International Journal of Robotics and Mechatronics

DE-based Robust Controller Design for Helicopter Cruise Control

Iwan Mahmud[†], Rini Akmeliawati[‡], and Agus Budiyono[§]

[†]Mechanical Engineering Deptartment, UCSI University, Cheras, Kuala Lumpur, Malaysia

[‡]Department of Mechatronics Engineering, International Islamic University, Malaysia

[§]Department of Aerospace Engineering, Konkuk University, Korea

Abstract**—**This paper presents a robust state feedback controller tuning via constrained optimization using DE (differential evolution). The controller gain is optimized based on the plant model such that the closed-loop system achieves maximum stability radius. The desired control performance is specified by assigning closed-loop poles region which is handled as a constraint in the DE-based optimization. The proposed controller design technique is applied to a longitudinal cruise control of a small-scale helicopter. The control of small-scale helicopter is a MIMO problem which increases the complexity of the controller design task. The proposed controller design attempts to simplify this complicated task where the automated design is employed via DE-based modern optimization. The effectiveness of the proposed controller design technique is presented by comparing its performance with that of conventional LQR-based controllers in computer simulation.

Keywords**—**Robust control design, differential evolution, helicopter, stability radius.

I. Introduction

DESIGNING a robust controller especially for MIMO (multi-outputs multi-inputs) systems is always interesting for control engineers. Particularly, flight control system design for a helicopter is a challenge because of nonlinear dynamics, strong coupling and poor model fidelity [1]. Achieving a verified model is obviously critical for designing a model-based autonomous control [2]. In addition, a special situation such as autonomous landing on a moving platform would add further complexity [3].

A MIMO system can be effectively represented in state-space variable form so that a state feedback controller can be designed with a common technique such as LQR (linear quadratic regulator). LQR controller presents interesting robustness properties, such as a phase margin larger than 60^o and an infinite gain margin [4]. However, it has been shown to be less robust especially to parametric uncertainty in real-time implementation [5]. This is due to the fact that uncertainty is modeled as Gaussian white noise, whereas uncertainty cannot always be modeled as white noise [6]. In addition, LQR controller is designed based on Q and R weighting matrices which need to be heuristically tuned. Studies of helicopter control using this technique can be found such as in [7].

On the other hand, H_∞ robust control technique provides better robustness to inherent uncertainty in the system model. However, the theory behind this technique is not trivial. In addition, it is not straightforward to formulate a practical control design problem into H_∞ design framework. The resulting controller using H_∞ technique is also often higher order which is not simple to implement [8]. The successful application of H_∞ robust control technique for helicopter control can be found such as in [9-11].

Some other works use intelligent techniques to compensate for the inherent nonlinearity of helicopter dynamics or combine them with conventional controller design technique to achieve an improved hybrid approach. Tee et al. [12] proposed adaptive neural network control for helicopters in vertical flight. Lee and Tsai [13] used recurrent neural network to successfully complete back-stepping controller design. Fuzzy logic was applied for flight control in [14] and [15]. However, the practical implementation of these techniques might be limited by the hardware issues. Therefore, they focused on the simulation studies.

Apart from the aforementioned techniques, other techniques such as feedback linearization, nonlinear controller design and adaptive control design were also used for helicopter control. There were a number of studies working with these approaches [12, 16-18].

This study focuses on the development of a fixed-order robust control design technique which is considerably simple, systematic and automated. This is motivated by the fact that in most of control design techniques, excessive parameters or functions tuning is required. For example, in H_∞ loop-shaping design [19], a heuristic tuning is required to obtain the suitable weighting functions in order to achieve the desired shape of frequency response.

Therefore, it is still interesting to study a robust control design that combines the advantages of a simple conventional controller structure and robust control theory to achieve a robust and practical control design. It is believed that dealing with controller parameters tuning is bound to involve some

Corresponding author: Rini Akmeliawati (e-mail: rakmelia@iium.edu.my).
This paper was submitted on February 28, 2014; revised on November 20, 2014; and accepted on November 25, 2014.

kind of optimization technique. The advanced progress of computational intelligent in modern optimization technique supports this conjecture. Among the candidates for modern optimization are genetic algorithms (GA), particle swarm optimization (PSO) and differential evolution (DE).

GA was used for tuning the fuzzy controller rules in an effective helicopter control [20]. In [21], GA was used to optimize the weighting functions for H_∞ controller. Meanwhile in [22], PSO algorithm was adopted as an optimization solver such that controller gains were selected to minimize the error between the desired response and the actual response of helicopter control system.

DE is basically a new version of evolutionary algorithm like GA which is considerably robust end efficient to solve global optimization [23]. In the proposed robust control design, DE is used to search for a set of robust controller gain such that the complex stability radius of the closed-loop system is maximized. Complex stability radius is a tool of measuring system robustness [24]. Furthermore, the desired time-domain performance is automatically handled by assigning closed-loop poles region which is defined as a constraint in the optimization.

The remaining of this paper is organized as follows. Section presents the helicopter model used for simulation. Section III briefly discusses DE algorithm. Section IV discusses the proposed controller design technique using DE. Section V presents the results and analysis. Section VI is conclusions and discussions.

II. System Description

The X-cell 60SE small scale helicopter model is used in simulation to evaluate the performance of the proposed controller. The dynamic modelling of this system has been developed together with detailed parameters derivation by Budiyono et al. [25]. The longitudinal flight dynamics is used for which the proposed DE-based feedback controller is applied. The longitudinal flight dynamics for 10mps speed is represented in state-space variable form as follows:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y = Cx(t) + Du(t) \quad (1)$$

where:

$$A = \begin{bmatrix} -0.2312 & -0.0010 & -9.9824 & 9.7711 & 11.5157 \\ -0.2629 & 0 & 0.0637 & 0 & 420.9436 \\ 1.9263 & 0 & -0.4668 & 0 & 0 \\ 0 & 1 & -0.0621 & 0 & 0 \\ 0.0025 & -1 & 0 & 0 & -8.3500 \end{bmatrix}$$

$$B = \begin{bmatrix} -4.4324 & 0 \\ -16.0645 & 0 \\ 117.7042 & 0 \\ 0 & 0 \\ 0 & 35.0700 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The state vector is $x = [u \quad w \quad q \quad \theta \quad a_1]^T$ and the input vector is $u = [\delta_{coll} \quad \delta_{lon}]$; where u and w denote the fuselage velocities (longitudinal and vertical components), q denotes pitch angular rates, θ denotes pitch angle, a_1 is longitudinal flapping angle, δ_{coll} and δ_{lon} are main rotor collective pitch and longitudinal cyclic inputs respectively. The model given in equation (1) is considered as nominal plant model.

III. Brief Overview of DE

Differential Evolution (DE) is a stochastic, population-based search algorithm developed by Storn and Price in 1995 [26, 27]. While DE shares similarities with other evolutionary algorithms (EA), it differs significantly in the sense that distance and direction information from the current population is used to guide the search process. Furthermore, the original DE algorithm was developed to be applied to continuous-valued landscapes. The original version of DE can be defined by the following parts [28].

A. Population

$$P_{X,j} = (X_{i,j}), \; i = 0,1,\dots,Np-1, \quad j = 0,1,\dots,j_{max} \quad (2)$$
$$X_{i,j} = (x_{k,i,j}), \; k = 0,1,\dots,D-1.$$

where N_p denotes the number of population vectors, *j* defines the generation counter, and *D* is dimension of the problem (number of variables).

B. Initialization of The Population

$$x_{k,i,0=}rand_k\,[0,1).(u_{b,j} - l_{b,j}) + l_{b,j} \quad (3)$$

The *D*-dimensionality initialization vectors, l_b and u_b indicates the lower and upper bounds of the variables vectors $X_{i,j}$. The random number generator, $rand_k[0,1)$, returns a uniformly distributed random number within the range $0 \le rand_k[0,1) < 1$. The subscript k indicates that a new random value is generated for each variable (candidate of solution).

The perturbation of base vector $y_{i.j}$ by using difference vector based mutation

$$v_{i,j} = y_{i,j} + F(X_{r1,j} - X_{r2,j}) \quad (4)$$

to generate a mutation vector $v_{1,j}$. The difference vector indices, r_1 and r_2, are randomly selected once per base vector. F_m is called mutation factor constant. Setting $y_{i.j} = X_{r0,j}$ defines what is often called classic DE where the base vector is also randomly chosen population vector. The random indicesr_0, r_1, and r_2 should be mutually exclusive. There are also variants of perturbation which are different from **Equation 4**.

C. Diversity Enhancement

The classic variant of diversity enhancement is crossover which mixes variables of the mutation vector $v_{1,j}$ and the so-called target vector $X_{i,j}$ in order to generate trial vector $u_{i,j}$. The most common form of crossover is uniform and is defined as

$$u_{i,j} = u_{k,i,j} = \begin{cases} v_{k,i,j} & if \ (rand_k[0,1) \leq C_r) \\ x_{k,i,j} & otherwise. \end{cases} \quad (5)$$

where C_r is crossover constant. In order to prevent the case $u_{i,j} = X_{i,j}$ at least one component is taken from the mutation vector $v_{i,j}$, a detail that is not expressed in equation (5). Other variants of crossover are given by [27].

D. Selection

DE uses simple one-to-one survivor selection where the trial vector $u_{i,g}$ competes against the target vector $x_{i,j}$. The vector with the lowest objective function value (minimization is assumed in the optimization) survives into the next generation $j+1$.

$$x_{i,j+1} = \begin{cases} u_{i,j} & if \ f(u_{i,j}) \leq f(x_{i,j}) \\ x_{i,j} & otherwise. \end{cases} \quad (6)$$

Along with DE algorithm came a notation to classify the various DE's variants [26]. The notation is defined by $DE/x/y/z$ where x denotes the base vector, y denotes the number of difference vectors used, and z represents the crossover method. For example, $DE/rand/1/bin$ is shorthand notation for equation (3) trough equation (6) $y_{i,j} = x_{r0,j}$. $DE/best/1/bin$ is the same except for $y_{i,j} = x_{best,j}$. In this case $x_{best,j}$ represents the vector with the lowest objective function value evaluated so far.

The rule of thumb values for these parameters given by Storn and Price [26] is: $F_m \in [0.5,1]$, $C_r \in [0.8,1]$ and $N_P = 10D$, where D is dimension of the problem. Furthermore, a study done by Ronkkonen et al. [29] recommends $C_r = 0.9$ when objective function is multi-modal and non-separable.

IV. Proposed Robust Controller Design via DE-based Optimization

A. Controller Structure

Consider a plant model of linear time-invariant continuous-time system as described in equation (1), where $x \in R^n$, $u \in R^m$ and $y \in R^p$ are state vector, control input and output vector respectively. It is assumed that the system given in equation (1) is completely state controllable and all state variables are available for feedback. One can use state feedback controller as shown in **Figure 1**. The controller gains, $K = [k_1, k_2, \dots, k_n]$, can be computed based on conventional methods such as LQR or pole placement method. These conventional methods however assume that there is no plant uncertainty.

In this study, a new approach for robust controller design via constrained optimization using DE is proposed. A set of robust controller gains are optimized such that the plant uncertainty is automatically handled with the use of stability radius that will be discussed in the next section. In addition, a closed-loop poles region is incorporated as the optimization constraint to specify the desired time-domain performance.

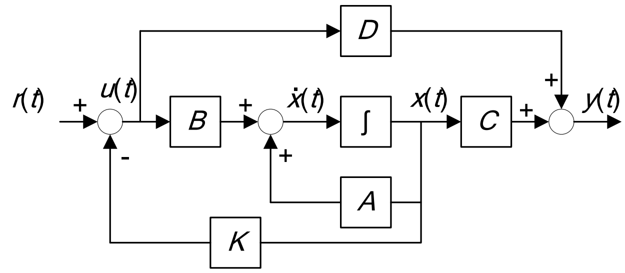


Figure 1 State feedback control scheme

B. Stability Radius

The objective of the optimization is to maximize complex stability radius [24]. Stability radius is a maximum distance to instability. In general, we can classify stability radius into two types; complex stability radius and real stability radius. We use complex stability radius which can handle wider class of perturbations.

The definition of complex stability radius is given as follows. Let $\mathbb{C}$ denotes the set of complex numbers. $\mathbb{C}_- = \{z \in \mathbb{C} | Real(z) < 0\}$ and $\mathbb{C}_+ = \mathbb{C} \backslash \mathbb{C}_-$ is the closed right half plane. Consider a nominal system in the form:

$$\dot{x}(t) = Ax(t). \quad (7)$$

$A(t)$ is assumed to be stable. The perturbed open-loop system is assumed as:

$$\dot{x}(t) = (A(t) + E\Delta(t)H)x(t) \quad (8)$$

where $\Delta(\cdot)$ is a bounded time-varying linear perturbation. E and H are scale matrices that define the structure of the perturbations. The perturbation matrix itself is unknown. The stability radius of equation (8) is defined as the smallest norm of Δ for which there exists a Δ that destabilizes equation (8) for given the perturbation structure (E, H).

For the controlled perturbed system in the form equation (8), let:

$$G(s) = H(sI - A)^{-1}E \quad (9)$$

be the "transfer matrix" associated with (A, E, H), then the complex stability radius is defined as:

Definition 1:

$$r_c(A, E, H, \mathbb{C}_+) = \left[\max_{s \in \partial\mathbb{C}_+} \|G(s)\|\right]^{-1}, \quad (10)$$

where $\partial\mathbb{C}_+ = \bar{\mathbb{C}}_- \cap \mathbb{C}_+$ is the boundary of $\mathbb{C}_+$. In other words, a maximum r_c can be achieved by minimizing the H_∞ norm of the "transfer matrix" G.

Proposition 1:

Using Definition 1, the complex stability radius of the closed-loop system as shown in **Figure 1** is given as:

$$r_c(\hat{A}, E, H, \mathbb{C}_+) = \left[\max_{s \in \partial\mathbb{C}_+} \left\| H(sI - \hat{A})^{-1} E \right\|\right]^{-1} \quad (11)$$

where $\hat{A} = A - BK$.

C. Regional Poles Placement

The proposed robust control design uses a feedback controller structure as shown in **Figure 1**. The problem of finding a set of robust controller gains is transformed into a constrained optimization problem using DE. As mentioned earlier, the objective of the optimization is to maximize the complex stability radius (r_c), however we will convert into minimization mode by putting negative sign. Based on our approach, the searching procedure using constrained optimization can be formulated as in **TABLE I**.

TABLE I CONSTRAINED OPTIMIZATION FORMULATION

Minimize:	$f(X) = -r_c(X)$	
Subject to constraint:	$\lambda_n(X) \in \psi$	for n=1,2,…
and boundary constraint:	$X \epsilon [l_b, u_b]$	

where $X = K = [k_1, k_2, \ldots, k_n]$ is the vector solutions such that $X \in S \subseteq R^{n+1}$. S is the search space, and $\psi \subseteq S$ is the feasible region or the region of S for which the constraint is satisfied. The constraint is the closed-loop poles region; in this feasible region, the controller gains are found such that the closed-loop poles (λ) lie within a wedge region (ψ) of a complex plane as given in **Figure 2**.

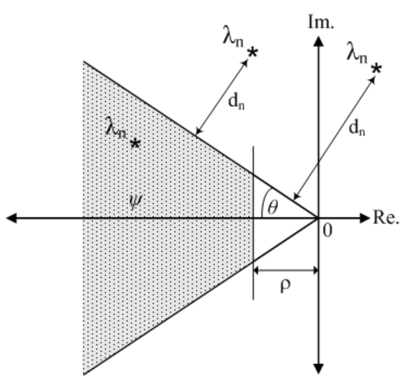


Figure 2 A wedge region for closed-loop pole placement

The wedge region can be specified by two parameters θ and ρ which are related to desired time-domain characteristics i.e.: damping factor ($\zeta = cos^{-1}\theta$) and settling time. For brevity of this paper, the detailed optimization process such as the constraint handling and stopping criterion can be found in [30].

V. RESULTS

A. Optimization Results

The optimization has been performed in MATLAB 2008 with Dual-Core Processor 2.1 GHz and 2 GB RAM. The optimization parameters are set (as shown in **TABLE II**) before optimization begins.

TABLE II DE PARAMETERS FOR OPTIMIZATION

Dimension of the problem	D	10
Population size	N_p	100
Mutation scaling constant	F_m	0.5
Crossover rate constant	C_r	0.9
Maximum iteration	j_{max}	200
Solution bounds	$[l_b, u_b]$	±10
Damping ratio	ζ	0.7
'transient margin'	ρ	-1

The obtained controller gains using DE-based optimization (DEFC) are shown in **TABLE III** together with two LQR-based controllers (LQR1 and LQR2) for benchmarking. The LQR-based controllers are respectively computed based on Q and R matrices respectively as follows:

$$Q_1 = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix} R_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 1000 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 100 & 0 \\ 0 & 0 & 0 & 0 & 100 \end{bmatrix} \quad R_2 = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}.$$

TABLE III FEEDBACK CONTROLLER GAINS

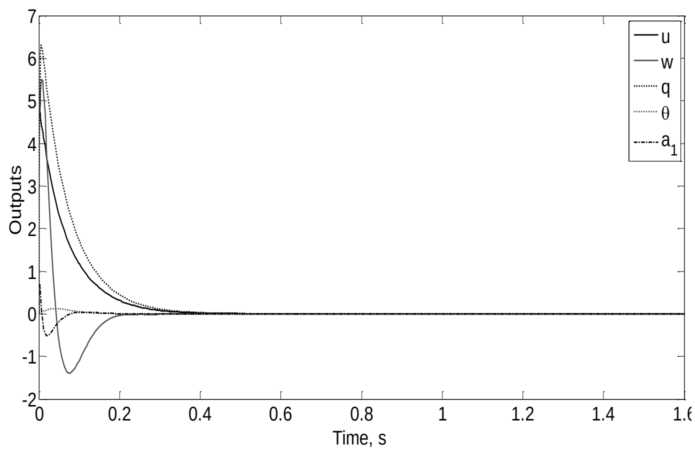
Controller	Controller gain
DEFC	$\begin{bmatrix} -7.25 & -0.04 & 5.24 & -6.22 & -0.40 \\ -5.45 & 0.59 & 3.76 & 8.16 & 5.45 \end{bmatrix}$
LQR1	$\begin{bmatrix} -3.09 & -0.28 & 3.12 & -3.06 & -0.21 \\ -0.04 & 3.14 & 0.36 & 3.87 & 9.00 \end{bmatrix}$
LQR2	$\begin{bmatrix} -9.64 & -2.45 & 2.97 & -3.82 & -1.09 \\ -2.38 & 9.67 & 0.91 & 2.86 & 15.23 \end{bmatrix}$

B. Controller Performance

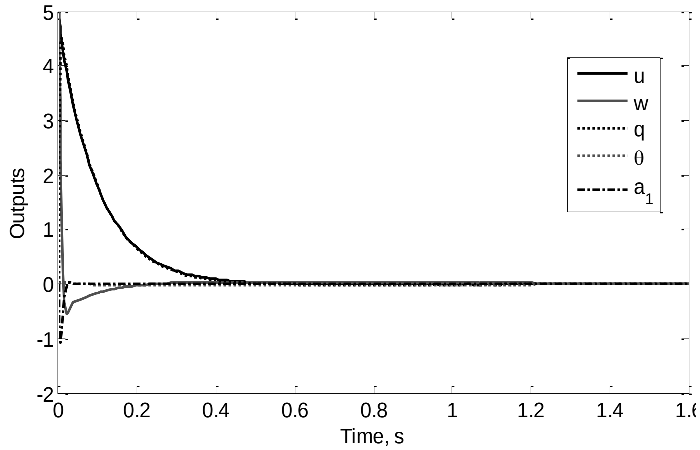
In order to test the controller performance, a disturbance is introduced to the system. Here, offset initial state values are chosen as disturbance as shown in **TABLE IV**. The plant model in **Equation 1** is a linearized model for longitudinal speed at 10 mph. This is the nominal plant model. The controller will also be tested for the perturbed plants which are the plant models at speed 4 mph and 16 mph.

TABLE IV INITIAL STATES VALUES

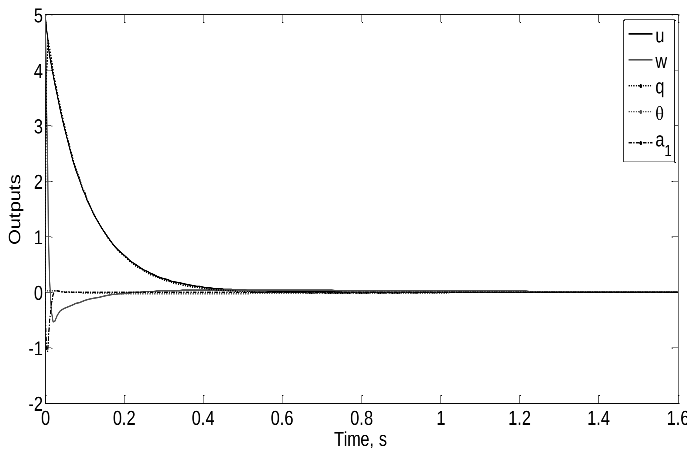
State	Name	Initial
u	Longitudinal velocity	5 mph
w	Heave velocity	5 mph
q	Pitch rate	0 rad/s
θ	Pitch angle	0 rad
a_1	Angle of flapping	0 rad



(a) Controlled using DEFC

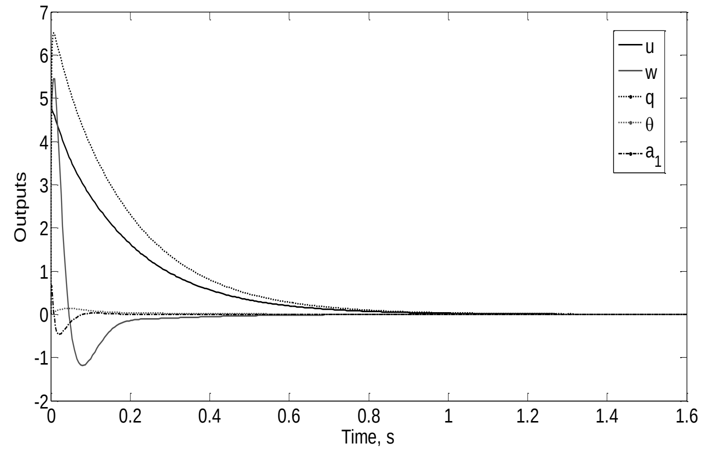


(b) Controlled using LQR1

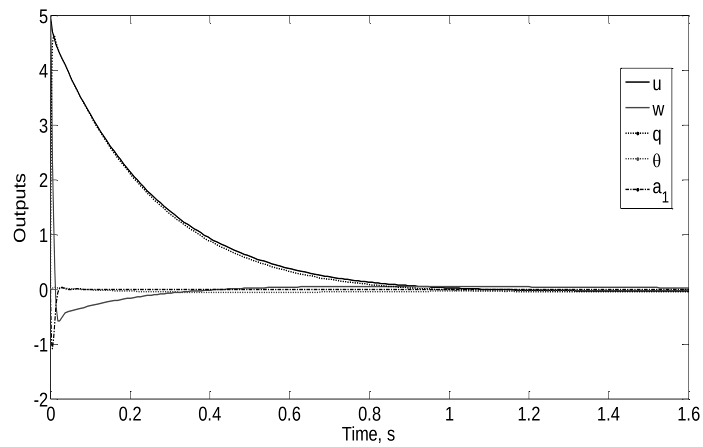


(c) Controlled using LQR2

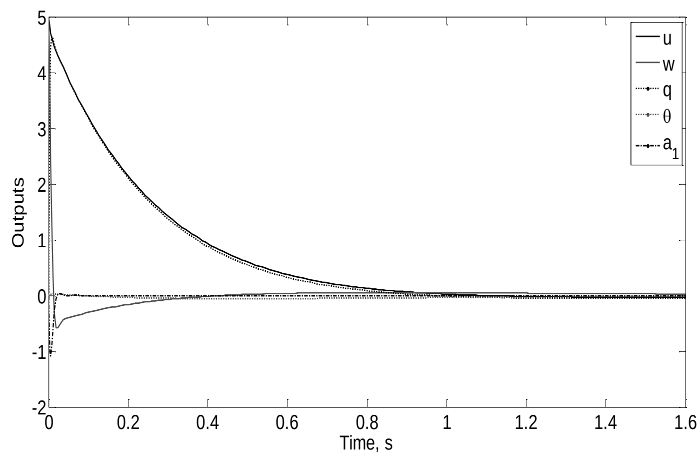
Figure 3 Response of nominal plant (at 10 mph)



(a) Controlled using DEFC

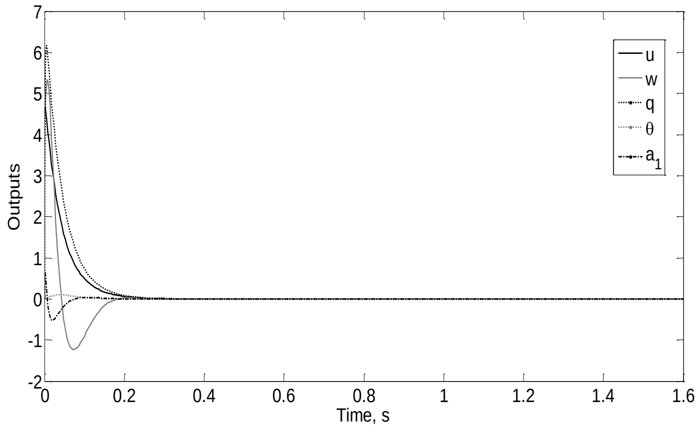


(b) Controlled using LQR1

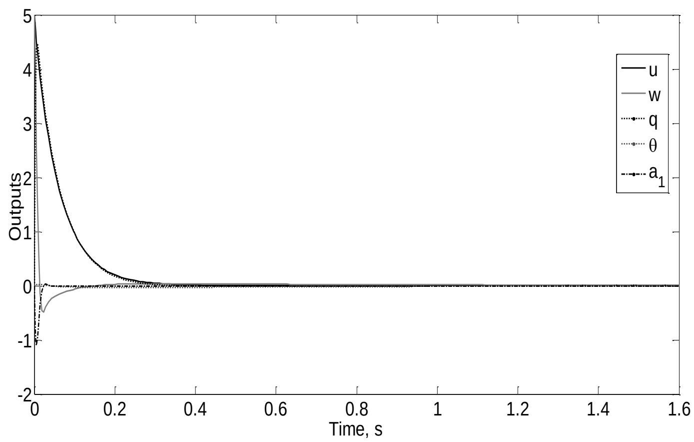


(c) Controlled using LQR2

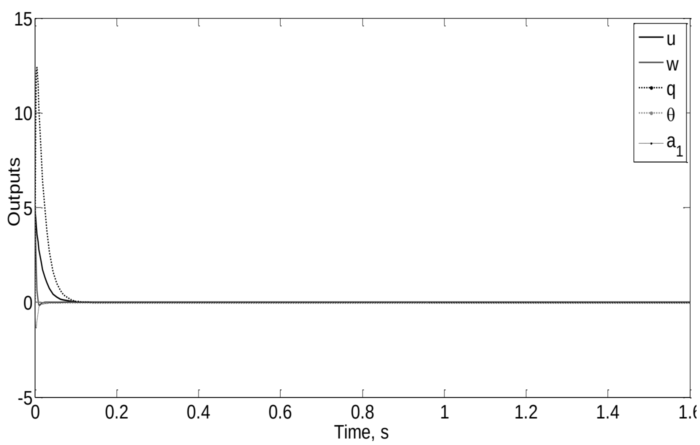
Figure 4 Response of perturbed plant (at 4 mph)



(a) Controlled using DEFC



(b) Controlled using LQR1



(c) Controlled using LQR2

Figure 5 Response of perturbed plant (at 16 mph)

Figure 3 shows the response of the nominal plant model with initial state values in **TABLE IV**. It can be seen there is no much different between the response of the system controlled by LQR1 and LQR2 (**Figure 3b** and **Figure 3c**). The response of both systems is slower compared to that of DEFC (**Figure 3a**). As the consequence, DEFC results in a bit more overshoot of w and q compared to both LQRs.

Furthermore, **Figure 4** – **Figure 5** show the perturbed plants where the linearized model is obtained for speed at 4 mph and 16 mph. It can be seen especially in **Figure 4b** and **4c** that steady state error exists at time 1s due to slow response of LQR1 and LQR2. However, in **Figure 5c**, it can be noticed that LQR2 results in quite high pitch rate (q) which shows that LQR has lack of robustness when tested for different speed conditions.

VI. CONCLUSIONS

A robust controller tuning via constrained optimization using DE has been presented. The controller design is performed by means of automated tuning using modern optimization algorithm. The proposed controller has been applied to control longitudinal flight control of a small scale helicopter. The simulation results has indicated that the proposed controller show its robustness property when tested for different condition of the helicopter speed. LQR controller may be able perform similarly to the proposed controller (DEFC). However, Q and R matrices must be tuned further manually.

REFERENCES

[1] E. Prempain and I. Postlethwaite, "Static H∞ loop shaping control of a fly-by-wire helicopter," *Automatica*, vol. 41, pp. 1517-1528, 2005. CrossRef

[2] S. S. Takahiro Ishii, Gennai Yanagisawa, Kazuki Tomita, Yasutoshi Yokoyama, "Multi-body Dynamics Modeling of Fixed-Pitch Coaxial Rotor Helicopter," *Journal of Unmanned System Technology*, vol. 1, 2013.

[3] C. Wu, J. Qi, D. Song, and J. Han, "LP Based Path Planning for Autonomous Landing of An Unmanned Helicopter on A Moving Platform," *Journal of Unmanned System Technology*, vol. 1, pp. 7-13, 2013.

[4] C. Olalla, R. Leyva, A. El-Aroudi, and I. Queinnec, "Robust LQR Control for PWM Converters: An LMI Approach," *Industrial Electronics, IEEE Transactions on*, vol. 56, pp. 2548-2558, 2009. CrossRef

[5] M. Castillo-Effen, C. Castillo, W. Moreno, and K. P. Valavanis, "Control Fundamentals of Small / Miniature Helicopters - A Survey," in *Advances in Unmanned Aerial Vehicles*. vol. 33, K. Valavanis, Ed., ed: Springer Netherlands, 2007, pp. 73-118.

[6] A. Stoorvogel, *H° control problem. A state space approach. The*: Prentice Hall., 1992.

[7] A. Budiyono and S. S. Wibowo, "Optimal tracking controller design for a small scale helicopter," *Journal of bionic engineering*, vol. 4, pp. 271-280, 2007.

[8] L.-H. Lin, F.-C. Wang, and J.-Y. Yen, "Robust PID controller design using particle swarm optimization," in *Asian Control Conference, 2009. ASCC 2009. 7th*, 2009, pp. 1673-1678.

[9] E. Prempain and I. Postlethwaite, "Static H_∞ loop shaping control of a fly-by-wire helicopter," in *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, 2004, pp. 1188-1195 Vol.2.

[10] R. Kureemun, D. J. Walker, B. Manimala, and M. Voskuijl, "Helicopter flight control law design using H∞ techniques," in *Proceedings of the 44th IEEE Conference on Decision and Control, and European Control Conference*, 2005, pp. 1307-1312. CrossRef

[11] G. Cai, B. M. Chen, X. Dong, and T. H. Lee, "Design and implementation of a robust and nonlinear flight control system for an unmanned helicopter," *Mechatronics*, vol. 21, pp. 803-820, 2011. CrossRef

[12] K. P. Tee, S. S. Ge, and F. E. H. Tay, "Adaptive neural network control for helicopters in vertical flight," *Control Systems Technology, IEEE Transactions on,* vol. 16, pp. 753-762, 2008. CrossRef
[13] C.-T. Lee and C.-C. Tsai, "Nonlinear adaptive aggressive control using recurrent neural networks for a small scale helicopter," *Mechatronics,* vol. 20, pp. 474-484, 2010. CrossRef
[14] T. G. B. Amaral and M. M. Crisostomo, "Automatic helicopter motion control using fuzzy logic," in *Fuzzy Systems, 2001. The 10th IEEE International Conference on*, 2001, pp. 860-863 vol.3.
[15] X. P. Zhu, Y. H. Fan, and J. Yang, "Design of Tiltrotor Flight Control System Using Fuzzy Sliding Mode Control," in *Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on*, 2010, pp. 1060-1063.
[16] J.-C. Avila Vilchis, B. Brogliato, A. Dzul, and R. Lozano, "Nonlinear modelling and control of helicopters," *Automatica,* vol. 39, pp. 1583-1596, 2003. CrossRef
[17] L. Marconi and R. Naldi, "Robust nonlinear control of a miniature helicopter for aerobatic maneuvers," in *Proceedings of the 32nd Rotorcraft Forum*, 2006.
[18] B. Song, J. K. Mills, Y. Liu, and C. Fan, "Nonlinear dynamic modeling and control of a small-scale helicopter," *International Journal of Control, Automation and Systems,* vol. 8, pp. 534-543, 2010. CrossRef
[19] D. McFarlane and K. Glover, "A loop-shaping design procedure using H∞ synthesis," *Automatic Control, IEEE Transactions on,* vol. 37, pp. 759-769, 1992. CrossRef
[20] C. Phillips, C. L. Karr, and G. Walker, "Helicopter flight control with fuzzy logic and genetic algorithms," *Engineering Applications of Artificial Intelligence,* vol. 9, pp. 175-184, 1996. CrossRef
[21] J. Dai and J. Mao, "Robust flight controller design for helicopters based on genetic algorithms," in *World Congress*, 2002, pp. 345-345.
[22] B.-M. Min, H.-S. Shin, and M.-J. Tahk, "Control System Design for an Autonomous Helicopter Using Particle Swarm Optimization," in *25th International Congress of the Aeronautical Sciences (ICAS)*, 2006.
[23] V. Feoktistov, *Differential evolution*: Springer, 2006.
[24] D. Hinrichsen and A. J. Pritchard, "Stability radii of linear systems," *Systems & Control Letters,* vol. 7, pp. 1-10, 1986. CrossRef
[25] A. Budiyono, T. Sudiyanto, and H. Lesmana, "Global linear modeling of small scale helicopter," in *Intelligent unmanned systems: Theory and applications*, ed: Springer, 2009, pp. 27-62.
[26] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization,* vol. 11, pp. 341-359, 1997. CrossRef
[27] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*: Springer, 2006.
[28] R. Storn, "Differential evolution research–trends and open questions," in *Advances in Differential Evolution*, ed: Springer, 2008, pp. 1-31.
[29] J. Ronkkonen, S. Kukkonen, and K. V. Price, "Real-parameter optimization with differential evolution," in *Proc. IEEE CEC*, 2005, pp. 506-513.
[30] M. I. Solihin, R. Akmeliawati, R. Muhida, and A. Legowo, "Guaranteed robust state feedback controller via constrained optimization using Differential Evolution," in *Signal Processing and Its Applications (CSPA), 2010 6th International Colloquium on*, 2010, pp. 1-6.