

# Design an Optimal PID Controller using Artificial Bee Colony and Genetic Algorithm for Autonomous Mobile Robot

Ramzy S. Ali, PhD  
Engineering College  
University of Basrah, Iraq

Ammar A. Aldair, PhD  
Engineering College  
University of Basrah, Iraq

Ali K. Almousawi  
Engineering College  
University of Kufa, Iraq

## ABSTRACT

Target tracking is a serious function for an autonomous mobile robot navigating in unknown environments such as disaster areas, projects sites, and any dangerous place which the human cannot reach. This paper deals with modified the parameters of PID controller using Artificial Bee Colony (ABC) and Genetic Algorithm (GA) for path tracking of autonomous mobile robot. Two PID control are designed, one for speed control and the other for azimuth control. The MATLAB program is used to simulate the autonomous mobile robot model with optimal PID controllers, ABC algorithm and GA. To test the effectiveness of the proposed controllers, two path trajectories have been chosen: circular path and sine wave path. The results have clearly shown the effectiveness and good performances of the PID controllers which are tuned using ABC algorithm than using GA.

## General Terms

Control system design for autonomous robot system.

## Keywords

Autonomous mobile robot, artificial bee colony, genetic algorithm, PID controllers.

## 1. INTRODUCTION

Mobile robot manipulators are mobile robot bases with at least one mounted robot arm which function in an integrated manner. The purpose of the mobile manipulator is to reach concrete locations in its environment and to pick up objects. There are two applications of using a mobile robot manipulator (MRM). The first one: using the MRM in unstructured environments, especially in the scenario that is unsuitable for human beings. The second application: using the MRM to transport objectives and tools in an already known industrial environment. Autonomous mobile robots are able to carry out many functions in dangerous sites where humans cannot reach, such as sites where harmful gases or high temperature are present, a hard environment for humans. Home assistant robots are expected to support daily activities at home. In all these examples robots have to move to their destination in order to perform their functions. For this purpose they need to be able to recognize the changes of environment using various sensors and cameras, and be equipped with a motion planning method in order to avoid collision with obstacles or other robots. In this work, the target tracking control issue are studied to improve the performances of the mobile robot.

The PID controllers are widely being used in the industries for process control applications. Even for complex industrial control system, the industries use the PID control module to build the main controller. The merit of using PID controllers

lie in its simplicity of design and good performance including low percentage overshoot and small settling time for normal industrial processes [1]. Several formulas have been suggested to select the suitable parameters of PID controller. Ziegler-Nichols tuning method was introduced in 1942 to tune the PID control parameters. This tuning method can be applied when the plant model is given by a first-order plus dead time. Many variants of the traditional Ziegler-Nichols PID tuning methods have been proposed such as the Chien-Hrones-Reswick formula, Cohen-Coon formula, refine Ziegler-Nichols tuning formula, Wang-Juang-Chan formula and Zhuang-Atherton optimum PID controller [2]. The main problem of use the PID controller is the correct choice of the suitable control parameters to improve the transient response of the controlled system. Using other words, the random setting for the parameters of PID controller, the controller may not provide the required control performance, when there are variations in the plant parameters and operating conditions.

Many authors proposed different algorithm to tune the parameters of PID controller. The genetic algorithm was used to modify the PID controller [3-5]. The particle swarm optimization method was used to tune the parameters of PID controller [6-8]. Some authors used Artificial Bees Colony (ABC) optimization algorithm for regulation parameters of PID controller [9].

In this paper, two algorithms (ABC and GA) are used to tune the parameters of PID controller to force the Autonomous mobile robot to follow the specific trajectory. Two PID control are designed, one for speed control and the other for azimuth control. The control signal is applied to mobile robot system for obtaining smooth path tracking of two wheeled differential which drive the mobile robot. The MATLAB program is used to simulate the autonomous mobile robot model with optimal PID controllers; ABC algorithm and GA.

## 2. AUTONOMOUS MOBILE ROBOT MODEL

Design, development, modification and control of a mechatronic system require an understanding and a suitable representation of a system; specifically, a "model" of the system is required. Any model is an idealization of the actual system. A mechatronic or robotic system may consist of several different types of components, and it is termed as a mixed system. One should use analogous procedures for modeling of such components. In this manner, the component models can be conveniently integrated to obtain the overall model.

Modeling of a differential drive mobile robot platform consists of kinematic and dynamic modeling. Each part of this system's modeling will be explained separately. After getting an accurate model, a complete system is simulated using an MATLAB program package.

## 2.1 Kinematic Model of Mobile Robot

Kinematics refers to the evolution of the position, and velocity of a mechanical system, without referring to its mass and inertia. The kinematic scheme of the mobile robot consists of a platform driven by two driving wheels mounted on same axis with independent actuators and one free wheel that is called a caster. The movement of mobile robot is done by changing the relative angular velocities of driving wheels. The assumptions are that the whole body of robot is rigid and motion occurs without sliding. Its wheel rotation is limited to one axis. Therefore, the navigation is controlled by changing speed on either side of the robot. The kinematics scheme of the differential drive mobile robot is as shown in Figure 1 where  $\{O, X, Y\}$  are the global coordinate and  $\{P, x_l, y_l\}$  are the local coordinate which are fixed to the robot with its centre P between the two wheels,  $r$  is the radius of each wheel and  $2L$  is the distance between two driving wheels,  $\omega$  is the angular velocity of the mobile robot,  $v_l$  and  $v_r$  are the velocity of the left and right driving wheel, the angle  $\theta$  indicates the orientation of the robot, C represented the centre of the mobile robot,  $d$  is the distance from center of mobile robot to origin P.

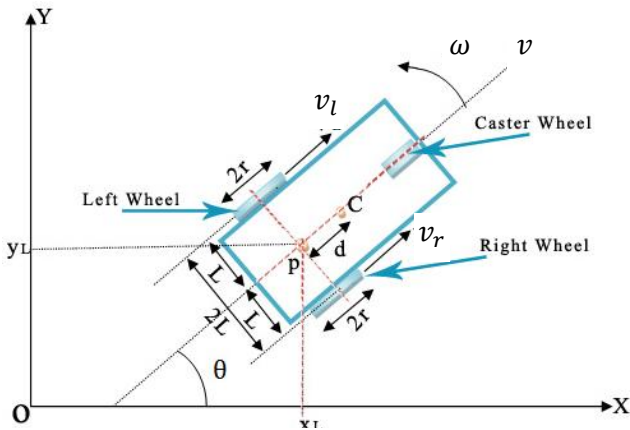


Fig. 1: Mobile Robot Model

According to the motion principle of rigid body kinematics, the motion of a two wheel differential drive mobile robot can be described using Eq. (1) and Eq. (2), where  $\omega_l$  and  $\omega_r$  are angular velocities of the left and right driving wheels respectively [10].

$$\begin{cases} v_l = r \omega_l \\ v_r = r \omega_r \end{cases} \quad (1)$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 \\ -1/2L & 1/2L \end{bmatrix} \begin{bmatrix} v_r \\ v_l \end{bmatrix} \quad (2)$$

The nonholonomic constraint equation of the robot is as following:

$$\dot{y} \cos \theta - \dot{x} \sin \theta = 0 \quad (3)$$

Substitution equation (1) in (2) yield :

$$\begin{cases} v = \frac{r}{2} (\omega_r + \omega_l) \\ \dot{\theta} = \frac{r}{2L} (\omega_r - \omega_l) \end{cases} \quad (4)$$

Additionally, we can define the dynamic function of the robot as follows.

$$\begin{cases} \dot{x} = v \cos \theta \\ \dot{y} = v \sin \theta \\ \dot{\theta} = \omega \end{cases} \quad (5)$$

Re-arranged equation (5) as a matrix yield:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \\ -\frac{r}{2L} & \frac{r}{2L} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \quad (6)$$

where  $\dot{x}$  and  $\dot{y}$  : denote the velocity of the robot in the direction of X-axis and Y-axis, respectively.

## 2.2 Dynamic Model of mobile robot

The design procedure will depend on the Lagrange-Euler differential equation of dynamics which is shown below [10]:

$$\mathbf{M}(q)\dot{w} + \mathbf{C}(q, \dot{q})w + \mathbf{F}(\dot{q}) + \mathbf{G}(q) = \mathbf{K}(q)\tau \quad (7)$$

$\mathbf{F}$  is a  $2 \times 1$  vector which holds the friction force and  $\mathbf{G}$  is a  $2 \times 1$  vector which holds the gravitation force acting upon the robot, but because of the assumptions that the robot does not slip and it moves on flat ground, both forces are set to zero.

With the assumptions that the robot does not slip, not disturbance, and it moves on flat ground; the third, fourth and fifth bound are set to zero. Then the equation (3.9) becomes:

$$\mathbf{M}(q)\dot{w} + \mathbf{C}(q, \dot{q})w = \mathbf{K}(q)\tau \quad (8)$$

where  $\mathbf{W} = [\omega_r \ \omega_l]^T$ , input torques on the wheels  $\tau = [\tau_r \ \tau_l]^T$ ,  $\mathbf{M}$  is a  $2 \times 2$  symmetric and positive definite inertia matrix, which describe as

$$\mathbf{M} = \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{11} \end{bmatrix}$$

where

$$m_{11} = I_W + \frac{R^2}{4L^2} (mL^2 + I)$$

$$m_{12} = \frac{R^2}{4L^2} (mL^2 - I)$$

$\mathbf{C}$  is a  $2 \times 2$  matrix which holds the centripetal forces which affect upon the wheels, where  $c$  equals the viscous friction of the wheels against the ground.  $\mathbf{K}$  is a  $2 \times 2$  input transformation matrix for the torque vector, where  $k$  is the gain constant for each torque input ( $\mathbf{K} = k\mathbf{I}$ ).

To convert Eq. (8) to the canonical state-space dynamics model with respect to the state vector  $w$ , the equation becomes:

$$[\dot{v} \ \dot{\omega}]^T M^{-1}(-c[v \ \omega]^T + k[\tau_r \ \tau_l]^T) \quad (9)$$

### 3. STRUCTURE OF PID CONTROLLER

The PID term refers to the first letter of the names of the individual terms that make up the standard three term controller. These are: P for the proportional term; I for the integral term and D for the derivative term in the controller. Appropriate tuning of these parameters will improve the performance of the plant, reduce the overshoot, eliminate steady state error and increase stability of the system.

The main problem of that simple controller is the correct choice of the PID gains. By using fixed gains, the controller may not provide the required control performance, when there are variations in the plant parameters and operating conditions. Therefore, the tuning process must be performed to insure that the controller can deal with the variations in the plant.

Because most PID controllers are adjusted on-site, many different types of tuning rules have been proposed in the literature. Using these tuning rules, delicate and fine tuning of PID controllers can be made on-site. Also, automatic tuning methods have been developed and some of the PID controllers may possess on-line automatic tuning capabilities.

In this work, two PID controllers are proposed for motion control of autonomous mobile robot. The first one of PID controller is used to control the velocity and another for controlling azimuth of the mobile robot. The velocity and azimuth of the robot are controlled by manipulating the torques for the left and the right-wheels which illustrate on the same Figure 2.

The  $x$  desired ( $x_d$ ),  $y$  desired ( $y_d$ ) and  $\theta$  desired ( $\theta_d$ ) are considered as the input signal to proposed system. Error of

The distance error  $e_d$  and the azimuth error  $e_\theta$  are considered as the inputs for PID controller, and the driving torques required for controlling the two wheels ( $u_r$  and  $u_l$ ) are considered as outputs.

In this work two intelligent algorithms are studied to find the optimal parameters of the PID controllers: first algorithm is GA and the second algorithm is ABC. Each algorithm is described in detail in the following subsection.

### 4. THE GENETIC ALGORITHM

The Evolutionary Algorithm (EA) is an optimization algorithm is used to search for optimal solutions to a problem. This algorithm operates on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution. Evolutionary algorithms provide a universal optimization technique that mimics the type of genetic adaptation that occurs in natural evolution. Unlike specialized methods designed for particular types of optimization tasks, they require no particular knowledge about the problem structure other than the objective function itself. At each iteration step, a new set of approximations is assumed by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators, such as mutation, crossover and selection, borrowed from natural genetics in order to generate the new generations [11]. The GA flowchart is shown in Figure 3.

In the case of using a GA method to tune the PID gains to design the control system for the mobile robot, the fitness function used to evaluate the individuals of each generation can be chosen to be Mean Square Error (MSE)

$$MSE = \frac{1}{N_s} \sum_{k=1}^{N_s} (e_d^2(t) + e_\theta^2(t)) \quad (12)$$

where  $N_s$ =No. of samples ,  $k$ : sample time.

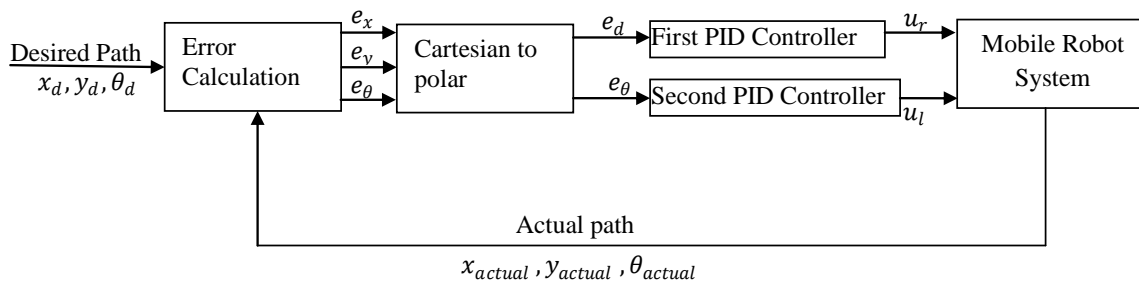


Fig. 2: Block diagram for mobile robot trajectory tracking control

$x, y$  and  $\theta$  are calculated from equation 10:

$$\begin{cases} e_x = x_d - x \\ e_y = y_d - y \\ e_\theta = \theta_d - \theta \end{cases} \quad (10)$$

where ;  $\{x, y, \theta\}$  : are the actual  $x, y$  and the actual azimuth of the robot, respectively. Then convert the Cartesian Coordinate into Polar Coordinate to calculate the distance error as shown below:

$$e_d = \sqrt{e_x^2 + e_y^2} \quad (11)$$

During the search process, the GA looks for the optimal setting of the PID controller gains which minimizes the fitness function (MSE).

This function is considered as the evolution criteria for the GA. The choice of this fitness function has the advantage of avoiding cancellation of positive and negative errors.

Each chromosome represents a solution of the problem and hence it consists of three genes: the first one is the Kp value, the second one is Ki value and last one is the Kd value; Chromosome vector = [Kp Ki Kd]. It must be noted here that the range of each gain must be specified. Figure 4 show the learning phase of the PID controllers which connected with autonomous mobile robot.

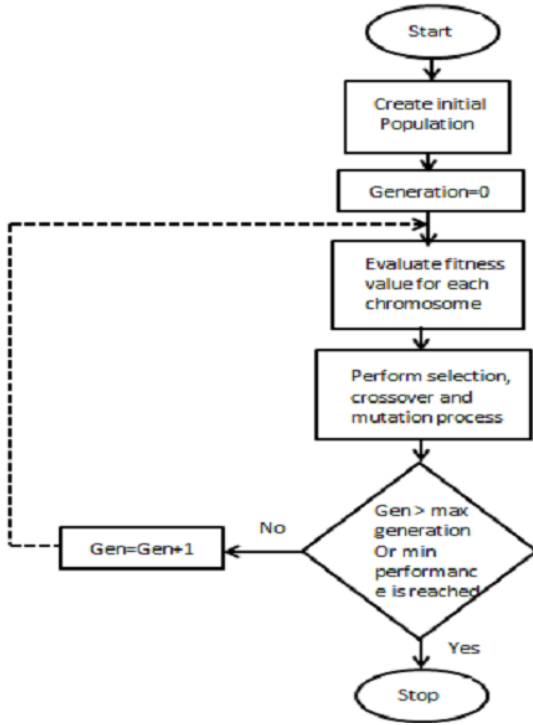


Fig. 3: Genetic Algorithm architecture

bees) and Scout bees (which carry out random search for new food source near the hive). Employed bees share information about food sources by dancing in the designated dance area inside the hive. The nature of dance is proportional to the nectar content of food source just exploited by the dancing bee. Onlooker bees watch the dance and choose a food source according to the probability proportional to the quality of that food source. Therefore, good food sources attract more onlooker bees compared to bad ones. Whenever a food source is exploited fully, all the employed bees associated with it abandon the food source and become scout. Scout bees can be visualized as performing the job of exploration, where as employed and onlooker bees can be visualized as performing the job of exploitation.

ABC algorithm, as an iterative algorithm, starts by associating each employed bee with randomly generated food source (solution). In each iteration, each employed bee discovers a food source in its neighborhood and evaluates its nectar amount (fitness) using Equation (13), and computes the nectar amount of this new food source:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \quad (13)$$

where  $x_{ij}$  is the  $j$ th food source,  $x_{kj}$  is the  $j$ th randomly selected food source,  $j$  is a randomly chosen parameter index and  $\phi_{ij}$  is a random number within the range [-1, 1]. The range of this parameter can make an appropriate adjustment on specific issues.

Onlooker bees observe the waggle dance in the dance area and calculate the profitability of food sources, then randomly select a higher food source. After that onlooker bees carry out randomly search in the neighbourhood of food source. The flowchart for ABC algorithm is given in Figure 5.

The quantity of a food source is evaluated by its profitability and the profitability of all food sources.  $P_i$  is determined by the formula:

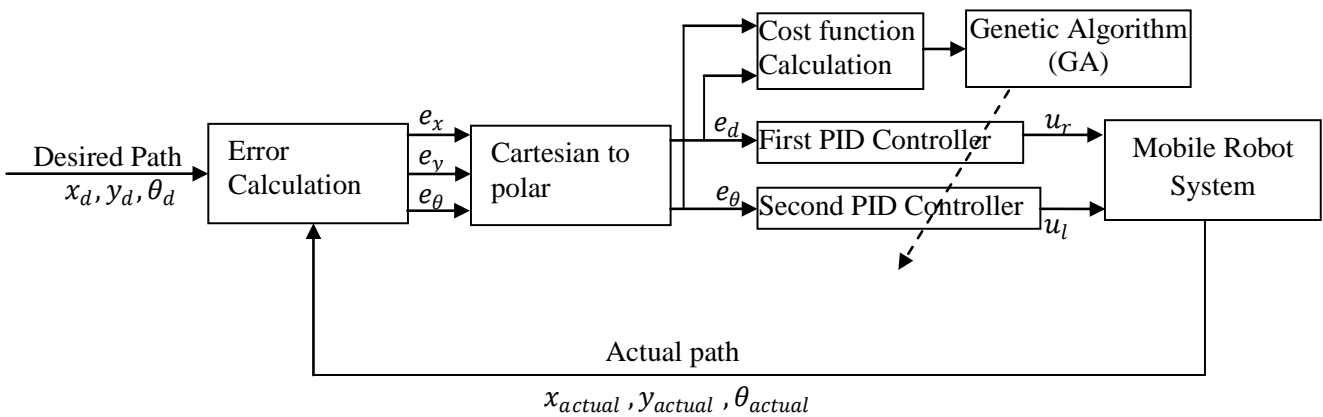


Fig. 4: The scheme of auto-tuning of the PID Controller by used GA

## 5. ARTIFICIAL BEE COLONY ALGORITHM

The Artificial Bee Colony (ABC) algorithm was recently proposed by Karaboga [12]. It proposed to simulate the foraging behavior of honey bee colonies. In this Algorithm, the foraging bees are divided into three categories: Employed bees (which go to the food source), Onlookers bees (which wait on the dancing area to making decision to choose a food source depend on the collecting information by employed

$$P_i = \frac{fit_i(x_i)}{\sum_{i=1}^{SN} fit_i(x_i)} \quad (14)$$

where  $fit_i(x_i)$  is the fitness value of  $x_i$ . This value is proportional to the nectar amount of the food source in the position  $i$  and  $SN$  is the number of food source which is equal to the number of employed bees.

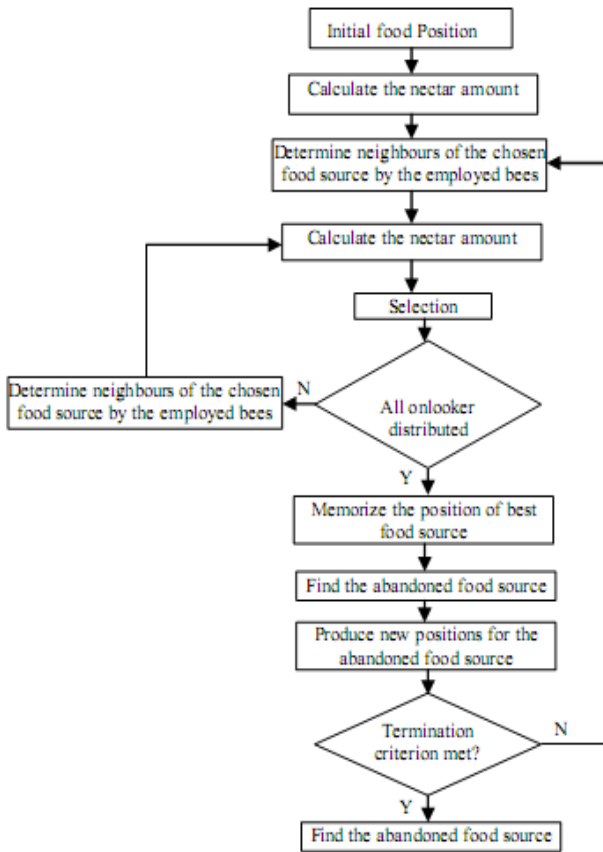


Fig. 5: The flowchart of ABC Algorithm

The employed bee becomes a scout bee when the food source which is exhausted by the employed and onlooker bees is assigned as abandoned. In that position, scout generates randomly a new solution by Eq. (13):

$$x_i^j = x_{min}^j + rand(0,1)(x_{max}^j - x_{min}^j) \quad (13)$$

Assume that the abandoned source is  $x_i$ , then the scout discovers a new food source to be replace with  $x_i$ .

## 6. SIMULATION AND RESULTS:

For the autonomous mobile robot system discussed in Section 2, the numerical values are used in this simulation are given in Table 1.

The simulation blocks that used for simulation trajectory tracking using GA-PID Controller is shown in Figure 5 (the same structure is used for simulation trajectory tracking using ABC-PID Controller). This Simulink is programmed as m file to achieve optimization with solving all the differential equations by using the 4<sup>th</sup> order Runge-Kutta method and the control sampling period is 0.01s. The search space consists of six dimensions, three dimensions specified for first controller (velocity controller) and other three dimensions for the second controller (azimuth controller). The GA parameters values are shown in Table 2, while optimal tuned parameters of PID controllers are shown in Table 3.

The ABC parameters values are shown in Table 4, while PID Controllers optimal tuned parameters using ABC algorithm are shown in Table 5.

Table 1. numerical values of the mobile robot

Parameters	Value	Unit
$v$	1	m/s
$r$	0.05	m
$L$	0.1	m
$d$	0.1	m
$m_c$	1.8	Kg
$m_w$	0.1	Kg
$m$	2	Kg
$I_m$	0.0025	Kg.m <sup>2</sup>
$I_w$	0.001	Kg.m <sup>2</sup>

Table 2. The GA parameters values

GA Parameters	Value
Population Size	30
Maximum iteration number	100
Dimension	6
Range of parameters	0 - 200
Mutation rate	0.2
Selection	0.5
Minimum Cost	$1e^{-15}$

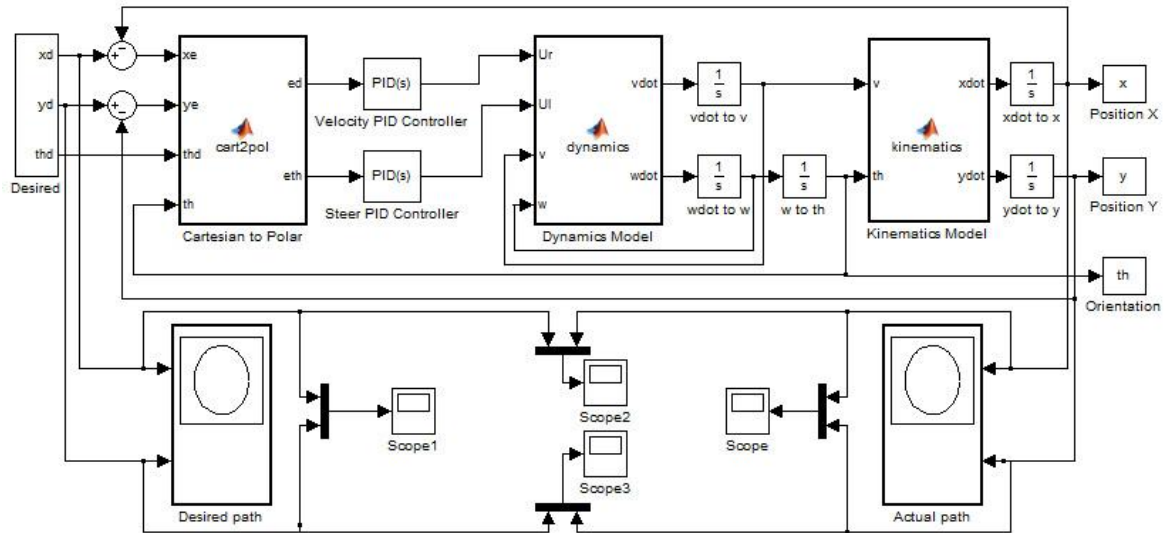


Fig. 5: Simulation model for mobile robot trajectory tracking control using PID controllers

Table 3 PID Controllers optimal tuned parameters using GA

Variables	Values
$k_{p1}$	99.4610
$k_{i1}$	0.1287
$k_{d1}$	4.5571
$k_{p2}$	191.7988
$k_{i2}$	1.3629
$k_{d2}$	8.9214

Table 5 PID Controllers optimal tuned parameters using ABC

Variables	Values
$k_{p1}$	100.0007
$k_{i1}$	0
$k_{d1}$	4.5167
$k_{p2}$	200
$k_{i2}$	0
$k_{d2}$	9.0333

Table 4. The ABC parameters values

ABC Parameters	Value
Colony Size	40
Maximum iteration number	100
Dimension	6
Limit	5
scout production	5

In order to investigate the performance of two proposed controller, we compare the results of these controllers for tracking at two different paths to determine the better controller.

- 1) The circular trajectory given by a linear velocity

$$v = 1 \text{ m/sec and desired azimuth given by}$$

$$\theta_d = 2 * \pi * \text{scale} * t / t_{\text{end}} \quad (15)$$

where  $t_{\text{end}} = 5 \text{ sec}$  and  $\text{scale} = 1$ . Figure 6 shows the actual and desired path for circular path trajectory. Figures 7, 8 and 9 show the  $e_x$ ,  $e_y$  and  $e_\theta$  respectively.

Table 6 illustrates the MSE for circular trajectory tracking for two controllers.

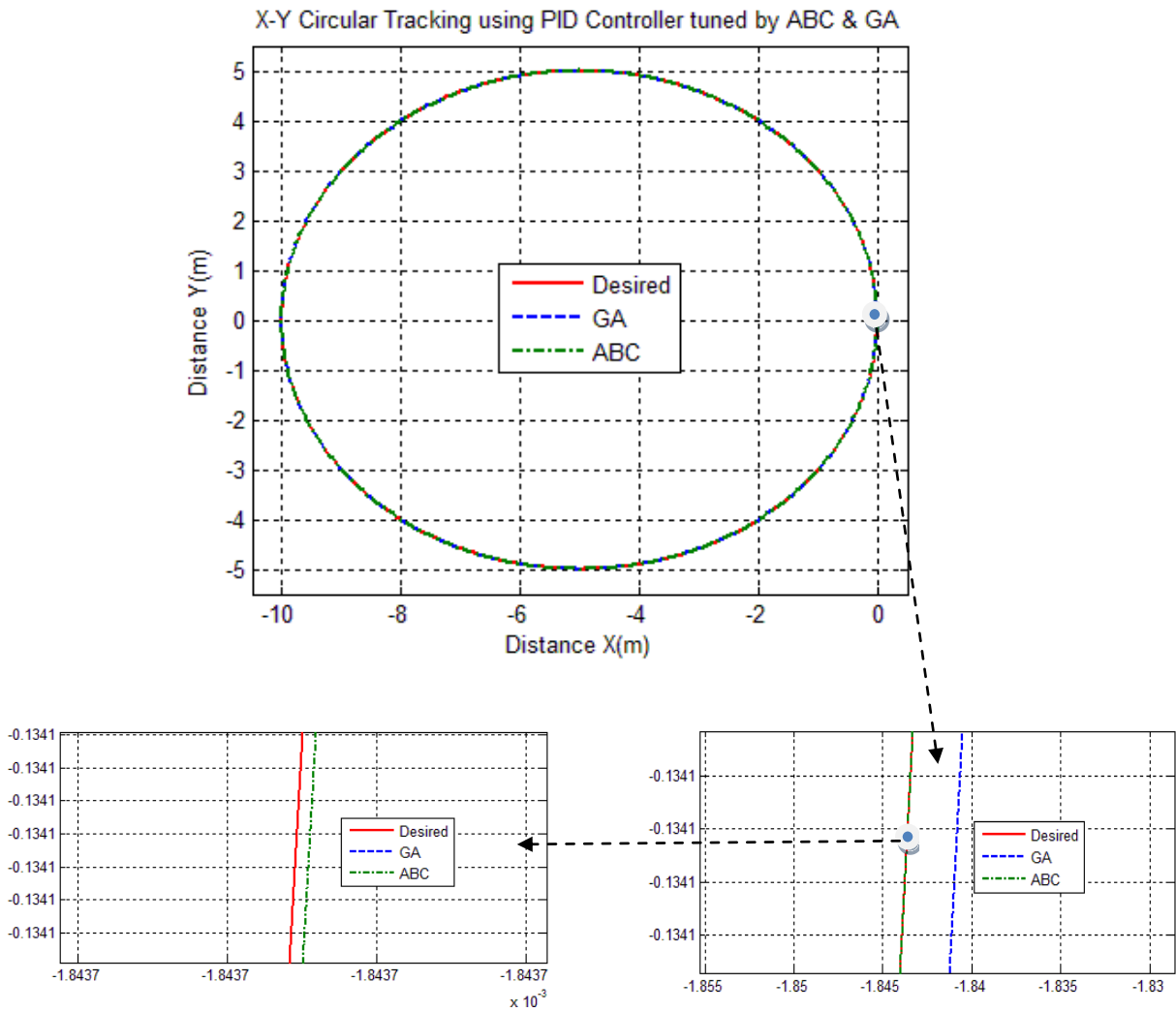


Fig. 6: Circular Trajectory using GA & ABC algorithm

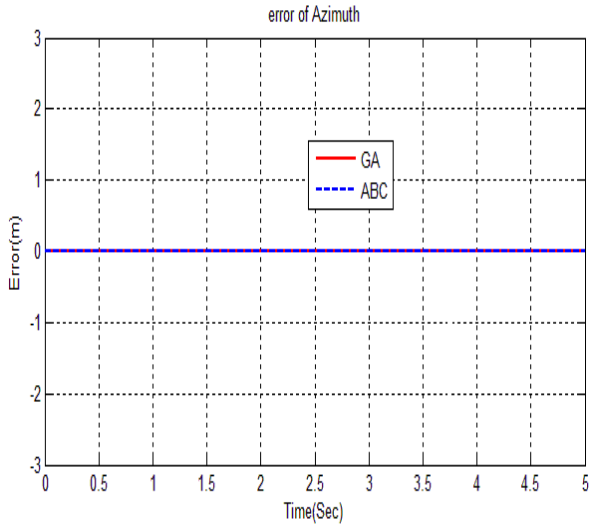


Fig. 9:  $e_\theta$  for Circular Trajectory using GA and ABC algorithm

Table 6 MSE for circular trajectory tracking for two controllers.

Parameter	Algorithm	
	Genetic	ABC
$e_x$	$3.0332e^{-06}$	$1.6586e^{-10}$
$e_y$	$6.2711e^{-05}$	$2.6396e^{-08}$
$e_\theta$	$1.1911e^{-05}$	$5.7856e^{-08}$
Total MSE	$1.2911e^{-05}$	$1.6446e^{-08}$

- 2) The sine wave trajectory given by a linear velocity  $v = 1 \text{ m/sec}$  and desired x position, desired y position, desired azimuth are given by

$$\left. \begin{aligned} \theta_d &= \tan^{-1}\left(\frac{2 * \pi * \cos(\phi)}{t_{end}}\right) \\ x_d &= t \\ y_d &= \sin(\phi) \end{aligned} \right\} (16)$$

Where  $\phi = 2 * \pi * \frac{t}{t_{end}}$  and  $t_{end} = 5 \text{ sec}$ .

Figure 10 shows the actual and desired path for sine wave path trajectory. Figures 11, 12 and 13 show the  $e_x$ ,  $e_y$  and  $e_\theta$  respectively.

Table 7 illustrates the MSE for sine wave trajectory tracking for two controllers.

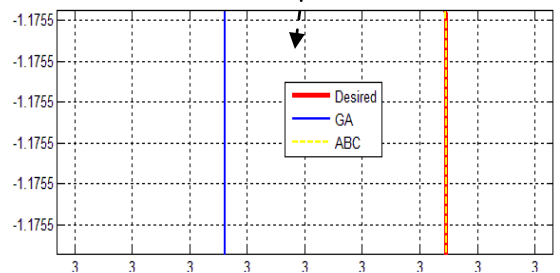
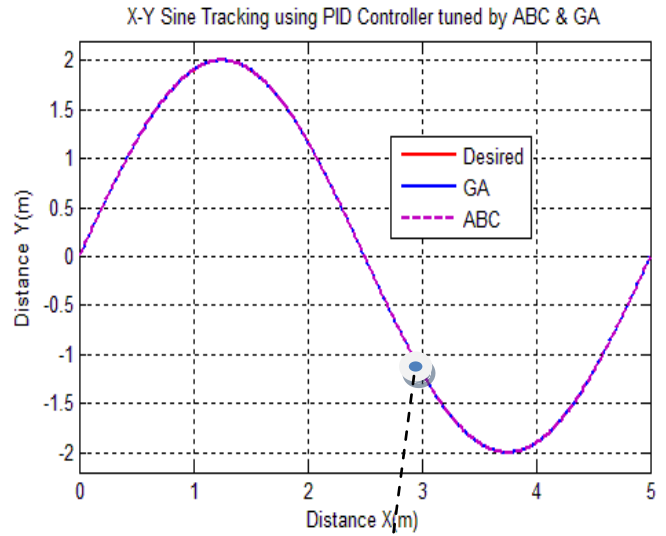


Figure 10 Sine wave Trajectory using GA and ABC algorithm

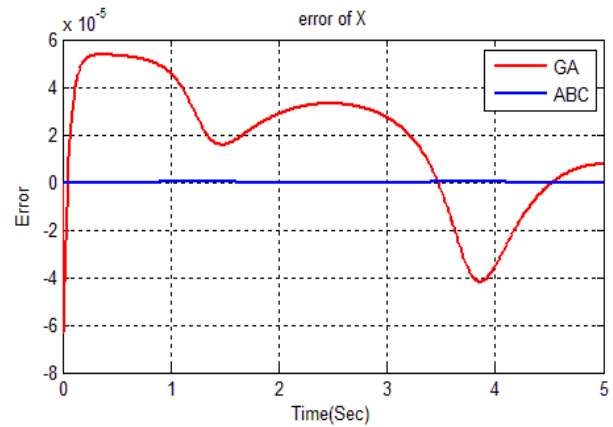


Figure 11  $e_x$  for sine wave trajectory using GA and ABC algorithm

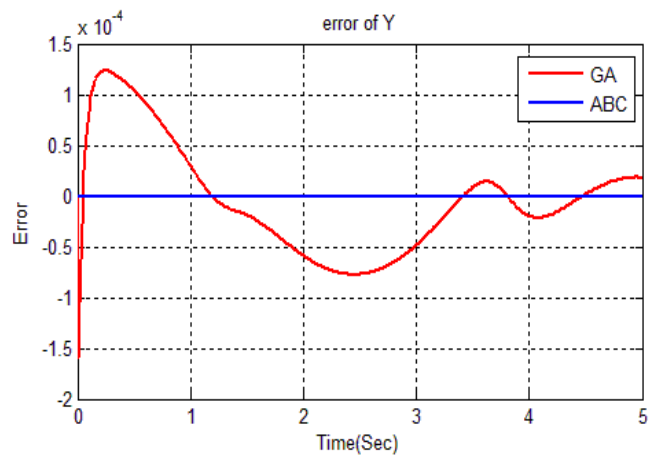


Figure 12  $e_y$  for sine wave trajectory using GA and ABC algorithm



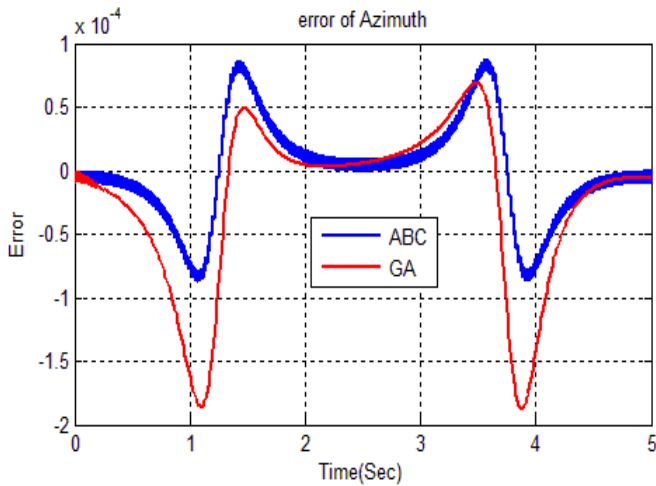


Fig. 13:  $e_{\theta}$  for sine wave trajectory using GA and ABC algorithm

Table 7. MSE for sine wave trajectory tracking for two controller(s).

Parameter	Algorithm	
	Genetic	ABC
$e_x$	$2.22 e^{-05}$	$1.9775 e^{-5}$
$e_y$	$3.7281 e^{-05}$	$2.09 e^{-05}$
$e_{\theta}$	$6.76 e^{-06}$	$7.37 e^{-09}$
Total MSE	$9.8481 e^{-05}$	$5.792 e^{-05}$

## 7. CONCLUSION

Two optimization algorithms are used to find the optimal values of two PID controllers to solve the trajectory tracking of autonomous mobile robot. The obtained results show that the ABC algorithm is more effective than the genetic algorithm. For both tested trajectories path (circular trajectory and sine wave trajectory) the MSE when ABC used to tune the PID controllers parameters is less than the MSE when genetic algorithm used.

## 8. REFERENCES

- [1] Astrom, K.J. and T. Haggund, *PID controller: Theory, Design and Tuning* 1995, USA: Instrument Society of America, Research Triangle Park.
- [2] Xue, D., Y. Chen, and D.P. Atherton, *Linear Feedback Control Analysis and Design with MATLAB*. 2007, USA: The society for Industrial and Applied Mathematics
- [3] Turki Y. Abdalla, S.J.A., *Genetic Algorithm Based Optimal of a Controller for Trajectory Tracking of a Mobile Robot*. Basrah Journal for Engineering Science, 2010. **1**(1): p. 54-65.
- [4] Ali, A.A., *PID Parameters Optimization Using Genetic Algorithm TEchnique for Electrohydraulic Servo Control System*. Intelligent Control and Automation, 2011. **2**: p. 69-76.
- [5] Gauri Mantri, N.R.K., *Design and Optimization of Controller Using Genetic Algorithm*. International Journal of Research in Engineering and Technology, 2013. **2**(6): p. 926-930.
- [6] Ali Tarique, H.A.G., *Particle Swarm Optimization Based Turbine Control*. Intelligent Control and Automation, 2013. **4**: p. 126-137.
- [7] Mahmud Iwan Solihin, L.F.T., Moey Leap Kean *Tuning of PID Controller Using Particle Swarm Optimization in Proceeding of the International Conference on Advanced Science, Engineering and Information Technology*. 2011: Bangi-Putrajaya, Malaysia.
- [8] Gaing, Z.-L., *A Particle Swarm Optimization Approach for Optimum Design PID Controller in AVR System*. IEEE Transaction on Energy Conversion, 2004. **19**(2): p. 384-391.
- [9] El-Telbany, M.E., *Tuning PID Controller for DC Motor: an Artificial Bees Optimization Approach* International Journal of Computer Applications, 2013. **77**(15): p. 18-21.
- [10] Reched Dhaouadi, A.A.H., *Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework*. Advances in Robotics and Automation, 2013. **2**(2): p. 1-7.
- [11] S. Sumathi, T.H., *Evolutionary Intelligence An Introduction to Theory and Applications with Matlab*, ed. S.V. Berlin. 2007, German.
- [12] D. Karaboga, B.B., *A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony ABC Algorithm*. . Journal Global Optimization, 2007. **39**: p. 459-471.