

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Development of Real-Time Hardware in the Loop Based MPC for Small-Scale Helicopter

Zahari Taha², Abdelhakim Deboucha¹,
Azeddein Kinsheel¹ and Raja Ariffin Bin Raja Ghazilla¹

¹Centre for Product Design and Manufacturing Department of Engineering Design,
Manufacture Faculty of Engineering- University of Malaya, Kuala Lumpur,

²Department of Manufacturing Engineering,
University Malaysia Pahang, Gambang, Pahang,
Malaysia

1. Introduction

In recent years, unmanned aerial vehicles (UAVs) have been shown a rapid development equipped with intelligent flight control devices. Many advantages could be offered by UAVs, due to their widely applications (Garcia and Valavanis, 2009).

Flight control is the principle unit for UAVs to perform a full autonomous mission without or with less interference of a human pilot. Numerous types of control have been developed for small-scale helicopters including classical, intelligent and vision controls.

The most conventional and common control methods that have been used by many researchers are the SISO controls, i.e., PI or PID control because their requirements are not highly dependent on the accuracy of the plant model. Two control approaches are proposed by Kim and Shim (2003), a multi-loop PID controller and a nonlinear model predictive control for the purpose of trajectory tracking. This strategy shows satisfactory results when applied to the Yamaha R-50. However, if large perturbations need to be compensated, or significant tracking abilities are required, this strategy may not be adequate.

Wenbing, et al (2007) presented a multiple-input-multiple-output (MIMO) neural network controller which has a structure of two inputs and two outputs to control a small-scale helicopter. The neural network controller is used with a simple adaptive PID configuration. The PID gains k_i , k_p and k_d are tuned online via the training of the proposed neural networks during the flight.

Srikanth, James and Gaurav (2003) combined vision with low-level control to perform the landing operation of the helicopter. The vision control navigates the commands to a low-level controller to achieve a robust landing. In their experiments, state initializations were set in the hover condition arbitrarily. The idea was to find the helipad, then align with it, and land on it. The low-level (roll, pitch lateral and heading) controls were implemented with a proportional controller. The altitude behaviour was implemented with a PI controller. To make altitude control easier, the PI controller was split into three: sub-hover control, velocity and sonar control.

In Montgomery et al (1995), the control system proposed in the USC architecture is implemented as PD control loops with gains tuned by trial and error. In hovering conditions, the system is assumed linear (or linearized), thus multivariable linear control techniques such as Linear Quadratic Regulator (LQR) and H_∞ can be applied. Edgar, Hector and Carlos (2007) propose a flight control structure by combining PID, fuzzy and regulation control, using a nonlinear MIMO model for an X-Cell mini-helicopter platform.

Recently, intelligent control methods have become popular and an alternative to conventional methods. Intelligent control methods can act efficiently with nonlinear and unstable systems. In general, these methods can be categorized into three main techniques: fuzzy control, neural networks approach and genetic algorithm. Furthermore, these techniques can be combined with each other or with conventional methods to become hybrid techniques.

The genetic algorithm based on floating point representation has been modified to tune the longitudinal controller gains parameters of a two-bladed XCELL helicopter platform by Mario, G. P. (1997). First principle modelling is used to model the longitudinal behaviour of the platform. The author applied and compared the proposed design in both time and frequency domains. This algorithm shows faster convergence of the system with less computational time. Kadmiry and Driankov (2003) propose a combination of a Fuzzy Gain Scheduler (FGS) and a linguistics (Mamdani-type) controller. The authors used the FGS to control the attitude stability of the helicopter, whereas the linguistics controller was used to generate the inputs to the fuzzy controller for the given attitudes (z , roll, pitch, and yaw). The proposed controller scheme contains two loops; outer loop and inner loop. The inner loop represents the attitude controller and the outer loop deals with the translational rate variables. The controller was obtained and simulated based on a real nonlinear dynamic model of the platform.

This paper addresses the control problem of the HIROBO model platform which is being developed by University of Malaya team. The details of the system hardware and data collection are presented by (Zahari, et al, 2008) and (Taha. Z, T et al, 2010), respectively. The black box Nonlinear Autoregressive Model (NARX) modelling and identification of the platform is presented by (Deboucha, A et al, 2010). The use of this model was preferred because of its ability to handle instability and nonlinearity of complex nonlinear dynamic and unstable systems such as the helicopter. The author estimated the NARX based on collected flight data test (Taha. Z, T et al, 2010). In this paper, the obtained model by Deboucha. A, et al (2010) is used as plant model to be controlled. Due to the complementary between Model Predictive Control (MPC) and NARX, the MPC algorithm is applied to control the stability of the helicopter. MPC algorithm differs from other control strategies in: firstly, its multi-variables feature and secondly the possibility of using constraints. Therefore, reasonable results are anticipated. To prove the capabilities of the latter control, it has been simulated as model in the loop using SIMULINK. Furthermore, an xpctraget rapid prototype is developed to implement and test the controller to play the role of hardware in the loop test (HIL).

2. Model description

In this section, a brief description of the NARX black box model is presented. As reported previously, the identified orientation model of a Hirobo scale helicopter is obtained by (Deboucha, .A, et al, 2010). A standard NARX discrete time nonlinear multivariable model system with m outputs and r inputs is a general parametric form for modelling Black-box nonlinear systems with one step ahead prediction, which can be described by the following formula (Zhang & Ljung, 1999).

$$\hat{y}_m(k) = N [y_m(k-1), \dots, y_m(k-n_a), u_r(k-n_k), \dots, u_r(k-n_k-n_b+1)] + e_m(k) \quad (1)$$

where, $y_m = [\emptyset \ \theta \ \varphi]^T$ are the orientation behaviour of the helicopter and $u_r = [u_{elev}, u_{aile}, u_{colle}, u_{ped}]^T$, are the swash-plate control input vector. n_a, n_b are the matrices of the past outputs and inputs involved in the system, respectively, n_k is a matrix of the input delays from each input to each output. $N(\cdot)$ represents unknown nonlinear function, which in this case is computed by the neural network technique for estimating the nonlinearity of the system. Since the input vector to the system is from the swash-plate, the dynamics of the actuators i.e. servo positions are not included. The dynamics from actuators position to the swash-plate control inputs is assumed linear. This mapping model is presented by (Deboucha, A, et al, 2010).

The model which is presented in LTI state space form is linearized about a specified input vector and treated in terms of stability to ensure the performance of the model.

The linearized LTI state space model of the orientation dynamics of the helicopter platform is given by:

$$\begin{cases} x_m(k+1) = A_m x_m(k) + B_m u_r(k) \\ y_m(k) = C_m x_m(k) + D_m u_r(k) \end{cases} \quad (2)$$

where, $u_r(k)$ is the manipulated variable (control inputs to the helicopter), $y_m(k)$ is the process output which are the Euler angles and $x_m(k)$ is the state vector.

The state space matrices have obtained by (Deboucha, A, 2011) as,

$$A_m = \begin{bmatrix} 0.9994 & 0.02606 & -0.8908 & -0.1565 & -1.019 & -0.0052 & -0.7776 & -1.774 & 0.5769 \\ 0.002018 & 1 & 0.4515 & -0.1531 & 0.1697 & -0.2016 & 0.1594 & 0.2529 & -0.3566 \\ 0.006199 & 0.01822 & -0.03431 & 0.8172 & -0.2367 & 0.03092 & -0.1036 & -0.165 & 0.05891 \\ 0.005672 & 0.01667 & 0.115 & 0.1182 & -0.2007 & 0.0178 & 0.03018 & -0.1787 & 0.1455 \\ -0.01055 & -0.03101 & -0.3068 & 0.1792 & 0.3507 & -0.03528 & -0.4128 & -0.6006 & -0.00849 \\ -0.001958 & -0.005753 & -0.05692 & 0.03324 & 0.08312 & 0.006115 & 0.3217 & -0.2741 & -0.9223 \\ 0.01661 & 0.04881 & 0.483 & -0.282 & -0.5697 & 0.04318 & 0.2607 & -0.4691 & 0.4382 \\ -0.002215 & -0.006509 & -0.0644 & 0.03761 & 0.07598 & -0.005758 & -0.03476 & 0.06961 & -0.0563 \\ 0.007343 & 0.02158 & 0.2135 & -0.1247 & -0.2519 & 0.01909 & 0.1152 & -0.2308 & 0.1866 \end{bmatrix}$$

$$B_m = \begin{bmatrix} -0.04578 & -0.07926 & -0.02391 & 0.1732 \\ 0.001391 & 0.002409 & 0.0007266 & -0.005263 \\ 0.266 & 0.06843 & 0.02064 & -0.1495 \\ -0.9375 & 0.0626 & 0.01889 & -0.1368 \\ -0.05145 & -0.1165 & -0.03514 & 0.2998 \\ -0.03485 & -0.02161 & -0.00652 & 0.07897 \\ 0.1059 & 0.1834 & 0.05529 & 0.5978 \\ -0.01412 & -0.02441 & 0.9926 & 0.05346 \\ 0.04682 & -0.9189 & 0.02449 & -0.1771 \end{bmatrix}$$

$$C_m = \begin{bmatrix} -2.473 & -7.36 & 3.98 & 0.2138 & -7.225 & -8.439 & 1.357 & 0.9117 & 4.069 \\ 1.657 & 0.04449 & -1.599 & -0.04054 & 0.1303 & 0.5443 & -1.073 & -3 & 0.8277 \\ -0.1586 & -6.489 & 1.333 & 3.037 & 5.237 & 0.1375 & -4.339 & 0.3119 & -1.192 \end{bmatrix}$$

$$D_m = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

3. Control design

The objective of the MPC in this study is to bring the helicopter to its equilibrium i.e. the hovering condition. The controller is designed in the case where the translation velocities are

decaying to zeros and the Euler angles are limited with specific constraints explained in the next section. With these parameter criteria, the helicopter tries to stabilize into a hover state.

To design an MPC control, the above matrices have to be updated following the procedure below addressed in (Liuping, 2008 and Jay, et.al, 1994).

By taking the difference operation in both sides of the formula (2)

$$x_m(k+1) - x_m(k) = A_m(x_m(k) - x_m(k-1)) + B_m(u_r(k) - u_r(k-1)) \quad (3)$$

Then by defining

$$\begin{cases} \Delta x_m(k+1) = x_m(k+1) - x_m(k) \\ \Delta x_m(k) = x_m(k) - x_m(k-1) \\ \Delta u(k) = u_r(k) - u_r(k-1) \end{cases} \quad (4)$$

the updated states model would be as follows:

$$\Delta x_m(k+1) = A_m \Delta x_m(k) + B_m \Delta u_r(k) \quad (5)$$

From (2) and (5), the relation between the outputs of the system and the state variables could be deduced as

$$\begin{aligned} \Delta y_m(k+1) &= C_m \Delta x_m(k+1) + D_m \Delta u_r(k+1) \\ &= C_m A_m \Delta x_m(k) + C_m B_m \Delta u_r(k) + D_m \Delta u_r(k+1) \end{aligned} \quad (6)$$

It can also defined as

$$\Delta y_m(k+1) = y_m(k+1) - y_m(k) \quad (7)$$

The augmented state space model has a new state defined by

$$x(k) = [\Delta x_m(k)^T \ y_m(k)^T]^T, \quad (8)$$

where, the predicted state space model is deduced as

$$\begin{bmatrix} \Delta x_m(k+1) \\ y_m(k+1) \end{bmatrix} = \begin{bmatrix} A_m & o_m \\ C_m A_m & I_{m \times m} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y_m(k) \end{bmatrix} + \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} \Delta u_r(k) + \begin{bmatrix} o_m \\ D_m \end{bmatrix} \Delta u_r(k+1) \quad (9)$$

Because D_m is a zero matrix, the last term in the above equation can be eliminated, where, o_m is zeros matrix and $I_{m \times m}$ is the identity matrix.

To predict the future behaviour of the system $x(k+1)$, the current information of the plant model should be given by $x(k)$. Thus, the future control signals can be expressed by the following:

$$\Delta u_r(k), \Delta u_r(k+1), \dots, \Delta u_r(k+N_c-1), \quad (10)$$

where N_c is the control horizon dictating the number of parameters used to capture the future control trajectory. With the given information of the model $x(k)$, the future state variables are predicted for N_p number of samples as

$$x(k+1|k), x(k+2|k), \dots, x(k+p|k), \dots, x(k+N_p|k), \quad (11)$$

where $x(k+p|k)$ is the predicted state variable at $k+p$ with given current plant $x(k)$.

Based on the predicted state space model with the matrices (A, B, C) ,

where $A = \begin{bmatrix} A_m \\ C_m A_m \end{bmatrix}$, $B = \begin{bmatrix} 0_m \\ I_{m \times m} \end{bmatrix}$, $C = \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}$

the forward state variables could be calculated sequentially and finalized for a sample Np as

$$x(k + Np | k) = A^{Np} x(k) + A^{Np-1} Bx(k) + A^{Np-2} Bx(k + 1) + \dots + A^{Np-N_c} Bx(k + N_c - 1) \quad (12)$$

Similarly, from the predicted output state variables (8), the predicted outputs are written as follows:

$$y_m(k + Np | k) = CA^{Np} x(k) + CA^{Np-1} x(k) + CA^{Np-2} B\Delta u_q(k + 1) + \dots + CA^{Np-N_c} B\Delta u_q(k + N_c - 1) \quad (13)$$

All predicted variables are formulated in terms of the information on current state variable $x(k)$ and the future control $\Delta u(k + j)$, where $j = 1, 2, \dots, N_c - 1$

$$y_m = [y_m(k + 1 | k) \ y_m(k + 2 | k) \ y_m(k + 3 | k), \dots, y_m(k + Np | k)]^T \quad (14)$$

$$\Delta U = [\Delta u_q(k) \ \Delta u_q(k + 1) \ \Delta u_q(k + 2) \ \dots \ \Delta u_q(k + N_c - 1)]^T \quad (15)$$

From the above formulas, the output vector is concluded as follows:

$$Y = Fx(k) + Q\Delta U \quad (16)$$

To sum up, the predictive model of the helicopter's attitude is updated in order to deal with the MPC design. In the next section the optimization algorithm of the MPC is treated based on a given set point (reference model).

3.1 Control optimization

For a given set point for Euler angles $Rs(k)$ at sample time k within a prediction horizon Np , the objective of the MPC system is to bring the predicted output behaviour of the helicopter as close as possible to the set-point signals, where firstly, assuming that the set-point signals remain constant in the optimization window. This objective is then translated into a design to find the 'best' control parameter vector ΔU such that an error function between the set-points and the predicted outputs is minimized.

Assuming the vector set-point data as $Rs = [r_1(k) \ r_2(k) \ r_3(k)]^T$,

where $r_1(k)$, $r_2(k)$ and $r_3(k)$ are the set points of roll angle, pitch angle and yaw angle, respectively. The cost function J which reflects the control objective is defined as

$$J = (Rs - Y)^T (Rs - Y) + \Delta U^T \bar{R} \Delta U \quad (17)$$

where the first term is linked to the objective of minimizing the error between the predicted output vector and the set-point signals, while the second term concerns the consideration of ΔU when the objective function J is as small as possible. \bar{R} is a diagonal matrix in the form that $\bar{R} = r_w I_{N_c \times N_c}$ ($r_w \geq 0$) where r_w is used as tuning parameter for the desired closed loop performance. The goal would be solely to make the error $(Rs - Y)^T (Rs - Y)$ as small as possible. In the case of large r_w the cost function is interpreted as the situation where would carefully consider how large the ΔU might be and cautiously reduce the error

$$(Rs - Y)^T (Rs - Y)$$

To find the optimal ΔU that will minimize the cost function J we have $y = Fx(k) + Q\Delta U$,

$$\text{where } F = \begin{bmatrix} CA \\ CA^2 \\ CA^3 \\ \vdots \\ CA^{N_p} \end{bmatrix} Q = \begin{bmatrix} CA & 0 & \dots & 0 \\ CAB & CA & \dots & 0 \\ CA^2B & CAB & \dots & 0 \\ \vdots & \vdots & \dots & 0 \\ CA^{N_p-1}B & CA^{N_p-2}B & \dots & CA^{N_p-N_c}B \end{bmatrix} \quad (18)$$

and, $J = (Rs - Fx(k))^T (Rs - Fx(k)) - 2\Delta U^T Q^T (Rs - Fx(k)) + \Delta U^T (Q^T Q + \bar{R})\Delta U$
From the first derivative of the cost function J

$$\frac{\partial J}{\partial \Delta U} = -2Q^T (Rs - Fx(k)) + 2(Q^T Q + \bar{R})\Delta U \quad (19)$$

The required condition is $\frac{\partial J}{\partial \Delta U} = 0$

From which the optimal solution for the control signal is found:

$$\Delta U = (Q^T Q + \bar{R})^{-1} * Q^T (Rs - Fx(k)) \quad (20)$$

where $(Q^T Q + \bar{R})^{-1}$ is the Hessian Matrix in the optimization.

The MPC is designed based on the above updated model and optimized cost function. One of the criteria of MPC, the constraints in both inputs and outputs has been chosen with regard to the behaviour of the vehicle during the flight test. Based on the flight test, the range outputs that guarantee the behaviour of the helicopter in hovering condition were approximately ± 15 degrees while the input ranges were approximately ± 10 degrees.

The second feature that has to be defined in designing an MPC is the selection of the prediction horizon N_p and the control horizon N_c . In this work, the chosen N_p value is 25 and the chosen N_c is five-control inputs horizon.

Based on these criteria and to satisfy the equation ($\frac{\partial J}{\partial \Delta U} = 0$), the best tuning output weights to stabilize the model were found to be the following matrix:

$$Q^T Q = \begin{bmatrix} 1.1735 & 0 & 0 \\ 0 & 1.1735 & 0 \\ 0 & 0 & 1.1735 \end{bmatrix} \quad (21)$$

While, the best input weights were found to be

$$\bar{R} = \begin{bmatrix} 0.08521 & 0 & 0 & 0 \\ 0 & 0.08521 & 0 & 0 \\ 0 & 0 & 0.08521 & 0 \\ 0 & 0 & 0 & 0.08521 \end{bmatrix} \quad (22)$$

To validate the previously designed controller that stabilizes the helicopter, the designed MPC is implemented into the same hardware described in the previous section. Using xpc-target software in SIMULINK, the model (in Fig 1) have been developed and deployed into the target PC (PC-104). The model contains the IMU sensor software, the MPC, and the corresponding C/T blocks for both capturing and generating PWM signals. The IMU DATA RECEIVE software reads the behaviour of the helicopter (angular position, acceleration...etc) and sends these data to the controller. The MPC generates the required swash-plate angles

as well the pedal control by setting the corresponding servo positions. The relationship between the swash plate which has 120° layout and the servo position is given by the transformation matrix from as described by (Deboucha et al, 2010):

$$e = \begin{bmatrix} 0.3333 & 0.3333 & 0.3333 \\ 0.5000 & 0 & -0.5000 \\ 0.3333 & -0.6667 & 0.3333 \end{bmatrix}$$

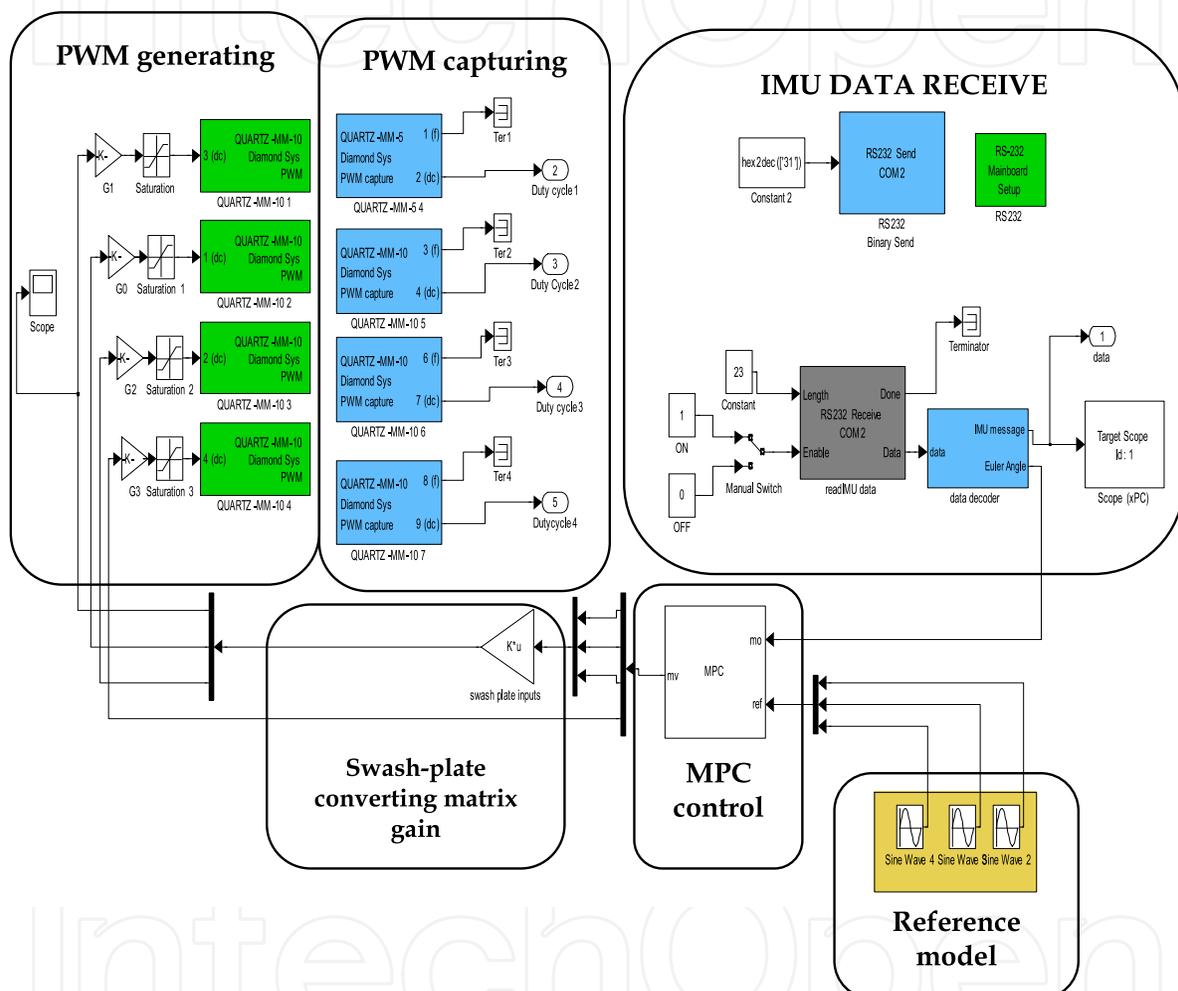


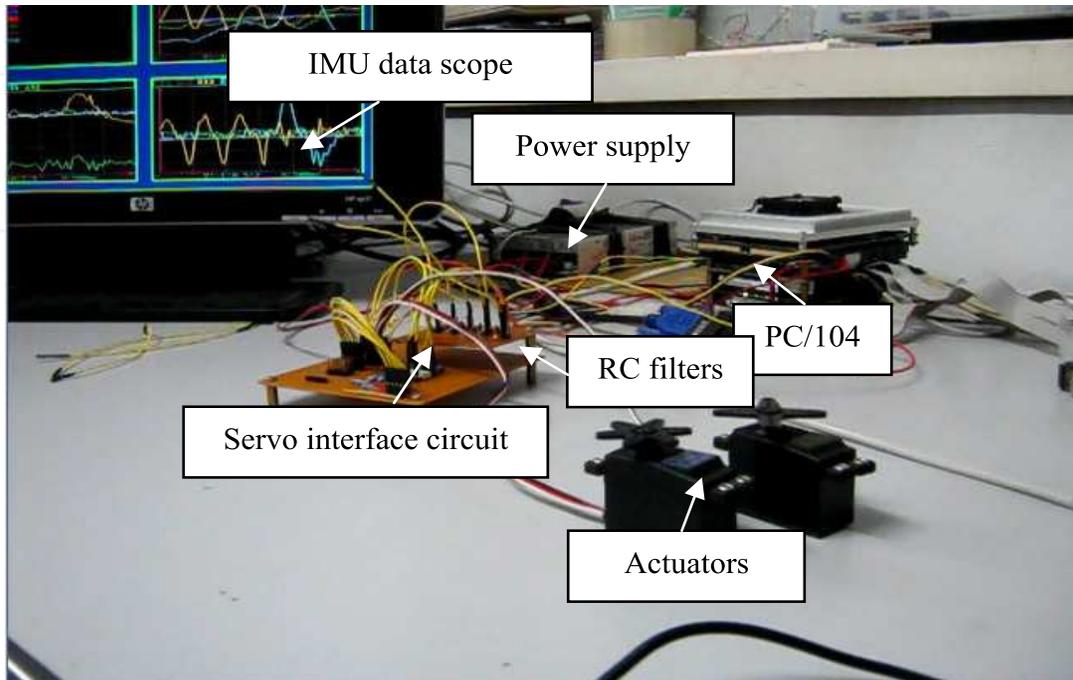
Fig. 1. Simulink Block Diagram model of the MPC implementation

The servos' positions are controlled through a set of PWM signals as described by (Deboucha et al, 2011). The pedal control has no effect on the swash-plate layout. Thus, its PWM signal is determined from the servo position. The corresponding C/T blocks (QUARTZ MM) in the SIMILINK model set the frequency of the PWM signals, which is 50 Hz. To capture the actual PWM signals, the QUARTZ PWM capturing block is used for each generated signal from the MPC. The saturation blocks are used to limit the duty cycle if any over-range of its values.

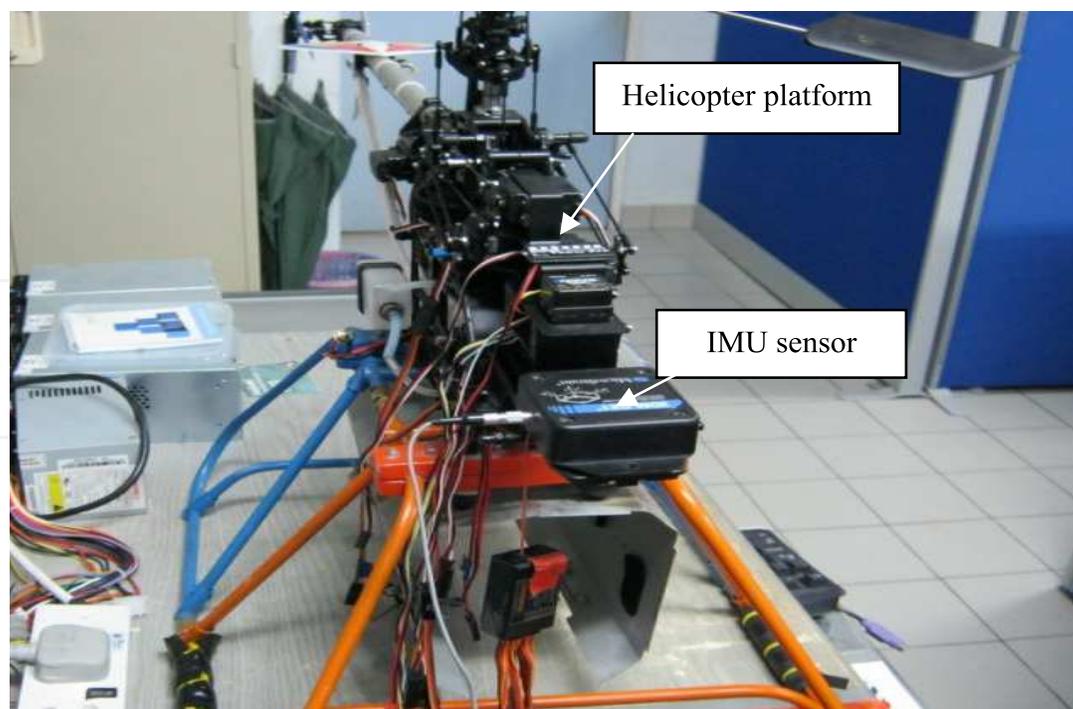
The main hardware used in this work are 1): A host computer, 2): PC-104, 3): Counter/Timer I/O board, 4): 3DM-GX1 Inertial measurement Unit, 5): two onboard servos (Futaba S3001 and Futaba S9254), and 6): Helicopter platform. The sampling time used for the experiment is 0.03s.

4. Experimental setup

A part of this work is to implement the above Simulink model to the target PC-104. Fig.2 presents the overall experimental prototype setup. The IMU sensor was mounted on the



(a): xpctarget prototype configuration



(b): Helicopter platform

Fig. 2. Experimental setup

nose of the helicopter and connected to the target PC-104 through a serial port. The corresponding pins of the I/O counter/timer board were connected to the servo actuators via a servo interface circuit. This circuit is an RC filter used to protect the I/O board from noises produced by the Helicopter components such as the actuators. The PC-104's processor runs the developed system in real time operating system.

5. Simulation results

To test the designed MPC, a simulation of the helicopter performance under different set-points is studied. The step response of the helicopter with the introduction of the disturbance in the roll angle is presented in Fig.3. The amplitudes of the roll, pitch and yaw angles are 12, 10 and 13 degrees, respectively. At 5 seconds, a disturbance with amplitude of 5 degrees is introduced. It can be seen that the controller damps down the amplitude of the angle to 13.3 out of 17 degrees in approximately 5 seconds. The effect of the disturbance on the other states is less and it would appear that there is a small steady state error in the yaw angle after 5 seconds of simulation.

The designed MPC has also been tested to track a square wave with a variety of amplitudes for Euler angles. The performance of the controller is good for all the utilized amplitudes, as illustrated in Figures 4, 5 and 6.

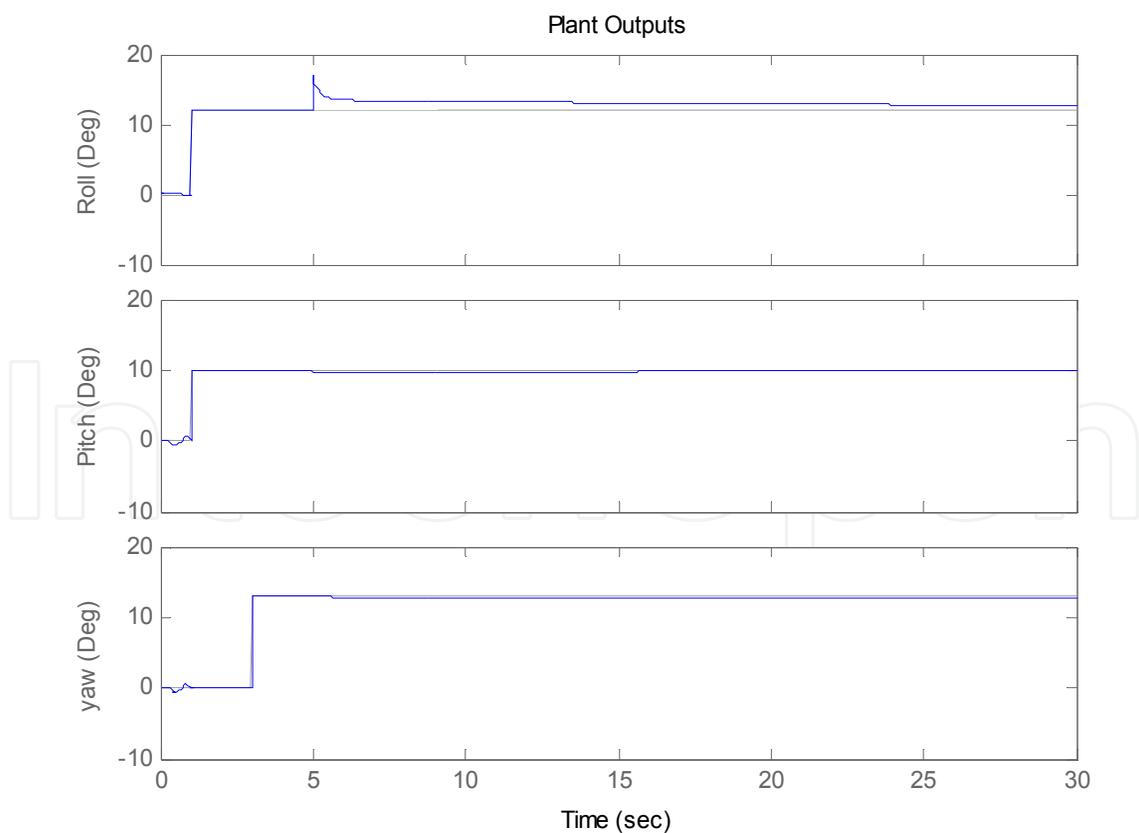


Fig. 3. Step responses

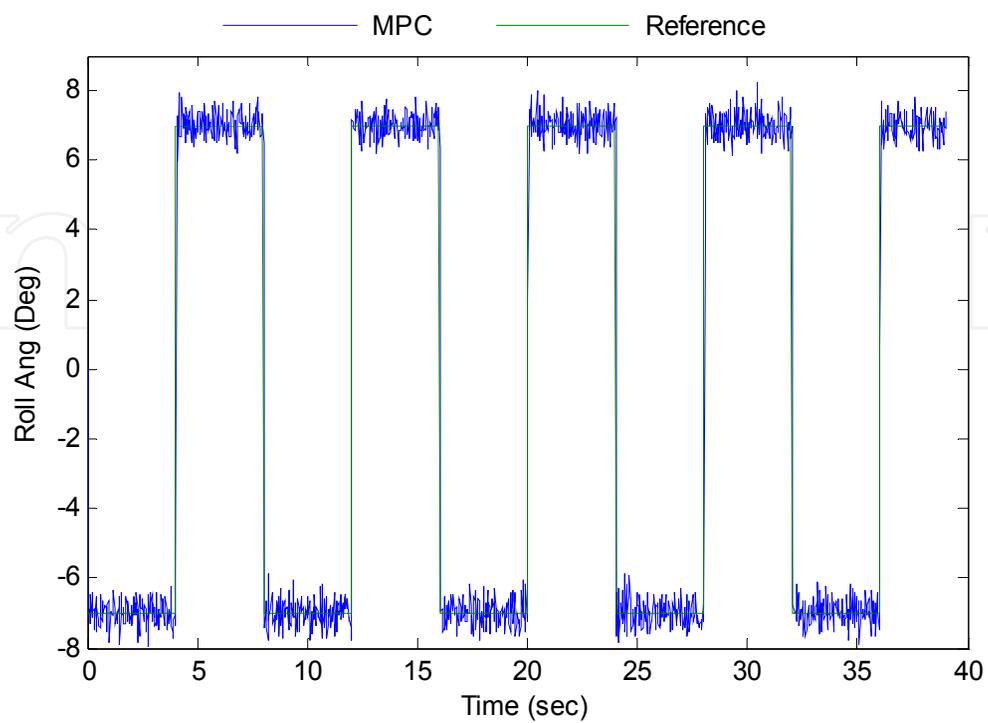


Fig. 4. Square wave roll angle tracking with the MPC controller.

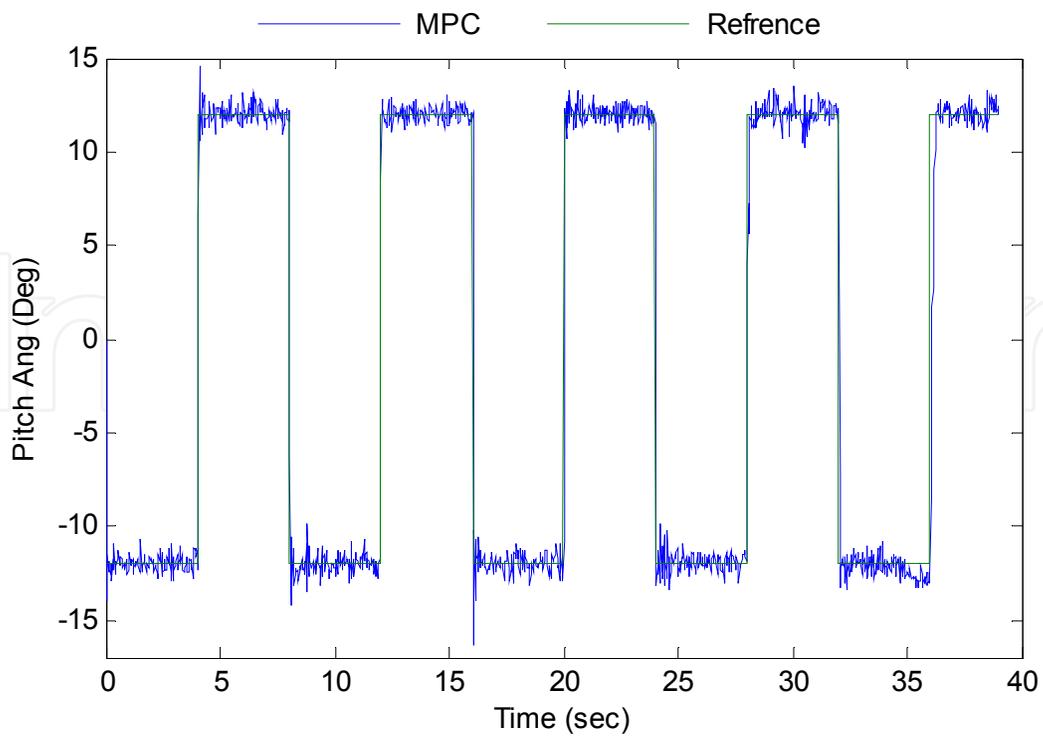


Fig. 5. Square wave pitch angle tracking with the MPC controller

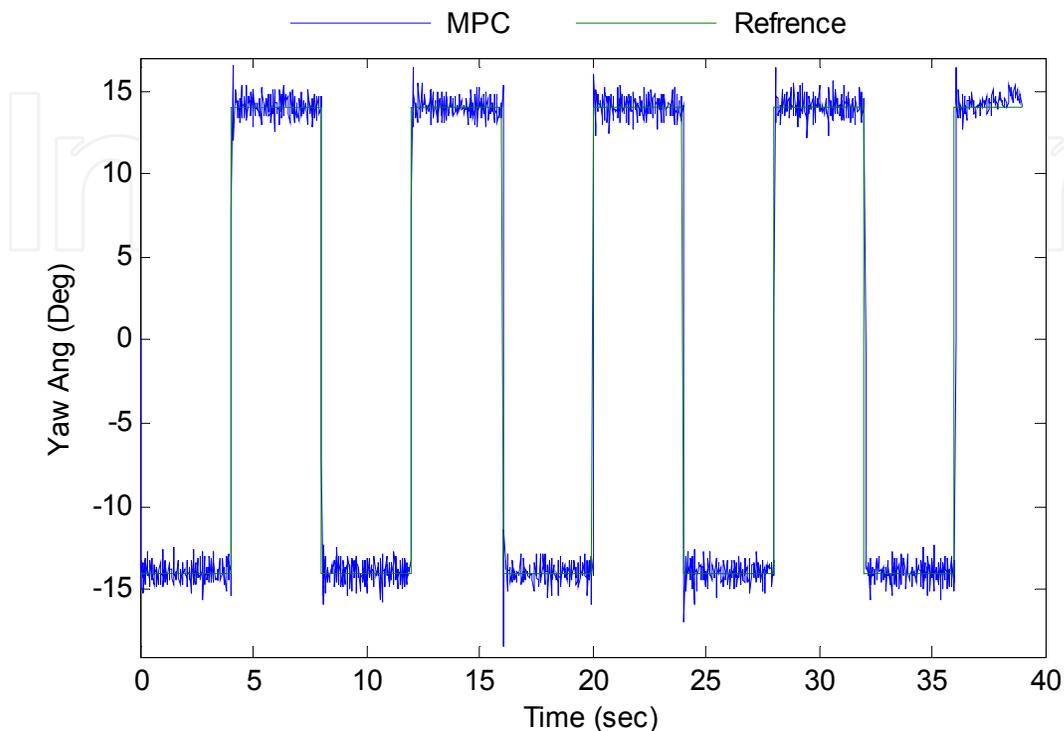


Fig. 6. Square wave yaw angle tracking with the MPC controller.

5.1 Experiment results

This section presents the implementation of developed hardware in the loop system. Two experiments were conducted in this work. The first is conducted where the flight test data were used as reference model and disabling the role of the IMU. Figures 4, 5 and 6 present the generated inputs by the real time MPC to the system to follow the reference model compared with the given inputs system during the flight test. From Fig 1, the collected PWM signals are collected as duty cycle; therefore it has to be transferred to the corresponding angles for each actuator. Instead of activating the IMU software, the feedback to the MPC is the reference model itself. It is noticeable that the generated inputs by the MPC do not follow closely the actual inputs used for modelling task. This is because the MPC is designed based on linearized model of the platform.

As preliminary step to investigate a real autonomous flight, a second experiment is carried out where the IMU software is enabled (fig1) to test its functioning and also to assess how the MPC is sensitive with disturbances. To achieve these criterions, the reference model is settled to zero and the nose of the helicopter is shaken slightly with small variation, the position of actuators change in order to bring back the system into the still condition i.e. the MPC gives the action to the system.

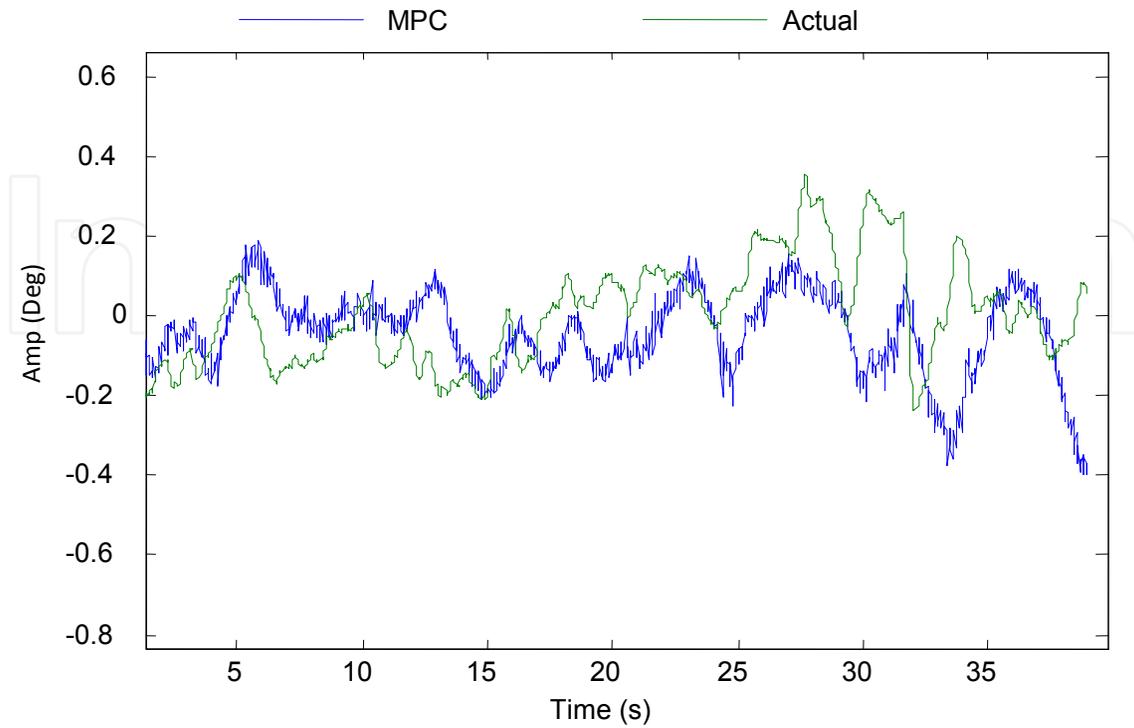


Fig. 7. generated lateral input (MPC) vs lateral command

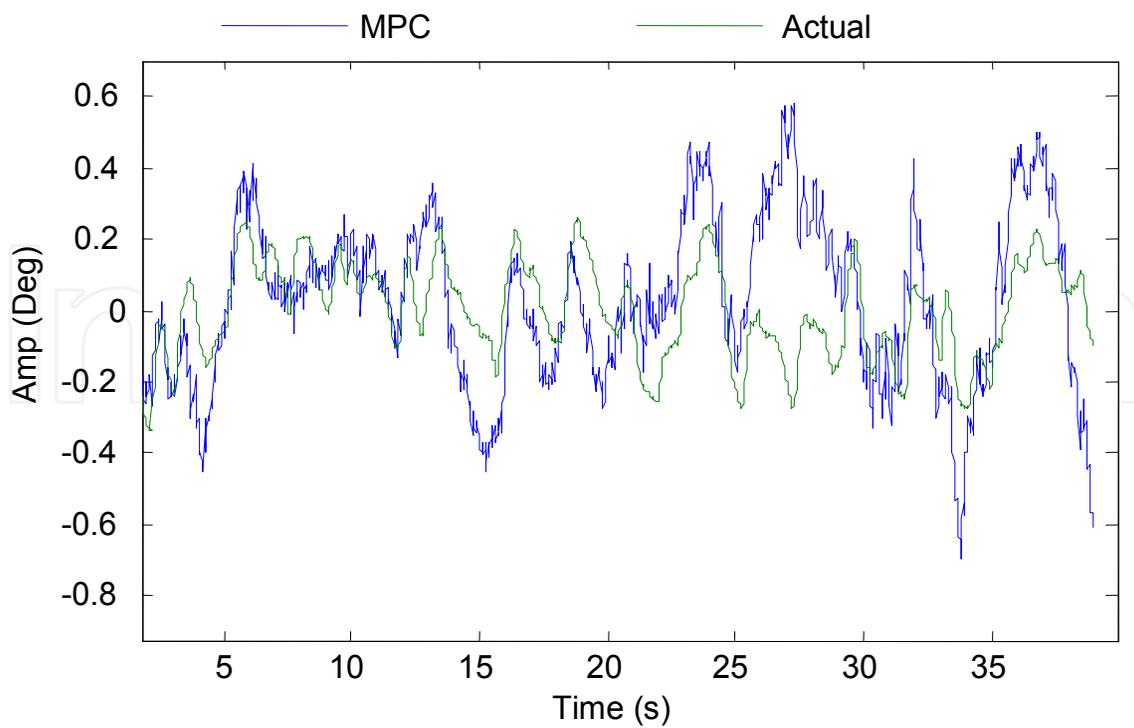


Fig. 8. generated longitudinal input (MPC) vs longitudinal command.

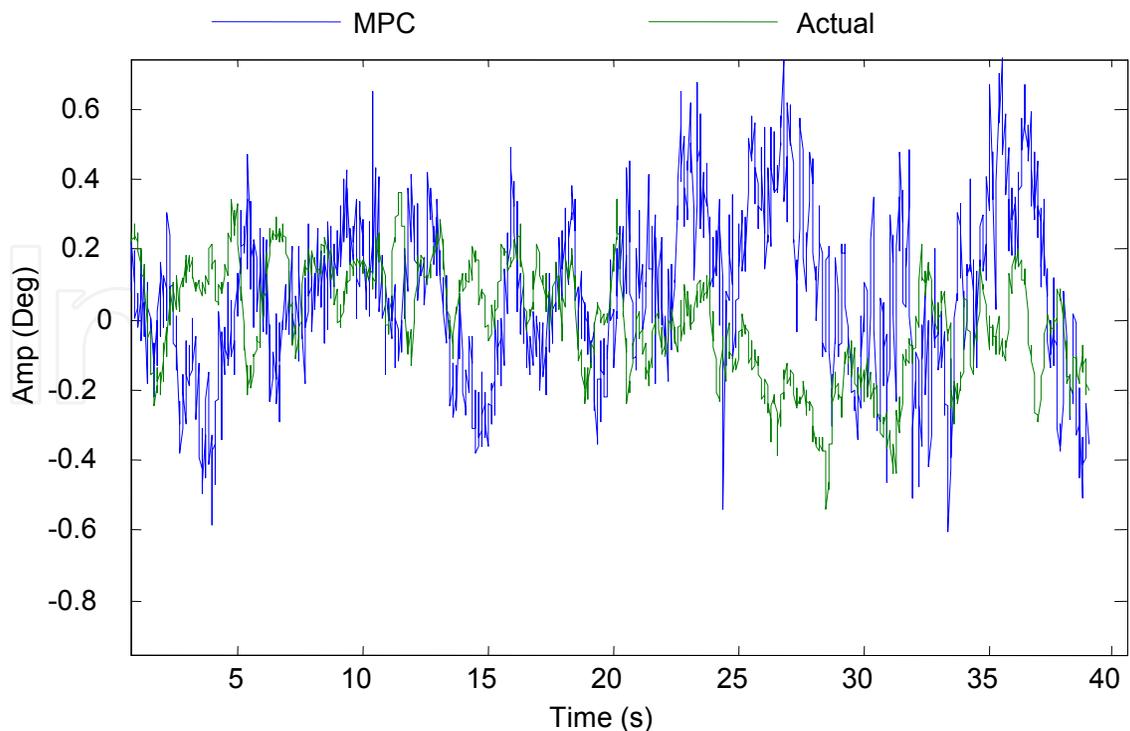


Fig. 9. generated pedal input by MPC vs pedal command.

6. Conclusion

In this paper, a MIMO model predictive control (MPC) system is implemented into hardware in the loop based xpc-target rapid-prototype system to guarantee the equilibrium of the helicopter platform. The MIMO MPC design was carried out using an experimentally estimated model of the Helicopter. The performance of the controller is tested in simulation and hardware in the loop using different set-point scenarios. Simulation results showed that the controller can efficiently stabilize the system under all the introduced disturbances. A real time controller based on xpc-target rapid prototype is developed to implement the proposed controller. The ground results proved that the proposed real time MPC can sufficiently stabilize the system in hovering conditions.

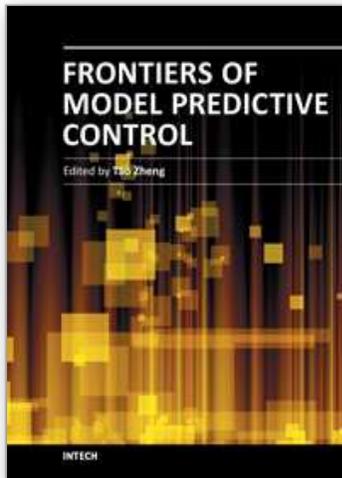
7. Acknowledgment

The authors gratefully acknowledge the support from MOSTI (Malaysia) Sciencefund: Hardware-in-the-Loop Simulation for Control System of Mini Scale Rotorcraft project No. 13-01-03-SF0024. The previous team researchers Mr Terran and KC Yap are gratefully acknowledged for their help.

8. References

Abdelhakim Deboucha, Zahari Taha, 2010. Identification and control of small-scale helicopter. *Journal of Zhejiang University Science A (Applied Physics & Engineering)*. Vol. 11 (12), pp. 978-985

- Abdelhakim Debooucha, 2011. Neural Network-based modelling and control design of small scale helicopter. *Master's Dissertation, University of Malaya, Malaysia.*
- Cai, G., Peng, K., Chen, B. M., Lee T. H., (2005). Design and assembling of a UAV helicopter system. *IEEE Proceedings of the 5th International Conference on Control and Automation* (pp. 697-702), Hungary.
- Del-Cerro, J., Barrientos. A., Campoy. P., & Garcia. P. (2002). An autonomous helicopter guided by computer vision for inspection of overhead power cable. *International Conference on Intelligent Robots and Systems* (pp. 69-78). Switzerland.
- Doherty, P., Granlund, G., Kuchcinski, K., Sandewall, E., Nordberg, K., Skarman, E., Wiklund, J., (2000). The WITAS unmanned aerial vehicle project. *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI)* (pp. 747-755). Berlin, Germany.
- Eck, C., Chapuis, J., & Geering, H. P. (2001). Inexpensive autopilots for small unmanned helicopters. *Proceedings of the Micro and Mini Aerial Vehicles Conference.* (pp. 1-8). Brussels, Belgium.
- Edgar, N. S., Hector, M. B., Carlos, M. V. (2007). Combining fuzzy, PID and regulation control for an autonomous mini-helicopter. *Information Sciences*, 177(10), 1999-2022. doi:10.1016/j.ins.2006.10.001.
- Garcia, R. D. & Valavanis, K. P., (2009). The implementation of an autonomous helicopter testbed. *Journal of Intelligent & Robotics Systems*, 54(3). doi: 10.1007/10846-008-9273
- Jay, H. L., Manfred. M., Carlos. E. G. (1994). State space interpretation of model predictive control. *Automatica*, 30(4), 707-714. doi:10.1016/0005-1098(94)90159-7
- Kadmiry, B., Driankov, D. (2004). A fuzzy flight controller combining linguistic and model-based fuzzy control. *Fuzzy Sets and Systems*, 146, 313-347.
- Kim, H. J., and Shim, D. H. (2003). A flight control system for aerial robots: Algorithms and experiments. *Control Engineering Practice*, 11, 1389- 1400
- Liuping, W. (2008). *Model predictive control system design and implementation using MATLAB.* Melbourne, Australia: Springer.
- Mario, G. P. (1997). A modified genetic algorithm for the design of autonomous helicopter control system. *American Institute of Aeronautics and Astronautics*, pp. 1-10.
- Montgomery, J. F., & Bekey, G. A. (1998). Learning helicopter control through "teaching by showing". *In Proceedings of the 37th IEEE Conference on Decision and Control.* Tampa, Florida, USA.
- Montgomery, J. F., Fagg, A. H., & Bekey, G. A. (1995). The USC AFV-I: A behaviour based entry. *1994 Aerial Robotics Competition IEEE Expert*, pp. 16-22.
- Taha. Z., Tang. Y. R., Yap. K. C. (2010). Development of an onboard system for flight data collection of small- scale helicopter. *Mechatronics, Vol.No*, 1-13.
- Wenbing, C., Xia, X., Ming, L., Yunjian, G., & Min, Z. (2007). Sensor based adaptive neural network control for small-scale unmanned helicopter. *International Conference on Information Acquisition* (pp. 187-191). Jeju City, Korea.
- Srikanth, S., James, F. M., & Gaurav, S. S. (2003). Visually guided landing of an unmanned helicopter. *IEEE Transactions on Robotics and Automation*, 19(3), 371-381. doi: 10.1109/TRA.2003.810239.
- Zahari, T., Yap. K. C., Tang, Y. R. (2008). Avionics box for small unmanned helicopter. *Proceedings of the 9th Asia Pacific Industrial Engineering & Management Systems Conference* (pp. 2841 - 2845). Bali, Indonesia.
- Zhang, Q. & Ljung., (2004). Multiples steps prediction with nonlinear ARX models. *Proceedings of the 6th international Federation of Automatic Control (IFAC). Nonlinear control systems* (pp. 309-314).



Frontiers of Model Predictive Control

Edited by Prof. Tao Zheng

ISBN 978-953-51-0119-2

Hard cover, 156 pages

Publisher InTech

Published online 24, February, 2012

Published in print edition February, 2012

Model Predictive Control (MPC) usually refers to a class of control algorithms in which a dynamic process model is used to predict and optimize process performance, but it is can also be seen as a term denoting a natural control strategy that matches the human thought form most closely. Half a century after its birth, it has been widely accepted in many engineering fields and has brought much benefit to us. The purpose of the book is to show the recent advancements of MPC to the readers, both in theory and in engineering. The idea was to offer guidance to researchers and engineers who are interested in the frontiers of MPC. The examples provided in the first part of this exciting collection will help you comprehend some typical boundaries in theoretical research of MPC. In the second part of the book, some excellent applications of MPC in modern engineering field are presented. With the rapid development of modeling and computational technology, we believe that MPC will remain as energetic in the future.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Zahari Taha, Abdelhakim Deboucha, Azeddein Kinsheel and Raja Ariffin Bin Raja Ghazilla (2012).
Development of Real-Time Hardware in the Loop Based MPC for Small-Scale Helicopter, Frontiers of Model Predictive Control, Prof. Tao Zheng (Ed.), ISBN: 978-953-51-0119-2, InTech, Available from:
<http://www.intechopen.com/books/frontiers-of-model-predictive-control/-development-of-real-time-hardware-in-the-loop-based-mpc-for-small-scale-helicopter>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the [Creative Commons Attribution 3.0 License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen