0.248905z^2 - 0.026393z - 0.000277,\\ U_4(z) &= z^7 - 0.363682z^6 + 0.257812z^5 - 0.166412z^4 + 0.096288z^3\\ &\quad -0.047987z^2 + 0.019244z - 0.005333.\end{aligned}$$

The difference equations of the systems are as follows:

$$y_1[k] = 0.967216y_1[k-1] + 0.098352u_1[k-1], \tag{23}$$

$$\begin{aligned}y_2[k] &= 1.414464y_2[k-1] - 0.616755y_2[k-2] + 0.027324y_2[k-3]\\ &\quad +0.057176u_2[k-1] + 0.107891u_2[k-2] + 0.009899u_2[k-3],\end{aligned} \tag{24}$$

$$\begin{aligned}y_3[k] &= 2.380647y_3[k-1] - 2.335256y_3[k-2] + 1.204551y_3[k-3]\\ &\quad -0.328146y_3[k-4] + 0.038774y_3[k-5] + 0.225545u_3[k-1]\\ &\quad +0.071233u_3[k-2] - 0.490390u_3[k-3] + 0.197875u_3[k-4]\\ &\quad +0.035971u_3[k-5],\end{aligned} \tag{25}$$

$$\begin{aligned} y_4[k] = {} & 0.363682 y_4[k-1] - 0.257812 y_4[k-2] + 0.166412 y_4[k-3] \\ & -0.096288 y_4[k-4] + 0.047987 y_4[k-5] - 0.019244 y_4[k-6] \\ & +0.005333 y_4[k-7] + 0.008550 u_4[k-1] + 0.102751 u_4[k-2] \\ & -0.123576 u_4[k-3] - 0.502080 u_4[k-4] - 0.248905 u_4[k-5] \\ & -0.026393 u_4[k-6] - 0.000277 u_4[k-7]. \end{aligned} \tag{26}$$

A training set consisting of 400 data for the 1st order and 2000 data for the 3rd, 5th, and 7th order plants is obtained using a random input, whose amplitude is uniformly distributed in the interval [–2.0, 2.0] for 0 initial conditions. To avoid a similar particular solution, all of the parameters to be estimated are initialized randomly over the range [–10.0, 10.0]. The proposed algorithm is run 10 times for each plant. For all of the plants, each parameter is estimated and the simulation results are presented in Table 1. The results in Table 1 were found using 10 employed bees for the 1st order and 25 employed bees for the 3rd, 5th, and 7th order plants, respectively.

**Table 1.** Simulation results of the estimated different order linear SISO processes.

| Plant | Parameters | Real system | Estimated system (by the ABC algorithm) |
|---|---|---|---|
| $G_1$(s) | $\beta_1$ | 0.967216 | 0.967216 |
| | $\alpha_0$ | 0.098352 | 0.098352 |
| $G_2$(s) | $\beta_1$ | 1.414464 | 1.114346 |
| | $\beta_2$ | –0.616755 | –0.188177 |
| | $\beta_3$ | 0.027324 | –0.151084 |
| | $\alpha_0$ | 0.057176 | 0.057323 |
| | $\alpha_1$ | 0.107891 | 0.124714 |
| | $\alpha_2$ | 0.009899 | 0.042447 |
| $G_3$(s) | $\beta_1$ | 2.380647 | 1.305520 |
| | $\beta_2$ | –2.335256 | –0.984349 |
| | $\beta_3$ | 1.204551 | 0.283957 |
| | $\beta_4$ | –0.328146 | –0.043367 |
| | $\beta_5$ | 0.038774 | –0.034219 |
| | $\alpha_0$ | 0.225545 | 0.225264 |
| | $\alpha_1$ | 0.071233 | 0.313395 |
| | $\alpha_2$ | –0.490390 | –0.140650 |
| | $\alpha_3$ | 0.197875 | 0.047479 |
| | $\alpha_4$ | 0.035971 | 0.033275 |
| $G_4$(s) | $\beta_1$ | 0.363682 | 0.260398 |
| | $\beta_2$ | –0.257812 | –0.497459 |
| | $\beta_3$ | 0.166412 | 0.169752 |
| | $\beta_4$ | –0.096288 | –0.060764 |
| | $\beta_5$ | 0.047987 | 0.012406 |
| | $\beta_6$ | –0.019244 | 0.004800 |
| | $\beta_7$ | 0.005333 | –0.005562 |
| | $\alpha_0$ | 0.008550 | 0.008500 |
| | $\alpha_1$ | 0.102751 | 0.103899 |
| | $\alpha_2$ | –0.123576 | –0.110693 |
| | $\alpha_3$ | –0.502080 | –0.485597 |
| | $\alpha_4$ | –0.248905 | –0.327795 |
| | $\alpha_5$ | –0.026393 | –0.207619 |
| | $\alpha_6$ | –0.000277 | –0.094800 |

The performance of the ABC algorithm was tested with the unit step input as well as the following input sequence consisting of mixtures of sinusoids and constant signals:

$$
\begin{aligned}
u(k) &= \sin(\pi k/25), \quad k < 250 \\
&= 1.0, \quad 250 \le k < 500 \\
&= -1.0, \quad 500 \le k < 750 \\
&= 0.3\sin(\pi k/25) + 0.1\sin(\pi k/32) + 0.6\sin(\pi k/10), \quad 750 \le k < 1000.
\end{aligned}
\tag{27}
$$

Responses of $G_1(s)$, $G_2(s)$, $G_3(s)$, and $G_4(s)$ are shown in Figures 1-4. The step responses of $G_1(s)$, $G_2(s)$, $G_3(s)$, and $G_4(s)$, plotted with the best values of the parameters estimated by the ABC algorithm in 10 runs, are shown in Figures 1a, 2a, 3a, and 4a, respectively. The sinusoidal input responses of $G_1(s)$, $G_2(s)$, $G_3(s)$, and $G_4(s)$, plotted with the best values of the parameters estimated by the ABC algorithm in 10 runs, are shown in Figures 1b, 2b, 3b, and 4b, respectively.

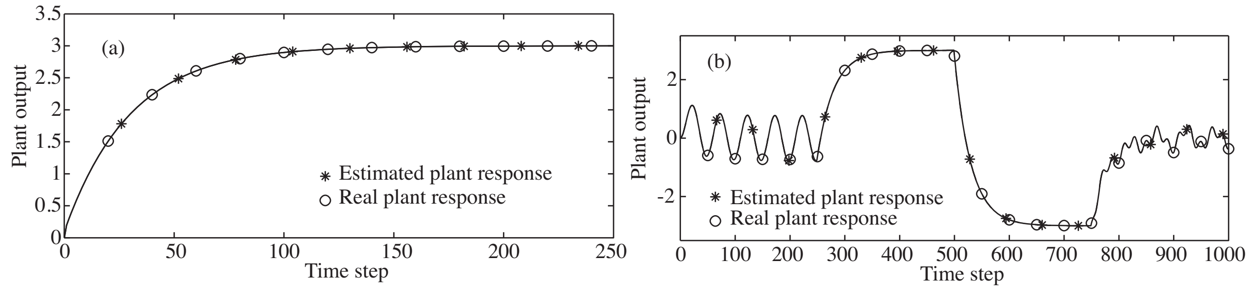


**Figure 1.** Responses of plant $G_1(s)$: a) step responses and b) sinusoidal responses.

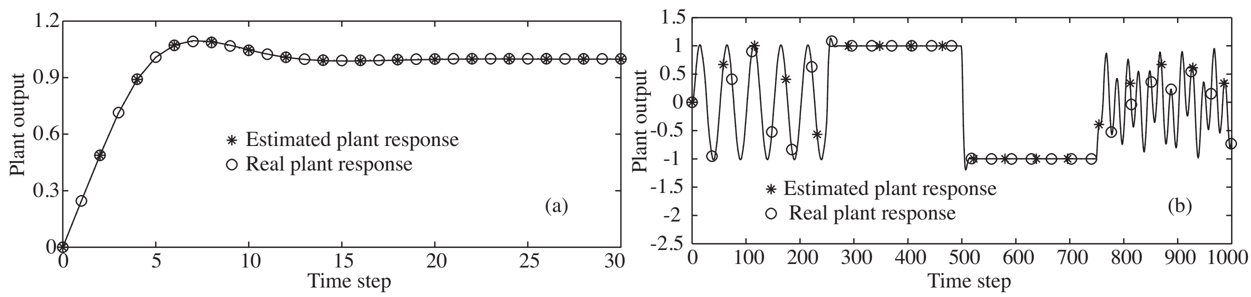


**Figure 2.** Responses of plant $G_2(s)$: a) step responses and b) sinusoidal responses.

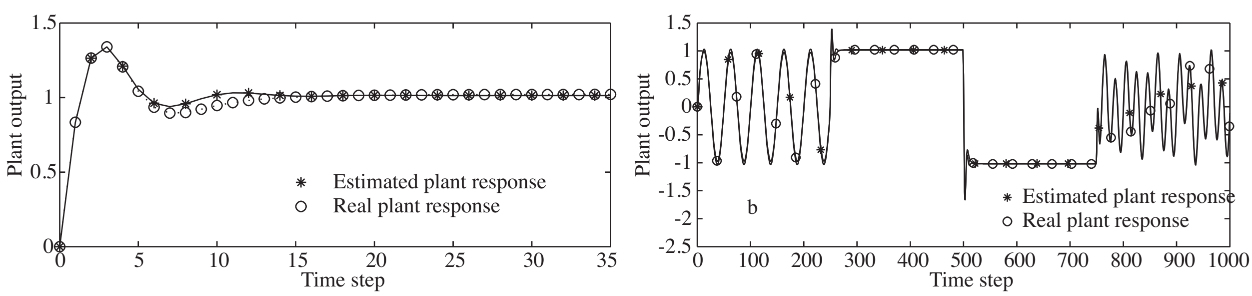


**Figure 3.** Responses of plant $G_3(s)$: a) step responses and b) sinusoidal responses.

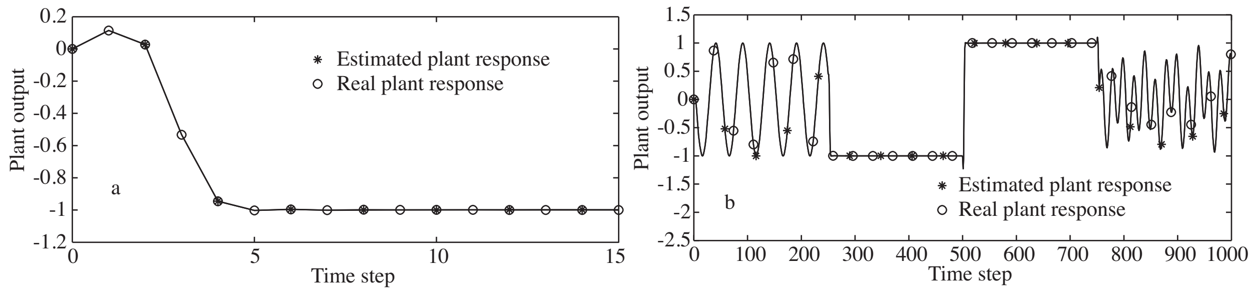


**Figure 4.** Responses of plant $G_4(s)$: a) step responses and b) sinusoidal responses.

The *RMSE* values for the unit step input and sinusoidal input are presented in Table 2. The results show that the value of the *RMSE* is quite small. The ABC algorithm showed satisfactory performance.

**Table 2.** *RMSE* for different order linear plants.

| Plant | *RMSE* for step response input | *RMSE* for sinusoidal input |
|---|---|---|
| G1(s) | 0.0 | 0.0 |
| G2(s) | 0.0019 | 0.0027 |
| G3(s) | 0.0104 | 0.0345 |
| G4(s) | 0.0021 | 0.0017 |

## 4.2. The ABC algorithm in systems identification for a DC motor

An identification method is presented here for a DC motor. A simplified mathematical model of the DC motor was used in order to build the DC motor's transfer function. There are differential equations of the electrical part and mechanical part of the DC motor model, and an interconnection also exists between them.

Using a simplified equivalent electromechanical diagram of the DC motor, illustrated in Figure 5, the differential mathematical model is written as [21]:

$$U_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + e_v(t), \tag{28}$$

$$e(t) = K_e \Omega(t), \tag{29}$$

$$C_m(t) = K_m i_a(t), \tag{30}$$

$$C_m(t) = J\frac{d\Omega(t)}{dt} + B\Omega(t), \tag{31}$$

where $C_m$ denotes the motor torque (Nm), $I_a$ denotes the rotor circuit current (A), $K_e$ denotes the electrical constant, $K_m$ denotes the mechanical constant, $L_a$ denotes the rotor circuit inductance (H), $R_a$ denotes the rotor circuit resistance (ohm), $U_a$ denotes the input voltage (V), $B$ denotes the damping ratio (Nms), $e_v$ denotes the electromotive voltage (V), $J$ denotes the rotor moment of the inertia (kg m$^2$), and $\Omega$ denotes the rotor speed (rad/s).

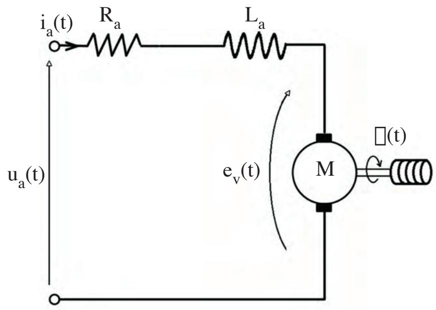


**Figure 5.** A DC motor equivalent circuit.

The transfer function of the speed model is obtained to allow the control of speed by the voltage input from the characteristic equations of the DC motor model. It is given by:

$$\frac{\Omega(s)}{U_a(s)} = \frac{K_m}{L_a J s^2 + (R_a J + L_a B)s + (R_a B + K_e K_m)}. \tag{32}$$

This transfer function makes it possible to simulate motor behavior to various inputs. The specifications of the motor used for simulation are given in Table 3. The DC motor transfer function in the $s$-domain is given by:

$$G_{DC\_MOTOR}(s) = \frac{0.1433}{5.2 \times 10^{-7}s^2 + 2.172 \times 10^{-4}s + 0.0227}. \tag{33}$$

The discrete time transfer function of the DC motor model of the 2nd order (sampling time = 0.001 s) is given by:

$$G_{DC\_MOTOR}(z) = \frac{0.12z + 0.1044}{z^2 - 1.623z + 0.6586}. \tag{34}$$

Its difference equation is given by:

$$y[k] = 1.623y[k-1] - 0.6586y[k-2] + 0.12u[k-1] + 0.1044u[k-2]. \tag{35}$$

A training set consisting of 400 data is obtained using a random input, whose amplitude is uniformly distributed in the interval [–2.0, 2.0] for 0 initial conditions. Simulations are carried out using employed or onlooker bees, $SN = 10$, with the maximum cycle number of $MCN = 500$ for the DC motor. To avoid a similar particular solution, all of the parameters are initialized randomly over the range [–10.0, 10.0]. The proposed algorithm is run 10 times for DC motor parameter estimation.

**Table 3.** Parameters of the motor [23].

| Parameters | Value |
|---|---|
| Armature circuit resistance ($R_a$) | 21.2 ohm |
| Armature circuit inductance ($L_a$) | 0.052 H |
| Back EMF constant ($K_m$) | 0.1433 kg m/A |
| Coefficient of friction (B) | $1 \times 10^{-4}$ Nms |
| Moment of inertia (J) | $1 \times 10^{-5}$ kg m$^2$ |
| Torque constant ($K_e$) | 0.1433 V/(rad/s) |

The plant given in Eq. (33) is tested with a unit step input as well as the input sequence consisting of mixtures of sinusoids and constant signals given in Eq. (27), to show the effectiveness and performance of the proposed method. A comparative graph of the actual and simulated dynamic responses with the identified parameters is illustrated in Figure 6, which shows a considerable agreement between the actual and identified plant responses using the estimated parameters. The real and estimated parameters of the DC motor are given in Table 4. The *RMSE* values for the step and sinusoidal inputs are given in Table 5.

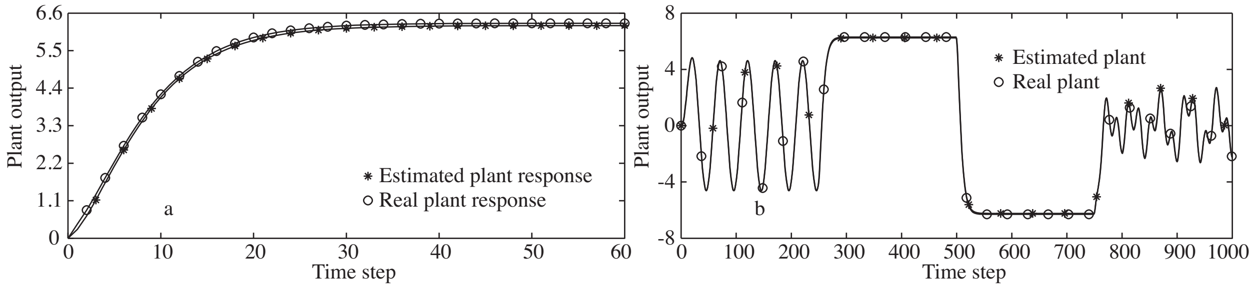


**Figure 6.** Response of the DC motor: a) step response and b) sinusoidal response.

**Table 4.** The real and estimated parameters of the DC motor.

| Plant | Parameters | Real system | Estimated system (by the ABC algorithm) |
|---|---|---|---|
| $G_{DC\_MOTOR}(s)$ | $\beta_1$ | 1.6230 | 1.5940 |
| | $\beta_2$ | –0.6586 | –0.6332 |
| | $\alpha_0$ | 0.1200 | 0.0101 |
| | $\alpha_1$ | 0.1044 | 0.2343 |

**Table 5.** *RMSE* for the DC motor.

| Plant | *RMSE* for unit step input | *RMSE* for sinusoidal input |
|---|---|---|
| $G_{DC\_MOTOR}$(s) | 0.0722 | 0.0844 |

# 5. Conclusion

In this paper, a novel parameter estimation method for linear system identification based on the ABC algorithm was developed. The ABC algorithm has been shown to be versatile when applied to parameter estimation, without requiring a detailed mathematical representation of the identification problem. The unit step and sinusoidal response performance of the ABC algorithm was tested with several orders of linear plants for system identification. It is well known that the ABC algorithm has good results in solving numerical optimization problems. Thus, the effectiveness of system identification using the ABC algorithm was researched and a satisfactory performance was obtained. The proposed method was also applied to estimate the parameters of a DC motor commonly used in industry. The proposed method was flexible and applicable in a wide range of optimization and identification problems. The simulation results show that the proposed method achieved a minimum tracking error and estimated the parameter values with high accuracy. Due to the fact that some stability criteria were taken into account in the system parameters estimation, the proposed method can thus be regarded as a general parameter estimation method that can be applied to a wide range of linear plant identification problems.

# References

[1] L. Lennart, System Identification: Theory for the User, New Jersey, Prentice Hall, 1997.

[2] A. Wittenmark, Computer Control Systems, New York, Addison Wesley, 1989.

[3] K. Ogata, Modern Control Engineering, New Jersey, Prentice Hall, 2002.

[4] P.J. Angeline, D.B. Fogel, "An evolutionary program for the identification of dynamical systems", Application and Science of Artificial Neural Networks III, Vol. 3077, pp. 409-417, 1997.

[5] R. Lippmann, "Review of neural networks for speech recognition", Neural Computation, Vol. 1, pp. 1-38, 1998.

[6] K. Rodriguez, P. Fleming, "Genetic programming for dynamic chaotic systems modeling", Proceedings of the 1999 Conference on Evolutionary Computation, pp. 22-28, 1999.

[7] P. Sastry, G. Santharam, K. Unnikrishnan, "Memory neuron networks for identification and control of dynamical systems", IEEE Transactions on Neural Networks, Vol. 5, pp. 306-319, 1994.

[8] D. Karaboga, B. Basturk, "On the performance of artificial bee colony (ABC) algorithm", Applied Soft Computing, Vol. 8, pp. 687-697, 2008.

[9] D. Karaboga, B. Basturk, "Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems", Proceedings of the 12th International Fuzzy Systems Association World Congress on Foundations of Fuzzy Logic and Soft Computing, pp. 789-798, 2007.

[10] D. Karaboga, B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", Journal of Global Optimization, Vol. 39, pp. 459-171, 2007.

[11] B. Akay, D. Karaboga, "Solving integer programming problems by using artificial bee colony algorithm", Proceedings of the 11th International Conference of the Italian Association for Artificial Intelligence Reggio Emilia on Emergent Perspectives in Artificial Intelligence, pp. 355-364, 2009.

[12] B. Akay, D. Karaboga, "Parameter tuning for the artificial bee colony algorithm", Proceedings of the 1st International Conference on Computational Collective Intelligence, Semantic Web, Social Networks and Multiagent Systems, pp. 608-619, 2009.

[13] D. Karaboga, B. Akay, "A comparative study of artificial bee colony algorithm", Applied Mathematics and Computation, Vol. 214, pp. 108-132, 2009.

[14] N. Karaboga, "A new design method based on artificial bee colony algorithm for digital IIR filters", Journal of the Franklin Institute, Vol. 346, pp. 328-348, 2009.

[15] R.V. Rao, "Multi-objective optimization of multi-pass milling process parameters using artificial bee colony algorithm", Artificial Intelligence in Manufacturing Research, New York, Nova Science Publishers, 2011.

[16] S.N. Omkar, J. Senthilnath, , R. Khandelwal, G.N. Naik, S. Gopalakrishnan, "Artificial bee colony (ABC) for multi-objective design optimization of composite structures", Applied Soft Computing, Vol. 11, pp. 489-499, 2011.

[17] P.J. Pawar, R.V. Rao, R. Shankar, "Multi-objective optimization of electro-chemical machining process parameters using artificial bee colony (ABC) algorithm", Proceedings of Advances in Mechanical Engineering, 2008.

[18] N. Karaboğa, M.B. Çetinkaya, "A novel and efficient algorithm for adaptive filtering: artificial bee colony algorithm", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 19, pp. 175-190, 2011.

[19] A. Akdağlı, M.B. Biçer, S. Ermiş, "A novel expression for resonant length obtained by using artificial bee colony algorithm in calculating resonant frequency of C-shaped compact microstrip antennas", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 19, pp. 597-606, 2011.

[20] N. Taşpınar, D. Karaboğa, M. Yıldırım, B. Akay, "PAPR reduction using artificial bee colony algorithm in OFDM systems", Turkish Journal of Electrical Engineering & Computer Sciences, Vol. 19, pp. 47-58, 2011.

[21] C.M. Ong, Dynamic Simulation of Electric Machinery, New Jersey, Prentice Hall Inc, 1998.

[22] V. Tipsuwanporn, W. Piyarat, C. Tarasantisuk, "Identification and control of brushless DC motors using on-line trained artificial neural networks", Proceedings of the Power Conversion Conference, pp. 1290-1294, 2002.

[23] N. Navidi, M. Bavafa, S. Hesami, "A new approach for designing of PID controller for a linear brushless DC motor with using ant colony search algorithm", Proceedings of the Asia-Pacific Power and Energy Engineering Conference, 2009.

[24] V. Tereshko, "Reaction-diffusion model of a honeybee colony's foraging behaviour", Proceedings of the 6th International Conference on Parallel Problem Solving from Nature, pp. 807-816, 2000.

[25] B. Akay, D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization", Information Sciences, Vol. 192, pp. 120-142, 2012.

[26] J. Schoukens, R. Pintelon, Y. Rolain, "Identification of nonlinear and linear systems, similarities, differences, challenges", Proceedings of the 14th IFAC Symposium on System Identification, pp. 122-124, 2006.

[27] E.E. Vladu, T.L. Dragomir, "On data management in elementary system identification approaches using genetic algorithms", Proceedings of SOFA, 2005.

[28] P.J. Pawar, R.V. Rao, J.P. Davim, "Optimization of process parameters of abrasive flow machining process using artificial bee colony algorithm", Proceedings of Advances in Mechanical Engineering, 2008.

[29] P.J. Pawar, R.V. Rao, J.P. Davim, "Optimization of process parameters of milling process using particle swarm optimization and artificial bee colony algorithm", Proceedings of Advances in Mechanical Engineering, 2008.

[30] J. Ronkkonen, S. Kukkonen, K.V. Price, "Real-parameter optimization with differential evolution", Proceedings of the 2005 IEEE Congress on Evolutionary Computation, Vol. 1, pp. 506-513, 2005.

[31] E.I. Jury, J. Blanchard, "A stability test for linear discrete-time systems in table form", Proceedings of the IEEE, Vol. 49, pp. 1947-1948, 1961.

[32] N. Stanarevic, M. Tuba, N. Bacanin, "Modified artificial bee colony algorithm for constrained problems optimization", International Journal of Mathematical Models and Methods in Applied Sciences, Vol. 5, pp. 644-651, 2011.