# Evolutionary Computation Based Control for Ball and Plate Stabilization System

H. GÖZDE

*Abstract*— **The platform stabilization systems used in marine, airborne or land vehicle applications are controlled with very different control methods basically including linear, nonlinear and artificial intelligence-based design techniques. Nowadays, evolutionary computation based optimization algorithms also provide new opportunities to engineers in order to design a gain scheduling controller. In this study, an evolutionary computation based gain scheduling controller is proposed for a ball and plate system so as to examine its control performances on a stabilization system. For this purpose, the swarm intelligence based algorithms are chosen due to their better performance than the other evolutionary computation algorithms. The results are comparatively investigated by using time domain and frequency domain analysis methods. Additionally, the robustness analysis is also applied to examine the tuning performances of these controllers in case of changing system parameters in the range of ±50%.**

*Index Terms*— **Platform stabilization, Ball and plate system, Gain scheduling control, Evolutionary computation, Particle swarm optimization algorithm, Differential evolution algorithm.**

## I. INTRODUCTION

NOWADAYS, the platform stabilization systems are commonly used in marine, airborne and land vehicle applications so as to direct satellite or radar antennas, cameras, missiles, guns etc. in the civil or military areas. These systems have been utilized since about 100 years in order to isolate motion of the vehicle from that of the platform by measuring the change of platform's motion and position continuously [1]. Specifically, despite the change of precision and accuracy depending on the application, the platform stabilization systems generally comprise three fundamental components: inertial and/or position sensor system, mechanical platform with two or more degrees of freedom and control system. The inertial or position sensor systems are required to determine the real position of the platform. In the case of employing inertial sensor system; the roll, pitch, and azimuth data of the platform measured by three accelerometers and gyroscopes in their own local reference frames are converted to the absolute three-dimensional position data in the global reference frame by combining with the direction data obtained by the magnetometer. However, in the applications where the geographical position data is not important, the well-known motor shaft position sensors such as potentiometers or rotary encoders can be used in order to measure only relative position of the platform in its local reference frame. The mechanical platform providing three dimensional movements is specially designed according to the intended use. The issues such as physical dimensions, weight bearing capacity, degree of freedom (DOF), controllability, place of use, vibrations, torsion strengths, ease of maintenance etc. are taken into account in the design procedure. The control system is especially designed as a closed loop position controller. Relative movement data of the platform is taken from the sensors and then is used in order to keep its stabilization according to global reference frame. In this process, since the control system should cope with the sensor noise, measurement errors, system uncertainties, dead-bands and environmental disturbances etc., the advanced noise filtering techniques and/or relatively complex control methods such as adaptive, optimal or robust controllers are commonly exploited in this type of control systems [2-7]. However, when the response speed of the actuators are essential as depending on the nature of application, complexity and memory requirement of the chosen controller turn into big challenges for implementing the system. In this case, fast microprocessors with large memory should be utilized for implementation or less complex but robust artificial intelligence-based control methods such as fuzzy control, genetic algorithm, swarm optimizations or their hybrids can be preferred with less complex microprocessors [8].

In this way, this system can be considered a basic platform which has similar mechanical and control characteristics with the advanced platform stabilizer systems discussed above. It can be generally defined as a typical multi-variable system and basically consists of plate, actuators, mechanical transmissions, ball and ball position sensors as depicted in Fig.1 [9]. Actually, there are some control challenges similar to ones belonging to the advanced platform stabilization systems discussed above. For instance, the sensor noise is fully effective on the ball position accuracy. This problem has been largely solved by using noise-free overhead camera in the ball and plate system, instead of using noise filters and advanced microprocessors. On the other hand, the control technique is still very important issue for transient and steady-state control performances of the system on account of stabilization and trajectory tracking control as well as mechanical impressions such as frictions and backlash.

**HALUK GÖZDE**, is with Department of Electronics and Communication Engineering, National Defence University, Turkish Military Academy, Ankara, Turkey, (e-mail: hgozde@kho.edu.tr).

https://orcid.org/0000-0003-3661-0794

Manuscript received October 1, 2018; accepted January 18, 2019.
DOI: 10.17694/bajece.466306

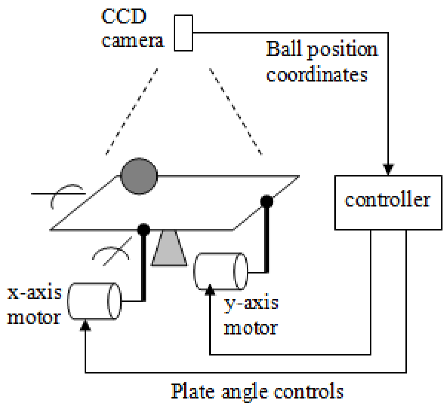



Fig.1. Basic scheme of ball and plate system [9].

If the literature related to control of ball and plate system is investigated, it is seen that various linear, nonlinear or artificial intelligence based control methods have been applied to the system since the beginning of 2000s. In 2004, Fan et al studied on the trajectory planning and tracking control of the ball moving on plate designed as a maze [10]. After that, Bai et al performed a fuzzy controller to control moving of the ball [11]. In 2008, while Hongrui et al designed a non-linear controller [12], Bai et al used an Adaptive Neuro Fuzzy Inference System (ANFIS) controller for the same purpose [13]. Dong et al realized a fuzzy neural network controller optimized by genetic algorithm, then applied the genetic algorithm-based neuro-PID controller to the same system in 2009 [14, 15]. In the same year, Casagrande et al designed a stable nonlinear controller based on Lyapunov function [16]. Moreno-Armend´ariz et al implemented a fuzzy controller on FPGA based hardware to control ball and plate system in 2010 [17]. In 2011, Dong et al used PSO based neuro-fuzzy controller [18], and one year later Han et al also applied PSO to tune the PID controller parameters for ball and plate system [19]. Mochizuki et al proposed a design method of PID controller based on the generalized Kalman-Yakubovich-Popov lemma in 2013 [20]. PSO was applied to the ball and plate system again by Roy et al for tuning the parameters of PD trajectory controller in 2014 [21]. Zhao et al designed a fuzzy multi-variable control combined neural networks in the same year [22]. In 2015, Han et al proposed a fuzzy based indirect adaptive controller [23], Oravec et al suggested model predictive controller [24], and Negash et al designed fuzzy based sliding mode controller [25] in order to control ball and plate system. Xiao et al implemented adaptive embedded controller for the same purpose in 2016 [26]. It can be observed from the reviewed literature that the artificial intelligence based controllers have been more preferable than the classical linear and non-linear controllers due to the multivariable and high order structure of the ball and plate system. The artificial intelligence tools provide great convenience to easy design and implement the optimal and adaptive controllers for this type of high order complex systems. For this reason, evolutionary computation based gain scheduling control method is proposed.

In detail, the aim of this study is to examine the tuning performance of evolutionary computation based optimization algorithms according to classical tuning for a platform stabilization systems discussed above on the example of ball and plate system. The evolutionary computation based algorithms are heuristic optimization tools which have short, algebraic and fast convergent program code and without derivative calculations. They are basically separated into two types: First one is the evolutionary algorithms based on Darvin’s theory of evolution, such as genetic and differential evolution algorithms etc. Second one is the swarm intelligence algorithms based on food searching behavior of organisms, such as particle swarm optimization and artificial bee colony algorithms etc. In this study, two types of evolutionary computation methods are applied to the gain scheduling controller to achieve adaptive control structure for ball and plate stabilization system against internal or external parameter variations.

The paper is organized as follows. The ball and plate system and its modeling are described in Section II. The proposed gain scheduling control system designed and the evolutionary computation based algorithms are explained in Section III. The simulation and the experimental results are represented in Section IV and Section V, respectively. The discussion is performed in Section VI. Finally, concluding remarks are presented in Section VII.

## II. Model Of Ball And Plate System

In this study, 2-DOF ball and plate stabilization system made by QUANSER is utilized due to its MATLAB/Simulink software support and user friendly interfaces.

Fig.2. Ball and plate system made by QUANSER [27].

This system basically consists of a plate, a ball, an overhead camera and two servo units as presented in Fig.2. While the servo units and plate allow rolling the ball about any direction, the overhead camera measures the position of the ball for feedback to the controller. There are two QUANSER SRV02 servo units under the plate, which are connected to the plate in order to provide 2-DOF gimbal actions [27].

The model of ball and plate system for only one axis is presented in Fig.3. Actually, the plate is capable of two axes movement. But, since both axes are symmetrical according to each other, their dynamics are also assumed same and the system can be modeled only in one axis. In this point of view,

$x$ is motion of the ball, $\alpha$ is angle of the plate, $m_b$ is mass of the ball, $F_{x,r}$ is force caused from the ball's inertia and $F_{x,t}$ is translational force generated by gravity. The friction and viscous damping are neglected. In order to model the one axis system, it can be assumed that $x$, $\alpha$ and $\theta_l$ are equal to zero while the ball is stationary in the center and the plate is parallel to the ground. At the same time, some forces caused from ball's momentum and gravity affects the ball for all conditions. In theory, the force from the ball's momentum must equal to the force produced by gravity along the plate surface in the $x$-axis as in Equation 3.

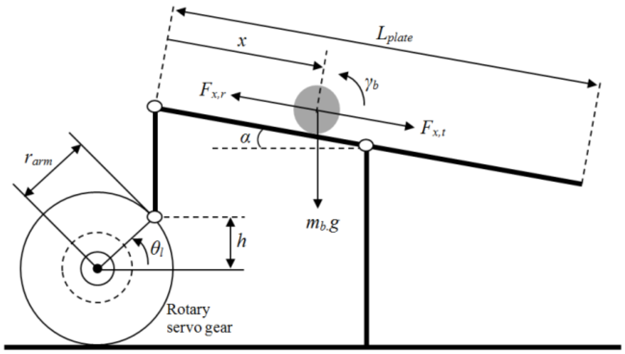



Fig.3. The one axis model of ball and plate system [27].

In addition, the mechanical parts of ball and plate system are considered into two parts: rotary servo gear mechanism and plate mechanism as depicted in Fig.4. In order to obtain the transfer function models of these parts, it can be started from Newton's first law [27];

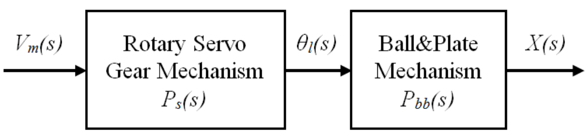



Fig.4. The mechanical parts of ball and plate system [27].

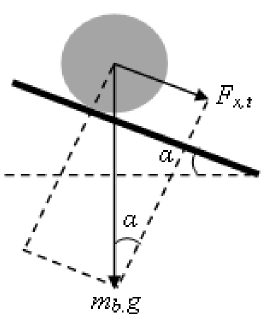



Fig.5. Gravitational forces affecting the ball.

$$F = m_b.a = m_b.\frac{d^2x}{dt^2} = m_b.\ddot{x} = F_{x,t} - F_{x,r} \quad (1)$$

where $F_{x,t}$ from Fig.5, and $F_{x,r}$ from Equation 3 are computed in Equations 2 and 4,

$$F_{x,t} = m_b.g.\sin\alpha \quad (2)$$

$$T_b = F_{x,r}.r_b = J_b.a_b = J_b.\ddot{\gamma}_b \quad (3)$$

$$F_{x,r} = \frac{T_b}{r_b} = \frac{J_b.\ddot{\gamma}_b}{r_b} \quad (4)$$

In these equations, $J_b$ is inertia of ball, $T_b$ is torque applied to the ball, $r_b$ is radius of ball and $\gamma_b$ is rolling angle of the ball. If the angular displacement $\gamma_b$ is transformed to linear displacement $x$ by Equations 5 to 6 as depicted in Fig. 6;

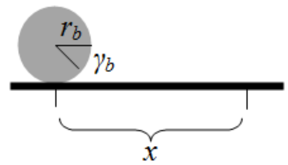



Fig.6. Transformation of angular and linear displacements.

$$2.\pi.r_b.\frac{\gamma_b}{2.\pi} = x \quad (5)$$

$$\ddot{\gamma}_b = \frac{\ddot{x}}{r_b} \quad (6)$$

$$F_{x,r} = \frac{J_b.\ddot{\gamma}_b}{r_b} = \frac{J_b.\ddot{x}}{r_b^2} \quad (7)$$

If Equation 2 and Equation 7 are substituted in Equation 1, then;

$$m_b.\ddot{x} = m_b.g.\sin\alpha - \frac{J_b.\ddot{x}}{r_b^2} \quad (8)$$

The acceleration of the ball on the plate is obtained from Equation 8;

$$a = \ddot{x} = \frac{m_b.g.\sin\alpha.r_b^2}{m_b.r_b^2 + J_b} \quad (9)$$

The relationship between the plate angle $\alpha$ and servo gear rolling angle $\theta_l$ can be written below, because the height h is equal to each other for both angles as represented in Fig. 7;

$$\sin\alpha = \frac{2.h}{L_{plate}} \quad (10)$$

$$\sin\theta_l = \frac{h}{r_{arm}} \quad (11)$$

From these equations,

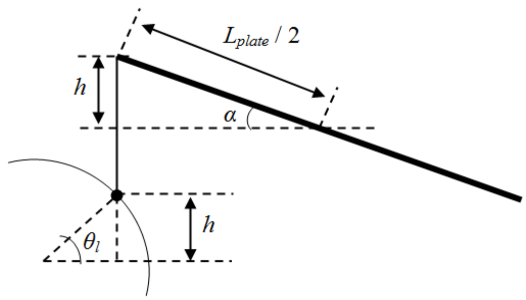



Fig.7. Relation between angle $\alpha$ and height $h$.

$$\sin\alpha = \frac{2.r_{arm}.\sin\theta_l}{L_{plate}} \tag{12}$$

If Equation 12 is substituted in Eq. 9, then the nonlinear acceleration equation of the ball on the plate is obtained below;

$$\ddot{x} = \frac{2.m_b.g.r_{arm}.r_b^2}{L_{plate}.(m_b.r_b^2 + J_b)}.\sin\theta_l \tag{13}$$

To linearize this equation, servo gear rolling angle $\theta_l$ can be taken near zero. In this case, the linear acceleration equation of the ball on the plate is obtained as Equation 15;

$$\theta_l \cong 0 \Rightarrow \sin\theta_l = \theta_l \tag{14}$$

$$\ddot{x} = \frac{2.m_b.g.r_{arm}.r_b^2}{L_{plate}.(m_b.r_b^2 + J_b)}.\theta_l \tag{15}$$

The transfer function model of the ball and plate system between servo gear rolling angle $\theta_l$ and motion of the ball $x$ is obtained from Equation 15 using Laplace transformation. In this equation, the constant multiplied by angle $\theta_l$ is defined as $K_{bb}$.

$$s^2.X(s) - s.x(0) - x'(0) = K_{bb}.\theta_l(s) \tag{16}$$

If the initial values of the motion are accepted zero, then the linear transfer function model of the ball and plate mechanism $P_{bb}(s)$ can be obtained as such [27];

$$\frac{X(s)}{\theta_l(s)} = \frac{K_{bb}}{s^2} = P_{bb}(s) \tag{17}$$

On the other hand, the rotary servo gear mechanism can be modeled as a two-order transfer function by using classical DC motor armature circuit and gear train modeling sequence as determined in Equation 18 [28].

$$P_s(s) = \frac{\theta_l}{V_m} = \frac{K}{s(\tau.s+1)} \tag{18}$$

where $V_m$ is DC servo motor input voltage, $K$ is servo gain and $\tau$ is servo time constant. As a result, the fourth-order linear complete transfer function model of complete ball and plate system can be computed as;

$$P(s) = P_s(s).P_{bb}(s) \tag{19}$$

$$P(s) = \frac{X(s)}{V_m(s)} = \frac{K}{s(\tau.s+1)}.\frac{K_{bb}}{s^2} = \frac{K.K_{bb}}{s^3.(s.\tau+1)} \tag{20}$$

The model parameters are presented in Table 1 [27, 28].

TABLE I
PARAMETERS OF THE BALL AND PLATE SYSTEM MODEL [27]

| **Parameters** | **Parameter values** |
|---|---|
| $m_b$ | 0.0252 kg |
| $r_b$ | 0.017 m |
| $g$ | 9.81 m/s$^2$ |
| $J_b$ | 2.89.10$^{-6}$ kg.m |
| $L_{plate}$ | 0.275 m |
| $r_{arm}$ | 0.0254 m |
| $K$ | 1.53 rad/V.s |
| $\tau$ | 0.0248 s |

## III. Proposed Control Method

The utilized ball and plate system is originally controlled by two sequential Proportional-Derivative (PD) and Proportional (P) control loops for ball position control and servo speed control, respectively [27]. Because each controller has constant control parameters which have previously calculated, the ball and plate system is completely sensitive to internal and external parameter variations. For this reason, the gain scheduling adaptive control method is suggested for the ball and plate stabilization system and comparatively discussed in this study. Basically, it is expected from this approach that controller parameters of the system are tuned to optimal values at normal condition and when the system parameters such as mass of the ball, and gain and/or time constant of the servo system, change in the range of ±50% for any reason at any time. Also, it is evaluated that the proposed controller will be able to be implemented as an embedded controller in the future.

### *3.1 Gain Scheduling Control*

Gain scheduling control is an adaptive control method tuning controller parameters according to different operating points where the system works. The gain scheduling control method is more suitable for nonlinear processes and time and/or parameter varying processes. In principle, all the desired system variables are measured firstly, and then the control parameters are tuned by a scheduling mechanism. The main superiority of this method is simpler design and implementation than the other adaptive control techniques as well as relatively fast adaptation ability to provide quick response to changes in system dynamics [29, 30].

In this study, it is suggested that the scheduling process is performed with evolutionary computation based optimization algorithms in order to examine the tuning performance of these types of algorithms according to classical tuning for a platform stabilization system on the example of ball and plate system. For this purpose, swarm intelligence based PSO and evolutionary algorithm based DE algorithms are chosen due to their better performance than others. The results are compared to the original P and PD controller given on the current system. The block diagram of proposed control method is represented in Fig. 8.

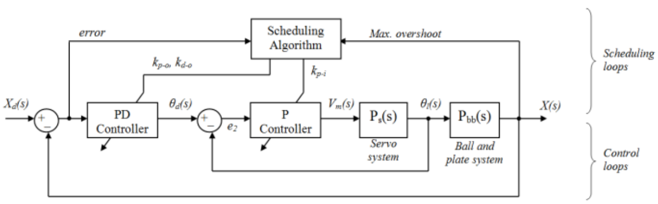



Fig.8. Proposed gain scheduling controller for 1-axis and 2-DOF ball and plate system.

### 3.2 Particle Swarm Optimization (PSO) Algorithm

PSO algorithm which was first introduced by Kennedy and Eberhart in 1995 is a swarm intelligence-based optimization algorithm [30]. Swarm intelligence is the study of evolutionary computation inspired by the collective food searching behavior of the bird flocks. It provides high speed convergence and high quality solutions with its short and algebraic program code without derivative calculations.

The algorithm basically models the food searching behavior of bird flocks or fish schools. Mathematically, it uses particles whose positions represent potential solutions of the problem and each particle flies in search space at a certain velocity which can be adjusted in light of preceding flight experiences. The positions of the particles are updated by the equations below, which include sufficient randomness;

$$v_i^{t+1} = w.v_i^t + c_1.r_1.(p_i^t - x_i^t) + c_2.r_2.(g_i^t - x_i^t) \qquad (21)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \qquad (22)$$

where, $i = 1, \ldots, n$ and n defines the size of swarm, $w$ is inertia weight decreased linearly for each iteration, $c_1$ and $c_2$ are positive constants for weighting local or global search activity, $r_1$ and $r_2$ are random numbers distributed uniformly between 0 and 1, superscript t is the iteration number, $p_i$ represents the best previous position of the $i^{th}$ particle and g represents the best particle position among all the particles in the swarm. At the end of the iterations, the best position $g$ will be optimum solution of the problem. Pseudo code of PSO algorithm is represented below;

```
Initialization
    Repeat
    Evaluate the fitness values of particles
    Compare the fitness values to determine the p and g
    Change velocity and position of the particles as to Eqs.21 and 22
Until (requirements are met)
```

### 3.3 Differential Evolution (DE) Algorithm

DE algorithm which was first suggested by Storrn in 1996 is the powerful stochastic optimization algorithm covered by the evolutionary algorithms. Evolutionary Algorithms is a branch of evolutionary computation concerned with computational methods inspired by the process of biological selection based on Darvin's theory of evolution [31]. In detail, DE algorithm also uses crossover, mutation and selection operators like genetic algorithms. But, the main difference of these algorithms is that while DE algorithm relies on mutation operation, the genetic algorithm is based on crossover operation. The algorithm is started with the first evaluation for the initial creation of the population. After that, recombination, evaluation, and selection processes are performed to the members of population, which are candidate solutions of the problem. The recombination is a process in which the new candidate solutions are created via selection process between two weighted random population members added to a third population member by using crossover process [32]. Pseudo code is represented below;

```
Initialization
    Evaluation
    Repeat
    Mutation
    Recombination
    Crossover
    Selection
Until (requirements are met)
```

## IV. Simulation Studies And Results

The simulation studies are realized in two parts on the one-axis Simulink model of ball and plate system depicted in Fig. 9. At the first part, the original P-PD controller given with QUANSER ball and plate system is investigated. After that, the proposed evolutionary computation based controllers are applied to the system at the second part of the study. Original and obtained $K_{p\text{-}o}$, $K_{d\text{-}o}$ and $K_{p\text{-}i}$ gains of the P-PD controllers are listed in Table 2. The model of P-PD controller which is designed originally and the tracking responses obtained from this and proposed controllers are depicted in Fig. 10 and Fig. 11. In the simulations performed with the handled algorithms, the upper and the lower bounds of the $K_{p\text{-}o}$, $K_{d\text{-}o}$ and $K_{p\text{-}i}$ gains are chosen as (2, 10), (2, 10) and (10, 20) respectively. The number of iteration and the population size are determined as 10 and 20 for both algorithms by trial and error method. Also, the scaling factor is taken as 0.02 for DE algorithm.

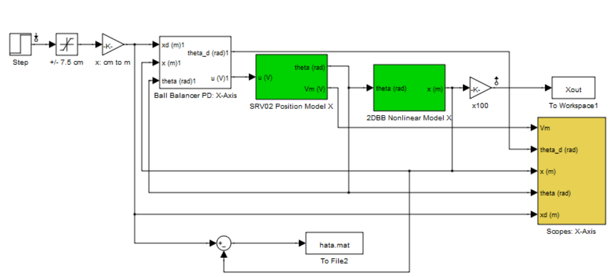


Fig.9. One-axis Simulink model of ball and plate system [27].

The integral of time weighted squared error (ITSE) function represented in Equation 23 is used as a cost function which will be minimized for determining the optimum values of tuned parameters. The purpose of choosing this function is to help to minimize settling time due to its dependency of errors on time. Additionally, the weighted maximum overshoot value computed from the result is also added to the cost function in order to minimize the overshoot of the system.

TABLE II
ORIGINAL AND OPTIMAL PD AND P PARAMETERS

| | | Classic P-PD | PSO based P-PD | DE based P-PD |
|---|---|---|---|---|
| **Outer loop** | $\mathbf{K_{p\text{-}o}}$ | 3.45 | 5.61 | 9.96 |
| | $\mathbf{K_{d\text{-}o}}$ | 2.11 | 2.69 | 3.42 |
| **Inner loop** | $\mathbf{K_{p\text{-}i}}$ | 14.00 | 14.98 | 16.52 |

$$ITSE = \int_{0}^{t} t.e^{2}.dt \tag{23}$$

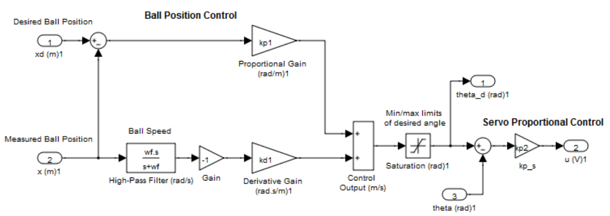



Fig.10. P-PD controller used in QUANSER ball and plate system [27].

The transfer functions of one-axis closed loop system according to the controller parameters obtained by the original, PSO based and DE based methods are represented in Equations 24, 25 and 26 respectively.

$$\frac{X(s)}{X_d(s)} = \frac{3237s + 50840}{5s^5 + 56.03s^4 + 1496s^3 + 13550s^2 + 34330s + 50840} \tag{24}$$

$$\frac{X(s)}{X_d(s)} = \frac{5629s + 88420}{5s^5 + 56.03s^4 + 1557s^3 + 14500s^2 + 48060s + 88420} \tag{25}$$

$$\frac{X(s)}{X_d(s)} = \frac{11030s + 173200}{5s^5 + 56.03s^4 + 1652s^3 + 15990s^2 + 70590s + 173200} \tag{26}$$

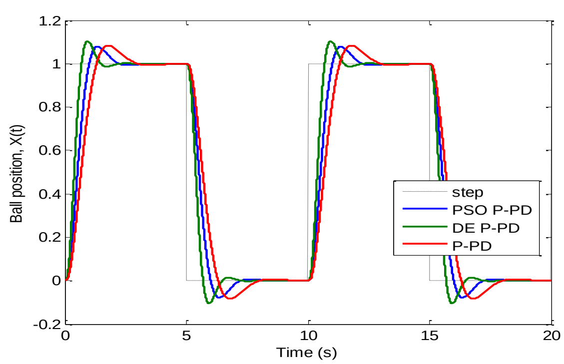



Fig.11. One-axis tracking response of the simulations.

At the end of the simulations, the tuning performance of the evolutionary computation methods according to classical method and each other is put forward by using time domain and frequency domain analysis methods as explained below. After that, the investigation of robustness of the system tuned by chosen algorithms is also examined.

*4.1 Step and Ramp Response Analysis*

The step response analysis is an investigation of system behavior during the time of the beginning and the steady state. It provides the data about relative stability and response speed of the closed loop system. For example, while the amount of maximum overshoot may be related to the relative stability, the settling and rise times show the response speed of the system. On the other hand, the ramp response gives an amount of deviation from the desired output in steady-state. The results obtained at the end of the analyses are represented in Table 3 and Figs. 12 and 13.

TABLE III
STEP AND RAMP RESPONSES OF THE CLOSED LOOP SYSTEM

| | Maximum Overshoot (%) | Settling Time [s] (±5% band) | Rise Time [s] | Peak Time [s] | Ramp Response [V] |
|---|---|---|---|---|---|
| **Classic P-PD** | 8.219 | 2.653 | 0.833 | 1.820 | 0.35 |
| **PSO based P-PD** | **7.830** | 1.965 | 0.625 | 1.372 | 0.28 |
| **DE based P-PD** | 10.300 | **1.220** | **0.422** | **0.932** | **0.21** |

The unit step response analysis shows that the performances of evolutionary algorithms tuned controllers are generally better than that of original P-PD controller.

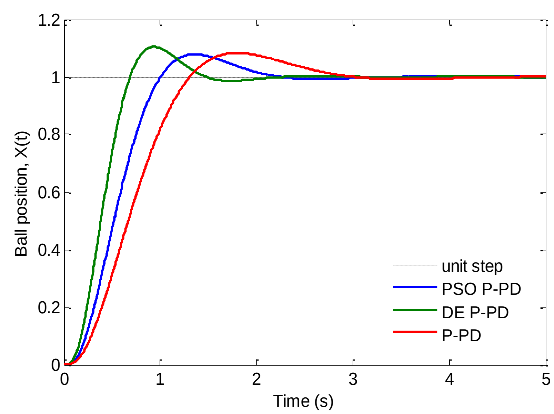



Fig.12. Unit step response of the closed loop system.

Especially, the settling time, rise time and peak time obtained with DE based controller are decreased about 50% with respect to the original P-PD controller. In this scope, it can be said that the response speed of the proposed method is better about 50% than the original one. However, it is observed that the maximum overshoot belongs to the system tuned by DE algorithm. On the other hand, the ramp responses show that the steady-state tracking error of the system is also decreased about 50% according to the original P-PD controller by using proposed P-PD controllers as represented in Table 3.

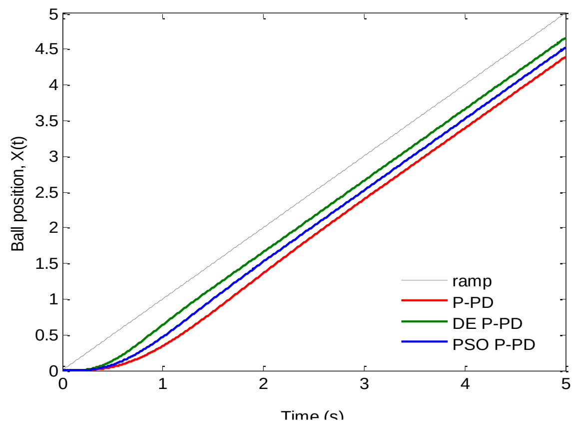



Fig.13. Ramp response of the closed loop system.

*4.2 Root Locus Analysis*

The root locus analysis is other time domain method which provides the data about stabilities of the closed loop system. Basically, while the placements of system poles can show whether the system is stable or not, the damping ratios of the poles emphasize information about degree of stability. The results of the analysis are presented in Table 4 and Figs. 14, 15, and 16.

TABLE IV
CLOSED LOOP POLES AND DAMPING RATIOS

| **Classic P-PD** | | **PSO based P-PD** | | **DE based P-PD** | |
|---|---|---|---|---|---|
| **Closed Loop Poles** | **Damping Ratio** | **Closed Loop Poles** | **Damping Ratio** | **Closed Loop Poles** | **Damping Ratio** |
| -21.3+j20.8 | **0.714** | -21.5+j22.1 | 0.697 | -21.7+j23.9 | 0.672 |
| -21.3-j20.8 | **0.714** | -21.5-j22.1 | 0.697 | -21.7-j23.9 | 0.697 |
| -10.6+j0 | 1 | -9.12+j0 | 1 | -7.51+j0 | 1 |
| -1.43+j1.83 | 0.618 | -1.99+j2.51 | 0.621 | -2.58+j3.94 | 0.548 |
| -1.43-j1.83 | 0.618 | -1.99-j2.51 | 0.621 | -2.58-j3.94 | 0.548 |

When the pole-zero maps of the closed loop system are investigated, all the system poles are at the left side of the s-plane. Hence, it can be said that all controllers provide absolute stability. In parallel, although close the each other, the better damping ratio belongs to the original P-PD controller. In contrast, the worst one belongs to the DE algorithm tuned controller in parallel to the overshoot results. These results exhibits that the original controller is more stable for given operating point.

*4.3 Bode Analysis*

The bode analysis which is the frequency domain analysis method gives information about the relative stability of the control system. The magnitude and phase plots are depicted in Figs. 17, 18 and 19. The peak gains and the delay, gain and phase margins are presented in Table 5.

To examine the frequency responses of the controllers, the gain and phase margins, peak gains and bandwidths are computed from the magnitude and phase curves for each controller. It is seen from these analyses that the gain and phase margins of the system based on DE algorithm have the worst values in parallel to the step response analysis. Also, the maximum bandwidth belongs to DE algorithm based system is greater about 50% than the original P-PD based system as expected from the results of step response analysis. These results mean that the DE algorithm based controller is less stable but faster than the others.

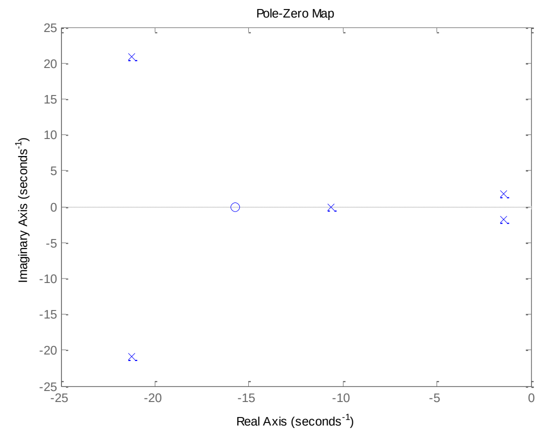



Fig.14. Pole-zero map of the original P-PD based system.

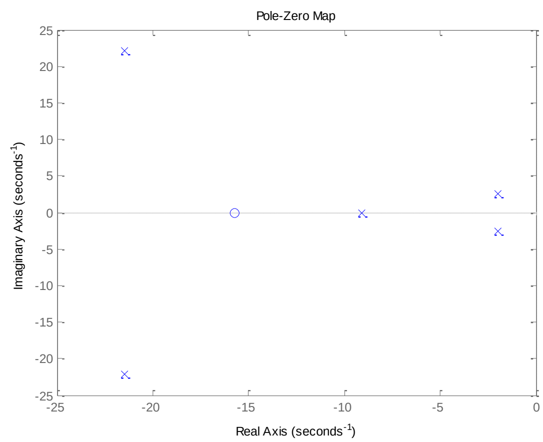



Fig.15. Pole-zero map of the PSO based system.

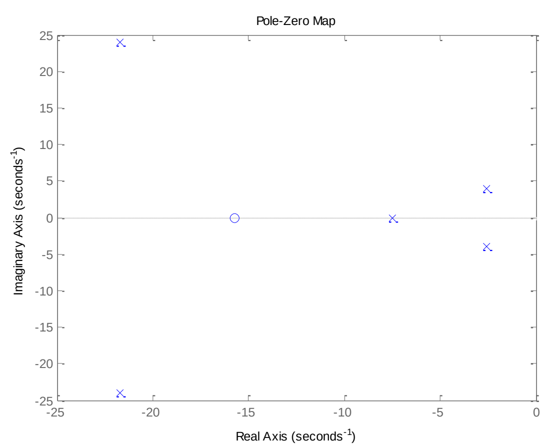



Fig.16. Pole-zero map of the DE based system.

*4.4 Robustness Analysis*

To examine the tuning performances of the evolutionary computation based controllers in case of the system parameter variations, a simple robustness analysis is applied to them. For this purpose mass of the ball is externally changed, servo gain and servo time constant are internally changed in the range of ±50% as represented in Table 6. At the end of the analysis, the controller parameters tuned by the PSO and DE algorithms are presented in Table 7. Additionally, the unit step responses obtained for these cases are also represented in Figs. 20, 21, 22, 23, 24 and 25 and in Table 8.

TABLE V
THE RESULTS OF BODE ANALYSIS

| | Peak Gain (dB) | Gain Margin (dB) | Phase Margin (deg.) | Bandwidth |
|---|---|---|---|---|
| **Classic P-PD** | 0.214 | **18.3** | 117 | 2.5718 |
| **PSO based P-PD** | 0.158 | 15.7 | **119** | 3.4552 |
| **DE based P-PD** | **0.341** | 12.0 | 97.8 | **5.1931** |

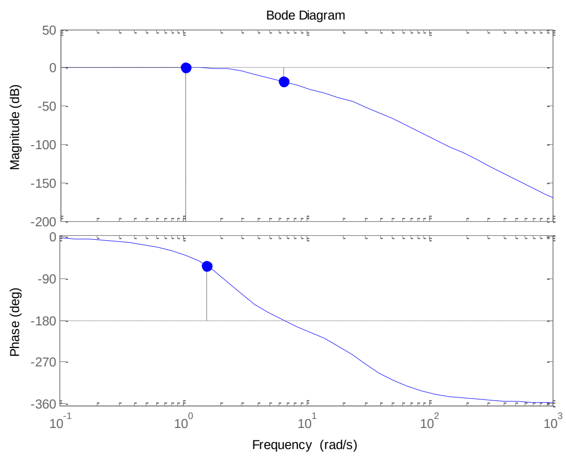


Fig.17. Bode curve of the original system.

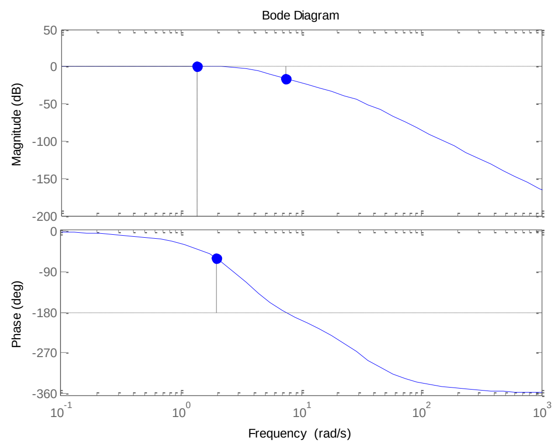


Fig.18. Bode curve of the PSO based system.

The analysis shows that although DE algorithm based controller has better tuning performance for given initial operating point, PSO algorithm exhibits better tuning capacity than DE algorithm in case of changing system parameters. In this analysis, the initial settings of the algorithms are not changed.

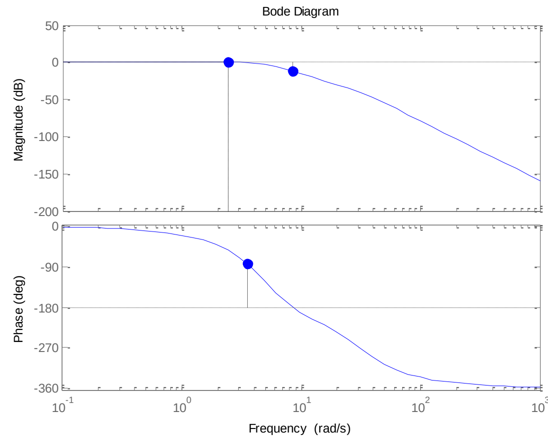


Fig.19. Bode curve of the DE based system.

TABLE VI
CASES OF SYSTEM PARAMETER CHANGING ACCORDING TO THEIR NOMINAL VALUES

| Cases | Disturbances | Disturbance types |
|---|---|---|
| Case-1 | 50% decrease of Servo gain ($K$) | Internal |
| Case-2 | 50% increase of Servo gain ($K$) | |
| Case-3 | 50% decrease of Servo time constant ($\tau$) | |
| Case-4 | 50% increase of Servo time constant ($\tau$) | |
| Case-5 | 50% decrease of Mass of the ball ($m_b$) | External |
| Case-6 | 50% increase of Mass of the ball ($m_b$) | |

TABLE VII
TUNED PARAMETERS DURING ROBUSTNESS ANALYSIS

| Cases | PSO based P-PD | | | DE based P-PD | | |
|---|---|---|---|---|---|---|
| | $K_{p\text{-}o}$ | $K_{d\text{-}o}$ | $K_{p\text{-}i}$ | $K_{p\text{-}o}$ | $K_{d\text{-}o}$ | $K_{p\text{-}i}$ |
| Case-1 | 14.66 | 5.96 | 17.22 | 6.78 | 2.67 | 17.96 |
| Case-2 | 19.81 | 5.30 | 18.61 | 2.89 | 3.14 | 17.86 |
| Case-3 | 14.61 | 5.23 | 18.03 | 7.18 | 4.30 | 19.44 |
| Case-4 | 13.82 | 5.40 | 15.56 | 4.14 | 5.00 | 15.36 |
| Case-5 | 19.99 | 6.63 | 18.85 | 8.53 | 4.03 | 10.28 |
| Case-6 | 14.63 | 5.77 | 10.08 | 9.33 | 7.06 | 14.62 |

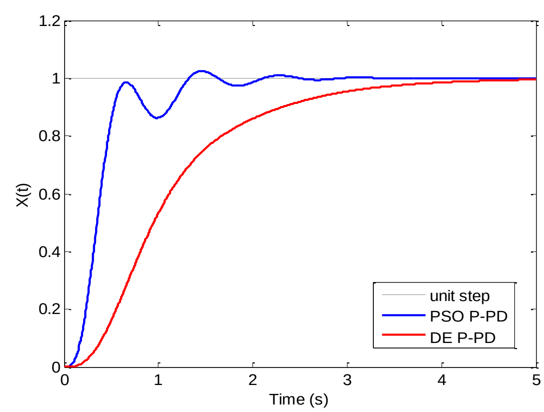


Fig.20. Performance comparison for 50% decrease of servo gain.

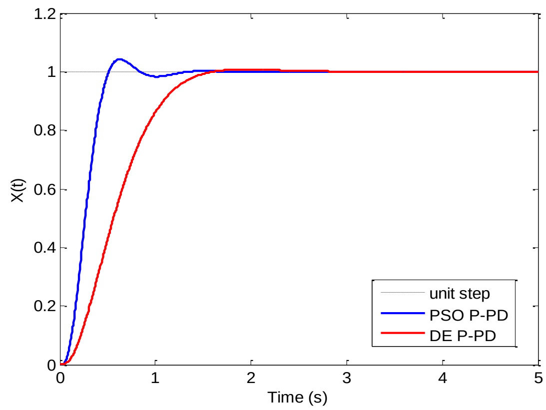


Fig.21. Performance comparison for 50% increase of servo gain.

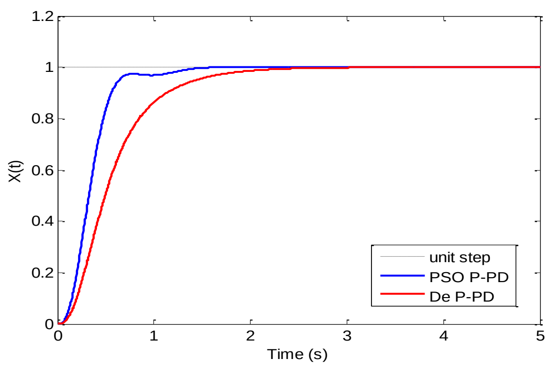


Fig.22. Performance comparison for 50% decrease of servo time constant.

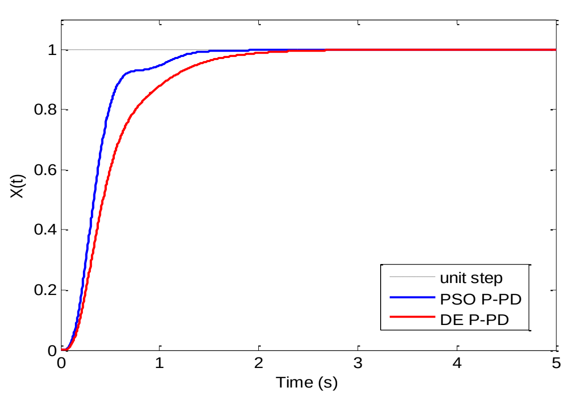


Fig.23. Performance comparison for 50% increase of servo time constant.

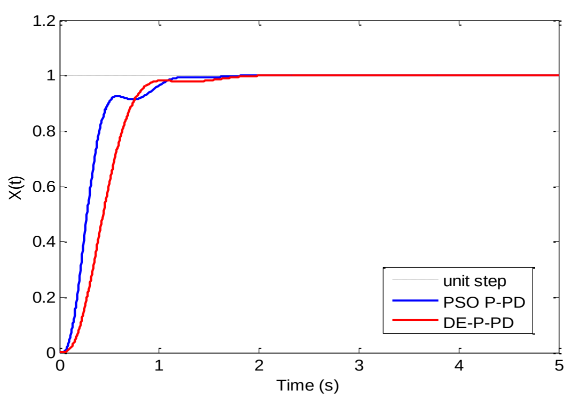


Fig.24. Performance comparison for 50% decrease of mass of the ball.

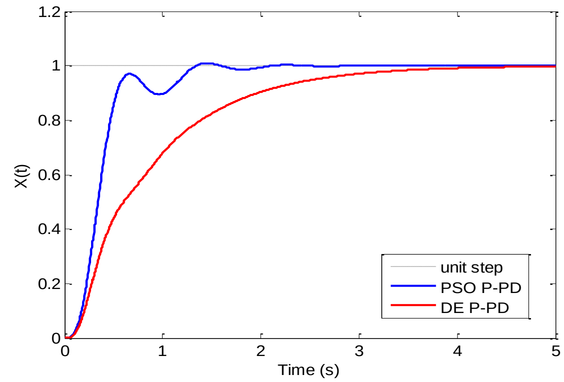


Fig.25. Performance comparison for 50% increase of mass of the ball.

TABLE VIII
UNIT STEP RESPONSE OF ROBUSTNESS ANALYSIS

| Cases | Maximum Overshoot (%) | | Settling Time (s) (5% band) | | Rise Time (s) | |
|---|---|---|---|---|---|---|
| | PSO based P-PD | DE based P-PD | PSO based P-PD | DE based P-PD | PSO based P-PD | DE based P-PD |
| Case-1 | 2.44 | **0** | **1.22** | 2.93 | **0.34** | 1.91 |
| Case-2 | 4.12 | **0.725** | **0.46** | 1.24 | **0.30** | 0.86 |
| Case-3 | 0 | 0 | **0.63** | 1.44 | **0.41** | 0.93 |
| Case-4 | 0 | 0 | **1.02** | 1.38 | **0.44** | 0.89 |
| Case-5 | 0 | 0 | 0.94 | **0.82** | **0.35** | 0.52 |
| Case-6 | 0.91 | **0** | **1.18** | 2.55 | **0.36** | 1.77 |

## V. Experimental Studies And Results

The controller parameters obtained by the simulation studies are applied to the real ball and plate system. The experimental hardware is represented in Fig. 26. The real one-axis tracking responses obtained by original controller and evolutionary computation based controllers are presented in Figs. 27, 28 and 29. In these figures, the black square curves are inputs and the red curves are system outputs. First of all the big magnitude oscillations are observed on all responses. Particularly, it is evaluated that they are particularly caused from the center point calibration error of the camera and inception angle errors of the plate. If the camera is not calibrated sufficiently to center point of the plate and/or if the plate is not fully parallel to horizontal plane, the controller will have to perform hysteresis behavior to keep the ball in balance, which results in various magnitude oscillations.

In this study, the tuning process is realized off-line by simulation studies. Then, the obtained controller parameters applied to the real ball and plate system. The experimental studies exhibit that the tracking responses of evolutionary computation based controllers are better than that of the original controller. Also, in fact that the response of PSO based controller has minimum oscillation means that this algorithm has better stability performance as observed from the simulation results before. On the other hand, the slopes of rising edges of these responses are investigated in order to compare the response speed of the actual results. Consequently, the slopes $m_P$ and $m_D$ related to PSO and DE algorithms are computed about 10.52 and 15.38 with respect to Equation 24, respectively as represented in Fig. 30. According to these results, the response speed of DE algorithm based controller is better about 20% than that of PSO algorithm based controller as presented in simulation studies.

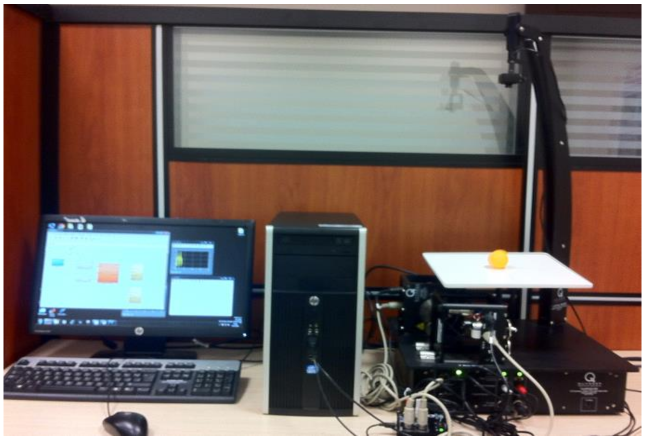


Fig.26. Ball and plate hardware for experimental study.

$$m = \frac{y_2 - y_1}{x_2 - x_1} \tag{24}$$

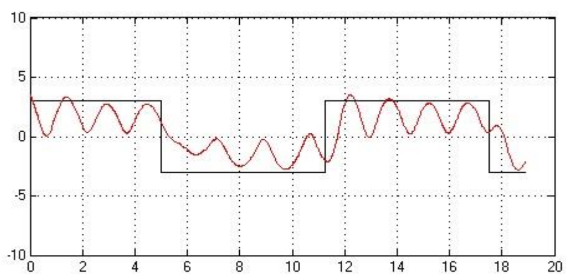


Fig.27. One-axis tracking response for experimental study of classic P-PD controller.

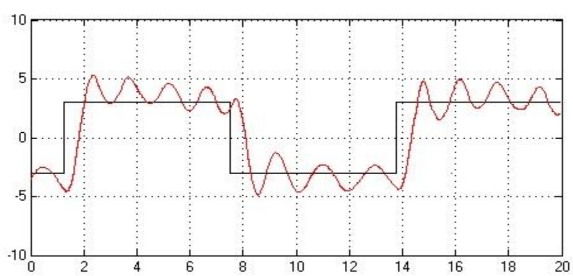


Fig.28. One-axis tracking response for experimental study of PSO based P-PD controller.

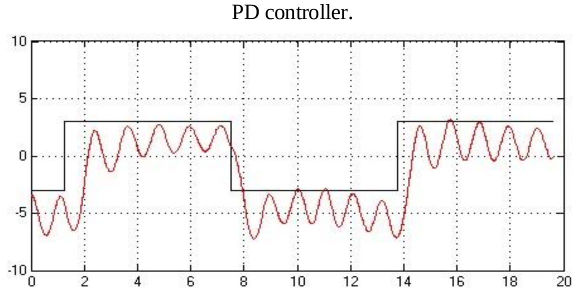
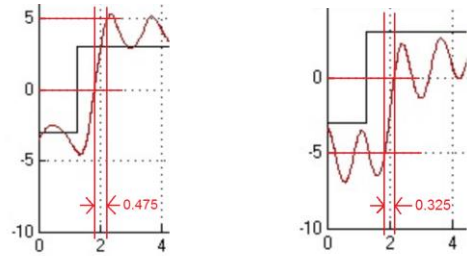


Fig.29. One-axis tracking response for experimental study of DE based P-PD controller.




(a) (b)

Fig.30. Slope of rising edge of tracking responses obtained from (a) PSO and (b) DE algorithms.

## VI. Discussion

The tuning performance of the evolutionary computation based optimization algorithms in gain scheduling control which is designed for ball and plate stabilization system is examined using time and frequency domains and robustness analyses. After that, the simulation results are verified by experimental studies. It is clear that a controller design is required compromise between stability and response speed. Nonetheless, the control performances of original and proposed controllers are investigated for both stability and response speed. In this point of view, the results of the analyses can be separated into system stability and response speed indicators as depicted in Figs. 31 and 32. Maximum overshoots and damping ratios which can be chosen as stability indicators show that although the damping ratio of original controller is a little big, the PSO based controller can work more stable than the original one for given operating point owing to its better overshoot performance and better settling time response informing about transition time to stable condition. On the other hand, the proposed gain scheduling controllers can be still stable when the system parameters change as evidenced by robustness analysis. Also, it can be said from Fig. 31 that the stability performance of PSO algorithm is better than DE algorithm due to smaller maximum overshoot and greater damping ratio.

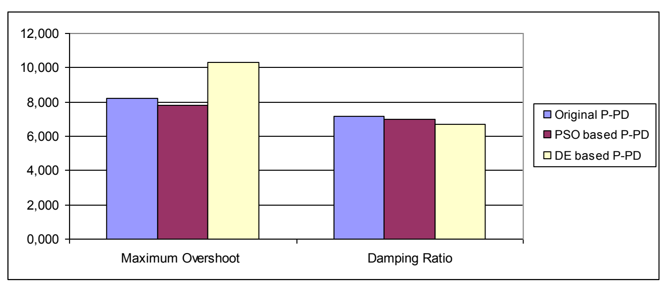



Fig.31. Results of system stability indicators.

When the response speed indictors such as settling time, rise time and bandwidth are examined, the results presented in Fig. 32 show that the proposed controllers respond faster about 50% than the original controller. Actually, this result is supported by the results of overshoot analysis. Also, the analysis shows that the response speed of DE algorithm is clearly better than that of PSO algorithm. Additionally, the DE algorithm based controller has minimum steady-state error as proved by ramp response analysis.

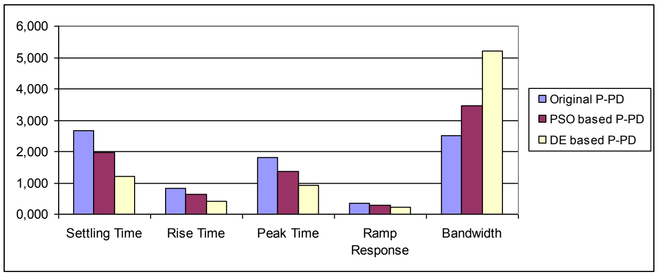



Fig.32. Results of response speed indicators.

On the other hand, the robustness analysis is performed to determine the behaviors of the proposed controllers in case of changing operating point caused by internal or external parameter variations. This analysis also exhibits interesting results about tuning performance of the chosen algorithms. According to them, the tuning performance of the swarm intelligence based PSO algorithm is more superior to that of DE algorithm based on evolution theory for different working conditions. It can be evaluated that this result is originated from better stability performance of PSO algorithm. The experimental studies largely verify these simulation results.

Finally, the evolutionary computation based optimization algorithms are successfully applied to the ball and plate stabilization system in order to tune the proposed gain scheduling P-PD controller parameters. The proposed controllers provide desired control system performances both for given operating point and for changing of system parameters. Also, it is proved by simulation and experimental studies that the results obtained by using these algorithms are better than the results belong to the original controller. In detail, while the swarm intelligence based PSO algorithm exhibits better stability performance, the evolutionary intelligence based DE algorithm indicates better response speed performance than the other.

## VII. CONCLUSION

In this study, the evolutionary computation based gain scheduling controller is designed in order to comparatively examine its control performance on ball and plate stabilization system. For this purpose, the swarm intelligence based PSO algorithm and the evolution theory based DE algorithm are chosen due to their better optimization performance than the others. The results taken by time and frequency domains and robustness analyses show that the proposed gain scheduling controller provides desired control system performances both for given operating point and for changing system parameters. In detail, PSO algorithm particularly exhibits better optimization performance in case of changing system parameters. The obtained simulation results are also verified by the experimental studies. On the other hand, designed controller is an off-line gain scheduling controller. In addition, the evolutionary computation based optimization algorithms can be easy embedded in a relatively simple microcontroller system due to their algebraic and short program codes. In the future study, the proposed gain scheduling controller will be implemented in a fully adaptive structure by designing an external microcontroller circuit.

## REFERENCES

[1] S. Leghmizi, S. Liu, "A Survey of Fuzzy Control For Stabilized Platforms", International Journal of Computer Science&Engineering Survey, vol.2, no.3, (2011).
[2] F.N. Barnes, "Stable Member Equations of Motion for a Three-Axis Gyro Stabilized Platform", IEEE Transaction on Aerospace and Electronic Systems, vol.7, no.5, 1971, pp. 830-842.
[3] T.H. Lee, E.K. Koh, M.K. Loh, "Stable Adaptive Control of Multivariable Servomechanisms with Application to Passive Line-of-Sight Stabilization System," IEEE Transactions on Industrial Electronics, vol. 43, no.1, 1996, pp. 98-10.
[4] J.A.R.K. Moorty, R. Marathe, "LQG/LTR Control Law for A Wideband Controller for Line of Sight Stabilization for Mobile Land Vehicles", Proceedings International Conference on Electro-Optics, 1999, pp.729–735.
[5] J.A.R.K. Moorty, R. Marathe, V.R. Sule, "H∞ Control Law for Line-of-Sight Stabilization for Mobile Land Vehicles", Society of Photo-Optical Instrumentation Engineers, Optical Engineering, vol.41, no.11, 2202, pp. 2935-2944.
[6] Y.J. Chen, S.H. Huang, W. Shanming, W. Fang, "DSP-Based Real-Time Implementation of A Neural Network Observer and Hybrid H∞ Adaptive Controller for Servo-Motor Drives", Proceedings of the 27th Chinese Control Conference, China, 2008, pp.130-134.
[7] F.R. Rubio, "Application of Position and Inertial-Rate Control to A 2-DOF Gyroscopic Platform", Robot Computer Integration Manufacturing, 2010, pp. 1-10.
[8] J.T. Spooner, K.M. Passino, "Stable Adaptive Control Using Fuzzy Systems and Neural Networks," IEEE Transactions on Fuzzy Systems, vol.4, no.3, 1996, pp. 339-359.
[9] P. Horacek, "Laboratory Experiments For Control Theory Courses: A Survey", Annual Reviews in Control, vol.24, 2000, pp. 151-162.
[10] X. Fan, N. Zhang, S. Teng, "Trajectory Planning and Tracking of Ball And Plate System Sing Hierarchical Fuzzy Control Scheme", Fuzzy Sets and Systems, vol.144, issue 2, 2004, pp. 297–312.
[11] M. Bai, H. Lu, J. Su, Y. Tian, "Motion Control of Ball and Plate System Using Supervisory Fuzzy Controller", Proceedings of the 6th World Congress on Intelligent Control and Automation, China, 2006.
[12] W. Hongrui, T. Yantao, F. Siyan, S. Zhen, "Nonlinear Control for Output Regulation of Ball and Plate System", Proceedings of the 27th Chinese Control Conference, China, 2008.
[13] M. Bai, Y. T. Tian, J. T. Su, "Position Control of Ball and Plate System Based on ANFIS", Complex Systems and Applications, vol.1, 2008, pp.165.
[14] X. Dong, Z. Zhang, J. Tao, "Design of Fuzzy Neural Network Controller Optimized by GA for Ball and Plate System", Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery,Vol.4, 2009, pp. 81-85.
[15] X. Dong, Z. Zhang, C. Chen, "Applying Genetic Algorithm to On-Line Updated PID Neural Network Controllers for Ball and Plate System", Fourth International Conference on Innovative Computing, Information and Control, 2009, pp.751-755.
[16] D. Casagrande, A. Astolfi, T. Parisini, "Switching-Driving Lyapunov Function and the Stabilization of the Ball-and-Plate System", IEEE Transactions on Automatic Control, vol.54, no.8, 2009, pp.1881-1886.
[17] M.A. Moreno-Armend´ariz, E. Rubio, C.A. P´erez-Olvera, "Design and Implementation of a Visual Fuzzy Control in FPGA for the Ball and Plate System", International Conference on Reconfigurable Computing, 2010, pp.85-90.
[18] X. Dong, Y. Zhao, Y. Xu, Z. Zhang, P. Shi, "Design of PSO Fuzzy Neural Network Control for Ball and Plate System", International Journal of Innovative Computing, Information and Control, vol.7, issue.12, 2011, pp. 7091-7103.
[19] K. Han, Y. Tian, Y. Kong, J. Li, Y. Zhang, "Tracking Control of Ball and Plate System Using An Improved PSO On-Line Training PID Neural Network", IEEE ICMA International Conference on Mechatronics and Automation, 2012, pp. 2297-2302.
[20] S. Mochizuki, H. Ichihara, "I-PD Controller Design based on Generalized KYP Lemma for Ball and Plate System", European Control Conference (ECC), Switzerland, 2013, pp.2855-2860.
[21] P. Roy, B. Kar, I. Hussain, "Trajectroy Control of a Ball In A Ball and Plate System Using Cascaded PD Controllers Tuned By PSO", Proceedings of Fourth International Conference On Soft Computing For Problem Solving, vol.2, 2014, pp.53.

[22] Y. Zhao, Y. Ge, "The Controller of Ball And Plate System Designed Based On FNN", Journal of Chemical and Pharmaceutical Research, vol.6, issue.6, 2014, pp. 1347-1352.
[23] K.W. Han, Y.T. Tian, Y.S. Kong, B.H. Zhao, C. Li, "Fuzzy Indirect Adaptive Control For Ball And Plate System", Kongzhi yu Juece/Control and Decision, vol.30, issue.2, 2015, pp. 303-310.
[24] M., Oravec, A. Jadlovska, "Model Predictive Control of a Ball and Plate laboratory Model", SAMI 2015 - IEEE 13th International Symposium on Applied Machine Intelligence and Informatics, 2015, pp. 165-170.
[25] A., Negash, N.P. Singh, "Position Control And Tracking Of Ball And Plate System Using Fuzzy Sliding Mode Controller", Advances in Intelligent Systems and Computing, vol.334, 2015, pp. 123- 132.
[26] J. Xiao, G. Buttazzo, "Adaptive Embedded Control for a Ball and Plate System", Proceedings of The Eighth International Conference on Adaptive and Self-Adaptive Systems and Applications IARIA, 2016, pp. 40-45.
[27] Quanser Inc., "2-DOF Ball Balancer Experiment for MATLAB/Simulink Users (Student Workbook)", 2013.
[28] Quanser Inc., "Rotary Servo Base Unit Workbook (Student Workbook)", 2011.
[29] D.J. Leith, W.E. Leithead, "Survey of Gain-Scheduling Analysis Design", International Journal of Control, vol.73, 1999, pp. 1001-1025.
[30] J. Kennedy, R.C. Eberhart, "Particles Swarm Optimization", IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.
[31] R. Storn, "On the Usage of Differential Evolution for Function Optimization", Proceedings Fuzzy Information Processing Society, Biennial Conference of the North American, 1996.
[32] S. Das, P.N. Suganthan, "Differential Evolution: A Survey of the State-of-the-Art", IEEE Transactions on Evolutionary Computation, vol.15, issue.1, 2011, pp. 4-31.

## BIOGRAPHIES

**HALUK GOZDE** received the B.Sc. degree in Electrical and Electronics Engineering from Karadeniz Technical University, Trabzon, Turkey, in 1997, and the M.Sc. and the Ph.D. degrees in Electrical and Electronics Engineering from Gazi University, Ankara, Turkey, in 2004 and 2010, respectively. Then, he is an Assoc.Prof. of the electrical and electronics engineering in National Defense University since 2016. His main research area consists of power system dynamics and control, artificial intelligence-based control methods, swarm intelligence-based optimization algorithms, autonomous mobile robots, swarm robotics and autonomous path planning algorithms.