PID vs LQ control techniques applied to an indoor micro quadrotor

PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor

Samir Bouabdallah, André Noth and Roland Siegwart
Autonomous Systems Laboratory
Swiss Federal Institute of Technology
Lausanne, Switzerland
Email: {samir.bouabdallah, andre.noth, roland.siegwart}@epfl.ch

***Abstract*— The development of miniature flying robots has become a reachable dream thanks to the new sensing and actuating technologies. Micro VTOL[1] systems represent a useful class of flying robots because of their strong abilities for small-area monitoring and building exploration. In this paper, we present the results of two model-based control techniques applied to an autonomous four-rotor micro helicopter called Quadrotor. A classical approach (PID) assuming a simplified dynamics and a modern technique (LQ), based on a more complete model. Various simulations were performed and several tests on the bench validate the control laws. Finally, we present the results of the first test in flight with the helicopter released. These developments are part of the OS4[2] project in our lab[3].**

I. Introduction

The important progress over the last years in sensing technologies, high density power storage, and data processing have made the development of micro unmanned aerial vehicles (UAV) possible. In the field of sensing technologies, industry can provide currently a new generation of integrated micro IMU[4] composed generally of MEMS[5] technology inertial sensors and magneto-resistive sensors. The last technology in high density power storage offers about 180W/kg which is a real jump ahead especially for micro aerial robotic. This technology was originally developed for handheld applications and is now widely used in aerial robotics. The cost and size reduction of such systems makes it very interesting for the civilian market in several applications like for small-area monitoring and building exploration. Simultaneously, this reduction of cost and size implies performance limitation and thus a more challenging control. Moreover, the miniaturization of the inertial sensors imposes the use of MEMS technology which is still less efficient than the conventional sensors because of noise and drift. The use of low-cost IMU is synonym of less efficient data processing and thus a bad orientation data prediction in addition to a weak drift rejection. On the other hand, and in spite of the latest progress in miniature actuators, the scaling laws are still unfavorable and one has to face the problem of actuators saturation. That is to say, even though the design of micro aerial robots is possible, the control is still a challenging goal.

A. The OS4 Project

This recent project, initiated at the Autonomous Systems Laboratory (EPFL), focuses on micro VTOL vehicles evolving towards full autonomy in indoor environments. The long term goal is to allow indoor navigation using various techniques. The approach advocated for this project is to simultaneously work on design and control. This original approach makes it possible to simplify the control by design adaptation, and vice versa. A Quadrotor configuration vehicle has been chosen for the experiments.

B. Quadrotor Configuration

The Quadrotor concept has been around for a long time. The Breguet-Richet Quadrotor helicopter *Gyroplane No.1* built in 1907 is reported to have lifted into flight [1]. One can describe the vehicle as having four propellers in cross configuration. The two pairs of propellers (1,3) and (2,4) turn in opposite directions. By varying the rotor speed, one can change the lift force and create motion. Thus, increasing or decreasing the four propeller's speeds together generates vertical motion. Changing the 2 and 4 propeller's speed conversely produces roll rotation coupled with lateral motion. Pitch rotation and the corresponding lateral motion result from 1 and 3 propeller's speed conversely modified as described in figure 1. Yaw rotation is more subtle, as it results from the difference in the counter-torque between each pair of propellers. In spite of the four actuators, the Quadrotor is still an under-actuated and dynamically unstable system.

Advantages and Drawbacks: The space and energy requirements are definitely the main disadvantages of the Quadrotor. However, this concept offers a better payload and is potentially simpler to build and to control. This could be a decisive advantage. Table I gives a rapid idea about Quadrotor's advantages and drawbacks.

TABLE I
Quadrotor main Advantages & Drawbacks.

Advantages	Drawbacks
Rotor mechanics simplification	Weight augmentation
Payload augmentation	High energy consumption
Gyroscopic effects reduction	

[1] Vertical Take-Off and Landing
[2] Omnidirectional Stationary Flying Outstretched Robot
[3] Autonomous Systems Lab
[4] Inertial Measurement Unit
[5] Micro Electromechanical Systems

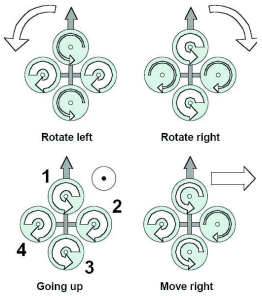

Fig. 1. Quadrotor concept motion description, the arrow width is proportional to propeller rotational speed.

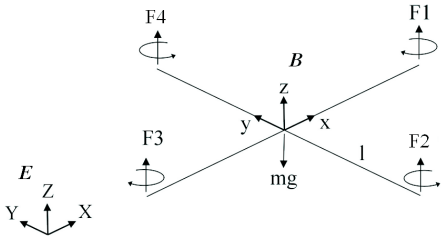

Fig. 2. Quadrotor configuration, frame system with a body fixed frame B and the inertial frame E.

II. QUADROTOR DYNAMIC MODELLING

The first step before the control development is an adequate dynamic system modelling [2], [3]. Especially for lightweight flying systems, the dynamic model ideally includes the gyroscopic effects resulting from both the rigid body rotation in space, and the four propeller's rotation. These aspects have been often neglected in previous works.

Let us consider earth fixed frame E and body fixed frame B, as seen in figure 2. Using Euler angles parametrization, the airframe orientation in space is given by a rotation R from B to E, where $R \in SO3$ is the rotation matrix. The dynamic model is derived using Euler-Lagrange formalism [4] under the following assumptions:

- The structure is supposed to be rigid.
- The structure is supposed symmetrical.
- The center of mass and the body fixed frame origin are assumed to coincide.
- The propellers are supposed rigid.
- The thrust and drag are proportional to the square of the propeller speed.

1) kinematics: For any point of the airframe expressed in the earth fixed frame, we can write:

$$\begin{cases} r_X = (c\psi c\theta)x + (c\psi s\theta s\phi - s\psi c\phi)y + (c\psi s\theta c\phi + s\psi s\phi)z \\ r_Y = (s\psi c\theta)x + (s\psi s\theta s\phi - c\psi c\phi)y + (s\psi s\theta c\phi + c\psi s\phi)z \\ r_Z = (-s\theta)x + (c\theta s\phi)y + (c\theta c\phi)z \\ c : cos, s : sin \end{cases} \tag{1}$$

The corresponding velocities are obtained by differentiation of (1), and thus the squared magnitude of the velocity for any point is given by:

$$v^2 = v_X^2 + v_Y^2 + v_Z^2 \tag{2}$$

Energy: From the equation (2), and by assuming that the inertia matrix is diagonal, one can extract the kinetic energy expression:

$$\begin{aligned} T &= \frac{1}{2} I_x (\dot{\phi} - \dot{\psi} s\theta)^2 \\ &+ \frac{1}{2} I_y (\dot{\theta} c\phi + \dot{\psi} s\phi c\theta)^2 \\ &+ \frac{1}{2} I_z (\dot{\theta} s\phi - \dot{\psi} c\phi c\theta)^2 \end{aligned} \tag{3}$$

And using the well known potential energy formula, one can express it in the earth fixed frame as:

$$\begin{aligned} V &= \int x dm(x)(-gs\theta) \\ &+ \int y dm(y)(gs\phi c\theta) \\ &+ \int z dm(z)(gc\phi c\theta) \end{aligned} \tag{4}$$

Equation of Motion: Using the Lagrangian and the derived formula for the equations of motion:

$$L = T - V \quad , \quad \Gamma_i = \frac{d}{dt}(\frac{\partial L}{\partial \dot{q}_i}) - \frac{\partial L}{\partial q_i} \tag{5}$$

Where $\dot{q}_i$ are the generalized coordinates and Γ_i the generalized forces. The three equations of motion are then:

$$\begin{aligned} \ddot{\phi} &= \dot{\theta}\dot{\psi}(\frac{I_y - I_z}{I_x}) \\ \ddot{\theta} &= \dot{\phi}\dot{\psi}(\frac{I_z - I_x}{I_y}) \\ \ddot{\psi} &= \dot{\phi}\dot{\theta}(\frac{I_x - I_y}{I_z}) \end{aligned} \tag{6}$$

On the other hand, the nonconservative torques acting on "OS4" result firstly from, the action of the thrust forces difference of each pair, see figure 2:

$$\begin{aligned} \tau_x &= bl(\Omega_4^2 - \Omega_2^2) \\ \tau_y &= bl(\Omega_3^2 - \Omega_1^2) \\ \tau_z &= d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{aligned} \tag{7}$$

Secondly from the gyroscopic effect resulting from the propellers rotation:

$$\begin{aligned} \tau_x^{'} &= J\omega_y(\Omega_1 + \Omega_3 - \Omega_2 - \Omega_4) \\ \tau_y^{'} &= J\omega_x(\Omega_2 + \Omega_4 - \Omega_1 - \Omega_3) \end{aligned} \tag{8}$$

The Derived Dynamic Model: The Quadrotor dynamic model describing the roll, pitch and yaw rotations contains then three terms which are the gyroscopic effect resulting from the rigid body rotation, the gyroscopic effect resulting from the propeller rotation coupled with the body rotation and finally the actuators action:

$$\begin{aligned} \ddot{\phi} &= \dot{\theta}\dot{\psi}(\frac{I_y - I_z}{I_x}) - \frac{J}{I_x}\dot{\theta}\Omega + \frac{l}{I_x}U_1 \\ \ddot{\theta} &= \dot{\phi}\dot{\psi}(\frac{I_z - I_x}{I_y}) + \frac{J}{I_y}\dot{\phi}\Omega + \frac{l}{I_y}U_2 \\ \ddot{\psi} &= \dot{\phi}\dot{\theta}(\frac{I_x - I_y}{I_z}) + \frac{1}{I_z}U_3 \end{aligned} \tag{9}$$

The system's inputs are posed U_1, U_2, U_3 and Ω as a disturbance, obtaining:

$$\begin{cases} U_1 = b(\Omega_4^2 - \Omega_2^2) \\ U_2 = b(\Omega_3^2 - \Omega_1^2) \\ U_3 = d(\Omega_1^2 + \Omega_3^2 - \Omega_2^2 - \Omega_4^2) \\ \Omega = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \end{cases} \quad (10)$$

where :

Symbol	definition
R	rotation matrix
ϕ	roll angle
θ	pitch angle
ψ	yaw angle
Ω_i	rotor speed
$I_{x,y,z}$	body inertia
J	propeller inertia
b	thrust factor
d	drag factor
l	lever

In this paper we focus on the rotational dynamics as the linear motion of the Quadrotor is a consequence of the rotations.

Rotor Dynamics: The rotors are driven by DC-motors with the well known equations:

$$\begin{cases} L\frac{di}{dt} = u - Ri - k_e\omega_m \\ \\ J\frac{d\omega_m}{dt} = \tau_m - \tau_d \end{cases} \quad (11)$$

As we use a small motor with a very low inductance, the second order DC-motor dynamics may be approximated:

$$J\frac{d\omega_m}{dt} = -\frac{k_m^2}{R}\omega_m - \tau_d + \frac{k_m}{R}u \quad (12)$$

By introducing the propeller and the gearbox models, the equation (12) may be rewritten:

$$\begin{cases} \dot{\omega}_m = -\frac{1}{\tau}\omega_m - \frac{d}{\eta r^3 J_t}\omega_m^2 + \frac{1}{k_m \tau}u \\ with: \\ \frac{1}{\tau} = \frac{k_m^2}{RJ_t} \end{cases} \quad (13)$$

The equation (13) can be linearized around an operation point $\dot{w}_0$ to the form $\dot{w}_m = -Aw_m + Bu + C$ with:

$$A = \left(\frac{1}{\tau} + \frac{2dw_0}{\eta r^3 J_t}\right), \quad B = \left(\frac{1}{k_m \tau}\right), \quad C = \frac{d\omega_0^2}{\eta r^3 J_t} \quad (14)$$

Symbol	Definition
u	motor input
k_e	back EMF constant
k_m	torque constant
ω_m	motor angular speed
τ_m	motor torque
τ_d	motor load
τ	motor time-constant
R	motor internal resistance
r	gear box reduction ratio
η	gear box efficiency

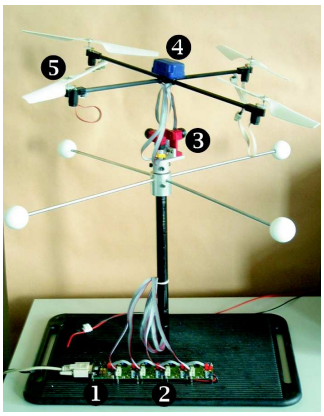

Fig. 3. OS4 test-bench for stabilization strategies testing, 3DOF are locked, the cross is made with carbon rods and the flying system weight is about 240g. 1)RS232 to I2C translator, 2)Motor modules, 3)3D captured universal joint, 4)Micro IMU, 5)Propulsion group.

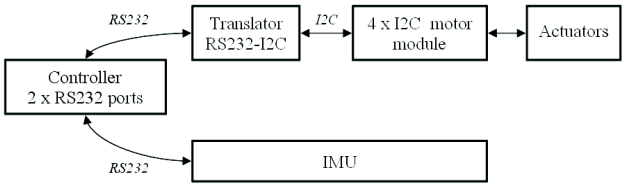

Fig. 4. OS4 test-bench block diagram

III. OS4 TEST-BENCH

The development of a control system for a flying robot requires the development of an adequate test-bench. This can help lock some number of degrees of freedom in order to reduce control complexity and to avoid system damage. For our control experiments, we use the test-bench in figure 3.

From a PC and through a standard RS232 port, one can send orders to the test-bench. The RS232 to I2C module translates the serial signals to the I2C bus motor modules. These modules integer a PID regulator on a PIC16F876 microcontroller. The MT9-B[6] IMU[7] estimates with a kalman filter the 3D orientation data and gives the calibrated data of acceleration and angular velocity. It weights about 33g and communicates at 115kbps. The OS4 test-bench has 4 propulsion groups, each one is composed of a 25g motor[8], a 6g gear box and a 6g propeller. To design the propulsion group, a test, evaluation and comparison method was developed.

IV. CLASSICAL CONTROL OF "OS4" VTOL SYSTEM

The dynamic model (9) presented above contains in addition to the actuators action, both the gyroscopic effects resulting from the rigid body, and the propellers rotation. The influence of these effects is in our case less important

[6]www.xsens.com
[7]Inertial Measurement Unit
[8]16G88 motor from: www.portescap.com

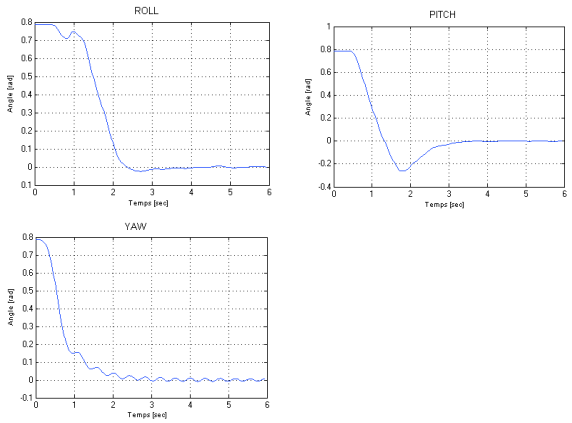

Fig. 5. Simulation: the system has to stabilize the orientation angles, starting from $\pi/4$ in roll, pitch and yaw as initial condition (P=0.8, D=0.4 for roll and pitch. P=0.8, D=0.5 for yaw angle).

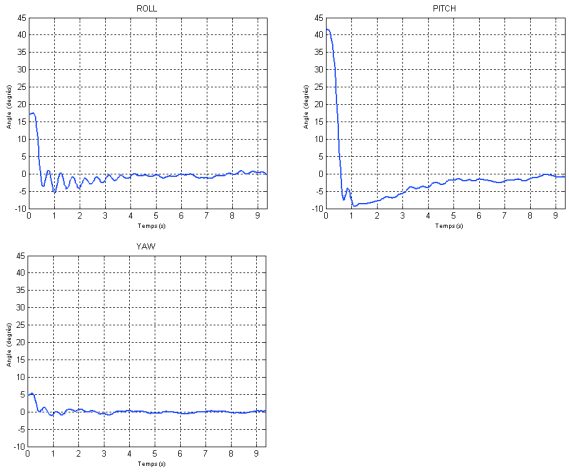

Fig. 6. Experiment: the system has to stabilize the orientation angles with a higher priority to roll and pitch angles, an integral term was added to eliminate the steady-state error (P=0.9, I=0.3, D=0.2 for roll and pitch. P=0.06, I=0.3, D=0.02 for yaw angle). This experiment includes a PID on each propeller to control the speed.

than the motor's action. Especially if we consider a near-hover situation. In order to make it possible to design multiple PID controllers for this system [5], one can neglect these gyroscopic effects and thus remove the cross coupling. The model (9) is then:

$$\ddot{\phi}, \ddot{\theta}, \ddot{\psi} = \frac{l}{I_{x,y,z}} U_{1,2,3} \tag{15}$$

If we include in (15) the rotor dynamics and rewrite the model in Laplace domain we obtain:

$$\phi(s) = \frac{B^2 bl}{s^2(s+A)^2 I_x}(u_2^2(s) - u_4^2(s))$$

$$\theta(s) = \frac{B^2 bl}{s^2(s+A)^2 I_y}(u_3^2(s) - u_1^2(s))$$

$$\psi(s) = \frac{B^2 d}{s^2(s+A)^2 I_z}(u_1^2(s) + u_3^2(s) - u_2^2(s) - u_4^2(s)) \tag{16}$$

Where A and B are the coefficients of the linearized rotor dynamics as described in (14). While C, too small comparing to B, is neglected.

A. PD Controller Synthesis and Simulation

Introducing a PD controller for each orientation angle:

$$U_{1,2,3} = k_{\phi,\theta,\psi}(\phi, \theta, \psi) + d_{\phi,\theta,\psi}(\phi, \theta, \psi) \tag{17}$$

In order to tune the controller parameters, and before implementing on the real system, we performed several simulations on Simulink using the complete model. The controller's task was to stabilize the orientation angles. For this simulations, the dynamic model (9) was used, obtaining the results showed if figure 5. The simulated performance was satisfactory regarding the simple control synthesis approach. We decided then to test on the real system.

B. PID Controller on The Real System

Finally, we implemented the controllers in C under Linux on a machine running at $450Mhz$ simulating the future integration of a Single Board Computer. The experiment has shown that the "OS4" was not completely stabilized, as a small steady-state error remains. An integral term was then added and the experiment was performed including a closed-loop speed control on each rotor. The results are shown in figure 6. The effect of the propellers speed control affects the general stabilization of the vehicle. In the closed-loop, the orientation stabilization is faster and the yaw angle is well controlled. Contrarily, in open-loop, the response is much more smooth. This highlights the importance of the actuators fast response. In both cases, the simulations and the experiments have shown that the Quadrotor can be controlled efficiently in hover using a classical approach. This is possible because the controller was tuned in simulation on the more complete model (9). Obviously, this controller will not be able to stabilize the robot in presence of strong perturbations.

V. OPTIMAL CONTROL OF "OS4" VTOL SYSTEM

Considering the general equations for state-space system, cost function and state feedback for a linearized system

$$\begin{cases} \dot{x} = Ax + Bu \\ J = \int (x^T Q x + u^T R u) dt \\ u(t) = -K_c x(t) \end{cases} \tag{18}$$

In this case, the necessary condition for optimality of the time derivative of the Hamiltonian function is:

$$K_c = R^{-1} B^T P \tag{19}$$

Where P obey to Riccati equation:

$$-PA - A^T P + PBR^{-1}B^T P - Q = \dot{P} \tag{20}$$

In order to solve Riccati equation, we first build the Hamiltonian matrix:

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^T \end{bmatrix} \tag{21}$$

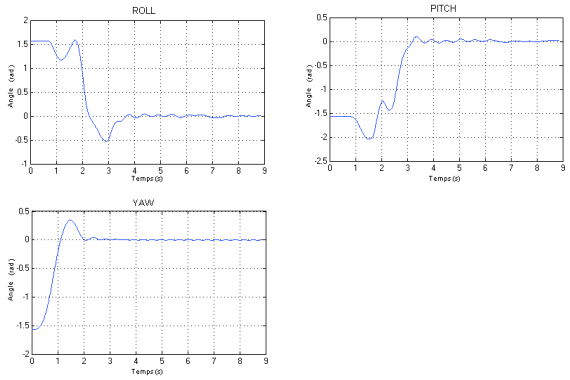

Fig. 7. Simulation: the system has to stabilize the orientation angles starting from $\pi/2$ with an LQ controller generated using Pearson method.

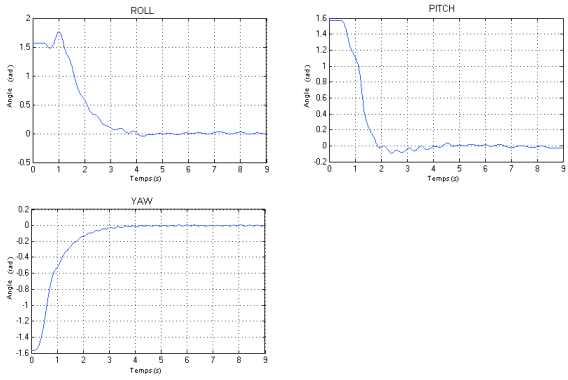

Fig. 8. Simulation: the system has to stabilize the orientation angles starting from $\pi/2$ with an LQ controller generated using Sage-Eisenberg method.

A. Adaptive Optimal Control

Applying the LQ control requires the system linearization to $\dot{X} = AX + BU$ form. In our specific system, a linearization around an equilibrium point will cause the model to be far form the reality (especially in large orientation angles) as all the couplings are neglected (gyroscopic effects). In order to allow the system optimization for a larger flight envelope, one can linearize around each state. Each coupled term is represented twice by fixing and varying each time one state. This leads to the following linear state-space system:

$$\dot{X}^T = \begin{pmatrix} \dot{\phi} & \ddot{\phi} & \dot{\theta} & \ddot{\theta} & \dot{\psi} & \ddot{\psi} \end{pmatrix}^T \tag{22}$$

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{I_y - I_z}{2I_x}\dot{\psi} & 0 & \frac{I_y - I_z}{2I_x}\dot{\theta} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{I_z - I_x}{2I_y}\dot{\psi} & 0 & 0 & 0 & \frac{I_{yz} - I_x}{2I_y}\dot{\phi} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & \frac{I_x - I_y}{2I_z}\dot{\theta} & 0 & \frac{I_x - I_y}{2I_z}\dot{\phi} & 0 & 0 \end{pmatrix} \tag{23}$$

$$B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{l}{I_x} & 0 & 0 & \frac{Jr}{I_x}\dot{\theta} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{l}{I_y} & 0 & 0 & \frac{Jr}{I_y}\dot{\theta} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I_z} & 0 \end{pmatrix} \tag{24}$$

The A and B matrix are now being adapted through the robot trajectory. The linearization is thus more valid.

B. First LQ Controller Synthesis and Simulation

If we consider Pearson method [6], we solve Riccati equation assuming that we zero the second term of (20), solve the equation and get the feedback gain matrix. A first simulation was performed on a model without the actuators dynamics, the results were very satisfactory, even if we start from a critical position as $\pi/2$ for the orientation angles. The same simulation including, this time, the actuator model was performed and showed the strong influence of the actuators dynamics as presented in figure 7.

C. Second LQ Controller Synthesis and Simulation

Considering a permanent solution to Riccati equation as simulated before gives medium results. Contrarily, Sage-Eisenberg method [6] proposes to consider a variable solution to Riccati equation and a fixed final condition $P(t_f) = 0$. Once discretized, Riccati equation can be rewritten as:

$$\begin{aligned} &-P_t(hA - I) - (hA^T)P_t \\ &+P_t(hBR^{-1}B^T)P_t - (hQ + P_{t+h}) = 0 \end{aligned} \tag{25}$$

Where : t_f: final time, $h = \frac{t_f}{n}$: iteration period, n: number of iterations.
The equation (25) represents correctly the system in the P_t to P_{t+h} interval. The control using this method was simulated at $100Hz$ under Simulink (see figure 8), with the full model including the actuators dynamics, the same Q and R matrix used in V-B and by taking $t_f = 0.3$ and $n = 10$. The gain matrix was then:

$$K = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 12.83 & 10.02 & 0 & 0 & 0 & 0 \\ 0 & 0 & 12.83 & 10.02 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12.86 & 10.01 \end{pmatrix} \tag{26}$$

Comparing with the previous simulation presented in V-B, Sage-Eisenberg method gives better results as it optimizes the cost function for every sub-trajectory in the P_t to P_{t+h} interval. According to Bellman principle [7], splitting an optimal trajectory generates several optimal sub-trajectories.

D. LQ controller on The Real System

In order to validate the previous simulations, we implemented the controllers on the same 450Mhz PC. It was problematic to find weight matrices which satisfy the control stability, in addition, a slight change in Q or R matrices introduces an important variation of the controller behavior. Hence, by choosing $t_f = 0.05$, $n = 10$ and an appropriate Q and R matrices, the system stabilizes as

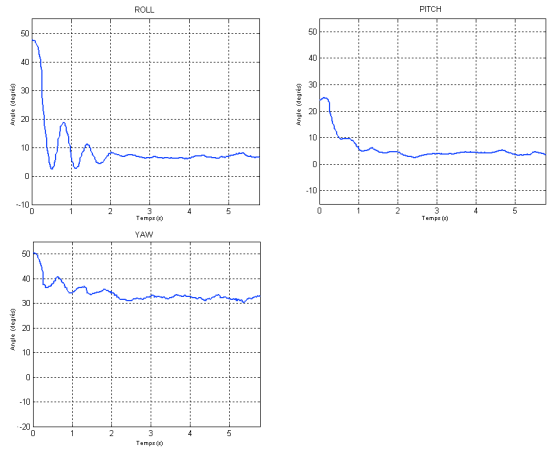

Fig. 9. Experiment: the system has to stabilize the orientation angles. The experiment was performed with an LQ controller using Sage-Eisenberg method.

shown in figure 9. The gain matrix K is then:

$$K = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1.059 & 0.391 & -0.001 & 0 & 0 & 0.001 \\ 0.0007 & 0 & 1.059 & 0.391 & 0 & -0.0004 \\ 0.005 & 0.002 & -0.0002 & -0.0001 & 0.015 & 0.028 \end{pmatrix} \quad (27)$$

As this can be seen from figure 9, a steady-state error remains on the three orientation angles, this is due to the slight differences of the propulsion groups and the disturbance introduced by the power and data cables. On the other hand, the fact that the LQ controller was developed without considering the actuators dynamics it is also responsible of the average performance. However, a new automatic test-bench for propellers is under construction, this will allow a better characterization of the propellers and the propulsion groups. However, one can try to introduce an integral term in an LQ controller as shown in [8]. This will be considered in a future development.

VI. EXPERIMENTAL AUTONOMOUS FLIGHT

After several simulations and experiments performed on the test-bench, it was time to test an autonomous flight. Once applied, the LQ controller brought-back average results for this experiment. In fact, a steady-state error remained because the actuator dynamics was not taken into account and the systematic slight differences in the propulsion groups. In addition, the LQ controller we obtained is experimentally less dynamic than the PID. Thus, we were not able to release "OS4" for a free flight. Contrarily, using the classical approach (PID), the autonomous flight was a success. The figure 10 shows the "OS4" orientation angles during an autonomous flight. Some perturbations were introduced by the power cables and by us while trying to prevent the robot from collisions with the walls. Obviously, there are still some episodic problems, especially with the sensors (drift, bad initialization,...) partly caused by the vibrations. We are rather happy with this result using the PID, but we are firmly convinced that the optimal control theory (LQ) should give better results.

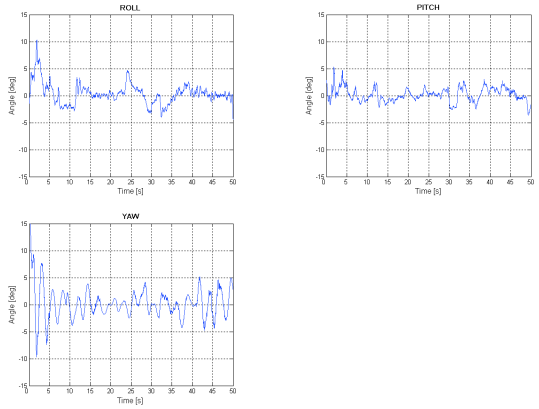

Fig. 10. Experiment: successful autonomous flight. The robot "OS4" has to stabilize the orientation angles by a PID controller.

VII. CONCLUSION

In this paper, we presented the application of two different control techniques (PID) and (LQ) to a micro Quadrotor called "OS4". As it can be seen from the experimental plots, the controller introduced using the modern approach provides average results, due to the model imperfections. It will be enhanced in a near future. On the other hand, the classical controller proves the ability to control the orientation angles in the presence of minor perturbation. The successful first autonomous flight validates the development. Our next goal is to enhance the control with position controller and to develop a fully autonomous vehicle. The positive results obtained in this development towards autonomous micro-VTOL, reinforce our conviction that, in spite of the natural high instability of these systems, a reliable control is still possible.

ACKNOWLEDGMENT

The authors would like to thank Dr. Philippe Müllhaupt, Dr. Mohamed Bouri and all the persons who helped in the successful first flight of "OS4" with there advices and fruitful discussions. Also, Mr. Andre Guinard for the realization of some mechanical components.

REFERENCES

[1] J.G. Leishman , *The Breguet-Richet Quad-Rotor Helicopter of 1907*. http://www.enae.umd.edu/AGRC/Aero/Breguet.pdf.

[2] M. Gäfvert, *Modelling of the ETH Helicopter*. Department of Automatic control, Lund Institute of Technology, CH, 2001.

[3] R. Mahony, et al. *Design of a Four-Rotor Aerial Robot*. Australasian Conference on Robotics and Automation, Auckland, Australia, 2002.

[4] P. Mllhaupt, *Analysis and Control of Underactuated Mechanical Nonminimum-phase Systems*. Phd thesis, Department of Mechanical Engineering, EPFL, 1999.

[5] R. Longchamp *Commande Numérique de Systèmes Dynamiques*. Presses Polytechniques et Universitaires Romandes, Lausanne, 1995.

[6] R. Longchamp *Commande Sous-optimale Adaptative de Systmes Nonlineaires*. PhD thesis, Swiss Federal Institute of Technology, Switzerland, 1978.

[7] P. Naslin *Introduction à la Commande Optimale*. Bibliotheque de l'Automaticien, Dunod, Paris, 1966.

[8] F. Franklin, et al. *Digital Control of Dynamic Systems (Second Edition)*. Addison Wesley, 1990.

[9] P. Castillo, R. Lozano, et al. *Real-time Stabilization and Tracking of a Four Rotor Mini-Rotorcraft*. accepted for IEEE Transactions on Control Systems Technology, 2003.