A Genetic Approach for Real-Time Identification and Control of a Helicopter SYSTEM

K. C. Tan†, C. Y. Cheong and Y. Peng

Department of Electrical and Computer Engineering, National University of Singapore
4 Engineering Drive 3, Singapore 117576
†*E-mail: eletankc@nus.edu.sg*

Abstract: *This paper presents a genetic approach for real-time identification and control of a helicopter plant. The proposed approach employs a dual-stage strategy to keep track of the dynamic system as well as to optimize the corresponding control parameters in the real-time control system. In order to evaluate the true potential of the online GA control, experiments are conducted on a 2-degree-of-freedom helicopter plant and the overall control system is implemented on the dSPACE DS1103 DSP controller board. Through different experiment setups, the robustness and effectiveness of the proposed evolutionary technique for online identification and control are validated.*

Keywords: *Genetic algorithm, parameter optimization, system identification, real-time control*

1. INTRODUCTION

Genetic algorithms (GAs) are a class of stochastic search techniques inspired by the principles of evolution and natural genetics. With a selection procedure based on the Darwinian principle of "survival of the fittest" and a recombination process that entails the exchange of information, GAs are recognized to have properties such as robustness, parallelism, and the ability to sample multiple potential solutions simultaneously. Therefore, it is not surprising that genetic techniques for real-world problems [2], [3], [13], [14], [16] have been gaining significant attention from researchers in various fields.

In the context of control system design, GAs are finding widespread applications in the design of control systems through parameter optimization [6], [7], [10], [15], [17], [19], [20]. The application of GAs to control engineering can be broadly classified into two main areas: 1) offline design and analysis and 2) online adaptation and tuning. In the case of offline applications, GAs can be employed as a search and optimization engine. For example, suitable control laws can be evolved for a known plant to satisfy the given performance criteria or to search for optimal parameter settings for a particular controller structure. On the other hand, genetic techniques may be used as a learning mechanism to identify characteristics of unknown or dynamic systems in the case of online applications.

Kristinsson and Dumont [8] conducted the initial research for applying standard GAs to online system identification. The algorithm is tried on a real plant, a tank system that has controllable inflow and measurable water height. Specifically, the GA is used to estimate the poles and zeros of the tank system with experimental data obtained online. Based on the estimates, an adaptive pole placement controller is then designed. However, they were not able to carry out any online control on the tank.

Lennon and Passino [9] studied a general genetic adaptive control system (GGAC) which essentially uses GAs for both identifying the plant model and tuning of the controller at the same time. Their particular application is the cargo ship steering control. In this application, GA evolves the model parameters by looking at the past input-output values and attempting to minimize the error between the cargo ship heading and the output of the cargo ship model. The best cargo ship model is passed directly to the genetic adaptive controller for its fitness evaluation of the population of controllers. Fixed plant models and controllers are incorporated into the population as a guarantee policy. On the other hand, assessment of the scheme is conducted by means of simulation.

According to Linkens and Nyongesa [11], there are three general approaches to the use of GAs for online control optimization: (1) by utilizing a process model, (2) by utilizing the process directly, and (3) by permitting restricted tuning of an existing controller. It should be noted that very few applications involving actual real-time use of GAs for control have been reported so far [4]. In one of such rarity, Ahmad *et al* [1] used GAs for online tuning of a Proportional-Integral (PI) controller for the temperature regulation in a heating system. Their stated objective is to achieve the desired temperature as quickly as possible with minimum or no overshoot and results are presented for both time-invariant and time-variant cases. In the latter case, the plant model is updated after a number of samples using a recursive least square (RLS) estimator. A single generation of GA, using a plant model, is evaluated between the sampling intervals. The best solution found for that generation is allowed to control the real plant. However, there are concerns about the computation limitation between samples which further confines the application of GAs to process control.

In this paper, a dual-stage genetic strategy is presented to keep track of the dynamic system as well as to optimize the corresponding control parameters in the real-time control system. The first stage is the genetic identification loop which evolves the system model that is used to evaluate the fitness of the candidate controllers. These controllers are actually evolved in the genetic control-tuning loop and the best controller will be implemented for real-time control. In order to validate the potential of GAs for real-time control applications, the proposed real-time genetic identification and control architecture is applied to a 2-degree-of-freedom (2DOF) helicopter plant.

The organization of the paper is as follows. Section 2 provides a brief overview of the 2DOF helicopter plant used to validate the performance of real-time genetic control. This is followed by the description of the system architecture as well as the real-time genetic identification and control stages in Section 3. In Section 4, experiments are conducted on the 2DOF helicopter plant to examine the performance of the proposed genetic approach. Conclusions are drawn in Section 5.

2. 2-DEGREE-OF-FREEDOM HELICOPTER CONTROL SYSTEM

The control objective of this paper is to design a controller that is capable of (1) stabilizing the hovering helicopter model and (2) providing set-point tracking in the presence of modeling uncertainty. Therefore, the first issue to be addressed is the identification of a linear time-invariant (LTI) model which captures the main dynamical features of the hovering helicopter.

The test bed used in the experiment is a 2DOF flight simulator consisting of a helicopter model mounted on a fixed base. The helicopter model has two propellers driven by DC motors. The pitch propeller and the yaw propeller are used to control the pitch and yaw of the model, respectively. Motion about the two degrees of freedom is measured using two encoders. The control objective is to command a desired pitch and yaw angle. The coupling between the pitch and yaw motor torques results in a coupled 2-input-2-output system. Electrical signals and power from the pitch encoder and the motors are transmitted via a slip ring, which allows for unlimited yaw and eliminates the possibility of wires tangling on the yaw axis.

Consider the 2DOF diagram shown in Figure 1. The pitch propeller is driven by a DC motor whose speed is controlled via input voltage V_p. The speed of rotation results in a force

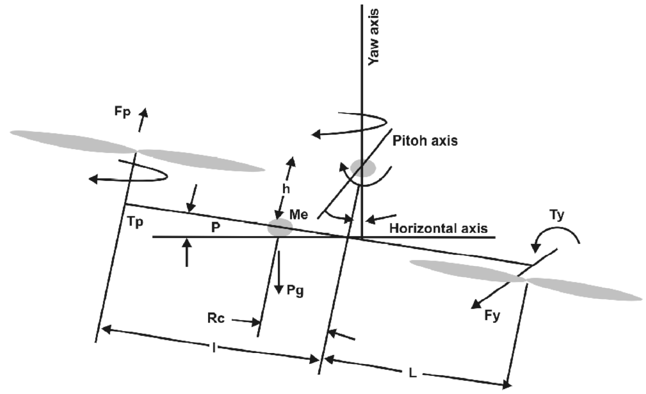


Figure 1: Transformation Frames of 2DOF Helicopter

that acts normal to the body at a distance R_p from the pitch axis. The rotation of the propeller, however, also causes a load torque T_p on the motor shaft which is in turn seen at the yaw axis (parallel axis theorem). Thus, rotating the pitch propeller does not only cause motion about the pitch axis but also about the yaw axis. Similarly, the yaw motor causes a force F_y to act on the body at a distance R_y from the yaw axis as well as a torque T_y which is experienced about the pitch axis. The simplified equations of motion are then given by,

$$\begin{aligned} J_{pp}\,\ddot{p} &= R_p F_p + G_p(T_y) + G_d(p) \\ J_{yy}\,\ddot{y} &= R_y F_y + G_y(T_p) \end{aligned} \tag{1}$$

where p and y are the pitch angle and yaw angle; F_y and F_p are the forces generated by the propellers; R is the horizontal distance of the center of mass from the pivot point; T_y and T_p are the torques at the propeller axes; G_y and G_p are the nonlinear functions representing the coupling; G_d is a gravitational constant disturbance; J_{pp} and J_{yy} are the moments of inertia of the body about the pitch and yaw axes, respectively.

3. REAL-TIME GENETIC IDENTIFICATION AND CONTROL

3.1. System Architecture for Real-Time Genetic Identification and Control

The real-time identification and control architecture of the 2DOF helicopter is based on the general genetic adaptive controller (GGAC) developed by Lennon and Passino [9]. The system architecture is represented by the block diagram in Figure 2. The system can be viewed as having two layers. Starting from the first layer, we have a typical control feedback loop comprising of the helicopter plant and its decoupled PID controllers. Adaptation and optimization of control parameters are achieved by means of the genetic learning layer. This layer is composed of two concurrent GA loops: (1) the identification loop which evolves the best model for the real-time plant and (2) the control-tuning loop which evolves the PID controller parameters based on the evolved plant model.

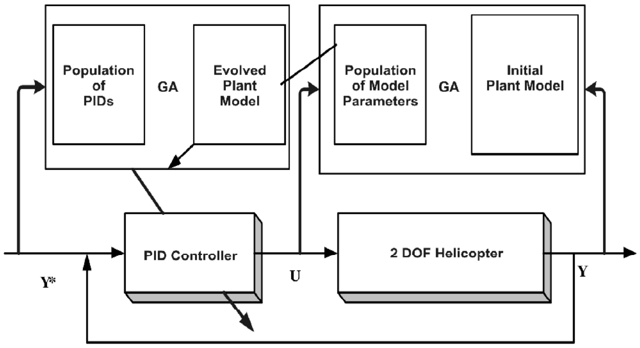


Figure 2: Block Diagram of Real-time GA Adaptive Control System

3.2. Genetic Identification

The evolved model from the identification loop is used for the evaluation of candidate controller parameters in the GA control-tuning loop. The use of a simulation model is necessary because real system evaluation is not possible in this case. Intuitively, the fundamental issue here is the accuracy of the evolved model as well as the amount of effort devoted to the optimization process. In this paper, identification is performed online and the simulation model is replaced whenever substantial difference between the current plant model and best model in the evolving population is detected.

The genetic identification procedure takes the measurements of two motor voltage inputs, V_p and V_y, and the output degrees of pitch and yaw motion, p and y. The purpose of genetic modeling here is to identify a linear time-invariant (LTI) model, which captures the main dynamical features of the hovering helicopter. Since the helicopter has significant cross coupling, a state-space multi-input-multi-output (MIMO) identification method is preferred. Linearization of standard nonlinear aero-dynamical equations at the hovering state provides a suitable structure for the model. Given that there is a gravitational disturbance term, integrators are needed in the loop. Thus, two new states, d and z, are defined and the augmented state space representation is given by,

$$\begin{bmatrix} \dot{p} \\ \dot{y} \\ \ddot{p} \\ \ddot{y} \\ \dot{\delta} \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ y \\ \dot{p} \\ \dot{y} \\ \delta \\ \zeta \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_{11} & k_{12} \\ k_{21} & k_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_p \\ V_y \end{bmatrix} \quad (2)$$

where k_{11}, k_{12}, k_{21}, k_{22} are the unknown parameters to be evolved by the genetic identification loop.

The simplicity of the model structure described above allows us to focus the identification on just the relevant frequency band. Specifically, the experimental data is pre-processed to reduce the effects of the trim. Once a data range is selected, the initial condition is removed and the data is sent through a 4^{th} order Butterworth bandpass filter F above 0.3 Hz as depicted in Figure 3.

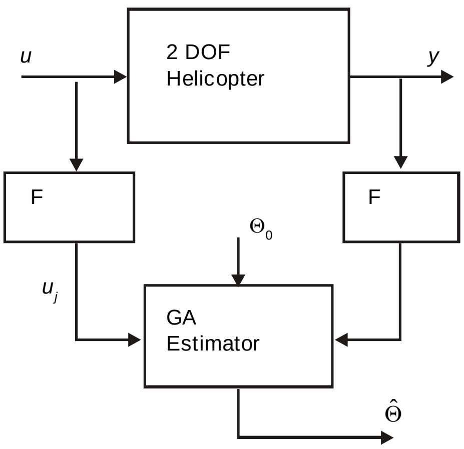


Figure 3: Identification Scheme

The overall system can be modeled by the transfer function,

$$y(\hat{t}) = G(\Theta, q)u(t) \quad (3)$$

where Θ denotes the set of parameters to be identified and q is the standard forward shift operator. The corresponding z operator will be omitted for simplicity. Given a description in (3) properly parameterized by the specific form and the input-output data, u and y, the prediction error e can be computed as follows,

$$e(t) = y(t) - G(\Theta, q)\, u(t) \quad (4)$$

For multi-output systems, the identification method consists of determining the parameter estimates by minimizing the following quadratic criterion using GA,

$$\hat{\Theta} = \arg\min_{\Theta} \det\left[\frac{1}{N}\sum_{i=1}^{N} e(t)e^T(t)\right] \quad (5)$$

where $e(t)$ is the prediction error and N is the size of the model estimation window or the number of time steps the fitness is accumulated.

3.3. Genetic Controller Tuning

Since it is very difficult, if not impossible, to find an exact representation of the actual dynamic model, the goal of the control-tuning loop is to evolve a robust set of control parameters. In general, robust controllers can be evolved by taking into account the presence of uncertainties in the real system. In dynamic model identification, uncertainty is inherent to the evolved model due to the noisy nature of the sensors. Apart from the continuous identification process maintained by the genetic identification loop to reduce the mismatch between the evolved model and the real plant, the change for a set of new control parameters and simulation model is also triggered when the discrepancy between the desired and actual response exceeds a certain threshold.

In this paper, the PID controller is considered for the control of the helicopter plant. It should be noted that the same approach can be easily extended to other control schemes. The digital PID controllers are governed on the discrete-time set $\{0, T, 2T, \ldots, kT, \ldots\}$ by control law equations of the form,

$$u(k)T = K_p\left(e(k)T + K_i z(k)T + K_d \frac{e(k)T - e(k-1)T}{T}\right) \quad (6)$$

$$e(k)T = v - y(k)T \quad (7)$$

$$z(k+1)T = z(k)T + Te(k)\ T \in R^+ \quad (8)$$

where T is the sampling period, e is the error, v is the set-point, and $z \in R^+$ is the integral of error. From (6), we note that the task of the control-tuning loop is to evolve the set of proportional, integral, and derivative gains $\{K_p, K_i, K_d\}$ that optimizes system performance. The fitness is evaluated using a time-weighted integral of the absolute value of error (ITAE) given as,

$$ITAE = \int_0^T t\left|e(t)\right| dt \quad (9)$$

This criterion balances error size and duration and avoids positive and negative errors canceling.

3.4. Implementation

When applying GA for online optimization in the real-time control experiment, conditions have to be made regarding some problems such as low computational efficiency, low convergence rate, and premature convergence. In order to reduce computational overhead due to the background operations of real-time identification and control-tuning, the evolutionary process is halted when satisfactory system performance is attained. On the other hand, the optimization procedure can be triggered whenever performance deterioration is detected. Therefore, considering one optimization instance of the two GA optimization loops involved in the experiment, the tuning time can be computed as,

$$T = (N \cdot T_S) + T_M + T_C \quad (10)$$

where N is the model estimation window size, T_S is the data sampling time, T_M and T_C are the computational time for the two genetic loops.

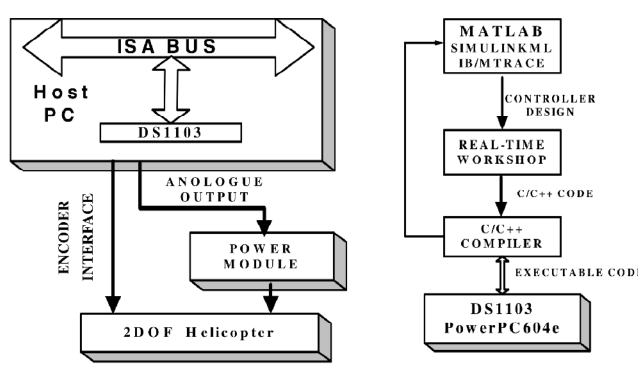
Real-coded representation, where the parameters of the plant model and the controllers are coded in floating point and concatenated to form an individual in the GAs, is adopted in both genetic loops since it is more efficient, provides increased precision, and allows for a continuous domain [18]. In this paper, elitism is implemented by allowing the best individual to survive into the next generation, otherwise the best individual may disappear due to sampling error, crossover, or mutation. Individuals are selected to a mating pool through a binary tournament selection of the evolving population. The selection criterion is based on the fitness function described in (5) and (9) for the identification and control-tuning loop, respectively.

3.5. Real-Time System Setup

Real-time hardware and software interfaces are accomplished by means of dSPACE DS1103, MATLAB, Real-Time Workshop, and Control Desk. Figure 4(a) illustrates the functional connections between the control system utilizing dSPACE hardware and the helicopter plant while the complete design process is illustrated in Figure 4(b). The different software components, such as MATLAB, Real-Time Workshop, MICROTEC C/C++ compiler, and the software interface utility are run on the host PC. The helicopter control system is first modeled on the ground with block diagrams using MATLAB/Simulink. The Real-Time Interface (RTI) provides additional Simulink blocks for the connection of I/O channels to the controller model as shown in Figure 5. Then, the real-time C code for the complete system is automatically generated by the Real-Time Workshop in conjunction with the dSPACE RTI. The download utility initializes the dSPACE hardware, loads the application executable file into the DS1103's memory area, and initiates program execution. Another dSPACE software utility, MLIB/MTRACE, allows online adjustment of Simulink schematic parameters while the executable code for the schematics is running on the dSPACE board without interrupting the experiment. Incremental encoder interfaces and D/A outputs make the board a powerful tool for rapid control prototyping [5], [12].

4. RESULTS

In order to evaluate the effectiveness of the proposed real-time genetic controller, simulations are carried out for different cases in this section. Specifically, the performance



(a) (b)

Figure 4: (a) DS1103 Real-time Control System and (b) Real-Time Software Design Process

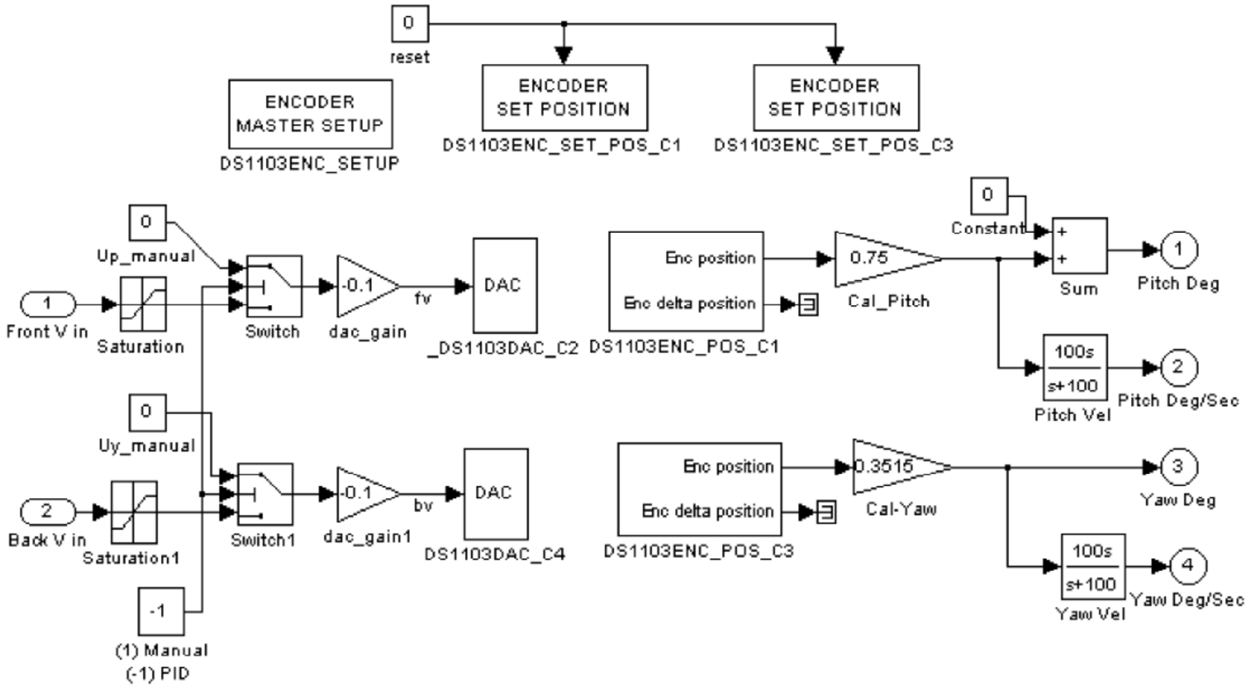
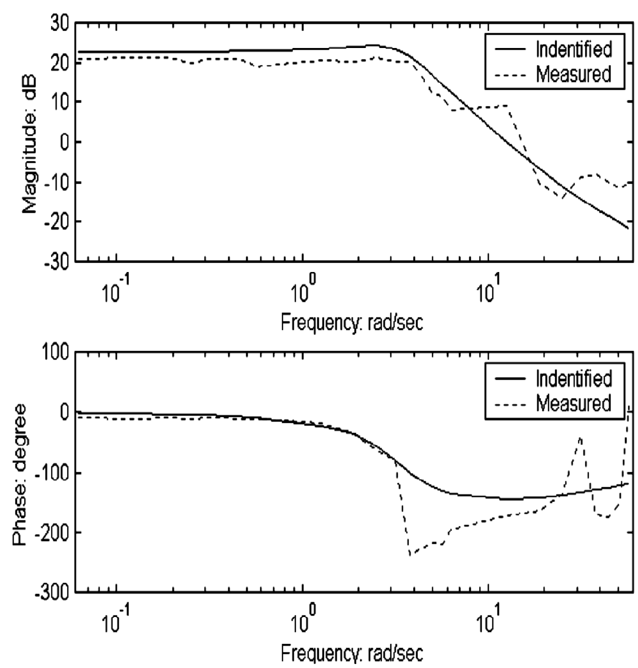


Figure 5: dSPACE Real-time Interface with Actual Systems

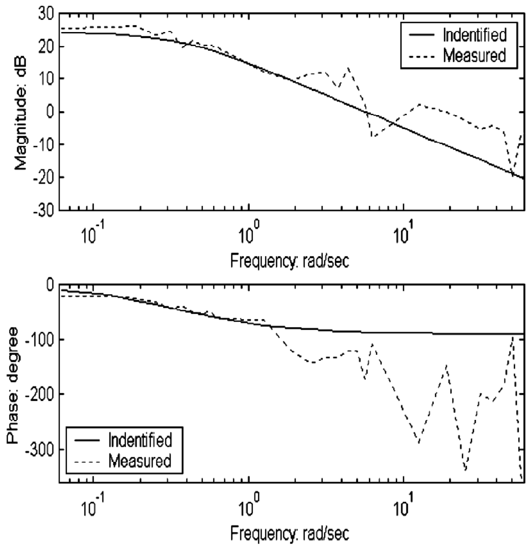
of the identification loop is analyzed with respect to the measured model while the control-tuning loop is examined based on its ability to reject noise and track set-point changes. The parameter setting used in the experiments is tabulated in Table 1. The real-time genetic identification and control are implemented in MATLAB and all the simulations are performed on an Intel Pentium III 933 MHz computer.

The frequency response of the final evolved model and the associated open-loop response for one optimization instance with respect to the real helicopter plant response are shown in Figure 6(a)-(d) and Figure 7(a)-(d), respectively. The performance of the evolved controller under the influence of external disturbances and changing set-points are illustrated in Figure 8(a)-(b) and Figure 9(a)-(b), respectively. In the simulation studies, the model estimation is performed over a 10s time window with N = 1000 and T_S = 0.01 and the average computational time is T_M = 20s for plant modeling and T_C = 30s for controller tuning.

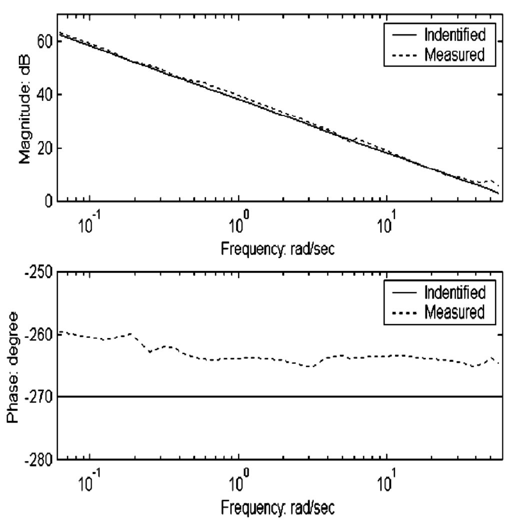
From Figure 6, it can be noted that the genetic identification loop is capable of evolving a model that can match the measured data well in the frequency range of 0 to 10 rad/s. In addition, from Figure 7, we are able to validate that the characteristics of the open-loop step responses of the evolved model and the real plant are very similar. From the system response under the influence of external disturbances in Figure 8(a)-(b), it can be observed that the evolved controller is inherently robust and capable of returning to the desired set-point. Furthermore, by comparing the system response to the application of external disturbances at different times, it can be noted that the evolving controller has the potential of adapting quickly to the environment. Similar behavior can be observed from the system response to set-point changes in Figure 9(a)-(b).



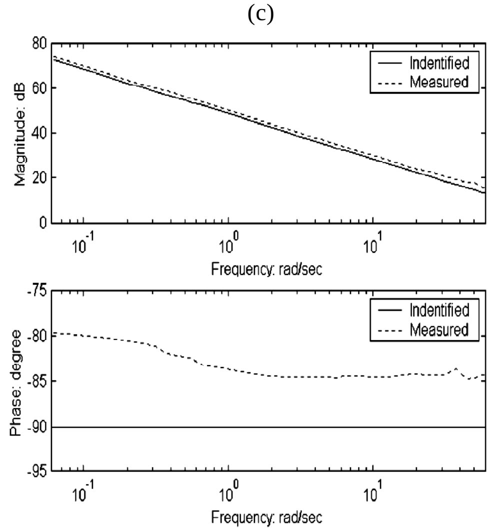
(a)



(b)

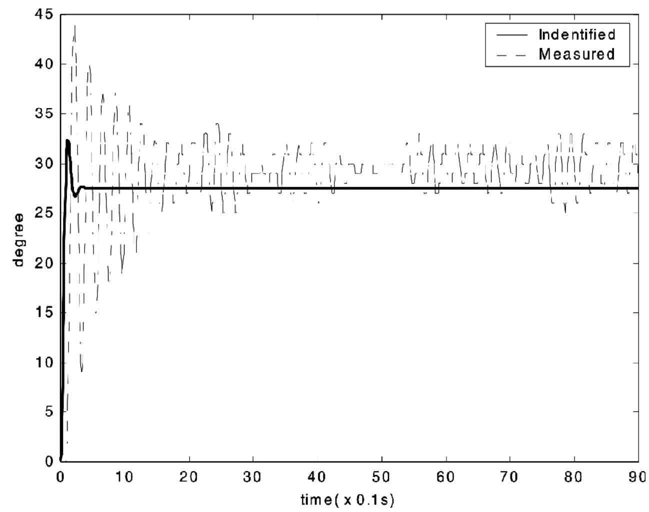


(c)

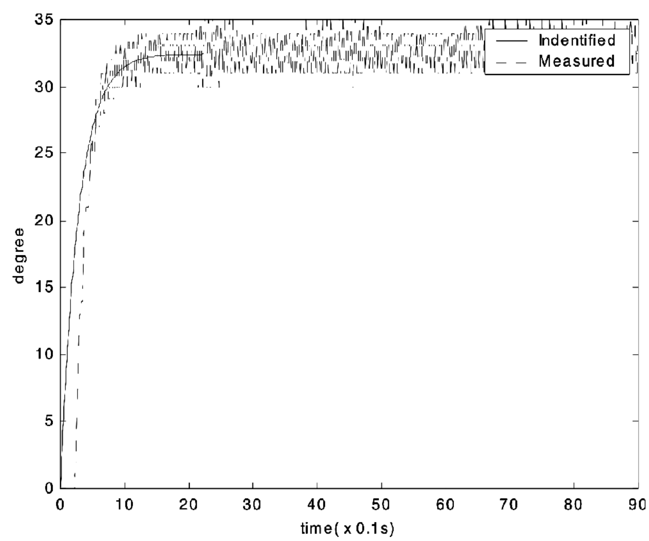


(d)

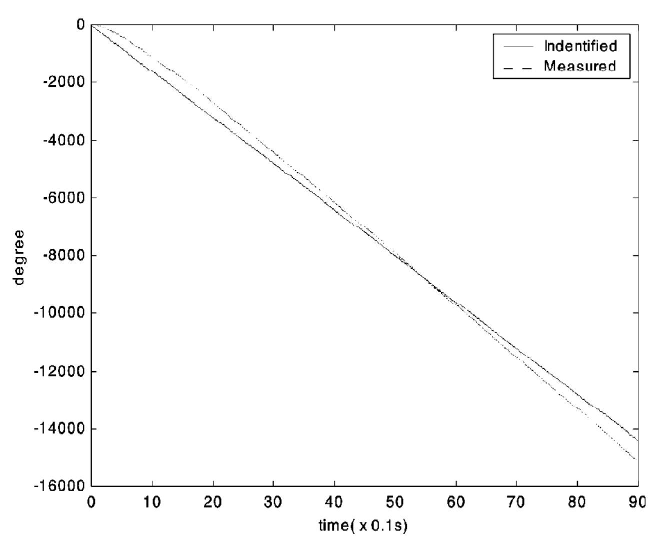
Figure 6: Frequency Response of Identified model vs. Real Plant (a) $G_{p11}(s)$ (Pitch-Up), (b) $G_{p22}(s)$ (Pitch-Uy), (c) $G_{p21}(s)$ (Yaw-Up), and (d) $G_{p22}(s)$ (Yaw–Uy)



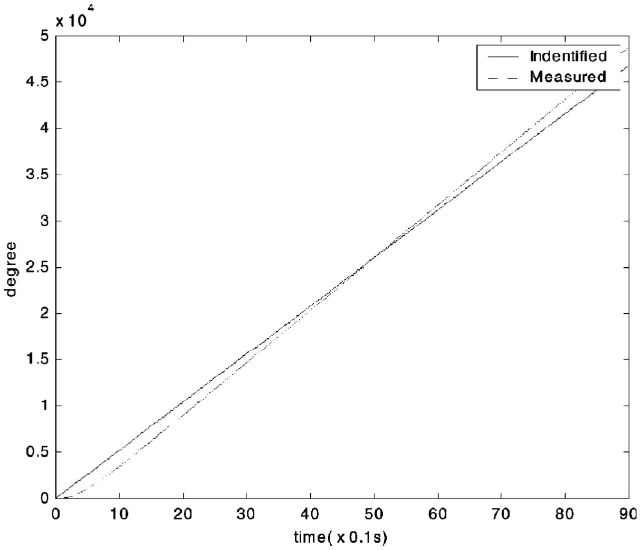
(a)



(b)

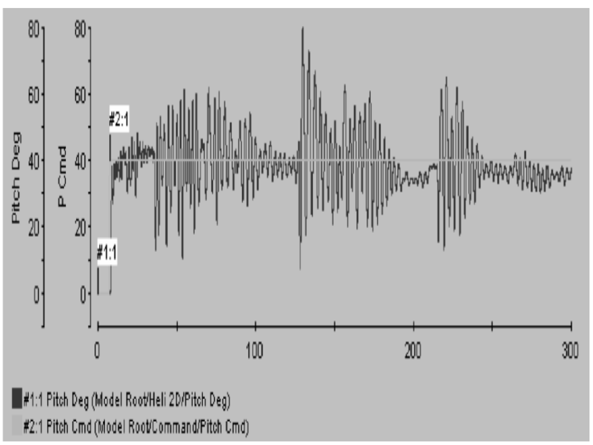


(c)

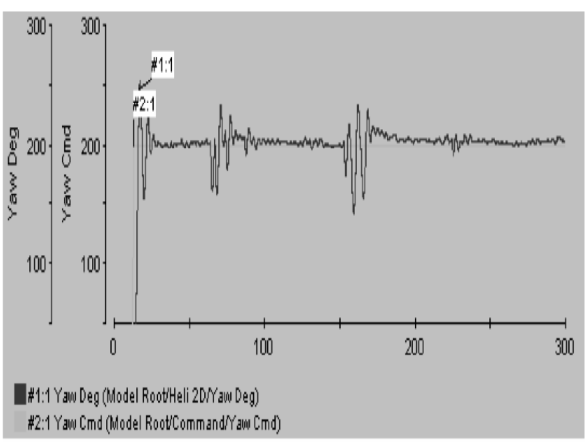


(d)

Figure 7: Open-loop Step Response of Identified Model vs. Real Plant (a) $G_{p11}(s)$ (Pitch-Up), (b) $G_{p22}(s)$ (Pitch-Uy), (c) $G_{p21}(s)$ (Yaw-Up), and (d) $G_{p22}(s)$ (Yaw–Uy)

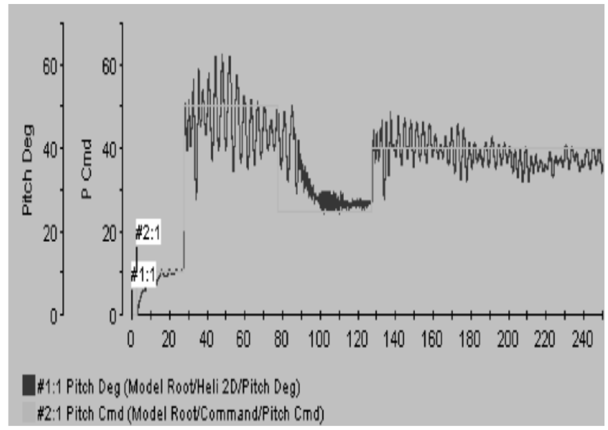


(a)

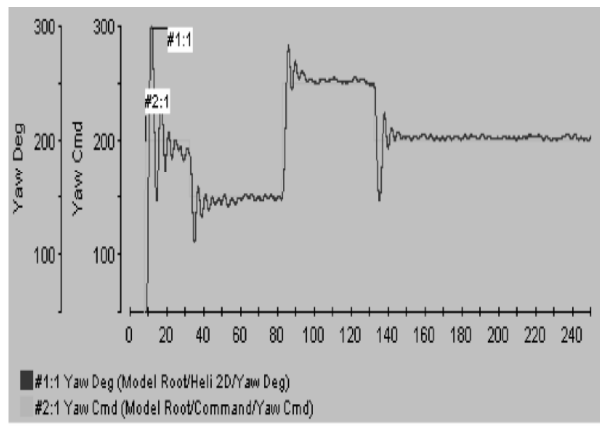


(b)

Figure 8: Evolved Controller under the Influence of External Disturbance for (a) Pitch and (b) Yaw



(a)



(b)

Figure 9: Evolved Controller Tracking Changing Set-point for (a) Pitch and (b) Yaw

Table 1
Parameter Setting for the Different Genetic Loops for One Optimization Instance

	Evaluation	*Population Size*	*Crossover rate*	*Mutation rate*
Identification loop	5	10	0.8	0.02
Control-tuning Loop	20	30	0.8	0.02

5. CONCLUSION

In this paper, a real-time genetic identification and control architecture is presented. This system architecture employs a dual-stage genetic strategy to update the simulation model of a dynamic system and optimize the associated control parameters for real-time control. In the first stage, a genetic identification loop, which evolves the system model, is employed. In the second stage, which is the genetic control-tuning loop, candidate controllers are evolved and evaluated based on the evolved plant model. Subsequently, the best controller will be implemented for real-time control whenever certain conditions are satisfied.

In order to validate the potential of GAs for real-time control applications, the proposed approach is applied to a 2DOF helicopter plant. In the experiments, it is noted that the real-time genetic identification and control architecture is capable of monitoring the system dynamics as well as set-point variations. Furthermore, it is able to adapt to external disturbances in the system by updating the model parameters using data collected from the real plant and the evolution of the corresponding controllers. This implies the applicability of the proposed method to the control of dynamic systems that may be exposed to large disturbances.

REFERENCES

[1] Ahmad, M., Zhang, L. and Readle, J. C., "Online Genetic Algorithm Tuning of a PI Controller for a Heating System," in *Proceedings of Genetic Algorithms in Engineering Systems: Innovations and Applications*, pp. 510–515, 1997.

[2] Cheong, C. Y., Tan, K. C. and Veeravalli, B., "A Multi-objective Evolutionary Algorithm for Examination Timetabling," *Journal of Scheduling*, to appear.

[3] O'Donnell, T. H., Altshuler, E. E. and Best, S. R., "The Significance of Genetic Representation in Genetic Antenna Design," *IEEE Antennas and Propagation Society International Symposium (Digest)*, **1**, 149–152, 2003.

[4] Fleming, P. J. and Purshouse, R. C., "Genetic Algorithms in Control Systems Engineering," Research Report No. 789, Dept. of Automatic Control and Systems Engineering, Univ. of Sheffield, UK, 2001.

[5] Hanselmann, H., "DSP in Control: The Total Development Environment," in *Proceedings of the 1996 IEEE IECON 22nd International Conference on Industrial Electronics, Control, and Instrumentation*, **3**, 1647–1654, 1996.

[6] Hsu, M. R., Ho, W. H. and Chou, J. H., "Stable and Quadratic Optimal Control for TS Fuzzy-model-based time-delay Control Systems," *IEEE Transactions on Systems, Man and Cybernetics – Part A: Systems and Humans*, **38**(4), 933–944, 2008.

[7] Kaitwanidvilai, S. and Parnichkun, M., "Position Control of a Pneumatic Servo System by Genetic Algorithm Based Fixed-structure Robust H_∞ Loop Shaping Control," in *Proceedings of 30th Annual Conference of IEEE Industrial Electronics*, **3**, 2246–2251, 2004.

[8] Kristinsson, K. and Dumont, G., "System Identification and Control using Genetic Algorithms," *IEEE Transactions on Systems, Man and Cybernetics*, **22**(5), 1033–1046, 1992.

[9] Lennon, W. K. and Passino, K. M., "Genetic Adaptive Identification and Control," *Engineering Applications of Artificial Intelligence*, **12**(2), 185–200, 1999.

[10] Linkens, D. A. and Nyongesa, H. O., "Genetic Algorithms for Fuzzy Control - part I: Offline System Development and Applications," *IEE Proceedings Control Theory and Applications*, **142**(3), 161–176, 1995.

[11] Linkens, D. A. and Nyongesa, H. O., "Genetic Algorithms for Fuzzy Control - part II: Online System Development and Applications," *IEE Proceedings Control Theory and Applications*, **142**(3), 177–185, 1995.

[12] Rubaai, A., Castro-Sitiriche, M. J. and Ofoli, A. R., "Design and Implementation of Parallel Fuzzy PID Controller for High-performance Brushless Motor Drives: An Integrated Environment for Rapid Control Prototyping," *IEEE Transactions on Industry Applications*, **44**(4), 1090–1098, 2008.

[13] Tan, K. C., Cheong, C. Y. and Goh, C. K., "Solving Multi-objective Vehicle Routing Problem with Stochastic Demand via Evolutionary Computation," *European Journal of Operational Research*, **177**(2), 813–839, 2007.

[14] Tan, K. C., Wang, M. L. and Peng, W., "A P2P Genetic Algorithm Environment for the Internet," *Communications of the ACM*, **48**(4), 113–116, 2005.

[15] Tan, K. C. and Li, Y., "Performance-based Control System Design Automation via Evolutionary Optimization," *Engineering Applications of Artificial Intelligence*, **14**, 473–486, 2001.

[16] Vaidyanathan, S., Broadhurst, D. L., Kell, D. B. and Goodacre, R., "Explanatory Optimization of Protein Mass Spectrometry via Genetic Search," *Analytical Chemistry*, **75**(23), 6679–6686, 2003.

[17] Wai, R. J., "Supervisory Genetic Evolution Control for Indirect Field-oriented Induction Motor Drive," in *Proceedings of IEE Electric Power Applications*, **150**(2), 215–226, 2003.

[18] Wright, A. H., "Genetic Algorithms for Real Parameter Optimization," *Foundations of Genetic Algorithms*, G. J. Rawlins (ed.), Morgan-Kauffmann, San Mateo, CA, 205–218, 1991.

[19] Wu, T. B., Cheng, Y., Tan, J. F. and Zhou, T. Y., "The Application of Chaos Genetic Algorithm in the PID Parameter Optimization," in *Proceedings of 2008 3rd International Conference on Intelligent System and Knowledge Engineering*, 230–234, 2008.

[20] Yan, J. G., Yuan, D. L., Xing, X. J. and Jia, Q. L., "Kalman Filtering Parameter Optimization Techniques based on Genetic Algorithm," in *Proceedings of the IEEE International Conference on Automation and Logistics*, Qingdao, China, 1717–1720, 2008.