Dhaka Univ. J. Sci. 59(2): 181-186, 2011 (July)

Designing of Controller: a Multi-objective Genetic Optimization Approach

M. A. Habib, M. M. Islam and M. S. Alam

Department of Applied Physics, Electronics and Communication Engineering, Dhaka University, Dhaka-1000, Bangladesh

Received on 24. 11. 2009. Accepted for Publication on 22.03.2011

Abstract

Controller design for any system requires multiple objectives to be minimized or optimized. This involves the satisfaction of multiple objectives which are conflicting such as Rise time and Maximum overshoot. It is difficult to meet these two conflicting objective simultaneously using conventional design methods. This paper presents an investigation on design and development of Multi-Objective Genetic algorithm (MOGA) based controller to achieve that goal. Two aerial vehicle systems have been investigated in this work. Performance of conventional technique such as Ziegler-Nichols (Z-N) method has been compared and analyzed with the intelligent tuning technique MOGA. Simulation results confirm the superiority of this MOGA approach over conventional techniques. Moreover, it is proposed that if an objective value does not satisfy its corresponding goal value, a penalty can be imposed onto its competing objective in proportion to the extent of violation. This approach enhance the generation of the desired goal solutions.

Keywords: Multi-objective Genetic Algorithms, PID, Pitch controller, Twin rotor system

I. Introduction

Optimization refers to finding one or more feasible solutions which correspond to extreme values of one or more objectives. Aerial vehicle system is subjected to variation in parameters and parameter perturbations, which when significant makes the system unstable. So the control engineers are looking for automatic but effective tuning procedures. Proportional-Integral-Differential (PID)[1] controllers are widely used in such systems because it is simple and robust. Designing of a PID controller involves three separate parameters; Proportional gain (K_p), Integral gain (K_i) and the Derivative gain (K_d). Transfer function of a PID controller[1] in Laplace transforms form is as follows:

$$G_o(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s \text{ ......(1)}$$

The challenge behind designing such controller is the adjustment of its control parameters to the optimum values for the desired control response. There are several methods for tuning a *PID controller*. Some of them are Mathematical criteria, Cohen-coon Method, Trial and error method, Continuous cycling method, Relay feedback method, Ziegler-Nichols (Z-N) method and Kappa-Tau tuning method.[2] Other than these techniques there are soft computing techniques which are proved to be more simple and reliable.

Controller design of a system naturally involves the satisfaction of multiple performance measures, or objectives. Such search problems can often benefit from an effective use of parallelism, in which many different possibilities are explored simultaneously in an efficient way. Multi-objective Genetic Algorithm (MOGA)[3] has this ability to find multiple optimal solutions in one single simulation run. From the above mentioned methods, one manual tuning method, Z-N method, and one soft tuning method using MOGA, have been selected and the obtained results are compared. In this work, a modified version of conventional MOGA algorithm is also proposed to improve the performance of searching ability of optimization process.

II. Multi-Objective Genetic Algorithm

MOGA is a stochastic optimization algorithm inspired by the evolutionary biology such as inheritance, mutation, selection, and crossover. Binary-coded Genetic algorithms are implemented as a computer simulation in which a population of abstract representations (called chromosomes or the genotype) of candidate solutions (called individuals or phenotypes) to an optimization problem evolves toward better solutions. For a given set of solutions, those members which are not dominated by any other member of the set belong to an elite solution set called non-dominated solutions. When the set of solutions is the entire search space, the resulting non-dominated set is called Pareto-optimal set[1]. The steps involved in MOGA optimization are as follows[3]:

- Create a random set of solutions, called population.
- Each individual of the population is evaluated using the objective functions.
- Non-dominated solutions are searched and stored.
- Each individual is ranked according to their degree of dominance.
- Fitness to a solution is assigned based on its rank and diversity.
- The fittest individuals are selected.
- Pair-up bias the way in which individuals are paired for Reproduction[4]
- Create one or two new offspring.
- The mutation operator makes small, random changes to the genetic coding of the individual.
- If the termination criteria reached, the process ends. Otherwise the cycle continues.

III. Aerial Vehicle Systems

The proposed control scheme has been tested in simulation environment on two aerial vehicle systems, namely, 1) Twin Rotor MIMO system and 2) Pitch Controller of an aircraft. A brief description of each system is provided below:

Twin Rotor MIMO System (TRMS)

Twin Rotor MIMO system (TRMS)[5] is a laboratory set-up designed for control experiments. The TRMS consists of a beam pivoted on its base in such a way that it can rotate freely both in the horizontal and vertical planes. At both ends of the beam there are rotors (the main and tail rotors) driven by DC motors. The main rotor moves the system in the vertical plane and the tail rotor moves the system in the horizontal plane. A counterbalance arm with a weight at its end is fixed to the beam at the pivot. The pivot point allows the helicopter to move simultaneously in both the horizontal and vertical planes. It is said to have two degrees-of-freedom (DOF). The measured signals are: position of the beam, that is, two position angles, and angular velocities of the rotors. The schematic diagram of TRMS[5] is shown in figure 1. The transfer function characterizing the vertical movement[6] is:

$$\frac{Y(s)}{U(s)}=\frac{-0.08927s^3+2.249s^2+45.57s+595.1}{s^4+3.469s^3+519.6s^2+35.98s+2189}\text{......(2)}$$

Where U(s) represents the main rotor input (volt) and Y(s) represents pitch angle (radians)

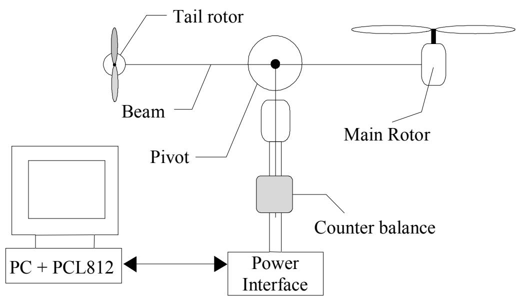

Fig. 1. Twin rotor MIMO system

Pitch Controller

Figure 2 shows the schematic diagram of an airplane. Pitch is the up and down motion of an airplane. The pitching motion is being caused by the deflection of the elevator of this aircraft. The elevator is a hinged section at the rear of the horizontal stabilizer. There is usually an elevator on each side of the vertical stabilizer. To climb, the elevators are put in the up position. This pushes the tail down and the nose up. To dive, the elevators are put in the down position. This pushes the tail up and the nose down. The transfer function of Pitch Controller

$$\frac{\theta(s)}{\delta_e(s)}=\frac{1.151s+0.1744}{s^3+0.739s^2+0.921s}\text{..........(3)}$$

The input (elevator deflection angle, δ_e) will be 0.2 rad (11 degrees), and the output is the pitch angle (theta).

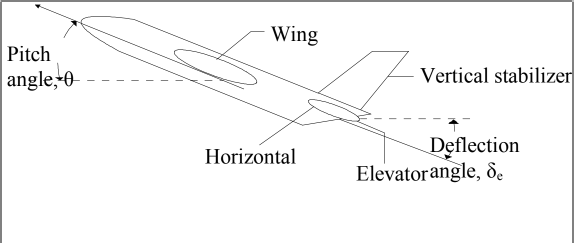

Fig. 2. Schematic diagram of an airplane

IV. Proposed Control Scheme

A MOGA-based PID controller is proposed to meet the design objectives and associated goals as demanded by an aerial vehicle system. Figure 3 shows the conceptual representation of the MOGA-based PID controller. Initially a random population of PID controller parameters is created. Each individual is sequentially considered at a time for configuring the controller. A unit step input is fed into the closed loop system and the resultant response is recorded. A detail time domain analysis on the response gives the values of objectives such as rise time, overshoot, settling time etc. Based on this performance measuring characteristic fitness is assigned to each individual.

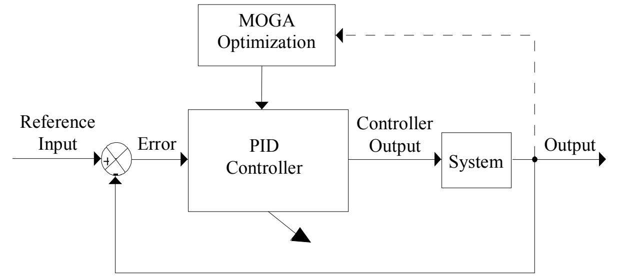

Fig. 3. Schematic diagram of MOGA based PID controller

The selection operator sorts out good solutions according to the fitness values. Multiple copies of these solutions are created which replace solutions having low fitness value. The population of the solutions is then modified by two genetic operators, recombination and mutation, and a new (better) population is created. At this stage one generation is completed and the next cycle is initiated. The cycle is repeated for several times and better offspring are produced at the end of each cycle. All MOGA routines and several objective functions are implemented in Matlab.[7 - 8]

Parameter Encoding

The initial population consists of a number of solutions (individuals) generated at random and distributed uniformly in a predefined range. Each solution consists of three separate strings representing three parameters: K_d, K_p and K_i. Parameter encoding can be elaborated in following three steps:

Step-1: Randomly generated binary codes of dimension 50×3×20 are created, where 50 represents the number of individuals, 3 the number of parameters and 20 the number of bits to form one parameter to ensure sufficient precision for the design procedure.

Step-2: The binary strings are considered as Gray code and converted to decimal numbers.

Step-3: The three parameters are designated as Kd, Kp and Ki of the PID controller respectively.

MOGA Optimization process

The optimization process consists of standard generational GA with multi-objective ranking, fitness sharing and mating restriction evaluated in the objective domain[3]. After evaluating different objective functions, multi-objective ranks are computed according to the current preferences. The fitness assigned to individuals with the same multi-objective rank is averaged, and fitness shared within each rank before selection takes place. Selection uses Baker's stochastic universal sampling (SUS)[9] algorithm, which is optimal in terms of bias and spread. Once the parents of the next generation have been selected from the old population, they are paired up and recombined with high probability (0.8). Mating restriction is implemented by forming pairs of individuals within a distance of each other in the objective space, where possible. Reduced-surrogate shuffle crossover[10] is used for recombination. The mutation rate for this optimization process was set at 0.01%.

Objectives and Goal values

Case 1: TRMS

The goal value of Objective-1 corresponds to reduction in rise time of 90% lower than that of Z-N tuned system. The goal value of maximum peak from steady level (objective-2) was chosen 10% higher than that with step response of Z-N tuned system. Table 1 shows the objective functions and associated goal values used in the optimization process.

Table. 1. Two objective and goal values

Objective	Parameter	Goal value
Objective-1	Rise time	≤5sec
Objective-2	Maximum Overshoot	≤10%

Case 2: Pitch Controller

MOGA optimization process as discussed earlier is extended to include two more performance measures, namely, a) settling time, and b) Steady state error to form a four-objective optimization problem. The aim is to investigate the relationship among these objectives, and the extent to which and if they are conflicting or non-conflicting with one another. The outcome may provide important information for the design procedure and parameters that may be taken into consideration on priority basis. The objective functions and corresponding goal values are shown below:

Table 2. Four objective and Goal values

Objective	Parameter	Goal value
Objective-1	Rise time	≤ 2sec
Objective-2	Maximum Overshoot	≤ 10%
Objective-3	Settling time	≤ 10sec
Objective-4	Steady State Error	≤ 2%

Goal Value Generation Enhance Technique

Constraint handling[11] is one of the major concerns when applying MOGA to solve constrained optimization problems. TRMS is not a constrained optimization problem so there is no need to employ constraint handling mechanism in the algorithm. But in order to improve the number of solutions satisfying goal values in the final solution set, a similar strategy of constraint handling is employed. The goal value chosen for TRMS is shown in Table 1. The approach is to impose penalty onto solution that does not satisfy the goal value. The penalty imposed depends on the extent to which the performance of the objectives violates the goal value. Such penalty values are added before fitness is computed. Since this mechanism modifies the objective value, special precaution is taken so that those solutions do not make to the non-dominated set. Otherwise they would be presented in the objective space with a wrong value. After evaluation of the objectives for a solution, the solution is compared with a value slightly greater than the goal value. For maximum overshoot and rise time the critical value chosen is 20 and 10 sec respectively. Penalty is imposed to those solutions which have at least one objective value that does not satisfy its corresponding critical value. The mathematical formulation of this mechanism is shown below

For objective functions $f_1(x)$: Overshoot & $f_2(x)$: Rise time, where x ∈ solution set

$$penalty:\begin{cases} f_1(x) > 20,\ f_2(x) = f_2(x) + f_1(x) - 20 \\ f_2(x) > 10,\ f_1(x) = f_1(x) + f_2(x) - 10 \end{cases} \quad ...(4)$$

In TRMS control, two conflicting design objectives are considered while finding the parameters of PID controller.

When one of the objective values violates its corresponding critical value, an extra penalized cost is added to the performance value of the other rival objective. This prevents the solution from getting selected to the non-dominated set. The extra penalized cost is large enough to cause all these solution to have a worse fitness value than that of the other solution whose performance are near to the goal value. Thus the mechanism enhances the formation of goal solutions.

V. Results

Case 1: TRMS

MOGA with a population size of 50 individuals was run on this problem 5 times, each time for 50 generations. Non-dominated solution, termed as 'ND-sol' and solutions which are both non-dominated and satisfying all the goal values as 'G-sol' were recorded. Average values of ND-sol and G-sol after every 10 generations are shown in figure 4. After generation 10, out of a total of 2500 solutions, evaluated in each run, only 11 were non-dominated relative to each other, of which, 4 solutions are satisfying. The numbers of both ND-sol and G-sol slightly change in following generations as shown in figure 4. The Pareto optimal set of run-5 for TRMS is shown in figure 5. To validate the solution sets, three solutions are selected on the Pareto front, two on either extreme point of the two objectives, the other at approximately middle. The responses of PID controller with solutions obtained from Z-N method and proposed technique are shown in figure 6. It is clearly evident from figure 6 that the responses obtained using MOGA are far better than Z-N tuned PID control system. Moreover, it is also observed that different solutions (sol-1, sol-2 and sol-3) trade-off between two conflicting design objectives.

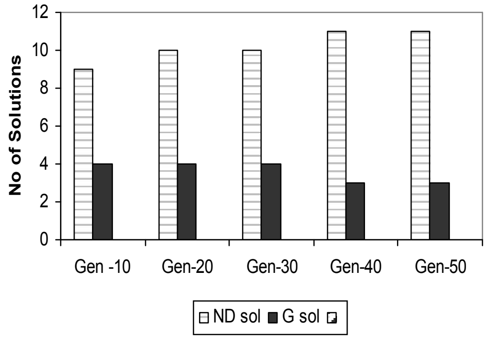

Fig. 4. Non-dominated and satisfying solution set for TRMS

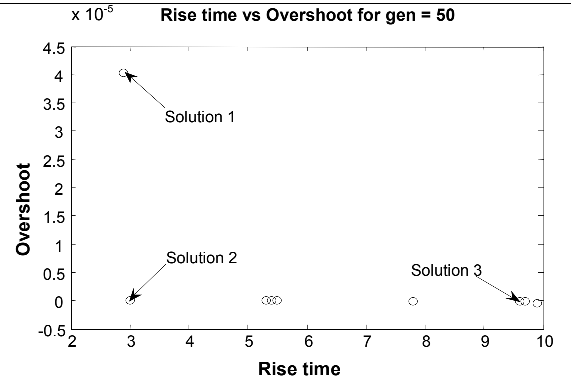

Fig. 5. Pareto optimal set for TRMS

Case 2: Pitch Controller

A GA with a population size of 50 individuals is run on this problem 10 times, each time for 50 generations. Average values of ND-sol and G-sol after every 10 generations are shown in figure 7. It is evident from the figure that the number of preferable solutions gradually increases as the algorithm proceeds. Out of a total of 2500 points evaluated in each run, around one-eighth is non-dominated relative to each other. Of these, more than 137 points are satisfying.

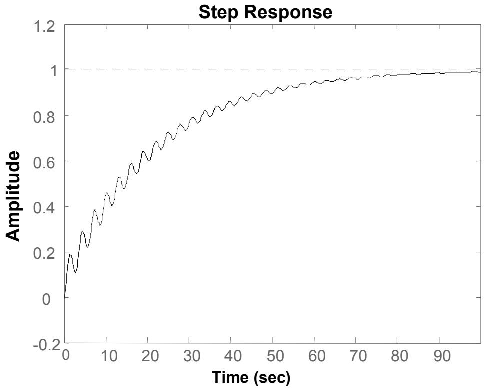

Fig. 6(a). Step response of Z-N tuned PID control

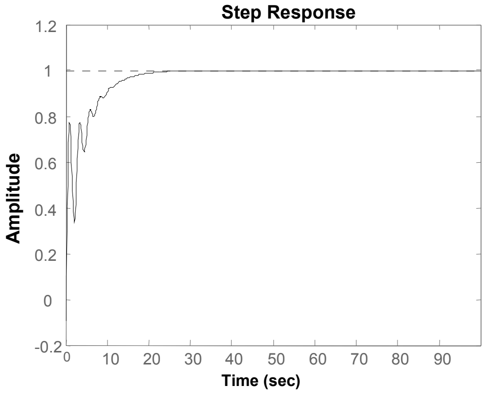

Fig. 6(b). Step response for sol-1 values

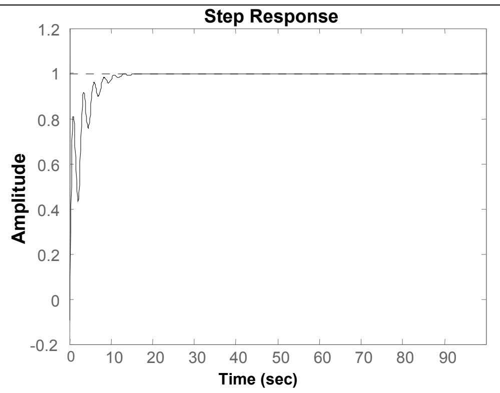

Fig. 6(c). Step response for sol-2 values

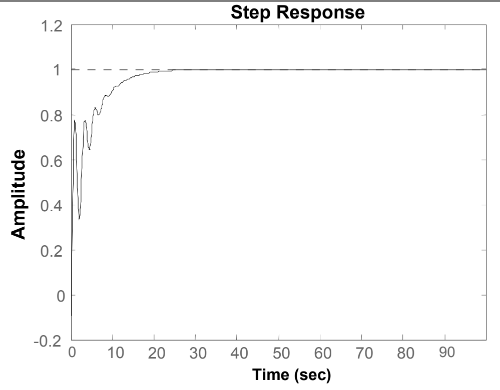

Fig. 6(d). Step response for sol-3 values

To highlight the competing features of different objectives, the same solutions are redrawn with normalized cost function along the y-axis. The preferable solutions found at the end of the run reveal tradeoffs between several objectives within the bounds imposed by the goals associated with them. A trade-off between adjacent objectives results in the crossing of lines between them, whereas nonconcurring lines indicate that those objectives are non-competing. It is evident from figure 8 that only settling time (objective-3), steady state error (objective-4) appears to be non-competing whereas the other two objectives are heavily competing.

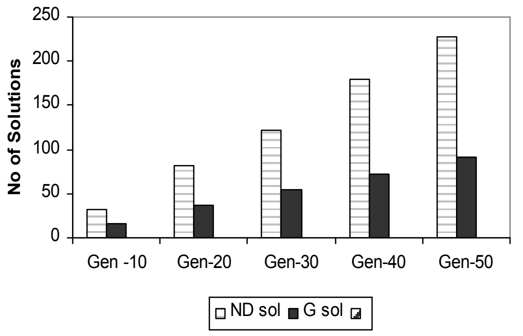

Fig. 7. Non-dominated and goal solution for Pitch Controller

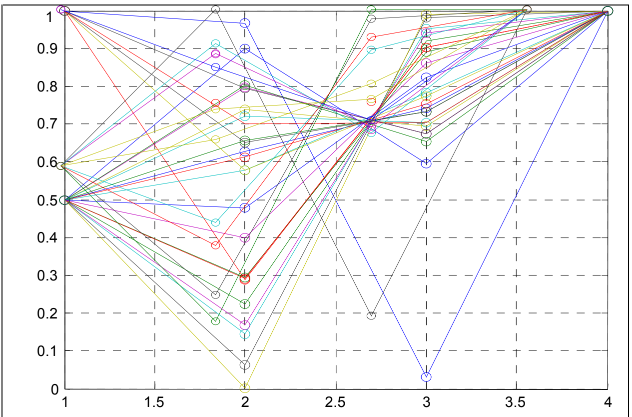

Fig. 8. Selected non-dominated solutions after generation -50 (cost function in normalized form along y-axis, design objectives along x-axis)

For example, rise time (objective-1), overshoot (objective-2) seem to compete heavily. There is a clear trade-off between these two objectives overshoot (objective 2) and settling time (objective-3). To prove the effectiveness of the four-objective optimization procedure in designing Pitch Controller, one non-dominated solutions, termed as sol-1 is selected and tested on the system. Performance of the solution is presented in figure 9. It is evident from the figure that the solution yield a very low rise time at the cost of very small overshoot and Steady state error is almost negligible.

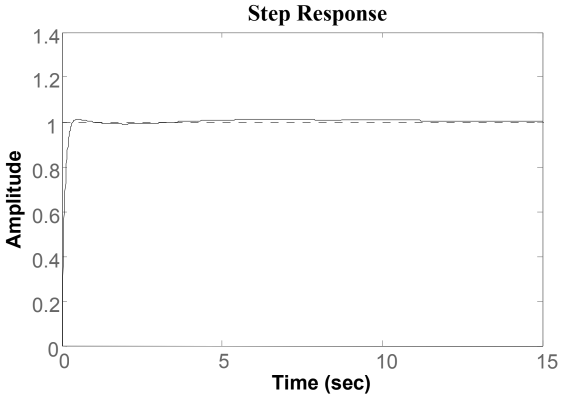

Fig. 9. Step response for sol-1 values

Performance of the New Goal Value Generation Enhance Technique

Comparison is made between the numbers of goal value generated with and without the incorporation of the new technique in the MOGA algorithm as shown in figure 10.

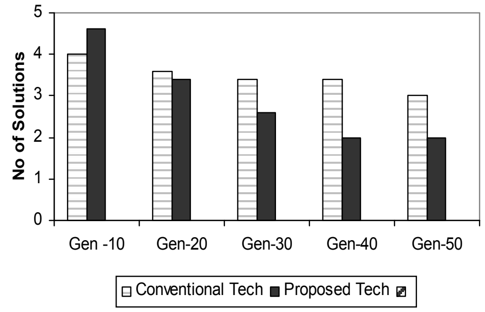

Fig. 10. Goal solutions for TRMS

It is evident that the number of preferable solutions after each 10 generation is greater for MOGA with the new technique incorporated than that which is not, suggesting the new technique help to generate more goal values.

VI. Conclusion

This paper has presented a MOGA-based control design method for aerial vehicle system where multiple conflicting design objectives are either to be met simultaneously or traded-off within acceptable limits. PID controller is chosen for its proven effectiveness in aerial vehicle systems and MOGA is used to find suitable sets of controller parameters that trade-off design objectives and satisfy associated goal values. TRMS controller design presents a two-objective optimization problem whereas Airplane pitch controller design is a four-objective one. Z-N method is widely used in designing and tuning PID controller whose performance has been compared with the proposed technique for same systems. In case of TRMS, the proposed technique for a PID controller considerably reduced the overshoot and rise time.

In this paper a new technique to enhance the generation of goal solution after each generation is also proposed. Simulation results confirm the superiority of this new approach over standard MOGA techniques.

The MOGA algorithm for PID controller tuning presented in this work offers several advantages. One can use a high-order process model in the tuning, and the errors resulting from model reduction are avoided. It is possible to consider several design criteria in a balanced and unified way. Approximations that are typical to classical tuning rules are not needed. Soft computing techniques are often criticized for two reasons[2]: algorithms are computationally heavy and convergence to the optimal solution cannot be guaranteed. PID controller tuning is a small-scale problem and thus computational complexity is not really an issue here. Compared to conventionally tuned system, MOGA tuned system has good steady state response and performance indices.

……

1. Ogata, K., 1995, *Modern Control Engineering*, 2nd ed., Prentice Hall of India.
2. Nagaraj, B., S. Subha and B. Rampriya, 2008, Tuning Algorithms for PID Controller Using Soft Computing Techniques, *IJCSNS International Journal of Computer Science and Network Security*, **8**(**1**), 278-281.
3. Fonseca, C. M. and P. J. Fleming, 1993, Genetic algorithms for multiobjective optimization: formulation, discussion and generalization, *Genetic Algoritms: Proceeding of the Fifth International Conference,* 416-423, San Mateo, CA.
4. Goldberg, D. E., 1989, *Genetic algorithms in search, optimization and machine learning*, Addison Wesley Longman, Publishing Co. Inc., New York.
5. Alam, M. S. and M. O. Tokhi, 2008, Dynamic Modeling of a Twin rotor System: a Genetic Algorithm Optimization Approach, *Dhaka University Journal of Science*, **56**(2)183-187.
6. Feedback Instruments Ltd, 1996, *Twin Rotor MIMO System Manual 33-007-0*, Sussex, UK.
7. Chipperfield, A., P. Fleming, H. Pohlheim and C. Fonseca, 1994, Genetic algorithms toolbox user's guide, *Research Report 512, Department of Automatic Control and Systems Engineering, University of Sheffield*, Sheffield, UK.
8. *MATLAB Reference Guide,* The Math Works, Inc., 2006
9. Baker, J. E., 1987, Reducing bias and inefficiency in the selection algorithm, *Proceedings of the Second International Conference on Genetic Algorithms,* Morgan Kaufmann Publishers, Inc., 14-21, San Mateo, CA.
10. Booker, L.B., 1987, Improving search in genetic algorithms, Davis, L. (Ed.), *Genetic Algorithms and Simulated Annealing,* Morgan Kaufmann, 61-73, Los Altos, CA.
11. Deb, K., 2001, Multi-Objective Optimization using Evolutionary Algorithms, 1st ed., John Wiley and Sons Ltd.