



IMAV 2011

International Micro Air Vehicle Conference and Flight Competition

summer edition

➤ Proceedings

 **TU**Delft



THALES

Main sponsors:



Koninklijke Landmacht
Ministerie van Defensie

THALES



Korps landelijke politiediensten

Parrot
AR.Drone
YOUR REALITY BECOMES A VIDEO GAME

Supported by:



ONR



USAITCA



EORD

Sub sponsors:



Nationaal Lucht- en Ruimtevaart
Laboratorium



TU Delft



Saxion Hogeschool Enschede

Conference subsidised by:



KNAW

Koninklijke Nederlandse
Akademie van Wetenschappen

Preface

It is our pleasure to present the proceedings of the International Micro Air Vehicle conference and competitions 2011 (IMAV 2011) Summer Edition that was held in 't Harde, the Netherlands from September 12th through 15th. This event has been organized by Thales and Delft University of Technology, and aims at presenting an overview of state-of-the-art research in the emerging field of Micro Air Vehicles (MAVs). The received submissions show that the field of MAV research is very active all over the world. The topics addressed by the articles in the proceedings range from the autonomous mapping of unknown environments to the investigation of vortex rings on both a butterfly and a flapping-wing MAV. These proceedings contain the peer-reviewed scientific articles presented at the IMAV 2011 Summer Edition.

The research on MAVs is making swift progress, as shown by both the research and practical demonstrations at the IMAV 2011. The competitions, which had a record number of 23 participating teams, illustrated the technological advances being made and the challenges that are ahead. For example, it is noteworthy that the IMAV 2011 Summer Edition was the first IMAV-event to have six teams participating in the "pylon challenge" with an autonomously flying MAV. The pylon challenge requires the MAVs to fly in 8-shapes around two brightly colored orange pylons. Despite the predefined color of the pylons, this challenge is not as easy as it may seem at first sight: with the exception of one system that used a laser scanner, all other autonomous systems used a single, relatively small field-of-view camera. As a consequence, most of the systems based their flight on monocular visual cues. In particular this implied that the systems had to deal with the presence of distractors, and well-known problems of varying light conditions. The difficulty of the challenge is reflected in the final ranking. Despite a considerable bonus for autonomous flight, the indoor pylon challenge is led by a very well-designed tiny MAV controlled via video goggles, followed by the autonomous MAV with a laser-scanner, and on the third place a team that used monocular vision.

The outdoor mission challenge, in which the participating teams had to monitor a simulated "drugs deal", was also remarkable for two reasons. First, it was the first time at an IMAV that teams used multiple MAVs at the same time. While multi-MAV research has received considerable attention in the past years, the practical demonstration of such systems in a competition was novel. This year, teams with multiple MAVs typically had one ground station per MAV. Exceptions include the team behind the open source Paparazzi system, in which multiple systems were controlled from one ground station. The use of multiple MAVs is a promising step towards groups of MAVs that exploit self-organization to accomplish tasks given by the operator. Such a self-organizing multi-MAV system is typically referred to as a 'swarm'. A second remarkable observation is that there were at least two MAVs this year that can both hover and transition to forward fixed-wing flight. Such systems form a promise for missions in which the MAV will have to perform a detailed observation mission at a long distance from the take-off location. The design of these systems requires a tight integration of knowledge and expertise in the areas of structures, aerodynamics and control.

We would like to thank the program committee for their efforts in reviewing all the IMAV 2011 articles:

H. Bijl	Q.P. Chu	C. De Wagter
F. Groen	J. van den Herik	E. Johnson
E.-J. van Kampen	K. Kondak	L. van der Maaten
D. Moormann	J.-M. Moschetta	M. Ol
B. van Oudheusden	E.O. Postma	R. Ruijsink
W. Shyy	S. Shkarayev	A. Visser
P. Vörsmann	S. Watkins	V. Trianni

Finally, we want to express our hope that the open access to all the articles in these proceedings will contribute to the advances in this young and exciting research area.

On behalf of the IMAV 2011 organization committee,

Guido de Croon and Matthijs Amelink,

program chairs of the IMAV 2011.

Table of Contents

Wing flexibility effects in clap-and-fling <i>M. Percin, Y. Hu, B.W.van Oudheusden, B. Remes and F.Scarano</i>	5
Dynamic behaviors of a vortex ring on a butterfly and a small flapping robot <i>M. Fuchiwaki, T. Kuroki, K. Tanaka, and T. Tabata.</i>	13
Analysis of tail effects in flapping flight <i>W.B Tay, H. Bijl and B.W. van Oudheusden</i>	18
Application of electro-active materials to a coaxial-rotor NAV <i>C. Thipyopas, A.B. Sun, E. Bernard and J.-M. Moschetta</i>	24
Numerical investigation of disc-wing MAV with propeller in a wing slot <i>N. D. Ageev</i>	30
An engineering development of a novel hexrotor vehicle for 3D applications <i>D. Langkamp, G. Roberts, A. Scillitoe, I. Lunnon, A. Llopis-Pascual, J. Zamecnik, S. Proctor, M. Rodriguez-Frias, M. Turner, A. Lanzon and W. Crowther</i>	35
Closing the gap between simulation and reality in the sensor and motion models of an autonomous AR. Drone <i>Arnoud Visser, Nick Dijkshoorn, Martijn van der Veen and Robrecht Jurriaans</i>	43
PLASE: A novel planar surface extraction method for the autonomous navigation of micro-air vehicle <i>Rafid Siddiqui, Mohammad Havaei, Siamak Khatibi and Craig Lindley</i>	51
BioMAV: bio-inspired intelligence for autonomous fight <i>Paul K. Gerke, Jurriaan Langevoort, Sjoerd Lagarde, Laurie Bax, Tijl Grootswagers, Robert-Jan Drenth, Vincent Slieker, Louis Vuurpijl, Pim Haselager, Ida Sprinkhuizen-Kuyper, and Guido de Croon</i>	59
Practical aspects of trirotor MAV development <i>Andrzej Ryś, Roman Czyba and Grzegorz Szafranski</i>	67
Different approaches of PID control UAV type quadrotor <i>G. Szafranski and R. Czyba</i>	73
Carr Equilibrium transition study for a hybrid MAV <i>M. Itasse, and J.M. Moschetta, Y. Ameho, and R.</i>	79
Vortex-lift modeling provides reliable force predictions for flapping-wing micro air vehicles <i>W. Thielicke, A.B. Kesel and E.J. Stamhuis</i>	87
Aerodynamic analysis of the wing flexibility and the clap-and-peel motion of the hovering DelFly II <i>T. Gillebaart, A.H. van Zuijlen, and H. Bijl</i>	95
Aerodynamics study of fixed-wing MAV: Wind tunnel and flight test <i>C. Thipyopas and N. Intaratep</i>	103
One useful propeller mathematical model for MAV <i>S.V. Serokhvostov and T. E. Churkina</i>	111
Transient analysis of NYLON 6/6 for a thin shell structure by FEM <i>H.R. Montazer Hojjat</i>	118
Inverse dynamics approach to adaptive damage-tolerant control for unmanned aerial vehicles <i>Alexey Kondratiev and Yury Tiumentsev</i>	122
Design approach for selection of wing airfoil with regard to micro-UAVs <i>Vladimir Brusov and Vladimir Petruchik</i>	129
Flight data acquisition system for small unmanned aerial vehicles <i>Vladimir Brusov, Józef Grzybowski, and Vladimir Petruchik</i>	135
An architecture with integrated image processing for autonomous micro aerial vehicles <i>Christian Dernehl, Dominik Franke, Hilal Diab, Stefan Kowalewski</i>	141

Wing flexibility effects in clap-and-fling

M. Percin¹, Y. Hu^{1,2}, B.W.van Oudheusden¹, B. Remes¹ and F.Scarano¹

1. Delft University of Technology, Delft, The Netherlands

2. Beihang University, Beijing, PR China

ABSTRACT

The work explores the use of time-resolved tomographic PIV measurements to study a flapping-wing model, the related vortex generation mechanisms and the effect of wing flexibility on the clap-and-fling movement in particular. An experimental setup is designed and realized in a water tank by use of a single wing model and a mirror plate to simulate the wing interaction that is involved in clap-and-fling motion. The wing model used in the experiments has the same planform with the Delfly II wings and consists of a rigid leading edge and an isotropic polyester film. The thickness of the polyester film was changed in order to investigate the influence of flexibility. A similarity analysis based on the two-dimensional dynamic beam equation was performed to compare aeroelastic characteristics of flapping-wing motion in-air and in-water conditions. Based on the experimental results, the evolution of vortical structures during the clap-and-peel motion is explained. The general effects of flexibility on vortex formations and interactions are discussed. It was observed that the flexibility affects the behavior and orientation of the vortices in relation to the deformation of the wing and interaction with the mirror plate.

1 INTRODUCTION

Flapping-wing aerodynamics has been of interest to the researchers recently due to increasing design efforts in the field of Micro Aerial Vehicles (MAVs). MAVs are small unmanned air vehicles with overall dimensions not larger than 15cm [1]. Recent developments in several technological fields have enabled the possibility of using MAVs as mobile and stealth sensing platforms capable of gathering intelligence in hazardous and physically inaccessible areas. To accomplish such missions, MAVs should be capable of maneuvering with ease, staying aloft and propelling themselves efficiently. However, conventional means of thrust and lift generation become inefficient in terms of required capabilities at these scales and hence flapping-wing propulsion becomes a necessity. In contrast to the conventional mechanisms of aerodynamic force production, flapping-wing mechanisms are associated with vortices separating from the leading and trailing edge, which create low pressure region that can be used to create higher lift and thrust [1].

The phenomenon of force production as a result of flapping motion has been studied extensively in the literature originating from the pioneering studies of Knoller [2] and Betz [3], who pointed out that flapping wing motion generates an effective angle-of-attack that results in lift production with a thrust component, which is known as Knoller-Betz effect. Thenceforward, further investigations have clarified the underlying aerodynamic mechanisms, different flow topologies, and effective parameters for simplified two-dimensional flapping rigid wing motions, i.e. pitching, plunging or combined pitching-plunging. Discussion of these topics is outside the scope of the present

paper and the reader is referred to [4]-[10] for more detailed information.

Natural flapping is a three-dimensional phenomenon which combines pitching, plunging, and sweeping motions [11]. Moreover, birds and insects benefit from several different unsteady aerodynamic mechanisms, among them the clap-and-fling motion, which is the particular topic of the present study.

Clap-and-fling is a lift enhancement mechanism which was first described by Weis-Fogh [12]. This relates to the wing-wing interaction phenomenon, which takes place at dorsal stroke reversal (Figure 1). During the clap phase, the leading edges of the wings come together and pronation about the leading edges occurs until the v-shaped gap between the wings disappears (see Figure 1 A-C). Subsequently, in the fling phase, the wings rotate about their trailing edges forming a gap in between. Following, the translation of the wings occurs (see Figure 1 D-F). Investigations on birds and insects showed that as well as being used continuously during the flight, some species utilize this mechanism for a limited time in order to generate extra lift, especially while carrying loads or during the take-off phase [13]. The insect experiments of Marden [14] showed that use of clap-and-fling mechanism results in generation of 25% more aerodynamic lift per unit flight muscle than conventional flapping-wing motions.

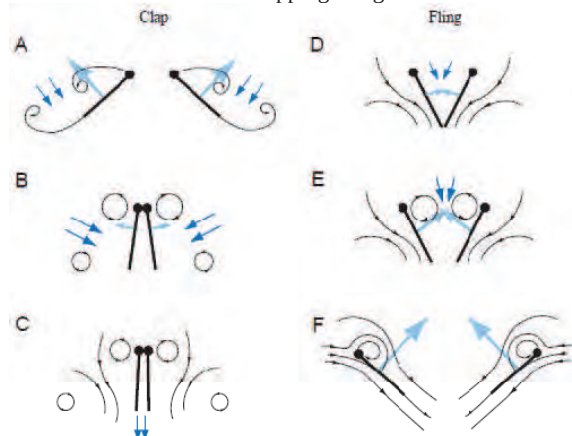


Figure 1: Schematic representation of clap-and-fling mechanism. Black lines represent flow lines, dark blue arrows show induced velocity and light blue arrows represent net force exerting on the airfoil, adapted from Sane [11].

Several studies have attempted to provide an explanation of the enhanced force generation mechanism of the clap-and-fling motion. Weis-Fogh [12] indicated that during the clap phase, as the gap between the wings vanishes progressively, the opposing circulation of both wings cancel each other out. This attenuates the starting vortex at the onset of fling and diminishes the Wagner effect. By doing so, circulation will build up more rapidly and the benefit of lift over time will be extended in the fling phase [11]. Moreover, a

downward momentum jet formed at the end of clapping motion can work in favor of lift generation [15]. On the other hand, during the fling phase, as the leading edges move apart, the fluid rushes into the low pressure region between two wings, which results in generation of massive leading edge vortices. This mechanism enhances circulation at the onset of fling phase and hence increases lift. This phenomenon was experimentally verified by Lehmann et al. [16]. They performed instantaneous PIV and force measurements on dynamically scaled rigid fruit fly wings in order to investigate the effects of the clap-and-fling motion on the force production. They pointed out that clap-and-fling motion, depending on the stroke kinematics, may enhance the force production up to 17%. Detailed PIV analysis revealed that the existence of a bilateral image wing increases the circulation induced by the leading edge vortex during the early fling phase, obviously correlated with a prominent peak in both lift and drag. Furthermore, it was shown that trailing edge vorticity shed during the clap phase of the motion is considerably reduced with respect to the single flapping wing case.

It is obvious that the majority of these studies focus on the flapping motion of rigid wings and the effect of flexibility has received relatively little attention. However, studies on the mechanical properties of insect wings report complicated variations in their stiffness and identify them absolutely flexible [17], [18]. Although aerodynamic benefits of flexibility for the insect are not completely clear [19], there is a growing evidence that wing deformation during the flapping motion boosts thrust and lift production considerably [20]. Vanella et al. [21] carried out a computational study on a hovering two-dimensional flexible wing model for Reynolds number (Re) ranging from 75 to 1000. They concluded that flexibility can enhance the aerodynamic performance and the best performance was achieved when the wing was flapped at $1/3$ of the natural frequency. Heathcote and Gursul [19] performed water tunnel experiments to investigate the effect of chord-wise flexibility on the propulsive efficiency of a heaving airfoil for Re of 9000 to 27000. They concluded that a certain degree of flexibility enhances the thrust coefficient and propulsive efficiency. Heathcote et al. [22] also studied the influence of spanwise flexibility and they found out that introducing a degree of spanwise flexibility affects the vortex mechanism and increases the thrust efficiency. They added that the range of Strouhal number (Sr) in which spanwise flexibility was beneficial overlaps with the range observed in nature ($0.2 < Sr < 0.4$). Based on above discussion, it can be inferred that MAVs might benefit from aerodynamic contributions of flexibility, in addition to the intrinsic low weight of flexible structures.

Regarding the clap-and-fling motion, it was shown that with the effect of flexibility the fling phase occurs more like a peel, while the clap phase can be considered as reverse-peel [23]. That is the reason why clap-and-fling motion is called clap-and-peel motion for flexible wing case. It has been speculated that flexible wings increase lift by enhancing the circulation in the fling phase and boosting the strength of downward momentum jet in the clap phase [15]. Moreover, it was indicated that flexibility reduces drag by allowing the wing to bend or reconfigure under the aerodynamic loading [24]. Miller and Peskin [24] investigated this phenomenon computationally by use of an immersed boundary method for Re of 10. They found that

clap-and-fling with flexible wings produces lower drag and higher lift with respect to clap-and-fling with rigid wings.

As indicated earlier, the clap-and-fling mechanism is the particular research interest for the present paper, because of its relevance to the DelFly II, which is a bi-plane flapping-wing MAV that was designed and built at Delft University of Technology. It has four wings, of anisotropic flexible construction, undergoing clap-and-fling motion. Therefore, it is important from a design optimization point of view to obtain a better understanding of the effect of flexibility on the clap-and-fling motion.

Experimental research has been performed on the DelFly II to reveal the effect of flexibility on the aerodynamic performance. De Clercq [25] captured the instantaneous flow field in the vicinity of the flapping wings of DelFly II in hover condition via stereo-PIV and performed simultaneous thrust (viz. lift) force measurements. She showed that flexibility causes the wings to peel apart at the onset of the fling and to flex at the maximum outstroke. She also found that the peeling phase makes the major contribution to the lift thanks to the generation of massive leading edge vortices. A subsequent wing optimization study by Bruggeman [26] for the hovering condition resulted in a wing layout that provides a 10% increase in thrust-to-power ratio. Consequently, Groen [27] performed phase-locked stereo-PIV measurements to compare the deformations and flow fields for the original and the improved wing. The acquired wing shapes revealed that the improved wing in general is less flexible with respect to the original wing. This condition manifests itself also in the flow fields as shown in Figure 2. He pointed out that the LEV stays closer to the wing surface for the improved wing which was attributed to the increased suction during the peeling phase of the motion.

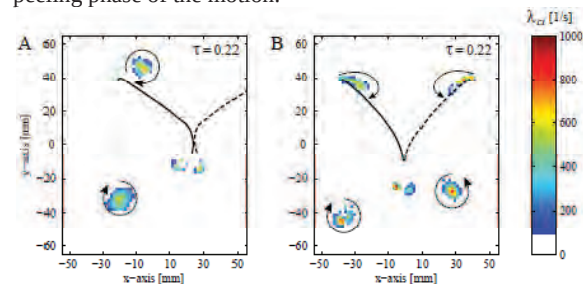


Figure 2: Comparison of swirling strengths for the original (A) and improved (B) DelFly wings flapping at 11 Hz at the spanwise position of $0.86R$, adapted from Groen [27]

Notwithstanding the extensive measurements performed on the DelFly wings, it proved difficult to link the effect of flexibility to the flow field characteristics and performance improvements, due to the anisotropic characteristics of the DelFly wings and the associated complicated aeroelastic behavior. Based on these considerations, in this project it is aimed to investigate the influence of wing flexibility in a more generic configuration. The experiments were performed in a water tank to permit high-quality and time-resolved three-dimensional flow field measurements via time-resolved Tomographic Particle Image Velocimetry (TOMO-PIV) [28]. The wing model used in the experiments consists of a rigid leading edge and an isotropic flexible wing surface made from polyester film; therefore it is assumed that the wing is flexible in only the chordwise

direction. The flexibility was varied in the experiments by changing the thickness of the polyester film.

2 SIMILARITY ANALYSIS

The determination of scaling laws is an important step in order to identify characteristic properties of the system under consideration and to establish which combination of parameters is of importance for the given conditions. In the current study, analysis of structural dynamics as well as fluid dynamics becomes necessary, because of the different deformation characteristics of a flapping flexible wing for in-air and in-water conditions. The Euler-Bernoulli dynamic beam equation was utilized for the analysis of similarity parameters. The general form of the equation after neglecting the rotary inertia and deformations due to shear is shown below.

$$(1) \quad m \frac{\partial^2 w}{\partial t^2} + EI \frac{\partial^4 w}{\partial x^4} = f(x, t)$$

It should be noted that deformation only in the chordwise direction x is considered, in accordance with the model wing used in the experiments. In Equation-1, m is the mass per unit chord length, E is the Young's modulus, and I is the moment of inertia. The term $f(x, t)$ on the right hand side is the forcing function term, consisting of aerodynamic and inertial contributions, as given in Equation-2, for a sinusoidally heaving leading edge.

$$(2) \quad f(x, t) = -C_{aero} \left[-\omega \bar{h} \sin(\omega t) \right] \left(-\omega \bar{h} \sin(\omega t) \right) - m \left(-\omega^2 \bar{h} \cos(\omega t) \right)$$

Here, C_{aero} is the aerodynamic multiplication factor ($C_{aero} = 0.5 \rho_f \bar{c}$, where $\bar{c}$ is the mean chord length, and ρ_f the fluid density), ω is the circular frequency ($\omega = 2\pi f$, with f the flapping frequency), and $\bar{h}$ is the amplitude of the heaving motion. Dimensional analysis of Equations-1 and 2 yields two similarity parameters, which are the ratio of elastic to aerodynamic forces (bending stiffness parameter, Equation-3), and the ratio of inertial to aerodynamic forces (Equation-4).

$$(3) \quad \lambda = \frac{EI}{\rho_f f^2 \phi^2 b^3 c^3}$$

$$(4) \quad \tau = \frac{\rho_w t}{\rho_f b \phi}$$

In this, ϕ is the angular flapping amplitude, b the span length, c the chord length (taken as the mean chord in the present calculations), ρ_w the density of the wing material, and t the thickness of the wing.

In addition to the derivation of similarity parameters, analytical solutions of the dynamic beam equation were used for comparison of the deformation characteristics for in-air and in-water conditions. The solution was performed for a sinusoidally plunging airfoil which is assumed to represent the cross-section of the flapping wing. Firstly, a wing of isotropic flexibility was defined as an equivalent-DelFly-wing. Secondly, a model wing for in-water

conditions was defined, operating at the same Reynolds number and with similar flexibility as the (equivalent) DelFly wing.

Assumptions involved in the definition of the equivalent DelFly wing are: (1) it has the same geometry ($b = 0.14$ m, $\bar{c} = 0.08$ m, and $AR = 1.75$) and mass (88 mg for a single wing excluding the rigid leading edge) as the DelFly wing; (2) it has isotropic structural characteristics; (3) it flaps in-air at 13 Hz and (4) the maximum deformation (trailing edge displacement with respect to leading edge) it experiences is equal to that of the DelFly wing. The maximum deformation for the reference case was observed to be approximately 60% of the chord length at 71% of the span length, based on the wing shape visualization performed by Groen [27].

The analytical solution was then used to determine the required flexural stiffness of the equivalent-DelFly-wing. The maximum deformation of 60% of chord length is achieved at a flexural stiffness value of 1.06×10^{-4} Nm². The order of flexural stiffness agrees with the study of Combes and Daniel [18] who measured the flexural stiffness (both in the spanwise and chordwise directions) of 16 different insect wings. Based on their correlation, the chordwise flexural stiffness for the chord-length of 8 cm is estimated as approximately 10^{-4} Nm².

Deformation characteristics for in-water conditions were acquired for a scaled down model-wing ($b = 0.1$ m, $\bar{c} = 0.057$ m, and $AR = 1.75$). Polyester sheet was used as a wing material and, apart from reproducing the Reynolds number, it was aimed to satisfy two constraints: first to have approximately the same amount of maximum deformation during the flapping motion for in-air and in-water conditions; second to keep the bending stiffness parameters of the equivalent-DelFly-wing and the model-wing as close as possible. These conditions were satisfied for the case of model wing with a thickness of 100 μ m flapping at 0.75 Hz in water. The resultant bending stiffness parameters were calculated as 0.60 for equivalent-DelFly-wing and 0.55 for the model-wing. Deformations for a flapping period are compared in Figure 3. It should be noted that although it might be possible to satisfy the above-mentioned conditions with a different flapping frequency and wing thickness, current values were selected based on the limitations regarding to the experimental setup and material availability.

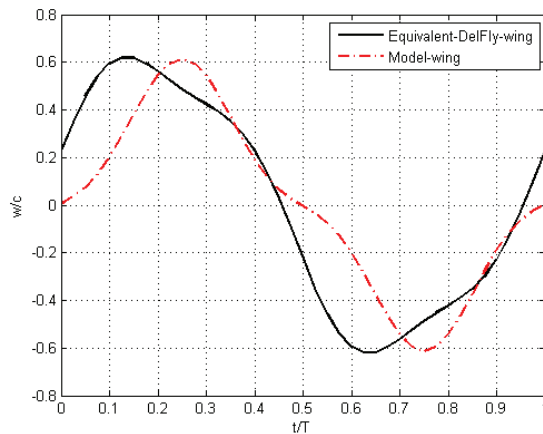


Figure 3: Comparison of total deformations (trailing edge displacement with respect to leading edge) for the equivalent-DelFly-wing flapping in air and model-wing flapping in-water

It is clear that the deformation characteristics are different from air to water, as already expected for. The major difference can be approximated by a phase shift of 10% the flapping period. In order to clarify the reason of this difference, the deformation contributions of the aerodynamic and inertial effects are shown separately in Figure 4. It is obvious that inertial forces are substantially diminished conducting experiments in water rather than air, as also evident from Equation-4. It can be concluded that the effect associated to the wing-inertia term is largely reduced when the wing flaps in water, which has similar density. As a result, it is difficult to achieve the same deformation characteristics for air and water experiments. Therefore, it is aimed to achieve order of scales of deformation and flexural bending stiffness as similar as possible to the equivalent-DelFly-wing in this study.

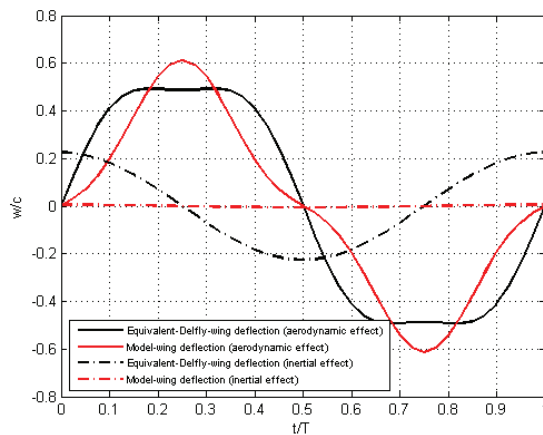


Figure 4: Comparison of deformations (trailing edge displacement with respect to leading edge) due to aerodynamic and inertial forces separately for the equivalent-DelFly-wing flapping in air and model-wing flapping in-water

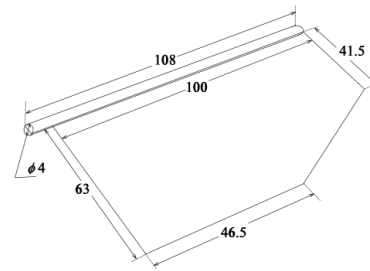


Figure 5: Dimensions of the wing model.

3 EXPERIMENTAL SETUP

Wing model and setup

The experiments were conducted in a water tank at the Aerodynamic Laboratory of Delft University of Technology (TUDelft). A single actuated wing was used in combination with a mirror plate to simulate the clap-and-fling motion of DelFly II in hovering flight. The wing model consists of a rigid leading edge and a flexible wing surface. The leading edge is made of two D-shape carbon rods of 4 mm diameter and the wing is manufactured from transparent polyester film (see Figure 5). Four different polyester wings with thickness of 50, 100, 175 and 250 μm were used to investigate the effect of flexibility.

As shown in Figure 6, a stainless steel rod with a diameter of 15 mm, connecting with the wing model, stands vertically in the octagonal water tank (600 mm of diameter and 600 mm of height). The existing water tank is made of Plexiglass allowing full optical access for illumination and tomographic imaging to study turbulent jets [29]. The distance between the rod axis and the surface of the mirror plate is 10 mm. When the motor actuates the crank-arm mechanism, the wing will flap with an amplitude of 50° and a frequency of 0.5-1.0 Hz. Based on the mean wing tip velocity and the mean chord length, the Reynolds number for hovering flight [30] is 6700.

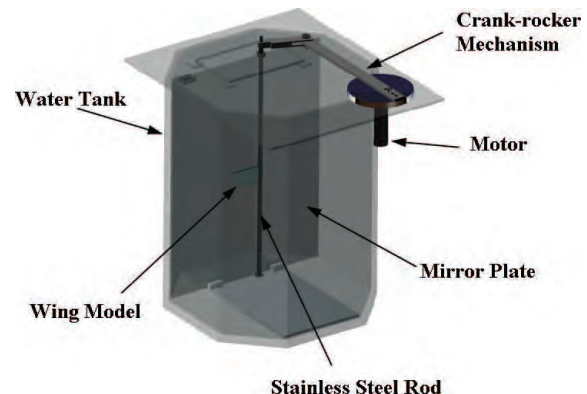


Figure 6: schematic representation of experimental set-up.

Time-resolved tomographic PIV

High-speed tomographic-PIV measurements were carried out to acquire time-resolved three-dimensional quantitative information of the flow around the wing model. Polyamide spherical particles of 56 μm diameter were employed as tracer particles at a concentration of 0.04 particles/ mm^3 . An

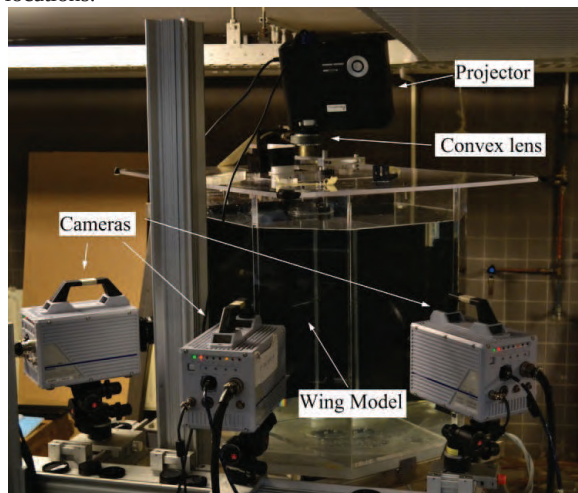
alternative method to laser was applied for volume illumination, using the light beam from a normal projector, Acer PD125D DLP, which is converged by a convex lens. Position, color and size of the beam could be modified easily by Microsoft PowerPoint software on the computer. The light with lumen of 2000 ANSI from the projector is adequate to collect sufficient light scattered by the particles. The measurement volume of $100 \times 100 \times 40 \text{ mm}^3$ in size, oriented as indicated in Figure 7(b) where the distance $D = 70 \text{ mm}$, is captured by three high-speed cameras (1024×1024 pixels) arranged along different azimuthal directions in a horizontal plane as shown in Figure 7.

The digital resolution is 10 pixels/mm and the average particle image density is approximately 0.015 particles per pixel (ppp). The tomographic recording system is composed of three CMOS cameras with a resolution of 1024×1024 pixels at 5.4 kHz. The pixel pitch of the cameras is 20 μm . Each camera is equipped with a Nikon 60 mm focal objective with numerical aperture $f\# = 11$. Scheimpflug adapters are used on two cameras to align the mid-plane of the measurement volume with the focal plane. Image sequences of tracer particles are recorded with recording frequency of 250 Hz (exposures time is 1/300 s).

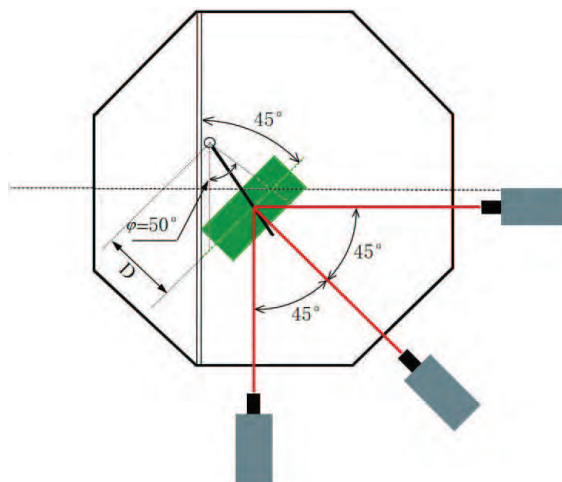
Tomographic reconstruction

Besides synchronization of the cameras and image acquisition, DaVis 7.4 (LaVision) was also used in the image pre-processing, volume calibration, self-calibration [31], reconstruction, and three-dimensional cross-correlation based interrogation that yields the velocity vector fields.

In this experiment, the measurement volume is calibrated by scanning a plate with 9×10 dots through the volume in depth of 20 mm with steps of 5 mm. In each of the calibration planes, the relation between the physical coordinates (X, Y, Z) and image coordinates is described by a 3rd order polynomial fit. Linear interpolation is used to find the corresponding image coordinates at intermediate z -locations.



(a) Experimental arrangement in the water tank.



(b) Schematic representation of top view.

Figure 7: Optical arrangement for the TR-PIV experiments.

The reconstruction process is improved by means of image pre-processing with background intensity removal, particle intensity equalization and a Gaussian smooth (3×3 kernel size). Reconstructed volumes are discretized at 10^3 voxels per mm^3 . The particle images are interrogated using windows of final size $64 \times 64 \times 64$ voxels with an overlap factor of 75 %, resulting in a vector spacing of about 1.6 mm in each direction. A dataset of order $62 \times 62 \times 25$ velocity vector volumes is acquired in each region.

4 RESULTS

Four polyester wings with different thickness (ranging from 50 to 250 μm) were used to investigate the effect of flexibility. In this section, the evolution of vortical structures at the clap-and-peel phase of the flapping motion is discussed. For brevity, only the cases of 100 and 250 μm wings flapping at 0.75 Hz are represented here.

In Figure-8, iso-surfaces of vorticity magnitude during the clap-and-peel phase are shown for the wing with a thickness of 100 μm . The end of clap is labeled as $t=0$ and the rear surface of the illumination volume is represented with a green slice. The wing position with respect to symmetry plane is schematized on the left corner of each phase. The vortex formation mechanism can be explained as follows: After the onset of instroke, the wing starts to move towards the wall with the formation of leading end trailing edge vortices ($t=-0.12T$). The separating shear layer from the trailing edge forms into a discrete trailing edge vortex. Trailing and leading edge vortices grow in size as the rotation goes on ($t=-0.06T$). At the end of in-stroke, the trailing edge vortex sheds into the wake of the wing, whereas the leading edge vortex stretches downward behind the wing ($t=0$). During the supination ($t=0.06T$), the stretched leading edge vortex moves towards the tapered part of the wing and interacts with the newly formed trailing edge vortex ($t=0.12T$). The outstroke continues with the formation of a massive leading edge vortex ($t=0.16T$) thanks to the inrush of fluid into the gap between the wing and the wall during the peeling phase.

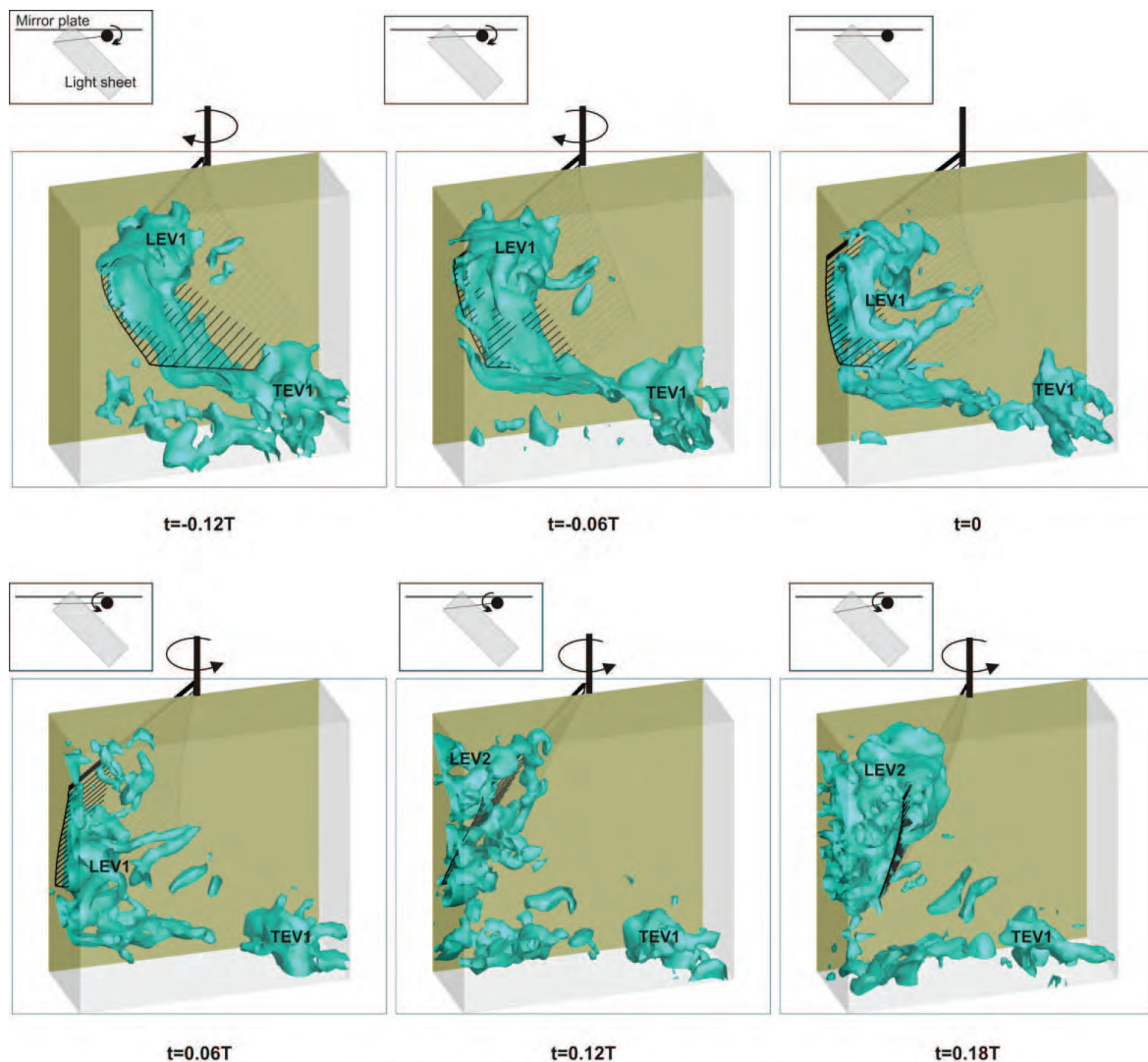


Figure 8: Isosurfaces of vorticity magnitude during the clap-and-peel phase ($t=100 \mu\text{m}$, $f=0.75 \text{ Hz}$)

The wing flexibility manifests its effects in general vortex formation-interaction mechanisms as well as in the deformation of the wing during the flapping motion. In Figure-9, contours of z -vorticity are plotted in the cross section plane at 70% of the span at the early clap phase ($t=-0.12T$), for the wings with thickness of 100 and 250 μm .

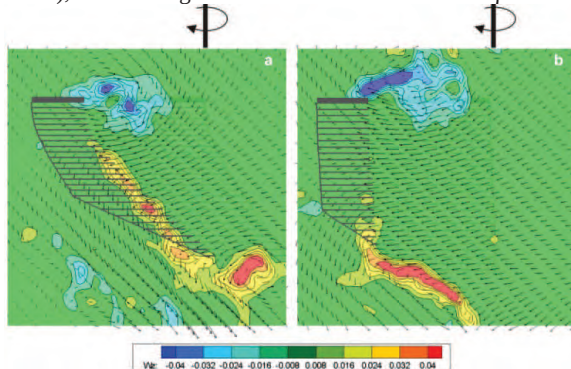


Figure 9: Contours of z vorticity for the wings at 70% of the span lengths of the wings with thicknesses of (a) 100 μm and (b) 250 μm at $t=-0.12T$

It is clear that the flexible wing curves towards its own wake whereas the other wing is almost rigid. In the case of the rigid wing, a stronger leading edge vortex is present but rather detached from the wing surface. It is possible to indicate that this condition will result in lower lift generation than the flexible wing case where the leading edge vortex stays near the curved surface of the wing and creates suction that works in favour of lift generation. On the other hand, the separating shear layer of the rigid wing trailing edge is curved downwards most probably due to increased downwash generated. In contrast, for the flexible wing, the trailing edge vortex stays inside the wake.

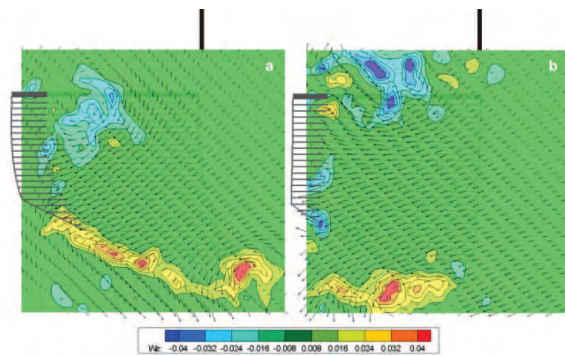


Figure 10: Contours of z vorticity for the wings at 70% of the span lengths of the wings with thicknesses of (a) 100 μm and (b) 250 μm at $t=0$

The vortical formations for the rigid and flexible wings are compared at the end of the clap ($t=0$) in Figure-10.

It is evident that due to the rigidity of the wing, the interaction with the wall results in stronger downwash and upwash (due to finite distance between the rigid leading edge and the wall) which in turn cause the vortices formed during the previous instroke to convect downward and upward, respectively. As noted earlier, contrary to the rigid wing case, the leading edge vortex in the case of flexible wing stays in the wake.

The comparison of z-vorticity contours at the late peel phase (Figure-11) reveals the fact that the rigid wing supinates into relatively stationary fluid region with respect to flexible wing case. It can be speculated that this condition will result in increase of the power requirement to fling the wing apart.

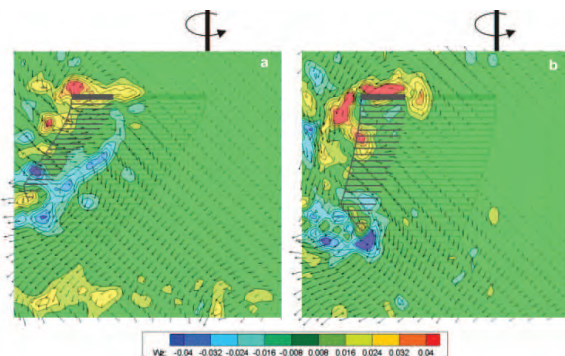


Figure 11: Contours of z vorticity for the wings at 70% of the span lengths of the wings with thicknesses of (a) 100 μm and (b) 250 μm at $t=0.18T$

5 CONCLUSION

The applicability of time-resolved tomographic PIV for the experimental investigation of vortical structures and effects of flexibility for the clap-and-fling type flapping motion was studied in this investigation. Experiments were performed in a water tank with a model wing that consists of a rigid leading edge and an isotropic flexible polyester film. The thickness of the polyester film was changed to investigate the influence of flexibility. A similarity analysis was performed by use of the two-dimensional dynamic beam equation. This revealed the different deformation characteristics for in-air and in-water conditions. Experimental results obtained with the tomo-PIV technique allow to characterize the three-dimensional structure of the flow field around the flapping wing model. The general

vortex formation and interaction mechanisms are explained during the clap-and-peel phase of the flapping motion. It was shown that in the case of more rigid wing case, leading and trailing edge vortices stay detached from the surface of the wing and they convect upwards and downwards due to relatively strong up-downwash generated as a result of interaction with the wall. Therefore, at the stroke reversal, the more rigid wing cannot benefit from the phenomenon of the wake capture. However, in the more flexible wing case, vortices from the previous stroke interact with the wing and newly generated vortices.

REFERENCES

- [1] J. Young. Numerical simulation of the unsteady aerodynamics of flapping airfoils. *The University of New South Wales Ph.D. thesis*, Sydney, 2005.
- [2] R. Knoller. Die gesetze des luftwiderstandes. *Flug- und Motortechnik (Wien)*, 3(21): 1–7, 1909.
- [3] A. Betz. Ein beitrag zur erklärung des segelfluges. *Zeitschrift für Flugtechnik und Motorluftschiffahrt*, 3: 269–272, 1912.
- [4] J.M. Anderson, K. Streitlien, D.S. Barrett, and M.S. Triantafyllou. Oscillating foils of high propulsive efficiency. *Journal of Fluid Mechanics*, 360: 41–72, 1998.
- [5] J.C.S. Lai, and M.F. Platzer. Jet characteristics of a plunging airfoil. *AIAA Journal*, 37(12): 1999.
- [6] J. Young, and J.C.S. Lai. Oscillation frequency and amplitude effects on the wake of a plunging airfoil. *AIAA Journal*, 42(10): 2042–2052, 2004.
- [7] I.H. Tuncer, and M. Kaya. Optimization of flapping airfoils for maximum thrust and propulsive efficiency. *AIAA Journal*, 43(11): 2005.
- [8] J. Young, and J.C.S. Lai. Vortex lock-in phenomenon in the wake of a plunging airfoil. *AIAA Journal*, 45(2): 2007.
- [9] J. Young, and J.C.S. Lai. Mechanisms influencing the efficiency of oscillating airfoil propulsion. *AIAA Journal*, 45(7): 2007.
- [10] M.F. Platzer, K.D. Jones, J. Young, and J.C.S. Lai. Flapping-wing aerodynamics: progress and challenges. *AIAA Journal*, 46(9): 2008.
- [11] S.P. Sane. The aerodynamics of insect flight. *The Journal of Experimental Biology*, 206: 4191–4208, 2003.
- [12] T. Weis-Fogh. Quick estimates of flight fitness in hovering animals, including novel mechanisms for lift production. *The Journal of Experimental Biology*, 59: 169–230, 1973.
- [13] F.O. Lehmann. The mechanisms of lift enhancement in insect flight. *Naturwissenschaften*, 91(3): 101 – 122, 2004.
- [14] J. Marden. Maximum lift production during take-off in flying animals. *The Journal of Experimental Biology*, 130:235–258, 1987.
- [15] C.P. Ellington. The aerodynamics of insect flight. IV. Aerodynamic mechanisms. *Philosophical Transactions of the Royal Society B*, 305:79–113, 1984.
- [16] F.O. Lehmann, S.P. Sane, M. Dickinson. The aerodynamic effect of wing-wing interaction in flapping insect wings. *The Journal of Experimental Biology*, 208: 3075–3092, 2005.
- [17] R.J. Wootton. Support and deformability in insect wings. *Journal of Zoology: Proceedings of the Zoological Society of London*, 193: 447–468, 1981.
- [18] S.A. Combes, and T.L. Daniel. Flexural stiffness in insect wings I. Scaling and influence of wing venation. *The Journal of Experimental Biology*, 206: 2979–2987, 2003.
- [19] S. Heathcote, and I. Gursul. Flexible flapping airfoil propulsion at low Reynolds numbers. *AIAA Journal*, 45(5): 1066–1079, 2007.
- [20] P. Gopalakrishnan, and D.K. Tafti. Effect of wing flexibility on lift and thrust production in flapping flight. *AIAA Journal*, 48(5): 865–877, 2010.
- [21] M. Vanella, T. Fitzgerald, S. Preidikman, E. Balaras, and B. Balachandran. Influence of flexibility on the aerodynamic performance of a hovering wing. *The Journal of Experimental Biology*, 212: 95–105, 2009.
- [22] S. Heathcote, Z. Wang, and I. Gursul. Effect of spanwise flexibility on flapping wing propulsion. *Journal of Fluids and Structure*, 24: 183–199, 2008.
- [23] C.P. Ellington. The aerodynamics of insect flight. III. Kinematics. *Philosophical Transactions of the Royal Society B*, 305:41–78, 1984.
- [24] L.A. Miller, and C.S. Peskin. Flexible clap and fling in tiny insect flight. *The Journal of Experimental Biology*, 212: 3076–3090, 2009.

- [25] K.M.E. De Clercq. Flow visualization and force measurements on a hovering flapping wing MAV DelFly II. *Delft University of Technology M.Sc. thesis*, 2009.
- [26] B. Bruggeman. Improving flight performance of DelFly II in hover by improving wing design and driving mechanism. *Delft University of Technology M.Sc. thesis*, 2010.
- [27] M. Groen. PIV and force measurements on the flapping-wing MAV DelFly II. *Delft University of Technology M.Sc. thesis*, 2010.
- [28] G.E. Elsinga, F. Scarano, B. Wieneke, and B.W. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41: 933-947, 2006.
- [29] D. Violato, K. Bryon, P. Moore, and F. Scarano. Application of Powell's analogy for the prediction of vortex-pairing sound in a low-Mach number jet based on time-resolved planar tomographic PIV. *16th AIAA/CEAS Aeroacoustics Conference*, 2010.
- [30] C.P. Ellington. The novel aerodynamics of insect flight: applications to Micro-Air Vehicles. *The Journal of Experimental Biology*, 202: 3439-3448, 1999.
- [31] B. Wieneke. Volume self-calibration for 3D particle image velocimetry. *Experiments in Fluids*, 45: 549-556, 2008.

Dynamic Behaviors of a Vortex Ring on a Butterfly and a Small Flapping Robot

M. Fuchiwaki¹, T. Kuroki¹, K. Tanaka¹, and T. Tabata²

Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, Japan

Kagoshima National College of Technology, 1460-1 Shinkou Hayato, Kirishima, Japan

ABSTRACT

Butterflies fly combining wing flapping and gliding efficiently and have beautiful flight patterns. Micro-Air-Vehicles (MAVs) and micro-flight robots that mimic the flight mechanisms of insects have been attracting significant attention in recent years. A number of MAVs and micro-flight robots that use various devices have been reported. However, these robots were not practical. A number of studies on the mechanism of butterfly flight have been carried out. Moreover, a number of recent studies have examined the flow field around an insect wing, such as a leading edge vortex (LEV) structure using the first digital particle image velocimetry (DPIV). We have carried out the PIV measurement around a butterfly wing and have visualized a vortex ring formed on the wing clearly. On the other hand, we developed a small flapping robot without tail wings, which is similar to the butterfly. The purpose of the present study is to clarify the dynamic behaviors of the vortex rings formed on the wings of the butterfly and the small flapping robot. A vortex ring is formed over the wings of the butterfly and small flapping robot with stable flight when the wings flap downward. An another vortex ring is formed below the butterfly wings when the wings flap upward and it was smaller than that in downward flapping motion because of the elastic deformation of the wings. Both vortex rings pass through the butterfly completely at the top and bottom dead position in the flapping motion.

1 INTRODUCTION

Micro-air-vehicles (MAVs) and micro-flight robots have been actively developed in recent years. These technologies are being developed for use in search and rescue operations in areas with a high risk of secondary disasters and for monitoring security risks. Several MAVs and micro-flight robots that employ various devices have been reported. Among them, flapping robots that mimic the flight mechanism of insects and that have good flying abilities have been attracting considerable attention [1][2]. Tanaka et al. [3][4] developed a very small and lightweight butterfly-type ornithopter (BTO) to investigate butterfly flight. Their BTO has a mass of 0.4 g, a wingspan of 140 mm, and a flapping frequency of 10 Hz. However, these robots require a tail wing to achieve stable flight. It is difficult to develop flapping robots that do not have a tail wing like butterflies. One reason for this is that the flying mechanism of insects has not yet been sufficiently clarified.

Several recent studies have investigated the mechanism of butterfly flight [5][6]. Dickinson et al. [7] identified three important motions; delayed stall, rotational circulation, and wake capture. Senda et al. [8] analyzed these motions and

measured the forces of the butterfly *Parantica sita* by observing its flight.

Several studies have examined the flow field around an insect wing [9]. Michael et al. [10] used flow visualization and instantaneous force measurements of tethered fruit flies, *Drosophila melanogaster*, to study the dynamics of force generation during flight. They presented a reconstruction of a vortex loop during the downstroke and ventral flip. Richard et al. [11] used first digital particle image velocimetry (DPIV) to investigate flow around the wings of an insect. They investigated the moth *Manduca sexta* and reported the relationship between the structure of the leading edge vortex (LEV) and force production. They [12] employed smoke visualization techniques to analyze the aerodynamic mechanisms of free-flying bumblebees and reported the independent LEVs on both wings. However, the dynamic behavior of the vortex that forms on the insect wing and its growth process have not been clarified.

By observing the flight of the butterfly *Cynthia cardui* we have clarified the behavior of its wings in flight. By spatial evaluation of the wing during flapping flight, we found that the flapping angle, lead-lag angle and feathering angle of the butterfly are periodic. Moreover, we have clarified that the wing deforms elastically in both the wing span and chord directions [13]. Based on these results, we developed a small flapping robot that does not have tail wings and hence resembles a real butterfly [14]. We found that elastic deformations of the wing cause twisting motions of the wing, which enable stable flight [15]. Moreover, we performed particle image velocity (PIV) measurements around a butterfly wing and clearly visualized a vortex ring formed on the wing; this vortex ring was formed over the wings for all the butterfly species examined [16]. However, the dynamic behavior of the vortex ring is not sufficiently well understood. We are particularly interested in the dynamic behavior of the vortex ring during up and down flapping.

The purpose of the present study is to clarify the dynamic behavior of the vortex rings that form above the wings of the butterfly and the small flapping robot. We performed PIV measurement around the wings and investigated the dynamic behavior of the vortex rings in upward and downward flapping motions.

2 EXPERIMENTAL SYSTEMS

2.1 Butterfly and small flapping robot

Figure 1(a) and (b) respectively depict the butterfly and small flapping robot used in the present study. Table 1 lists

Email address: futiwaki@mse.kyutech.ac.jp

(a) Butterfly, *Idea leuconoe*

(b) Small Flapping robot

Figure 1: Butterfly and small flapping robot in our study

Table 1: Dimensions of butterfly and small flapping robot

	Butterfly	Small flapping robot
c [mm]	60	80
l [mm]	140	240
AR	2.3	3.0
m [g]	0.4	1.9
W/S [N/m ²]	0.6	1.5
f [Hz]	6	10

their wing chord length c , wing span length l , aspect ratio of the wing AR , mass m , wing load w/s , and flapping frequency of the wing f .

The butterfly shown Fig. 1(a) is *Idea leuconoe*; it was captured on our campus. The small flapping robot has a motor and battery and it weights approximately 1.9g. A crank mechanism is used to convert the rotary motion of the motor into flapping motion of the two wings with a flapping frequency of about 10 Hz. The battery can power flight for about 20 minutes. Over the time, the flapping frequency decreases to 9 Hz due to the voltage drop of the battery.

Figure 2 shows flight trajectories of the small flapping robot. The flying trajectories were recorded using a fixed digital video camera with a frame rate of 30 [fps]. The time when the flapping robot leaves the hand is taken to be $t = 0.0$

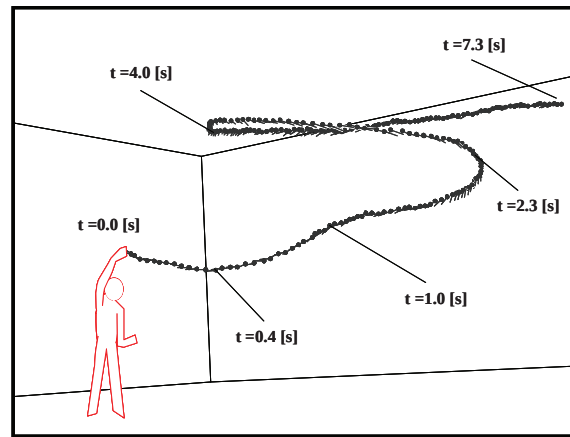


Figure 2: Flight trajectories of our small flapping robot

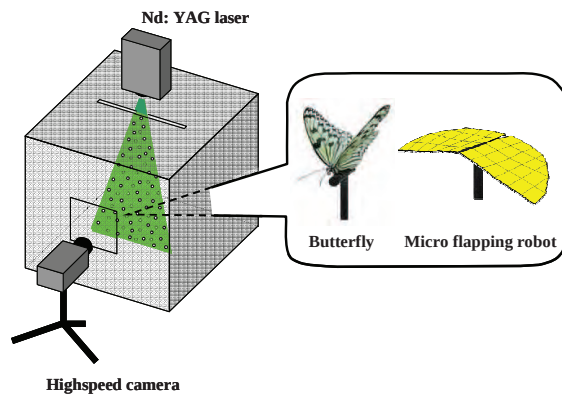


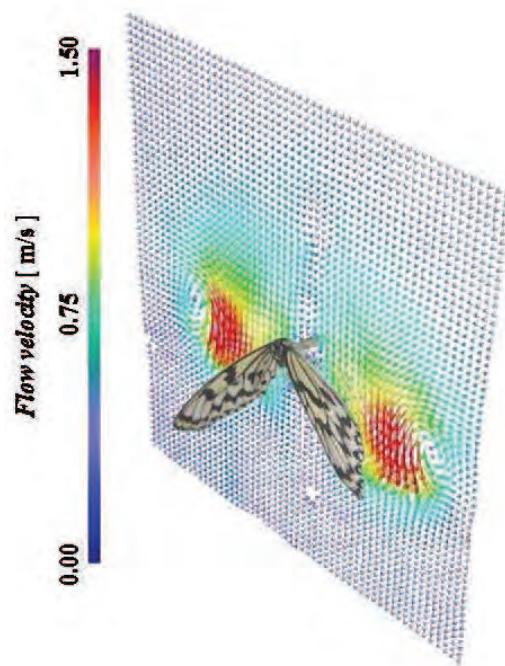
Figure 3: PIV measurement system in our study

[s] and flight trajectories are depicted up to $t = 7.3$ [s].

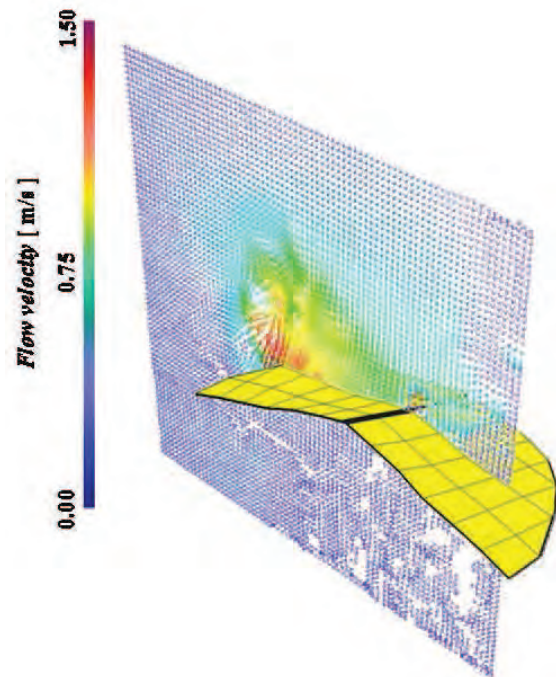
The flying posture of the small flapping robot becomes unstable and its height decreases from $t = 0$ to $t = 0.4$ [s]. The flying posture continues to be unstable until $t = 1.0$ [s], but the height of the robot increases. After $t = 1.0$ [s], the flying posture becomes almost constant and the robot ascends while performing large turns to the left. The small flapping robot flew stably for 20 minutes (i.e. for the life of the battery). The small flapping robot ascends for 7-8 minutes while slewing and then descends while slewing. This is because the revolution speeds of the motor decreases with decreasing voltage of the battery and consequently the flapping frequency generated by the crank mechanism decreases. By reversing the rotary motion of the motor applied to the crank mechanism, we found that the slewing direction in the flight of the small flapping robot is counterclockwise.

2.2 PIV measurement system

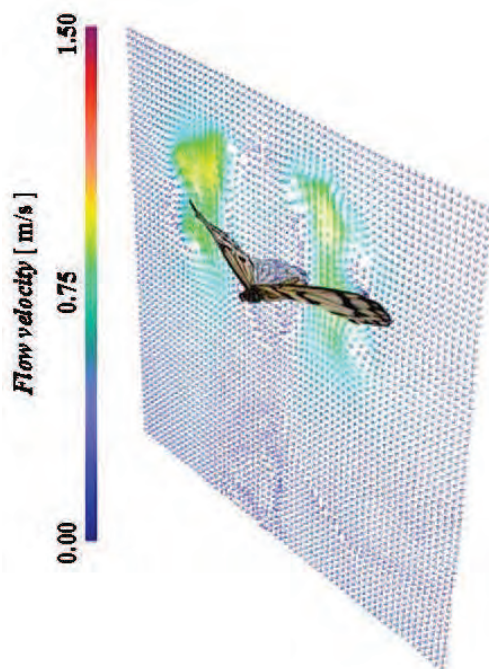
We performed PIV measurements using an acrylic container, a high-speed camera, a continuous-wave Nd:YAG laser, and a butterfly (see Fig. 3). The test section in the acrylic container is 900 [mm] long, 900 [mm] wide, and 900 [mm] deep. Expancel (Kanomax), which has a diameter of approximately 10 [μm] and a specific gravity of 0.7, was used as tracer particles. In the butterfly experiments, the legs



(a) Downward flapping motion

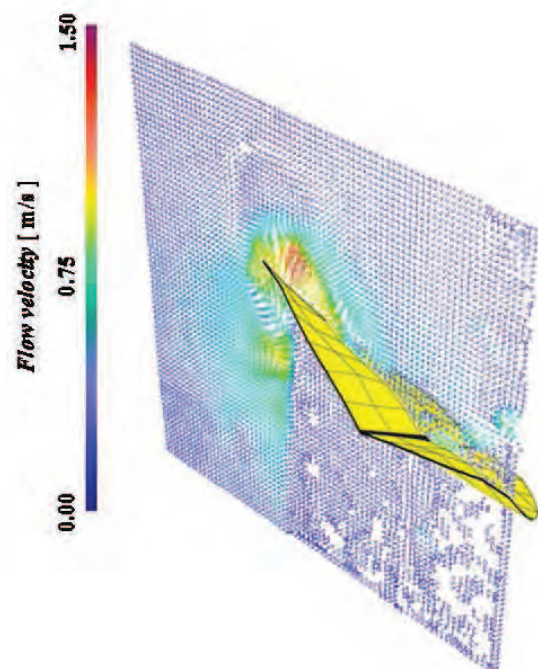


(a) Downward flapping motion



(b) Upward flapping motion

Figure 4: Velocity vectors around wings of the butterfly



(b) Upward flapping motion

Figure 5: Velocity vectors around wings of the small flapping robot

of the butterfly were fixed to a shaft in the test section without hindering the butterfly's flight motion. In the small flapping robot experiments, the body of the robot was fixed to a shaft in the test section without hindering its flight motion and the flapping frequency was 10 Hz. Measurements were performed with the illumination plane aligned with the

chord and with it aligned with the span. In the first configuration, 12 planes were studied at chordwise positions including the leading edge, the center of the wing chord, the trailing edge, and the wake. In the second configuration, four planes were studied at spanwise positions including the wing

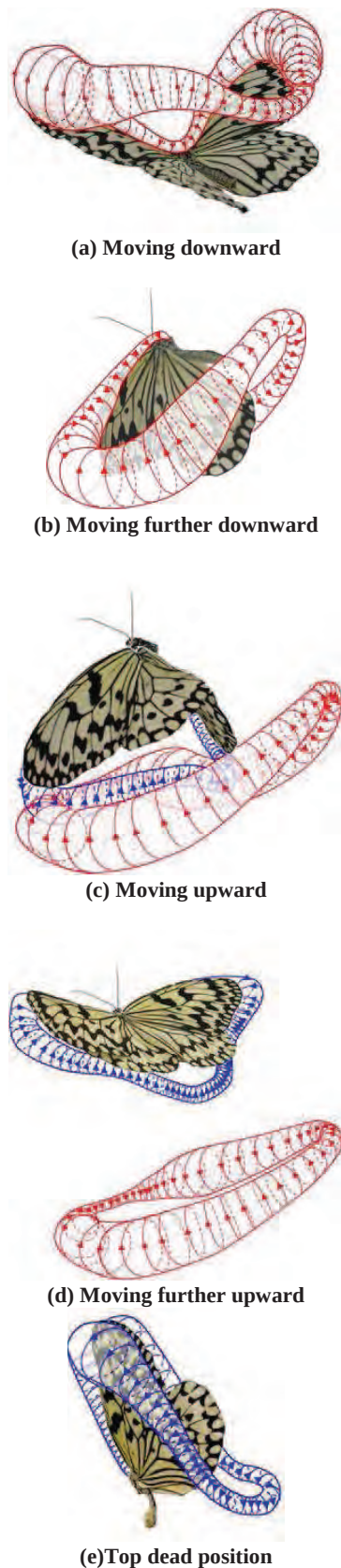


Figure 6: Pattern diagrams of the vortex on the butterfly wings obtained by the PIV measurement results

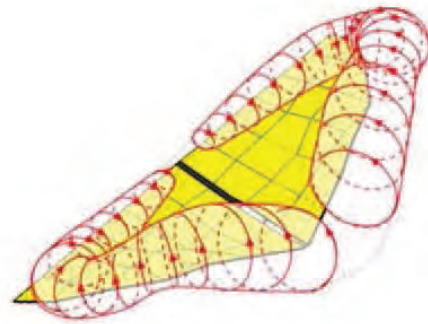


Figure 7: Pattern diagrams of the vortex on the wings of the small flapping robot obtained by the PIV measurement results

tip, the center of the chord, the basic point of the wing, and the body. Images were captured using a high-speed camera (2,000 [fps]) and the vortex flows that developed in the wing chord and span directions were investigated.

We investigated the wing tip traces of a fixed butterfly and found that the dynamic behavior of the wings was similar to that in take-off flight. Moreover, the dynamic behaviors of the butterfly wings and of the small flapping robot under fixed conditions are highly periodic. Therefore, the flow fields around the butterfly and small flapping robot are also expected to vary periodically. In the present study, we found a three-dimensional vortex above the wings by analyzing two-dimensional flow field at each point of the wings.

3 RESULTS AND DISCUSSIONS

3.1 Velocity vectors around wings of the butterfly and small flapping robot

Figures 4(a) and 4(b) respectively show velocity vectors around the butterfly wings with up and down flapping motion obtained by PIV measurements.

As the wings move downward (see Fig. 4(a)), a pair of vortices is clearly observed above the wings. This pair of vortices initially forms between the wings at the beginning of the downward flapping motion and then develops over the wing surfaces. Similarly, a pair of vortices is observed below the wing with upward flapping motion (see Fig. 4(b)). The size and velocity of vortices are smaller for downward flapping motion than for upward flapping motion. These vortices were observed at each point in the wing chord direction. That is, tube vortices form over the wings in the wing chord direction. Moreover, the tube vortices above the two wings are connected by one chord length. Furthermore, a LEV is observed in the wing span direction and it is connected with the tube vortices in the wing chord direction. As a result, a vortex ring formed over the wings of the butterfly [16].

Figures 5(a) and 5(b) respectively show velocity vectors around the wings of the small flapping robot with up and down flapping motion. Similarly to the butterfly, a pair of vortices is clearly observed above the wings of the robot (see Fig. 5(a)). Moreover, a pair of vortices is observed below the wings during upward flapping motion (see Fig. 5 (b)); this

pair of vortices is smaller than those generated during downward flapping motion, which is the same as for the butterfly wing.

3.2 Dynamic behaviors of a vortex ring

Figure 6 shows pattern diagrams of the vortex on the butterfly wings obtained from vorticity isosurfaces which were derived by analyzing the PIV results for each point in the wing chord and span directions. Figures 6(a-e) show the results obtained for the region moving downward, moving further downward, moving upward, moving further upward and at the top dead position, respectively.

A vortex ring begins to form when the wings flap downward and it is completely formed over the wings when the wings flap downward to the bottom dead position (see Fig. 6(a)). The vortex ring develops over the wings when the wings flap further downward (see Fig. 6(b)) and then completely passes through the butterfly and grows until reaching the wake at the bottom dead position (see Fig. 6(c)). Another vortex ring begins to form when the wings flap upward (see Fig. 6(c)) and it is completely formed below the wings when the wings flap upward to the top dead position (see Fig. 6(d)). The vortex ring formed during upward flapping motion completely passes through the butterfly completely, just as during downward flapping (see Fig. 6(e)).

Both vortex rings completely pass through the butterfly at the top and bottom dead positions during the flapping motion. However, the size of the vortex ring that forms during downward flapping is larger than that formed during upward flapping. The veins in butterfly wings have different diameters on the upper and lower sides of the wings, which important butterfly wings with different elastic deformations during upward and downward flapping motions. Consequently, the vortex rings have different size and dynamic behaviors during upward and downward flapping and the lifts and drags generated differ accordingly. Butterflies are thought to generate lift and drag through the induced downward and backward flows in the vortex ring.

Figure 7 shows pattern diagrams obtained by PIV measurements for the vortex ring formed above the wings of the small flapping robot.

A vortex ring forms above the wings of the small flapping robot during stable flight, just as for butterfly wings. However, the vortex ring formed above the wing during unstable flight has a strange shape. Specifically, the rear of the vortex ring is deformed, making it difficult to form a vortex ring. It was found that the formation of a vortex ring above the wing is an important requisite for stable flight. However, we have not yet clarified the dynamic behaviors of the vortex ring of the small flapping robot.

4 CONCLUSIONS

A vortex ring was clearly observed above the wings of the butterfly and small flapping robot with stable flight. It was found that the formation of a vortex ring above the wing is an important requisite for stable flight.

Another vortex ring formed below the butterfly wings when the wings flap upward to the top dead position. This vortex was smaller than that formed during downward flapping due difference in the elastic deformation of the wings during

upward and downward flapping. Both vortex rings completely pass through the butterfly at the top and bottom dead positions during the flapping motion.

ACKNOWLEDGMENT

The authors are grateful to the Mitsubishi Foundation for supporting the present work.

REFERENCES

- [1] T. N. Pornsin-sirirak, S. W. Lee, H. Nassef, J. Grasmeyer, Y. C. Tai, C. M. Ho and M. Keennon, Mems Wing Technology for a Battery-Powered Ornithopter, *13th IEEE Annual International Conference on MEMS*, pp. 709-804, 2000.
- [2] B. Singh and I. Chopra, Insect-based hover-capable flapping wings for micro air vehicles: experiments and analysis. *AIAA J* 46; pp.2115-2135, 2008.
- [3] H. Tanaka, K. Hoshio, K. Matsumoto and I. Shimoyama, Flight dynamics of a butterfly-type ornithopter. *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2706-2711, 2005.
- [4] H. Tanaka and I. Shimoyama, Forward flight of swallowtail butterfly with simple flapping motion. *Biosp Biomim* 5; 026003, 2010
- [5] S. Sunada, K. Kawachi, I. Watanabe, A. Azuma, Performance of a butterfly in take-off flight. *J Exp Biol* 183; pp.249-277, 1993
- [6] J. S. Lee, J. H. Kim and C. Kim, Numerical study on the unsteady force generation mechanism of insect flapping motion. *AIAA J* 46; pp.1835-1848, 2008
- [7] M. H. Dickinson, F. O. Lehmann and S. P. Sane, Wing rotational and aerodynamic basis of insect flight. *Science* 284(5422); pp.1954-1960, 1999
- [8] K. Senda, M. Sawamoto, T. Shibahara and T. Tanaka, Study on flapping-of-wings flight of butterfly with experimental measurement. *AIAA Atmospheric Flight Mechanics Conf. Exhibit*; pp.2004-5368, 2004.
- [9] L. Yuan and X. S. Gong XS, Three-dimensional flow structures and evolution of the leading-edge vortices on a flapping wing. *J Exp Biol* 211; pp.1221-1230, 2008
- [10] H. D. Michael and G. G. Karl GG, The wake dynamics and flight forces of the fruit fly *drosophila melanogaster*. *J Exp Biol* 199; pp.2085-2104, 1996
- [11] J. B. Richard, J. L. Nicholas, J. H. Nicholas, K. T. Graham and L. R. T. Adrian, The aerodynamics of *manduca sexta*: digitalparticle image velocimetry analysis of the leading-edge vortex. *J Exp Biol* 208; pp.1079-1094, 2005
- [12] J. B. Richard, K. T. Graham and L. R. T. Adrian, Smoke visualization of free-flying bumblebees indicates independent leading-edge vortices on each wing pair. *Exp Fluids* 46; pp.811-821, 2009
- [13] M. Fuchiawaki and K. Tanaka, Vortex flow on a butterfly wing. *12th Int Symp Flow Vis*; 12ISFV-156, 2006.
- [14] M. Fuchiawaki, I. Tadatsugu and K. Tanaka, Characteristics of butterfly wing motions and their application to micro flight robot. *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, 2009
- [15] M. Fuchiawaki and K. Tanaka, Dynamic behavior of butterfly wing and its vortex flow. *13th Int Symp Flow Vis*; 13ISFV-103, 2008.
- [16] M. Fuchiawaki, T. Kuroki, K. Tanaka and T. Tababa, Dynamic behavior of vortex ring formed on a butterfly wing and its structure, *International Conference on Jets, Wakes and Separated Flows, ICJWSF-2010*, 2010.

Analysis of tail effects in flapping flight

W.B Tay, H. Bijl, and B.W. van Oudheusden

Delft University of Technology, Kluyverweg 2, 2629 HT Delft, The Netherlands

ABSTRACT

Numerical simulations have been performed to examine the interference effects between an upstream flapping airfoil and a downstream stationary airfoil in a tandem configuration at a Reynolds number of 1000, which is around the regime of small flapping micro aerial vehicles. The objective is to investigate the effect of the distance of the tail and its angle of attack on the overall propulsive efficiency, thrust and lift. An immersed boundary method Navier-Stokes solver is used for the simulation. Results show that efficiency and average thrust can be increased up to 10% and 25% respectively when a stationary airfoil is placed downstream. The simulations reveal how the vortex-shedding pattern of the airfoils are affected by the interaction between them. As the angle of attack of this airfoil increases from 0 to 45°, high lift is generated at the expense of rapidly decreasing efficiency and thrust. The results are not very sensitive to the shape of the airfoil; similar results are obtained with a flat plate airfoil. Lastly, a simple optimization study is performed to obtain the configuration which gave the best performance based on the range of parameters studied. The results obtained from this study can be used to optimize the performance of small flapping MAVs.

INTRODUCTION

In recent years, micro-aerial vehicles (MAVs) are becoming increasingly important, especially in the area of military surveillance. MAVs can be classified into fixed wing, rotary or flapping wing [1][2] MAVs. In terms of maneuverability and efficiency at this low Reynolds number regime, flapping wing MAVs have a clear advantage. Most designs of flapping wing MAVs or ornithopters have a tail at the end of the fuselage. It is mainly used for pitch and direction control. Unlike fixed wing aircraft, the interference effects between the flapping wing and the tail are more complex and have been investigated in many ways [3-7]. One of the earliest investigations in this area is performed by Schmidt [3], who found that the overall efficiency, lift and thrust increase when a tail is placed behind a flapping wing. This is followed by Bosch [4]. His theoretical analysis shows that a lifting surface at rest located behind an oscillating airfoil considerably improves the propulsive efficiency (η) of the arrangement by almost 100%. Platzer [5] uses an unsteady panel code to do a systematic study of thrust and lift characteristics produced by oscillating airfoils and airfoil combinations in an incompressible inviscid flow. The efficiency increases from 48% to 60% in one of the tandem configuration, indicating that the vertical energy produced by the upstream airfoil can be exploited for thrust generation by the downstream airfoil. However, the predictive capabilities of a panel code for their type of flows is limited. Besides being based on inviscid flow model, it also cannot accurately describe simulations where vortices from the

upstream airfoil impinges on the downstream airfoil. This shows that x_{12} and α are important factors affecting the outputs. Tuncer [6] uses an overset multi-block Navier-Stokes solver to simulate flapping/stationary airfoils in a tandem configuration at $Re = 3.0 \times 10^6$. He varies the reduced frequency, plunging amplitude and x,y displacement of the downstream airfoil with respect to the upstream one. It is found that the efficiency depends heavily on the reduced frequency and amplitude of the flapping motion. Compared to a single flapping airfoil, the efficiency of the tandem airfoils increases by more than 40% depending on the flapping configuration. Moreover, thrust augmentation is highest when the two airfoils are closely aligned. Rival et al. [7] performed a particle image velocimetry (PIV) experiment at $Re = 3.0 \times 10^4$ on the tandem configuration of two airfoils. The flapping airfoil is in front while a downstream airfoil is held fixed with a specified angle of attack (0°, 8°) and vertical spacing (0, -0.25, -0.5) in the front airfoil's wake. It was found that the vortices strengthen due to the tandem configuration and the downstream airfoil's angle of attack determines whether the type of vertical interaction is inviscid or viscous. Moreover, the combined lift of the tandem configuration is doubled that of the original single airfoil case whereas the mean drag increases or decreases depending on the angle of attack and vertical spacing.

We have been developing our own flapping MAV, known as the Delfly for many years. Previous researches [1][8] have concentrated on the flapping wings. In this study, we are interested in the interference effects between the upstream flapping airfoil and the downstream airfoil or tail. As a preliminary step, we are simulating the tail with a single flapping airfoil in 2D instead of the biplane wing configuration of Delfly in 3D. An Immersed Boundary Method (IBM) [9] Navier-Stokes solver [10] is used for the simulations. The motivation for adopting this approach will be discussed in more details in the following section. In our study, we focus mainly on the effects of the distance and angle of the tail behind a flapping airfoil on the overall efficiency, lift and thrust. Unlike Tuncer [6], our simulations run at a much lower Re of 1000, which will help to limit the grid requirements as the non-conformal grid of the IBM requires a very high resolution grid. This value corresponds to the Reynolds number ranges in the range of insects' flight and small MAVs.

NUMERICAL METHOD

The unsteady viscous flow fields are computed by solving the non-dimensioned Navier-Stokes equations using the IBM. Due to the tandem arrangement of the airfoil and tail,

Email address: W.B.Tay@tudelft.nl

it is difficult to use conforming grids with Arbitrary Lagrangian–Eulerian (ALE) formulation since maintaining grid quality can be a challenge, especially when the distance between the airfoil and tail is very small. The overset grid solver is a viable choice but stability and conservation problems may arise [8]. The IBM used in this research is based on the scheme by Ravoux et al [10]. This method combines features from both the IBM and the volume of fluid (VOF) in order to compute flows past moving and deformable bodies.

We currently use a fully implicit scheme instead of the original fully explicit scheme to improve the stability of the solver. The unsteady flow results are analyzed in terms of efficiency, thrust, lift coefficients, while the flow behavior is visualized with the help of vorticity contour plots. The discretization and the time advancement scheme will be discussed in the next section. More details about the solver and its verifications at $Re = 1000$ can be found in [10] and [11].

Fractional step method

A fractional step method, which is based on an improved projection method [12], is used to solve the modified non-dimensional incompressible Navier-Stokes equations in Equation 1 and 2.

$$(1) \nabla \cdot \mathbf{u} = 0,$$

$$(2) \frac{\partial \mathbf{u}}{\partial t} = -\mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{Re} \nabla^2 \mathbf{u} - \nabla p + \mathbf{f}c$$

where $\mathbf{u}$ is the velocity, t is the time, p is the pressure, $\mathbf{f}c$ is the external body force, Re is the Reynolds number. A fully implicit scheme which uses the Crank Nicholson approximation for the viscous term and convective term is employed to improve the stability of the solver. All spatial derivatives are discretized using the second-order central difference scheme on a staggered grid.

Computational domain

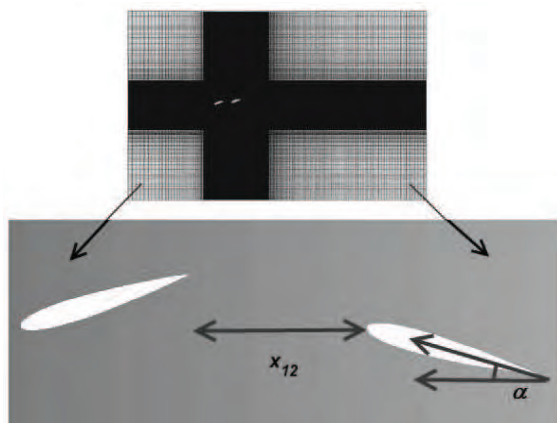


Figure 1: An example of the 1320×1120 Cartesian grid and its magnification

Figure 1 gives an example of the grid and its magnification. The upstream airfoil flapping motion is specified by

$$(3) h = h_0 \sin(2\pi f(t-t_0) - \phi), \quad \theta = \theta_0 \sin(2\pi f(t-t_0))$$

where h , h_0 and f are the instantaneous heaving position, heaving amplitude (normalized with c) and frequency respectively. Similarly, θ and θ_0 are the instantaneous pitch angle and pitch amplitude respectively.

The downstream (downstream) airfoil is stationary and simulates the tail of the airfoil. The pitching motion is about the quarter chord point. The phase angle is 90° in all simulations, so that the pitch and heave are one quarter period shifted in phase.

Boundary conditions

The boundary conditions prescribe on the domain on the boundaries are :

Inflow boundary - $u_x = U_\infty = 1, u_y = 0, dp/dx = 0$.

Top/bottom boundary - $u_y = 0, du_x/dy = 0, dp/dy = 0$.

Outflow boundary - $\frac{\partial u_i}{\partial t} + v_c \frac{\partial u_i}{\partial x} = 0$, where v_c is space-averaged streamwise velocity at the exit [13], $p = 0$ where the subscript i refers to the x and y directions.

RESULTS AND DISCUSSIONS

In this study, the objective is to investigate the effect of the distance of the tail x_{12} and its angle of attack on the overall efficiency, thrust and lift. The upstream NACA012 airfoil flaps with a prescribed motion. The Strouhal number St and the reduced frequency k are fixed at 0.2 and 0.4 respectively to isolate the effects of x_{12} and α . Moreover, these particular values are chosen because previous simulation results [14] showed that using too low values resulted in little or no shedding of vortex. This is because the energy input is too low. Tuncer [6] also observed this when low k and h_0 are used. On the other hand, high St and k may result in chaotic vortex shedding. This will hinder a systematic analysis of the effect of x_{12} and α . A description of the motion parameters for the detailed flapping configuration is given in Table 1.

Table 1: Flapping configuration

Re	1000
$St (fh_0/U_\infty)$	0.2
$k (fc/U_\infty)$	0.4
ϕ / degrees	90.0
θ_0 / degrees	17.5
x_{12} (normalized with c)	0.001 – 3.0
α / degrees	-30.0 – 45.0

The downstream NACA0012 airfoil, simulating the tail, is

placed behind the upstream airfoil in a tandem configuration. The range of values of x_{12} used in this study is between 0.001 and 3.0. It is found that beyond $x_{12} = 3.0$, the effect of the tail on the overall C_t and C_l is negligible. For the α , the range is between -30.0° and 45.0° , which is much larger than the angle used normally for the tail. The single airfoil is simulated under the same flapping configuration for comparison.

In this study, the power input of an airfoil is defined as

$$(4) P_{in} = -L(t) \frac{dh}{dt} - M(t) \frac{d}{dt}$$

where L , M are the lift and moment respectively. Propulsive efficiency, η , which is a measure of the energy lost in the wake versus energy used in creating the necessary thrust, is given by

$$(5) \eta = \frac{\bar{C}_t}{\bar{P}_{in}}$$

Effect of x_{12}

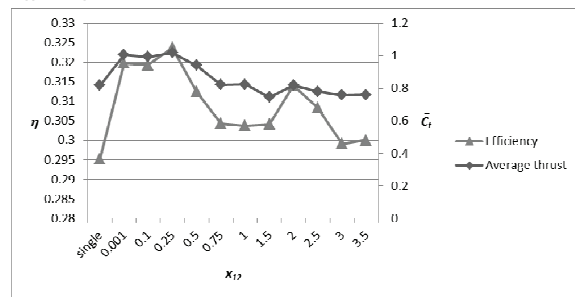


Figure 2: Plot of η and $\bar{C}_t$ against x_{12} , $\alpha = 0^\circ$

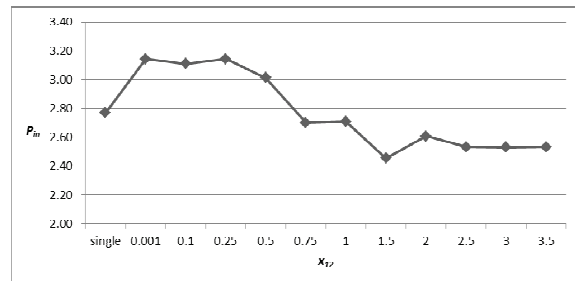


Figure 3: Plot of power input P_{in} against x_{12} , $\alpha = 0^\circ$

Figure 2 shows the efficiency (η) and average thrust coefficient ($\bar{C}_t$) for the single airfoil and the tandem airfoils at different x_{12} . Figure 3 shows the power input plot for the single airfoil and the tandem airfoils at different x_{12} . The outputs from the upstream and downstream airfoils are presented together because the thrust contribution from the downstream airfoil is small. Hence, it is sufficed to analyze the output of the tandem airfoils as a whole. In Figure 2, the $\bar{C}_t$ is much higher when x_{12} is small, and it decreases as x_{12} increases, until it is almost equal to that of the single airfoil. The trend for the η is similar to that of the $\bar{C}_t$, although at $x_{12} = 2.0$, there is a small increase. At $x_{12} = 3.0$, for the tandem airfoils is very close to that of the single airfoil.

Figure 4 shows the vorticity contour plot of the single airfoil. All the vorticity contour plots in this paper use the same

vorticity legend as Figure 4. A reverse Von Karmen vortex street indicating thrust can be seen. With a second airfoil (tail) behind, there is up to 25% increase in the overall $\bar{C}_t$ when $x_{12} \leq 0.75$. Comparing between Figure 4 and Figure 5, there is visually no difference in the vortex shedding of the upstream airfoil. The shed vortex from the upstream airfoil impinged on the leading edge of the downstream airfoil and induced in the shedding of its own vortex. The pair of vortices then shed diagonally. Since the upstream airfoil shed its vortex every half cycle, the diagonal vortex shedding also happened every half cycle towards the top and bottom right directions. Vorticity contour plots at other x_{12} also showed very similar shedding of the vortices. The main difference lies in the vortex shedding timing of the downstream airfoil relative to the upstream airfoil, as shown in Figure 6.

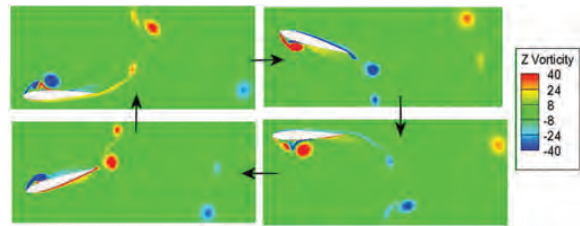


Figure 4: Vorticity contour plot of the single airfoil

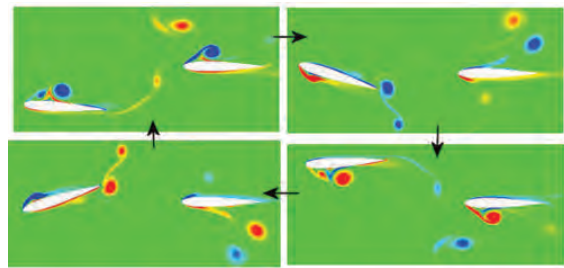


Figure 5: Vorticity contour plot of the tandem airfoils at $x_{12} = 1.0$, $\alpha = 0^\circ$

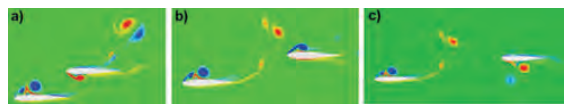


Figure 6: Difference in vortex shedding timing at $x_{12} =$ a) 0.001 b) 1.0 c) 2.0 when upstream airfoil is at its lowest plunging position

The thrust generated by the airfoil is reflected in the flow downstream and can be analyzed using wake velocity profiles. In this case, velocity profiles are extracted one chord length behind the tail. In Figure 7, the velocity profiles are plotted using the horizontal average component of $u_x - U_\infty$ against the vertical line between $-2.5 \leq y \leq 2.5$. Positive $u_x - U_\infty$ indicates thrust production. The center peak of the velocity profile for the single airfoil is due to the shedding of the vortices directly behind the airfoil, along the horizontal direction. For the tandem airfoils, the two peaks at either side of the central peak are due to the diagonal shedding of vortices. The sum of area under the curve indicates the total average thrust produced. It is 2.64×10^{-1} and 3.32×10^{-1} for the single and tandem airfoils at $x_{12} = 0.001$ respectively. Hence the average thrust for the tandem case is higher. The additional vortex shedding from the downstream airfoil helps to increase the total thrust.

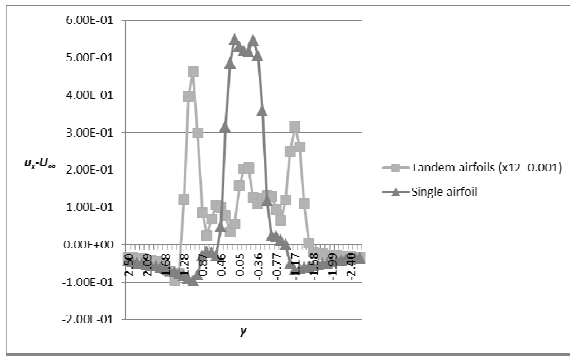


Figure 7: Average velocity profiles for the single airfoil and the downstream airfoil with $x_{12} = 0.001$, $\alpha = 0^\circ$

In Figure 2, the thrust decreases as x_{12} increases in the tandem configuration. This thrust decrease can be explained by comparing the intensity of the vortex shedding from the downstream airfoil for different x_{12} . When x_{12} increases, the shed vortex from the upstream airfoil travels a larger distance before impinging on the leading edge of the downstream airfoil. Its strength has decreased in the process. Figure 8 shows the shed pair of vortices at different x_{12} , at the instance where the upstream flapping airfoil is at its lowest plunging position. The red positive vortex from the upstream airfoil weakens in strength as x_{12} increases, hence giving lower thrust.

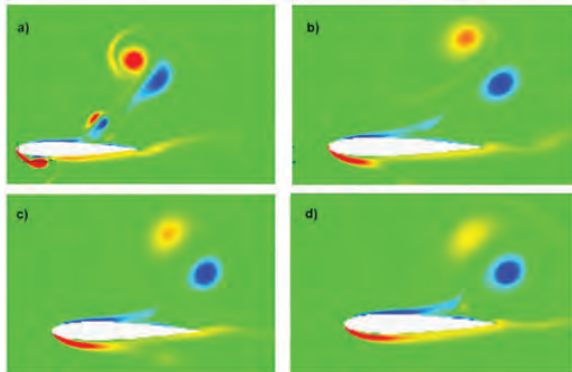


Figure 8: Vorticity strength comparison for different x_{12} = a) 0.001 b) 1.0 c) 2.0 d) 3.0, $\alpha = 0^\circ$

As shown in Figure 3, since the tail is stationary, P_{in} is entirely due to the motion of the flapping upstream airfoil. At $x_{12} \leq 0.75$, the P_{in} is larger than that of the single airfoil configuration. More work is needed to flap the upstream airfoil due to the blockage effect of the downstream airfoil in the near wake [6]. However, since $\bar{C}_t$ also increases, the P_{in} is also higher for $x_{12} \leq 0.75$. Figure 2 shows a maximum of 10% increase in the overall P_{in} when the tail is placed at $x_{12} = 0.25$. At $x_{12} > 1.5$, the P_{in} of the tandem configuration is almost the same as that of the single airfoil (neglecting the small increase at $x_{12} = 2.0$), indicating that the effect of the tail on the upstream airfoil is no longer significant.

In general, the $\bar{C}_t$ and $\bar{C}_l$ are higher when x_{12} is small, with the optimum at $x_{12} = 0.25$. The interference effects decrease as x_{12} increases and the outputs approach that of the single airfoil. The $\bar{C}_t$ increase is lower than that reported by Tuncer and Platzer [6]. They reported an increase of more than

40%. However, it is hard to compare directly because the Re and flapping parameters used are different.

Effect of

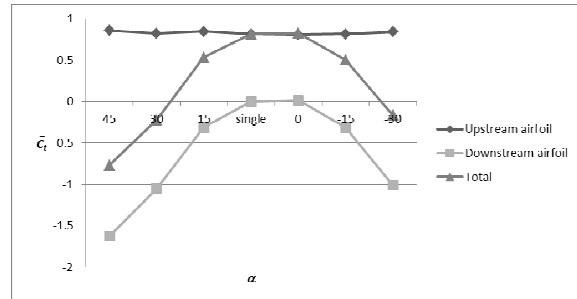


Figure 9: Plot of $\bar{C}_t$ against α at $x_{12} = 1.0$

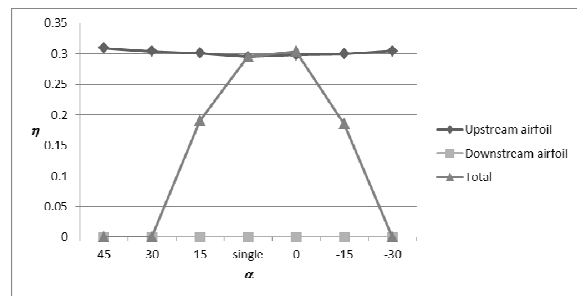


Figure 10: Plot of η against α at $x_{12} = 1.0$

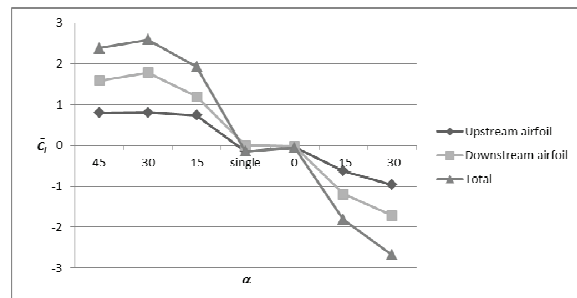


Figure 11: Plot of $\bar{C}_l$ against α at $x_{12} = 1.0$

In this analysis, x_{12} is fixed at 1.0 while α varies from -30° to 45° . The plots given in Figure 9, Figure 10 and Figure 11 show the $\bar{C}_t$, $\bar{C}_l$ and η of the upstream, downstream and tandem airfoils. The contributions of the upstream and downstream airfoils are shown separately; effect of α on the output is different for the downstream airfoil and the upstream airfoil. Figure 9 showed that the $\bar{C}_t$ of the upstream airfoil is the same for all α , indicating that the angle of attack of the downstream airfoil does not influence the upstream airfoil's $\bar{C}_t$. However, the $\bar{C}_t$ of the downstream airfoil decreases, producing drag, as it deviates away from $\alpha = 0^\circ$. $\bar{C}_t$ directly affects the η and hence it also drops rapidly when α deviates away from zero.

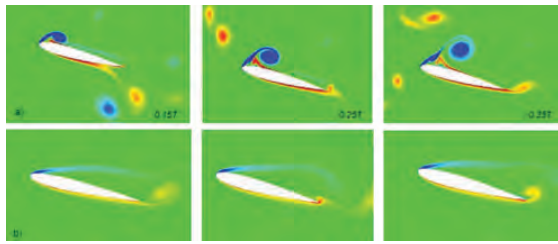


Figure 12: a) Downstream airfoil with a strong leading edge vortex (LEV) on top (upstream airfoil not shown) b) single airfoil at $\alpha = 15^\circ$ at time intervals of 0.15, 0.25 and 0.35T

The $\bar{C}_l$ of the single airfoil and the tandem configuration with $\alpha = 0^\circ$ are almost zero because the NACA0012 is a symmetrical airfoil. However, when α increases, the $\bar{C}_l$ increases both for the upstream and downstream airfoil, with the $\bar{C}_l$ of the downstream airfoil being almost twice as large. The reverse happens when α decreases.

Figure 12 shows a comparison of the vorticity plots between the single airfoil and the downstream airfoil (in a tandem configuration) at $\alpha = 15^\circ$. The top row of diagrams from Figure 12 shows that a LEV develops on top of the downstream airfoil as a result of the unsteady interaction with the upstream airfoil. It remains attached for about a quarter of the flapping cycle. This results in a low pressure region on top of the downstream airfoil, generating lift. There is also a LEV generated on the bottom of the downstream airfoil. However, it gets dissipated almost immediately. Similarly, when α decreases from 0° , the airfoil's bottom produces a LEV at the bottom of the airfoil, giving negative lift. The bottom row shows the single airfoil. Due to the high α , the flow is detached and there is no LEV present. It is clear that the vortex shed from the upstream airfoil helps to enhance the lift production of the downstream airfoil. There is no LEV on top of the single airfoil and the strength of its vortex is much weaker. In the tandem configuration, the vortex shed from the upstream airfoil imparts energy to the downstream airfoil and induces the generation of the LEV. The $\bar{C}_t / \bar{C}_l$ for the downstream and single airfoil are -0.31/1.18 and -0.27/0.69 respectively. Both their $\bar{C}_t$ are similar but the $\bar{C}_l$ of the downstream airfoil is almost twice that of the single airfoil. This improvement can be incorporated into the design of a flapping MAV like the Delfly to increase its lift.

Effect of using a thin plate for the downstream airfoil

The simulation is repeated using a thin plate instead of the NACA0012 airfoil as the downstream airfoil for the $x_{12} = 1.0$, $\alpha = 0^\circ$ and $x_{12} = 1.0$, $\alpha = 15^\circ$ cases. This is to investigate the importance of the shape of the downstream airfoils. Results on show that for α , $\bar{C}_t$ and $\bar{C}_l$, the differences between the tail and the NACA0012 are very small but in general, the NACA0012 gives slightly better performance.

Table 2: Comparison between the $\bar{C}_t$ and $\bar{C}_l$ of the thin plate and NACA0012 airfoil for the $x_{12} = 1.0$, $\alpha = 0^\circ$ and $x_{12} = 1.0$, $\alpha = 15^\circ$ cases

	Output	NACA0012	Plate
0°		0.30	0.29
	$\bar{C}_t$	0.82	0.77
	$\bar{C}_l$	-0.06	-0.07
15°		0.19	0.17
	$\bar{C}_t$	0.53	0.46
	$\bar{C}_l$	1.92	1.81

Optimization study

We also conduct a simple¹ optimization study on the configuration which gives the highest performance based on the combined values of x_{12} and α . Performance index I_p , used to determine the overall performance of the tandem airfoils in this case, is defined as:

$$(6) \quad I_p = \frac{(x_{12})}{\max(\text{all } (x_{12}))} + \frac{\bar{C}_t(x_{12})}{\max(\text{all } \bar{C}_t(x_{12}))}$$

for the x_{12} case study.

$$(7) \quad I_p = \frac{(\alpha)}{\max(\text{all } (\alpha))} + \frac{\bar{C}_t(\alpha)}{\max(\text{all } \bar{C}_t(\alpha))} + \frac{\bar{C}_l(\alpha)}{\max(\text{all } \bar{C}_l(\alpha))}$$

for the α case study.

Equation 6 does not include the lift component because it is a symmetrical case and the $\bar{C}_l$ for all the x_{12} cases are approximately zero. Based on Equation 6 and Equation 7, the x_{12} and α values which give maximum I_p are 0.25 and 15° respectively. Running the simulation at these two values gave $\alpha = 0.23$, $\bar{C}_t = 0.71$ and $\bar{C}_l = 1.61$, which indeed corresponds to a higher I_p than the values obtained in the x_{12} and α case studies.

CONCLUSION

Numerical flow simulations have been performed out to investigate the effects of the distance and angle of the tail behind a flapping airfoil on the overall efficiency, lift and thrust. The $\bar{C}_t$ and $\bar{C}_l$ increase up to a maximum of 10% and 25% respectively when $x_{12} \leq 1.0$ and the effect diminished as x_{12} extends beyond 1.0. The additional vortex shedding from the downstream airfoil helps to increase the total thrust, which can be observed in the vorticity contour and velocity profiles plots. When x_{12} is fixed at 1.0 while α increases from -30° to 45° , $\bar{C}_l$ increased up to a maximum of around 2.6. The high $\bar{C}_l$ is due to the strong LEV at the top of the downstream airfoil. As α deviates from 0° , $\bar{C}_t$ and $\bar{C}_l$ decrease rapidly, giving drag.

Simulations are also conducted to test the effect of changing the downstream airfoil's cross section profile to a flat plate. The differences in the $\bar{C}_t$ and $\bar{C}_l$ are found to be small, with the NACA0012 airfoil giving slightly better performance. Lastly, a simple optimization study shows that $x_{12} = 0.25$ and $\alpha = 15^\circ$ give the highest performance index I_p .

¹ It must be emphasized that the input variables x_{12} and α are varied independently by fixing one of the variables and changing the other. Hence, the entire parameter space is not investigated.

based on the range of parameters studied. The results obtained from this study show that by reducing the x_{12} , better efficiency and thrust can be obtained. Moreover, the downstream airfoil or the tail, besides giving pitch and direction control, can help to dramatically increase the overall lift. Applying these results to small flapping MAVs like the Delfly will help to improve their performance.

ACKNOWLEDGMENT

The authors wish to thank STW for the financial support. The project number is 11023.

REFERENCES

- [1] K.M.E.D. Clercq, R.D. Kat, B. Remes, B.W.V. Oudheusden, and H. Bijl, Flow visualization and force measurements on a hovering flapping-wing MAV 'DelFly II', *39th AIAA Fluid Dynamics Conference*, Texas: 2009, pp. 1-10.
- [2] T. Pornsin-sirirak, Titanium-alloy MEMS wing technology for a micro aerial vehicle application. *Sensors and Actuators A: Physical*. vol. 89. Mar. 2001. pp. 95-103.
- [3] W. Schmidt, Der wellpropeller, ein neuer antrieb für wasser-, land- und luftfahrzeuge. *Z Flugwiss Weltraumforsch*. vol. 12. 1965. pp. 472-479.
- [4] H. Bosch, Interfering airfoils in two-dimensional unsteady incompressible flow, *AGARD CP-227, Paper No. 7*, 1978.
- [5] M.F. Platzer, K.S. Neace, and C.-K. Pang, Aerodynamic Analysis of Flapping Wing Propulsion, *AIAA 31 st Aerospace Sciences Meeting & Exhibit*, Reno, Nevada: 1993.
- [6] I.H. Tuncer and M.F. Platzer, Thrust generation due to airfoil flapping. *AIAA Journal*. vol. 34. Feb. 1996. pp. 324-331.
- [7] D. Rival, R. Manejev, and C. Tropea, Measurement of parallel blade-vortex interaction at low Reynolds numbers. *Experiments in Fluids*. vol. 49. Jan. 2010. pp. 89-99.
- [8] K.M.E.D. Clercq, R.D. Kat, B. Remes, B.W.V. Oudheusden, and H. Bijl, Aerodynamic Experiments on DelFly II : Unsteady Lift Enhancement. *International Journal of Micro Air Vehicles*. vol. 1. 2009. pp. 255-262.
- [9] R. Mittal and G. Iaccarino, Immersed Boundary Methods. *Annual Review of Fluid Mechanics*. vol. 37. Jan. 2005. pp. 239-261.
- [10] J.F. Ravoux, A. Nadim, and H. Haj-Hariri, An embedding method for bluff body flows: interactions of two side-by-side cylinder wakes. *Theoretical and Computational Fluid Dynamics*. vol. 16. 2003. pp. 433-466.
- [11] K.B. Lim and W.B. Tay, Numerical analysis of the s1020 airfoils in tandem under different flapping configurations. *Acta Mechanica Sinica*. vol. 26. Oct. 2009. pp. 191-207.
- [12] D. Kim and H. Choi, A second-order time-accurate finite volume method for unsteady incompressible flow on hybrid unstructured grids. *Journal of Computational Physics*. vol. 162. 2000. pp. 411-428.
- [13] L.L. Pauley, P. Moin, and W.C. Reynolds, The Structure of 2-Dimensional Separation. *Journal of Fluid Mechanics*. vol. 220. 1990. pp. 397-411.
- [14] W.B. Tay and K.B. Lim, Analysis of non-symmetrical flapping airfoils. *Acta Mechanica Sinica*. vol. 25. 2009. pp. 433-450.

Application of Electro-Active Materials to a Coaxial-Rotor NAV

C. Thipyopas¹, A.B. Sun², E. Bernard² and J.-M. Moschetta²

¹Department of Aerospace Engineering, Faculty of Engineering, Kasetsart University, Thailand

²Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), University of Toulouse, France

ABSTRACT

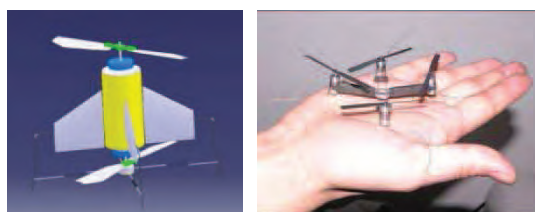
A new Nano Air Vehicle (NAV) configuration based on a coaxial nano rotor has been studied by ISAE. The coaxial rotor provides the thrust necessary for hovering and low speed translation flight. The major design challenge for rotary-wing NAVs is related to the difficulty of miniaturizing complex mechanisms such as a rotor cyclic pitch swashplate commonly used for controlling helicopters. The use of actuators made of smart materials is believed to allow for controlling rotary-wing NAVs in a much simpler and lighter way. The multidisciplinary subject of this complete system is separated into few parts. The studies of each subject are individually conducted before integrated in the future in order to optimize the global system. Those studies which are being carried out are shortly described in this paper.

1 INTRODUCTION

Micro Air Vehicles (MAVs) appeared in 1997 as a response to a DARPA program. Numerous institutions and universities have focused research in that topic. Over the past ten years, MAVs developed by ISAE have included RC-controlled vehicles such as the *MinusKiool* and the *Plaster* and autonomous vehicles such as the *LadyBug*, the *TYTO* [1], the *Vision'Air*, and the *MAVion* [2-3]. Pure rotorcraft concepts have also been studied [3]. Following the interest expressed by DARPA in 2006, ISAE has started the development of Nano Aerial Vehicles (NAV) [4] in 2007 under the supervision of Prof. Moschetta. With reference to DARPA, NAV has a limitation of size about a half of MAV. NAV is expected to have a dimension of 7.5 cm (3 in.), a total mass of about 10 grams, including a payload of 2 grams. The NAV should have a VTOL capability (Vertical Taking-off and Landing), the ability of sustaining autonomous hover flight as well as low speed forward flight. Due to very limited size and weight, designing and constructing NAV seems very difficult to achieve on the sole basis of available commercial components provided by RC-model companies. Military as well as civil applications are envisioned for NAVs, including recognition missions in complex and confined environment such as flying into a collapsed building after an earthquake or releasing a payload in a very narrow space [5]. As a consequence, there are many problems to be studied including very low Reynolds number aerodynamic difficulty, new miniature and light material, sensor and components.

For a design of tiny air vehicle, there are generally three concepts widely used that are, fixed-wing, rotary-wing and flapping-wing. Fixed-wing configuration is normally applied for outdoor and high speed flight requirement due to

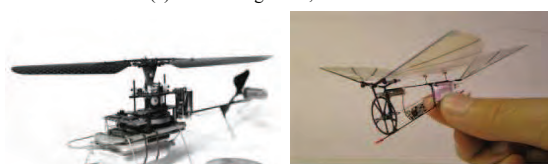
its horizontal flight efficiency. However, fixed-wing configurations are not suitable for NAV missions which include hover flight. Rotary-wing and flapping-wing concepts have been analyzed and compared by Liu [6], who concludes that the hovering efficiency of both concepts is comparable. Because flapping-wing mechanisms are more difficult to fabricate and control, a rotary-wing configuration has been selected, based on a Nano Coaxial Rotor (NACR). Coaxial Rotary wing concepts have been widely used because they intrinsically sustain hover flight without resorting to anti-torque rotors [7]. A compromise control method and vehicle configuration has to be adopted to achieve the objective of precise flight stability and satisfy the weight limitation of NACR.



(a) Coaxial Rotor NAV concept (b) Quad-rotor Indoor Flying RC



(c) Fixed Wing NAV, U. of Florida



(d) Prox dynamics's miniature flying machine

Figure 1: Present of Configuration used in Small UAVs.

The major design challenge for rotary-wing NAVs is related to the multidisciplinary design and the difficulty of miniaturizing complex mechanisms such as a rotor cyclic pitch swashplate commonly used for controlling helicopters. The use of actuators made of smart materials is believed to allow for controlling rotary-wing NAVs in a much simpler and lighter way.

Email address: fengcpt@ku.ac.th

This paper introduces and summarizes the present and coming future state of the NACR project. The concept and design overview will be discussed, followed by the study of each part of project. The test set up of smart actuator and test bench of propulsive characteristics will be presented. The fabrication method for the tiny structure or body of NACR will also be addressed.

2 DESIGN OVERVIEW

There are different solutions to control micro and nano aerial vehicle such as adding control surfaces just downstream the rotor or propeller as successfully demonstrated by MAVion [8], MiniVertiGo [9] and Vision'Air. Control alternatives include using additional propulsive or force generator system (ie, tail rotor) [10], center of gravity shifting [11], and tilting one or two rotors. Furthermore, an advanced concept using smart material may be considered such as a) smart active control intelligent rotor blade instead of changing cyclic pitch by swashplate, b) smart muscle structure. Applying surface control looks convenient for design but has drawback of additional drag and download force reducing flight autonomy. Using auxiliary force generator involves additional mass and may also increase the vehicle size. Based on present available technology, using smart material as an intelligent blade and a smart muscle structure is not feasible in the near future. Controlling the CG position has been implemented on micro air vehicle as reported in Ref. 11. However, this concept does not seem to be appropriate for the NAV size. Change CG position may be done by adjust the position of battery, but moment created is very small according to the small mass of battery represented in an estimated weight balance distribution (see Figure 5). After comparison, the control method of tilt rotor is selected in this NACR project, relying on the fast development of intelligent and light actuators over the past few years.

First Concept and Bench Model

The estimated maximum required thrust, normal force and tilt angle of upper rotor to reach 2m/s translation movement are 0.15N, 0.03N, and 10degrees, respectively. Based on these estimations, several light-weight actuators are commercially available. Table 1 summarizes some characteristics of miniature servos and actuators commercially available.

Commercial Actuator	Blue Arrow	Linear Servo	Smart Servo	Magnetic Actuator1	Magnetic Actuator2
					
Mass (g)	2.8	2	1	0.4	0.75
Size (mm)	19x20x8	21x13x9	38x9x3	7x8x4	9x10x8
Torque (g.cm)	150	160g force	15	0.4	0.8
Max. Motion	80deg	14mm	30deg	4mm	5.5mm
Time (sec)	0.22 (60deg)	0.15 (full)	-	-	-
Current (mA)	100	100	80	45	42

Table 1: Miniature servos commercially available.

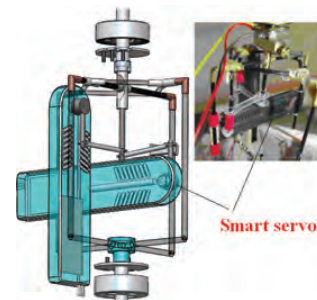


Figure 2: 1st NACR Bench Model.

Magnetic actuator is one choice because of its lighter weight of 0.4g. Considering its smaller torque, another bio-wire smart servo actuator, mass of 1g, was used in the first NACR bench model (see Fig.2), two smart servos are used to tilt the upper rotor for pitch and roll control. The structure is made of 1mm-carbon rods connected by a metal tube connector and just glued together. The total mass of that bench model is around 20 grams. A pair of rotors designed by Liu [12] has been installed. Measurement of total thrust performance is performed by new low friction magnetic test bench "Pop-Eye" shown in Figure 3. Yet, only thrust is measured, side force and pitching moment which are the most interesting parameters for the tilt-concept could be measured with that test bench.

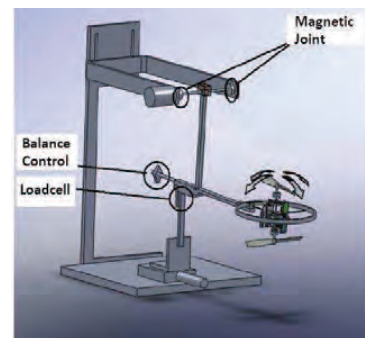


Figure 3: Pop-Eye Test Bench.

Current NACR Concept

Considering fast and precise stability requirement, new actuators with shorter response time, lighter weight and lower power consumption are required to replace such a complex configuration based on 2 SmartServos. EAPs (Electro-Active Polymers) with fast reaction speed (up to μ sec) on one side and low density with great resilience on the other side, compare favorably with SMA (Shape Memory Alloys) and EAC (Electro-Active Ceramics) [13] respectively. Among them IPMC (Ionic Polymer-Metal Composites) requiring only low drive voltages may be seen compatible with extremely uncluttered NAV power supply (reduced to one cell of 3.7V Lithium-Battery for the NACR). In addition, IPMC combine advantages towards miniaturization, light weight with large displacements and endurance. As an emerging technology, IPMC concentrate intensive research effort and still promises versatility in mechanical properties through the control of a wide range of

parameters such as ions and solvent nature, membrane thickness etc... IPMC have already been designed for actuators in flapping devices [14].

The relative limitations of IPMC low energy density and actuation force may be overcome from the overall mechanism design. To minimize the complexity of 2-DoF tilt rotor, only 1-DoF tilt rotor is first investigated in the present work. A pitch motion can be obtained by tilting the upper rotor while yaw motion can be achieved by controlling the rotation speeds of the two counter-rotating rotors. Primary configurations to control tilt-angle of upper rotor of NACR with IPMC actuators are illustrated in Fig.4. In Fig. 4-left, IPMC actuator belongs to the NACR mechanical structure, acting as a muscle, the upper rotor is directly supported by two parallel IPMCs. Axial, normal and torsion forces act on the IPMCs in that configuration. Another configuration is considered in Fig. 4-right, where the IPMC is only used to control the tilt-angle. In a further perspective, IPMC may also be considered in the building of smart/intelligent rotor blades as well. The expected total mass of NACR including 2g-payload is about 17g. It is noticeable that mass of propulsion system (battery, 2-motors, 2-rotors, and 2-speed controls) and mass of electronics parts are about one-third of total weight (6.5 and 6g, respectively).

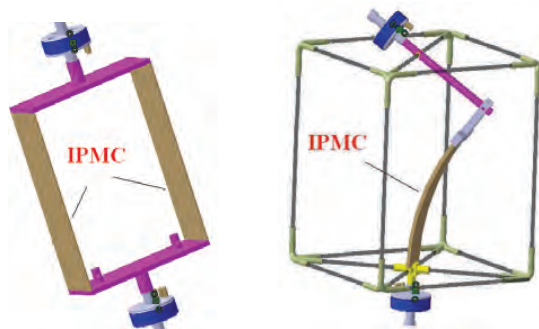


Figure 4: NACR Concept using Smart Actuator.
Left: Smart Muscle concept which upper rotor is support by IPMC,
Right: Tilt-rotor configuration controlled by smart IPMC actuator

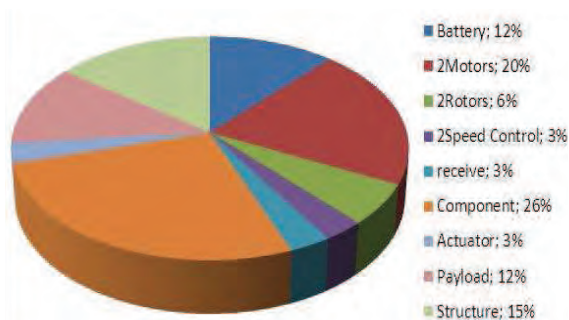


Figure 5: Expected mass distribution of NACR (total mass = 17 g)

3 PROPELLER DESIGN AND FABRICATION

Design of nano coaxial rotor is based on the optimization of hovering. Effect and interaction of each rotor to another rotor are taken into account in the optimized design preliminary calculated by XROTOR code of MIT. Induced velocity of the 1st rotor is inserted into the calculation of the

2nd rotor till the calculation converges [12]. Navier-Stokes numerical results are then compared using several turbulence models. Nano coaxial rotors are shown in Fig. 6.



Figure 6: Nano Coaxial Rotor designed by Liu



(a)



(b)

Figure 7: Fabrication Process of Nano Coaxial Rotor.

The rotor is then made. Its small dimension makes them difficult to fabricate. Accuracy in fabrication is critical and can affect the rotor performance. The blade thickness must be minimized to obtain the best performances while a thin section is convenient for the fabrication by composite material.

The 2-side molds are fabricated by a 3D printer. The first try, rotors are made using wet layup; 100g/m² carbon tissue (the thinness carbon tissue we have) and epoxy. The form of rotor is first cut by hand before layout on the molds. However the extremity is run out as showed in Figure 7(a). Then it was compressed by another mold. After drying for 15 hours, the edge of propeller has not good shape as desired. In an attempt to improve the shortcomings of carbon prototype, 42g/m² fiber glass is used. The fabrication processes are detailed in Figure 7(b). This time, tissue of fiber glass is layout on the molds without cutting the shape. Several layers of tissue are tried to get difference flexibility. The last layer is covered by peel piles in order to obtain smooth surface and also to absorb the exceed epoxy. After 15 hours, the shape of rotor is trimmed by cutter and sanded before being released from the molds.

4 AERODYNAMIC AND PROPULSION

To find precise flight dynamic modeling, thrust and torque of each rotor should clearly identified. Because of its very miniature size, characterizations of their aerodynamic and propulsive performance are challenging. The numerical simulation of very low blade-Reynolds number aerodynamics is strenuous. Experimental measurement of such a low force-moment is also very difficult. In the present project, the objective is to find an influence and interaction of each rotor on another rotor especially when the upper rotor is tilted. A normal force on the lower rotor is expected. Both numerical simulation and experimental test are performed and will be compared in the future.

Numerical Simulation

In this study, the overset structured meshes are used. Different from the meshes for MRF model, five isolated blocks in which four blocks conform four blades and a cylindrical block for background mesh are generated. Several rotor pitching cases are studied by Liu [12] such illustrated in Figure 8.

Test Bench for NACR Concept

The challenge of experiment of nano rotor is to measure precisely small thrust, torque and other parameters. As the expected maximum thrust of each rotor is order of 10 grams, sensor should have accuracy better than 0.5 grams force to achieve an error of less than 5%. Many single-rotor test benches had been developed and tested at ISAE since 2008 [18]. The study is satisfied but all of them are only two components (thrust and torque). ISAE developed new high precision 5-component nano sting balance in 2009. Thrust and torque of coaxial nano rotor were observed by this high precision nano sting balance but they are just measured together [12]. Now NACR is more and more challenging since the objective is to measure very petite influence of tilt-angle of the rotor in particular on the lower rotor. Rotor must be separately identified.

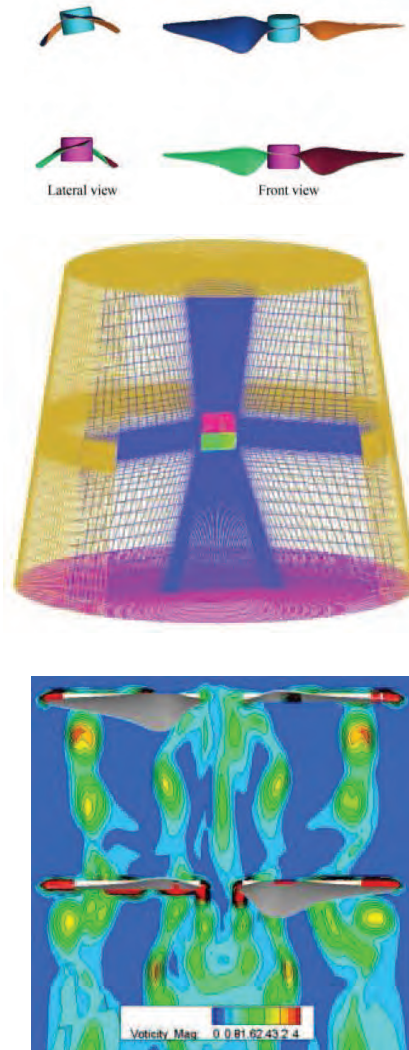


Figure 8: Numerical Simulation of Coaxial Nano Rotor done by Liu.

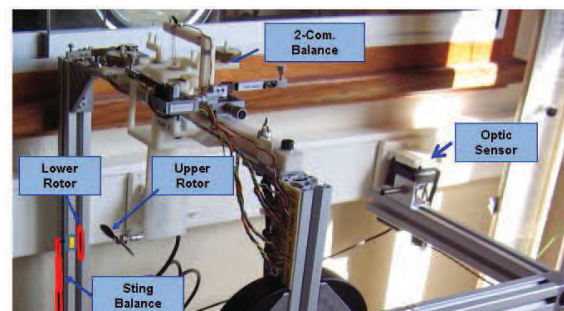


Figure 8: Experimental Set up of Tilt-Rotor in NACR project.

To successfully investigate the rotors, all forces and moments should be observed. Therefore, they have to be identified by the 5-component balance. Yet, only one 5-component balance is available. New solution should be found. Following the fact that blowing induced flow of the upper rotor is strongly faster than suction induced flow of the lower rotor, the presumption is applied. The upper rotor

gives only thrust aligned to the motor axis while the lower rotor may produce other lateral forces / moments. Thus, a new experimental test set up is installed. The upper rotor is mounted by the nano bench and the lower rotor is carefully supported by 5-component nano sting balance as illustrated in Figure 8. Both balances are installed together. The mechanical set up allows for adjustable distance and tilt-angle. So normal forces, side force, pitch and yaw moment of lower rotor can be measured beside thrust/torque of both rotors. All electric parameters and rotor speeds will be measured as well.

5 STRUCTURE

Main function of structure for *NACR* is holding and connecting motors and all other components. Nevertheless, since the components selected for *NACR* must be very light, they are usually fragile. Then, it is great to well protect them. Therefore, the structure of *NACR* is designed to hold and cover all other components. The light and stiff material must be carefully selected. On shelf carbon rod is often used for the structure of MAVs such as seen by *Vision'Air* and *Link MAV* in Figure 9. This material has already proved its stiffness.

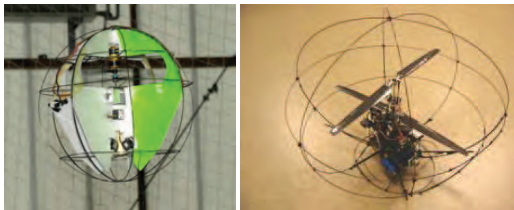


Figure 9: MAV.
Left: 1st version of *Vision'Air* of ISAE (2007)
Right: *Link MAV* of Linköping University (IMAV07, Toulouse)

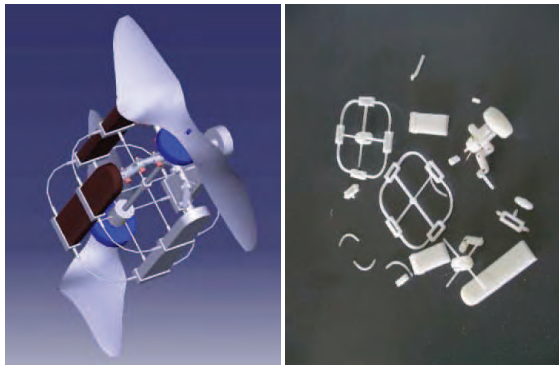


Figure 10: First Attempt of Structure Design.

However, it is not easy to find the right and optimized dimension of carbon rod because the diameter of vehicle is very small (around 8 cm). Off-the-shelves 0.5mm carbon rods cannot be shaped to a circle form, while that of 0.3mm is too small and soft. In addition, it is not easy to make the connection between each part by on-shelf carbon rod.

Due to the structure of *NACR* which has many connections, fabrication by rapid prototype machine has been carried out. The structure was designed by CATIA (Figure 10-left), and sent to a CNC machine. The outcome is presented in Figure 10-right. However, the structure is very

fragile since the machine follows Cartesian path lines.



Figure 11: Uni-Direction Carbon In-House NAV Structure Solution.

Finally, carbon material has been re-considered. Uni-direction carbon and epoxy is used for a wet layup. In the first attempt, the wet-carbon is formed by a simple cylindrical tube. As illustrated in Figure 11, a spiral structure is tried with the expectation of flexibility between two rotors. The weight of this first attempt is about 2.5g.

6 SMART ACTUATORS TEST

Experimental test is conducted to characterize the performance and efficiency of new smart actuators. As a start, simple and basic tests were performed as shown in Fig. 12(a). The accurate tip displacement under several external constant forces was measured by a high resolution laser device with the accuracy of 1cm per volt. Mechanical property of this IPMC can be analyzed. The experimental setup with micro balance measurement system [19] according to the reference is then proposed to be used for force testing of IPMC samples. Proper specifications for IPMC design according to NAV requirements are extracted. The IPMC tested in this study has a size of 40x10x0.3mm and a mass of 0.4g.

As the application of IPMC when installed to the prototype is not simple in the first test, the following work will be to test IPMCs under the generic environment test bench. The assumption is made that when the tilt-angle is increased, flight speed and normal force on rotor increase. Therefore, the constant external load measurement in the previous set up is not exactly true. New set up is represented in Figure 12(b). Known-property miniature beam is utilized to simulate the external force which is a function of tilt-angle and length of beam.

The numerical models of different structures are being set up in view to establish proper control loops to achieve the objective of precision control of IPMC actuator, which is a critical process for NAVs flight stability.

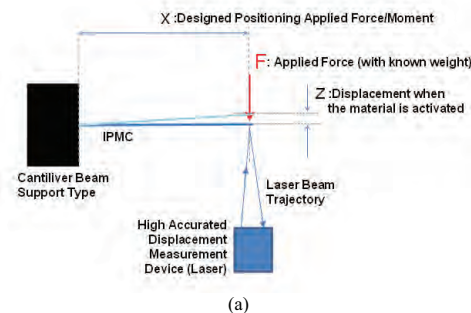


Figure 12: IPMC Characterization Test Bench.

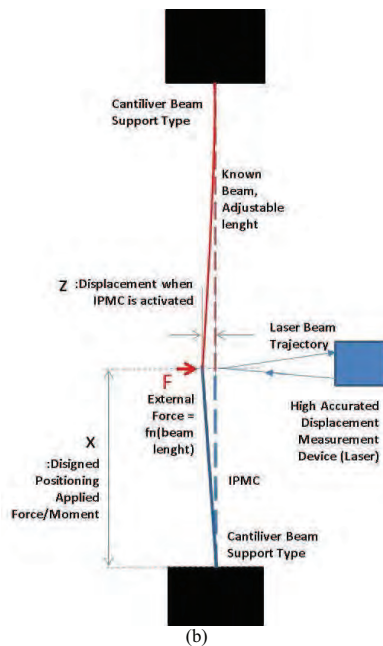


Figure 12: IPMC Characterization Test Bench (cont.)

7 ELECTRONIC COMPONENTS

This is another very important topic of *NACR*. Normally, electronic of flying rotary-wing RC consists of a receiver, speed controller, gyroscope, servos or actuators and the lightest mass of each component that can be found commercially are 0.3, 0.2, 2.3, 0.4g, respectively. *NACR* requires a pair of rotors and a minimum of one actuator so the total mass of electronic components is equal to 5.7g. Moreover, to achieve autonomous, controller and other sensor are required on board. Therefore it is not possible to use on-shelf components as electronics part of *NACR*. Small and miniature air vehicle needs special component multidisciplinary optimized design such as shape and structure to be as light as possible. All electronic components, such as speed control, sensors, actuator, and etc., may be integrated on board. PCB board of electronic can be a part of structure as well. The design of electronic components is being cooperated with other department of ISAE. The objective is to reach 5grams of electronic board including sensors.

8 CONCLUSION

A new concept of coaxial rotor Nano Air Vehicle (NAV) has been presented in this paper. The upper rotor can be tilted in order to provide roll and pitch control while differential throttle provides control in yaw. Among several electro-active actuators, IPMC has been selected because of the low driving voltage and the reasonable amplitude provided. Before designing a flying prototype, IPMC actuators need to be tested and compared with existing actuators. The propeller interaction will be studied using a new test bench devoted to extremely low thrust force as produced by a coaxial nano rotor. In the near future, a tilt-rotor concept will be evaluated by scaling up the platform.

The prototype by a factor of 2, called Micro Coaxial Counter-rotating Rotors (*MICCOR*), will be made at Kasetsart University. A radio-controlled prototype will be equipped with three gyroscopes.

ACKNOWLEDGMENT

Authors would like to thank for the financial funding of STAE (French Scientific and Technology for Aeronautic and Space Foundation) and EMMAV project. Without help and support of several colleges from different department at ISAE, the project could not be conducted. The first authors would like to thank also to Kasetsart University for the partial support of his visiting at ISAE during summer period.

REFERENCES

- [1] Thipyopas, C., and Moschetta, J.M., "A Fixed-Wing Biplane MAV for Low Speed Missions," International Journal of Micro Air Vehicles, Jan 2009, Vol. 1, no. 1.
- [2] Carr, R. et al., "A Tilt-Body Fixed-Wing Micro Air Vehicle for Autonomous Transition Flight," Proceedings of IMAV2010, Braunschweig, Germany.
- [3] Thipyopas, C. et al., "Aerodynamic Analysis of a Multi-Mission Short Shrouded Coaxial UAV: Part II – Transition Flight," 48th AIAA Aerospace Sciences Meeting, Jan 2010, Orlando, Florida.
- [4] Pines, D., "06-06 Proposer Information Pamphlet (PI) for Defense Advanced Research Project Agency (DARPA) Defense Sciences Office (DSO) Nano Air Vehicle (NAV) Program," Technical Report, DARPA DSO, 2005.
- [5] S., Zak, "Survey of UAV applications in Civil Marks," Proceedings of the 9th Mediterranean Conference on Control and Automation, Dubrovnik, Croatia, 2001.
- [6] Zhen, L. and Moschetta, J.M., "Rotary vs. Flapping –Wing Nano Air Vehicles: Comparing Hovering Power," Proceedings of IMAV2009, Delft, Netherland.
- [7] Lim, H. and Machida, S., "Mechanism and Control of Coaxial Double Contra-Rotation Flying
- [8] Thipyopas, C. et al., "Fixed-Wing VTOL MAV Design," Proceedings of IMAV2009, Pensacola, Florida.
- [9] Shkarayev, S. et al., Aerodynamic Design of Micro Air Vehicles for Hovering Flight," Journal of Aircraft, 2008, Vol. 45, No. 5, p. 1715-1724.
- [10] Bouabdallah, S. et al., "Design and Control of an Indoor Coaxial Helicopter," 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, China, Oct 2006.
- [11] Bernes, C. et al., "New Design of the Steering Mechanism for a Mini Coaxial Helicopter," 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, France, sept 2008.
- [12] Zhen, L. et al., "Evaluation of Nano Coaxial Rotors in Hover with a Validated Five-Component Balance," Journal of Aircraft, 2011, Vol. 48, No. 1, p. 220-229.
- [13] B.C., Yoseph, "Electroactive Polymer (EAP) Actuators as Artificial Muscle: Reality, Potential, and Challenges," SPIE Press, 2004, Chapter 1, p. 3-52.
- [14] S.G., Lee, et al., "Performance improvement of IPMC for a Flapping Actuator," International Journal of Control, Automation and Systems, 2006, Vol. 4, No. 6, p. 748-755.
- [15] B.L. Stevens and F.L. Lewis, Aircraft control and simulation, Jon Wiley & Sons, INC, 1992.
- [16] B. Mettler, Identification Modeling and Characteristics of Miniature Rotorcraft, Kluwer Academic Publisher, 2003.
- [17] Xu, M. and Zhen, L., "An Experimental and Theoretical Evaluation of Micro Rotor MCF3225 in Static Conditions," 2010 2nd International Conference on Mechanical and Electronics Engineering (ICMEE 2010), V2-37-41.
- [18] Zhen, L. et al., "Design of Test Benches for the Hovering Performance of Nano-Rotors," International Journal of Micro Air Vehicles, 2010, Vol. 2, No. 1, p. 17-32.
- [19] K. Jeahwan et al., "Blocked Force Measurement of Electro-Active Paper Actuator by Micro-Balance," Sensor and Actuators A 133, 2007, p. 401-406.

Numerical investigation of disc-wing MAV with propeller in a wing slot

N. D. Ageev,

Department of Aeromechanics and Flight Engineering of
Moscow Institute of Physics and Technology, Gagarin 16, Zhukovsky

ABSTRACT

Concept of disc-wing MAV with propeller in a wing slot has a number of advantages and disadvantages. The concept is used by MIPTEAM for solving the problem of fixed wing MAV indoor flight by means of lift maximization for wing with low aspect ratio. There are only few investigations of disc-wing MAV aerodynamics with propeller in a wing slot. Present work describes numerical investigation technique and aerodynamic characteristics obtained. The main results are: good stall characteristics of this concept, low lift/drag ratio (further improvement is possible) and sophisticated flow pattern. It was obtained that the main part of the lift is created by wing surface in front of the actuator disk. Also, leading edge separation area produces small impact at the lift even at high angles of attack. It was found that increasing of the propeller thrust decreases maximum lift/drag ratio. Flow features in numerical results correspond to the known experimental results and flight tests.

1 INTRODUCTION

1.1 Background

A disk wing provides maximum surface and, consequently, takeoff weight (at equal other conditions) at the fixed lateral dimension. So, it is of the great interest for the usage in MAV. The experimental aircraft with fixed disk wing was presented by MIPTEAM at the Indoor Dynamics Competition during the IMAV-2010 [1] (Figure 1). Its maneuverability is provided by three factors: usage of the low aspect ratio wing, low wing load and propeller-wing interaction that needs further investigation [2].

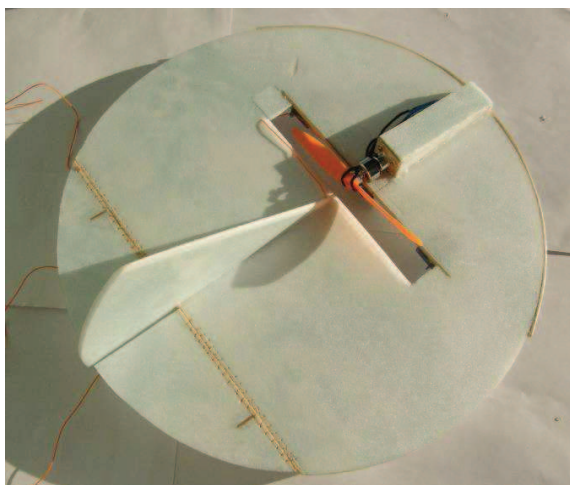


Figure 1: Disk-wing aircraft presented by MIPTEAM

Email address: n.d.ageev@gmail.com

Engineering methods along with flight experiment are usually used for calculation and improvement of the aircraft performance.

Further aerodynamic improvement of the apparatus requires more information about the flow pattern around the aircraft. Flow pattern can be obtained from experimental data (for example, by means of PIV) or from numerical modeling [3], [4]. Practically there are no experimental or numerical investigations of the disk-wing aircraft with the air screw situated at the slot in the wing. Therefore, numerical investigation technique is of great practical importance.

1.2 Notations

The following system of aerodynamics coefficients and approximations is accepted in this article:

L – Lift force
 D – Drag force
 F – Propeller thrust
 M – Mass of the aircraft
 V – Flight speed
 P – Static pressure
 ρ – Air density
 q – Initial dynamic pressure ($q = 0.5\rho V^2 = 14.8125 \text{ Pa}$)
 α – Angle of attack (AoA)
 S – Characteristic area ($S = 0.25\pi \text{diameter}^2 = 0.012 \text{ m}^2$)
 C_L – Lift Coefficient; $L = C_L q S$
 C_D – Drag Coefficient; $D = C_D q S$
 C_{D0} – Drag coefficient at zero lift
 C_p – Pressure coefficient; $C_p = (P - P_\infty)/q$
 $C_L(C_D)$ is approximated by $C_D = C_{D0} + A_1 C_L + A_2 C_L^2$

2 PROBLEM FORMULATION

2.1 General formulation

The problem is to calculate aerodynamics coefficients and flow pattern around the aircraft, represented by MIPTEAM [1], at the velocity equal to an average flight speed of the apparatus at the IMAV-2010. The characteristic time of every IMAV lap is 8 seconds. Length of the lap is about 40 m, average flight speed $V = 5 \text{ m/s}$. Reynolds number calculated at the chord of the wing (equal to 0.3 m) is 103 300. An estimate of the air screw thrust is also required. For the apparatus with mass $M = 70 \text{ g}$ with a C_L/C_D ratio 3.5 the required thrust is 0.2 N. The flight throttle is known to be about 40% of maximum thrust ($F_{\max} = 0.5 \text{ N}$). These two estimates agree well and give a thrust $F = 0.2 \text{ N}$. After preliminary calculations it was found that the required thrust in cruise mode was higher than estimated, second stage of calculations was carried out with thrust $F = 0.324 \text{ N}$.

2.2 Geometric model and mesh

The geometry of the apparatus was simplified (final geometry is shown in Figure 2). Control surfaces are fixed. The propeller is simulated by an actuator disk according to [4]. Thus, swirling of the propeller stream wasn't considered in the work.

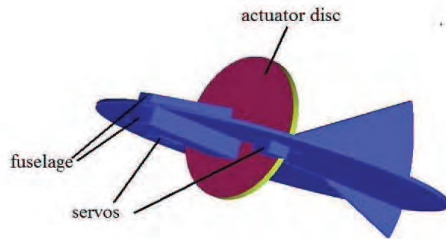


Figure 2: Geometric model

The thickness of the actuator disk is equal to the maximum thickness of the propeller. Computational domain around the unit has dimensions $20 \times 20 \times 20$ diameters of the wing. The model unit is located in the center of the computational domain. The selected size of the computational domain provides correct simulation of the vortex system of the wing at an acceptable computing cost.

Computational domain contains unstructured tetrahedral mesh with prismatic layers (Figure 3). This type of the mesh provides satisfactory quality of the flow separation simulation. The size of the cells on the model surface doesn't exceed 2 mm. General maximum cell size is equal to 128 mm. The cell size growth ratio is 1.08, which provides sufficiently detailed simulation. The total grid has 6 677 448 elements. Volume mesh is generated from surface mesh by improved Delaunay algorithm. There are 91 220 elements on the surface of the aircraft (without disk) and 14 473 elements at the surfaces of the actuator (which is much more than used in [4]). Prism layers are created on the surface of the model. Total thickness of the prism layer is 2.58 mm, which provides careful simulation of the boundary layer at given Reynolds number (103 300). Prism layer provides $Y_{+max}=3.56$.

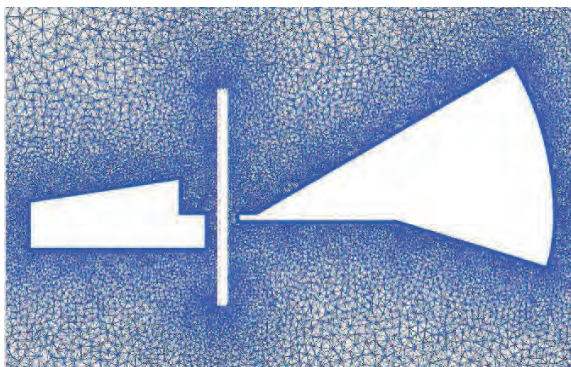


Figure 3: Mesh around the model

Test calculations showed good agreement between the results from this mesh and mesh with 14 millions elements.

2.3 Boundary conditions

Boundary conditions on the walls of the computational

domain are standard: front and bottom walls of the computational domain - inlet, rear and top - outlet, side - free slip wall. The air screw is modeled by an outlet at the front surface of the actuator, inlet at the rear surface. Mass flow rate is set through the actuator corresponding to the speed at the propeller, calculated by the theory of the active disk [5]. The side wall of the actuator is a free slip one. As shown by test calculations, this method allows the simulation of the propeller to be accurate enough to simulate thrust with preserving the continuity of the velocity through the actuator. Stream swirling is neglected in this approximation.

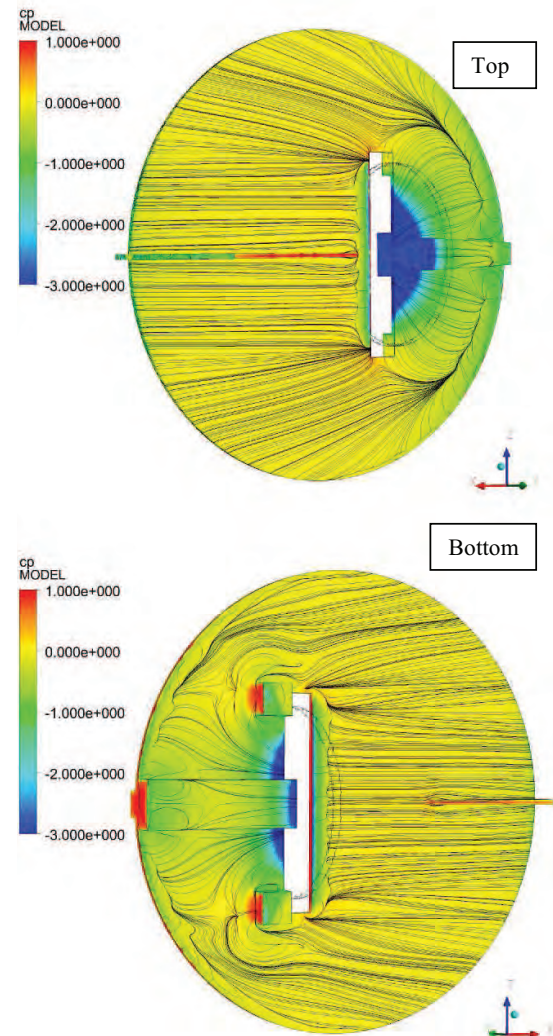


Figure 4: C_p distribution and surface shear stress streamlines at zero angle of attack

3 SOLUTION, VERIFICATION AND VALIDATION

3.1 Solution

RANS system of governing equations with the SST turbulence model was solved by the solver ANSYS CFX™.

Solving is carried out with an incompressible viscous fluid model with parameters corresponding to air at 25°C.

The problem is solved in a stationary regime with the approximation scheme of the second-order. One calculation takes 750 iterations in average. Stop condition is fixation of the third significant digit in magnitude of the aerodynamic forces acting at the model.

Preparation for calculations took near two weeks, one point extraction took 7 hours of calculation time at i7-960 (3.0 GHz, 4 physical cores) processor.

3.2 Verification

For verification shear stress lines are built on the model surface at zero angle of attack (Figure 4).

General flow pattern (at all investigated AoA) is in good accordance with experimental data [6]. Flow separation regions near sharp edges are simulated. Performance of the actuator stream corresponds to the active disk theory [5] qualitatively: the jet narrows both in front and besides of the actuator. Vortex system of the wing is present at any non-zero angle of attack and can be visualized by pressure isosurface with low static pressure (Figure 5). Distance of the vortex system trace modeling is large enough (about 2 m).

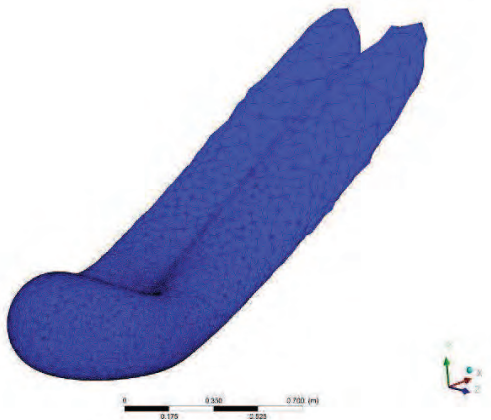


Figure 5: Isosurface of low pressure around the model at 40 degrees angle of attack

Boundary layer is simulated at all the surface of the wing. Low velocity areas besides the model are observed. Thus, the data agree well with the theory predictions and experiment [6], and can be subjected to further analysis.

3.3 Validation

Results of the computations show the good accordance between average pressure drop (that is about 25 Pa) on the actuator and the propeller thrust $F=0.324\text{N}$. Magnitude of the calculated lift of the vehicle let it fly stable at the angle of attack of 15 degrees and even perform aerobatics, as it was seen at the IMAV-2010. $A_2=0.4225$ calculated for the angles of attack 0-20 degrees is also in a good accordance with the experimental data obtained in [7] ($A_2=0.45$ for this wing).

4 RESULTS

$C_L(C_D)$ function is shown in Figure 6. $C_{D0}=0.2$ is greater than estimated for this type of MAV ($C_{D0}=0.03...0.1$). The

main part of the C_{D0} is pressure drag (about 85%). Large pressure drag can be explained by influence of the fuselage, servos, blunt edges with sharp corners and slot around the propeller.

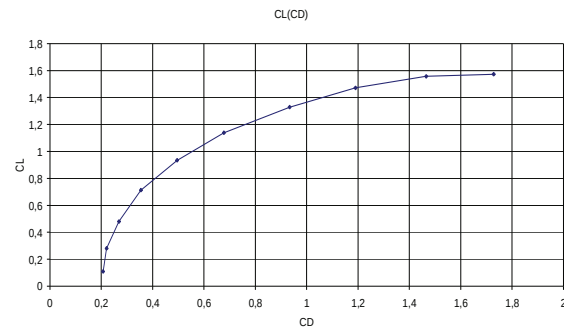


Figure 6: C_L as function of C_D

As one can see, the greatest values of the pressure are at leading edges of the wing and fuselage and at the leading slot edge.

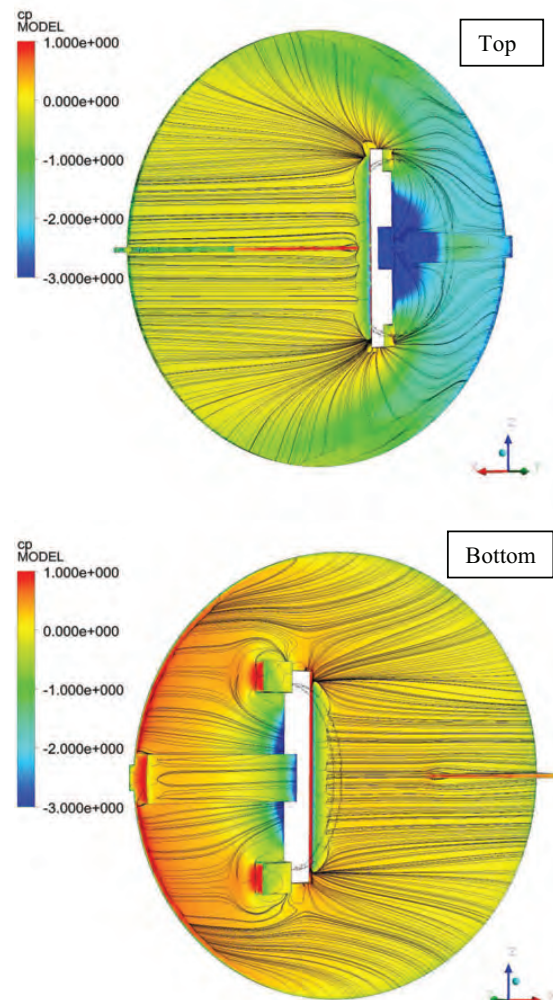


Figure 7: C_p distribution and surface shear stress streamlines at angle of attack of 15 degrees

Pressure distribution along with shear stress streamlines at cruise angle of attack (15 degrees) is shown in Figure 7. The static pressure decrease at the rear fuselage and servo surfaces is also caused by propeller work.

Thus, great amount of the drag at the wing is induced by propeller.

The air screw also loses some thrust due to low-speed zones besides the fuselage and the wing.

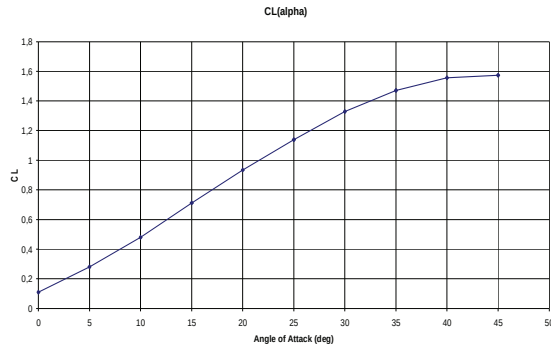


Figure 8: C_L as function of Angle of Attack

One of the most important results obtained is that the C_L (C_D) function coefficient A_2 calculated for the angles of attack 0-20 degrees is 0.4225 that is in a good agreement with the experimental data (0.45) [7]. But at the high angles of attack the A_2 becomes greater. It is caused by the intensification of the vortices and increase of the vortex lift role that can be proved by comparison of the flow patterns at different angles of attack and by investigation of the C_L as function from angle of attack. As it was expected, C_L () dependence has increasing derivative at angles of attack near 5-15 degrees (Figure 8). Non-zero lift at zero AoA is associated with an asymmetric air screw position relative to the longitudinal axis of the aircraft (air screw center is higher than the longitudinal axis; C_p distribution at zero AoA is shown in Figure 4).

Stall comes at the angle of attack near 45 degrees that is good, but perhaps can be better. Separation zones at this AoA are the largest ones. Thus, flow pattern at this angle of attack is the most interesting.

Flow structure above the wing at the angle of attack 40 degrees is visualized by the pressure distribution and shear stress surface streamlines in Figure 9. The flow has three main features: a strong vortex system of the wing (can be observed in Figure 5), low lift at the wing surface besides the air screw and weak effect of the leading edge separation (at C_p distribution and consequently C_L).

Strong vortex system of the wing is concerned with the low aspect ratio and the propeller-wing interaction. It increases AoA of stall, but decreases the C_L/C_D ratio.

Small lift at the wing area situated besides the air screw is associated with the rectifying of the flow besides the propeller and has three consequences. At first, total lift decreases. The other consequence is that aerodynamic focus shifts closer to the leading edge that makes the aircraft less stable. And the last consequence is that center of aerodynamic force application shifts to the leading edge, that reduces balance losses. So, this problem requires further investigation.

Little effect of the leading edge primary separation is concerned with the influence of the vortex system that creates a reverse flow zone in front of the wing and provokes a secondary separation besides of the primary one. Thus, efficiency of the deflected leading edge requires further investigation.

C_L/C_D ratio as function of C_L is presented in Figure 10. One can see that flight at the IMAV-2010 was carried out at the regime close to the maximum C_L/C_D mode. Low magnitude of the maximum C_L/C_D ratio caused by two reasons: high C_{D0} and high A_2 of the vehicle. The first reason is concerned with the drag of the blunt fuselage around the airscrew that can be reduced by slight geometry modification, the second one – with low aspect ratio that can't be sufficiently changed by geometry modification (in the concept of maximum lift at the fixed lateral dimension).

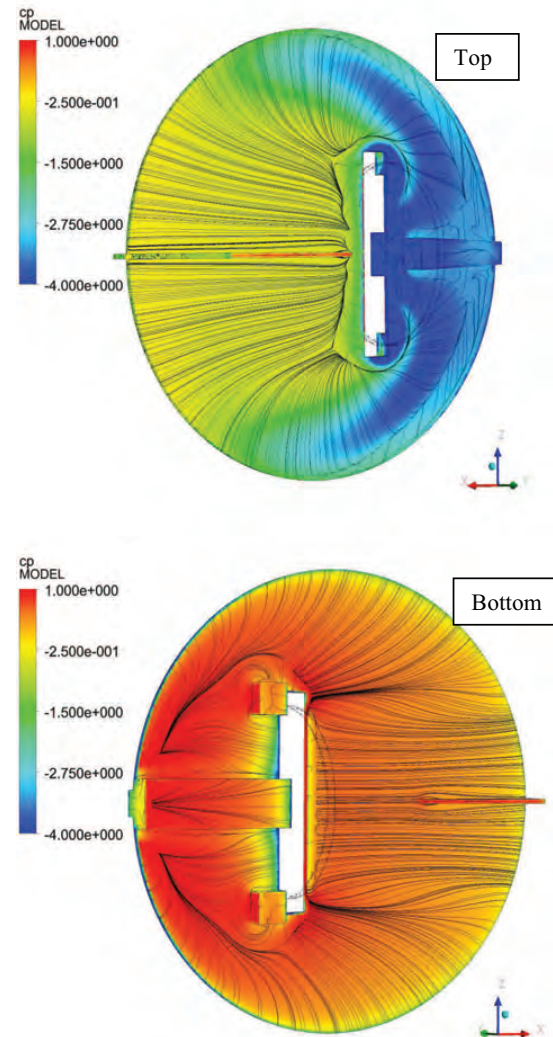


Figure 9: C_p distribution and surface shear stress streamlines at angle of attack of 40 degrees

If $F=0.2N$ maximum C_L/C_D ratio is equal to 2.4 and is reached at lower AoA, $C_{D0}=0.12$ that confirms existing of the relationships between the C_{D0} , A_2 and the propeller thrust.

5 CONCLUSION

Good agreement between the results obtained and experimental/flight test data is reached. Some interesting flow features around the vehicle are extracted: vortices, flow separation areas and small lift area besides of the actuator disk. The last feature requires further complex (dynamic and aerodynamic) investigation.

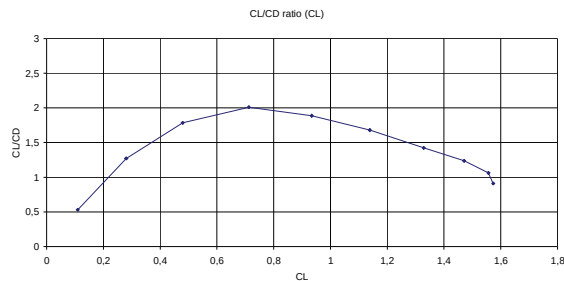


Figure 10: C_L/C_D ratio as function of C_L

Leading edge separation produces small impact at the lift of the vehicle. Increasing the propeller thrust reduces maximum C_L/C_D ratio. It is obtained that the lift can provide maneuverability of the apparatus. Main aerodynamic relationships were obtained and can be used in further aerodynamic research and development. It was found that further work on reducing the C_{D0} is appropriate. However, in the model used for calculations propeller stream was simplified by means of jet without swirling.

ACKNOWLEDGMENT

The present work is carried out under support of Russian "UMNIK" program. The author is grateful to prof. S. V. Serokhvostov, prof. I. V. Voronich, prof. A. A. Pavlenko and E. V. Birukova for the preliminary remarks.

REFERENCES

- [1] MAV Book. *IMAV-2010 proceedings*. Braunschweig, 2010
- [2] N. D. Ageev. Design of a highly maneuverable MAV able for controlled indoor flight. *Proceedings of the 53rd conference MIPT "Modern problems of fundamental and applied sciences" Part 6. Aeromechanics and Flight Engineering*. - Moscow: MIPT, 2010. p. 97-98. (in Russian)
- [3] Mark Groen, Bart Bruggeman, Bart Remes, Rick Ruijsink, Bas van Oudheusden, Hester Bijl. Improving Flight Performance of the Flapping Wing MAV DelFly II. *IMAV-2010 proceedings*. Braunschweig, 2010
- [4] Sungjin Choi and Jon Ahn. A Computational Study on the Aerodynamic Influence of a Pusher Propeller on a MAV. *AIAA 2010-4741. 40th Fluid Dynamics Conference and Exhibit 28 June - 1 July 2010, Chicago, Illinois*
- [5] Glauert, H. The elements of aerofoil and airscrew theory-2nd ed. - (Cambridge Science Classics Series), 1983, Cambridge University Press
- [6] J. R. Potts, W. J. Crowther. Visualization of the flow over a disc wing. *9th International Symposium on Flow Visualization, Heriot-Watt University, Edinburgh, 2000*
- [7] A. V. Kornushenko, S. V. Serokhvostov. Experimental studies of the aerodynamics of miniature aircraft. *"Air Fleet Tech" № 2-2008* p. 6-9. Moscow: TsAGI, 2008 (in Russian)

An engineering development of a novel hexrotor vehicle for 3D applications

D. Langkamp, G. Roberts, A. Scillitoe, I. Lunnon, A. Llopis-Pascual, J. Zamecnik, S. Proctor, M. Rodriguez-Frias, M. Turner, A. Lanzon and W. Crowther
University of Manchester, George Begg Building, Sackville Street, M13 9PL, UK

Abstract

This paper is about the development of a novel type of non coplanar hex rotor with the ability to decouple translational and attitude dynamics. The proposed design features six variable pitch rotors arranged in three different rotor planes in order to point the thrust and torque vectors independently of body attitude. It is envisaged that this design could be beneficial for operations in confined spaces where the requirements for translational control authority design outweigh the reduced hover efficiency compared to coplanar multi rotors, such as quad rotors.

The rotor arrangement leads to design and modeling challenges which are very different from those of conventional planar multi rotor vehicles. The key engineering challenge lies in the requirement to generate sufficient thrust for weight support with two rotors alone. This paper shows how this challenge was overcome by the use of high thrust/weight electric variable pitch units and a low airframe mass fraction. The design of avionics and indoor positioning solutions is discussed and the control strategies are outlined. The development and wind tunnel validation of a simulation model is discussed and operational modes are presented which satisfy the key constraints, linearise the thrust response and optimise hover efficiency.

The feasibility of the concept was demonstrated through the flight testing of fixed-pitch design in 2009 and a flight demonstration of the variable pitch design is planned for IMAV 2011.

1 INTRODUCTION

This paper presents the engineering design of a novel non planar hexrotor vehicle with the capability to decouple translational and attitude dynamics [6]. This is achieved by arranging the six rotors on three separate planes such that the thrust and torque vectors can be pointed independently.

The "Tumbleweed" hexrotor concept was originally developed as a rotary wing vehicle for operation in the interface zone between urban structures and the airspace adjacent to these structures. The capability to vector thrust independent of body attitude enables manoeuvring in confined spaces and provides the ability to land and take off at arbitrary body attitudes. These performance improvements come at the cost of reduced hover efficiency and increased mass compared to planar rotorcraft solutions and hence there has to be a suitable trade off between improved manoeuvrability and reduced payload, range and endurance.

A prototype vehicle with fixed pitch propellers and flown in a similar manner to a conventional planar rotary wing vehicle was flown in 2009. The present hexrotor work is distinct to previous published work on multi-rotor UAVs, e.g. [1-3], which considers only coplanar rotor systems and is predominantly based on fixed-pitch rotors. A similar operating principle to the hexrotor was introduced by

Salazar [4] who used a fixed-pitch quadrotor with four thrusters to provide side-forces. Although this concept also provides means to decouple attitude and translational dynamics it requires at least 8 rotors and can only be used over a limited range of body attitudes.

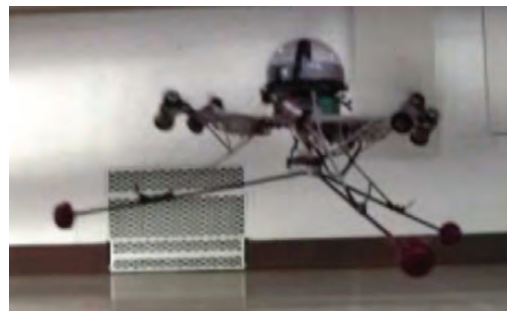


Figure 1: Test flight of a fixed-pitch hexrotor in November 2009

Our previous work on the hexrotor [5, 6] introduced the concept, described appropriate control strategies and demonstrated practical feasibility in near hover conditions. This paper goes further by presenting the engineering design and modelling of a variable pitch hexrotor system with increased control authority compared to the previous fixed-pitch design. It is shown how such a vehicle can be designed using off-the-shelf systems, open source electronics and rapid prototyping solutions. Furthermore, a collective-pitch/rpm envelope is presented that satisfies the design constraints whilst optimising the combined electric and aerodynamic efficiency for typical operating conditions. The feasibility of the proposed design is demonstrated through simulations and wind tunnel testing.

2 VEHICLE DESIGN SPECIFICATION

2.1 Systems concept

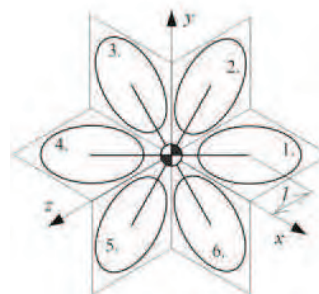


Figure 2: Orthogonal face-centred hexrotor configuration and body coordinate system

Email address: david.langkamp@postgrad.manchester.ac.uk

The way in which the six rotors are arranged into three rotor planes is a trade-off between hover efficiency and torque control authority (the interested reader is referred to [6] for details). However, the highest authority and most energy efficient thrust vectoring occur when the force characteristic axes are orthogonal such as in the configuration shown in Figure 2. This is the most energy efficient configuration for a hexrotor with decoupled translational and attitude dynamics, because it has the largest rotor disk area projected on the horizontal plane and hence the induced power requirement. The default orientation of the vehicle is such that equal weight support is provided by each rotor and the earth-z axis aligns with the resultant of $[-1, -1, -1]$ in body coordinates.

In order to provide full authority (4π steradians) of thrust and torque vectoring on a hover capable hexrotor configuration two key design constraints need to be met: a) the direction of thrust must be reversible and b) two rotors must be able to provide sufficient thrust to lift the vehicle. The latter proved to be the hardest constraint since it resulted in increased pressure on component performance and mass targets compared to those for typical planar multi-rotor designs. The approach taken was to design the propulsion group for maximum thrust/weight ratio at the expense of pure hover efficiency. There was also aggressive programme to minimise structural mass.

The system sizing was based on an electrically powered vehicle in the 1kg class suitable for indoor operations. A CAD illustration of the prototype is shown in Figure 3.

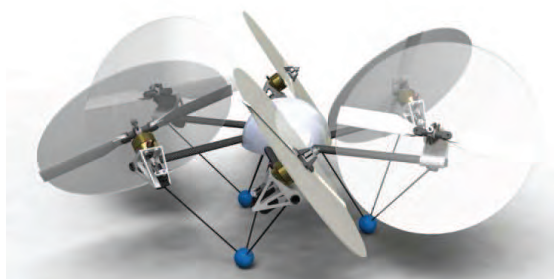


Figure 3: CAD rendering of prototype vehicle showing the orthogonal rotor planes. Rotor diameter 0.25m

2.2 Electric Variable Pitch Propulsion

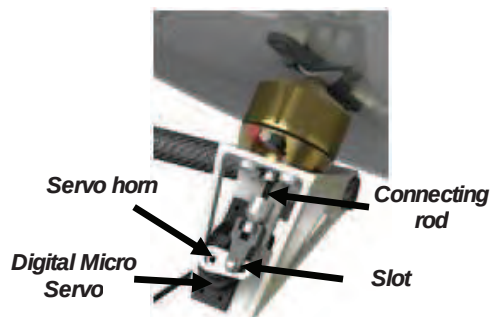


Figure 4: EVP unit driven by a digital servo

Electric Variable Pitch Propulsion (EVP) systems are commonly used for 3D aerobatic model aircraft. EVP systems typically consist of a DC electric motor and a hollow shaft. A connecting rod passes through the hollow shaft and is secured to a pitch horn so that the collective pitch of the symmetrical blades can be varied by pushing/pulling the connecting rod. The pitch actuation mechanism had to provide rapid pitch changes at a fine pitch resolution which is typically not required for fixed-wing aerobatic aircraft.

A pitch control mechanism consisting of a digital micro servo and slotted servo horn was chosen (Figure 4). The servo horn is slotted to prevent the transfer of lateral forces to the pin hence reducing the bending moment experienced by the connecting rod. A lower bending moment on the connecting rod means less friction between it and the hollow motor shaft, which leads to greater pitch control authority at high rotational speeds. The mechanism allows for a trade-off between pitch resolution and pitch response speed; a large servo moment arm gives a rapid pitch response but poor pitch resolution, while a small moment arm leads to the opposite. The mechanism also provides for an extremely compact geometry and easy to set-up design. The servo position was calibrated against the blade pitch under static conditions and the repeatability was tested using thrust and motor loading data.

Based on preliminary vehicle sizing and propulsion system mass fractions, to achieve flight a thrust/weight ratio of approximately 6.6 is required from each EVP system. Based on the EVP unit mass breakdown as shown in Table 1 this equates to a thrust of 6.3 N per EVP unit. Other important requirements are an identical performance when running in both clockwise and anti-clockwise rotary directions; a symmetrical thrust versus pitch curve with a large linear region to simplify control; small dimensions; and a mechanically simple design. The propulsion system thrust/weight ratio is at the limit of commercially available EVP systems, hence achieving the required maximum thrust is a key challenge. Finally, due to the lack of a significant incoming airflow heating issues were expected.

Component	Weight (g)
Brushless motor	45
Servo	8
EVP System, Hub and blades	19
25A ESC	17
Power connectors	7.5
Total =	96.5

Table 1: Mass breakdown of an EVP unit based on the AXI 2208/20

Table 1 shows the mass breakdown of the EVP unit. Whilst most components are comparable in mass to similar sized fixed-pitch systems, there is a mass penalty of about 10-20% due to the need for an additional actuator for blade pitch control and the increased mass of the rotor hub.

2.3 Structural design

The main constraint on the airframe structural design was to minimise mass in order to enable hovering on two rotors. The preliminary target mass for the airframe was defined as

10% (~100g) of the total takeoff mass. Further requirements were simplified construction and repair following damage.

The rotor arms are off the shelf carbon fibre tubing, cut to length using hand tools. The sizing of the carbon fibre tubing was based on static and dynamic finite element analysis and was checked experimentally, chosen for minimal bending deformation and a natural frequency above the motor rpm range.

All other components are manufactured from ABS plastic using an in-house low cost additive 3D printer (stereo-lithography). Using this technology complex parts, such as motor mounts ensuring the correct alignment of the rotor orientation, can be produced. This same method can be used to accurately align the rotors in the three rotor planes by printing components for an assembly jig. The strength to weight ratio of these components is sufficient for them to be evaluated on the vehicle. A strict mass reduction plan was followed using commercial finite element software to ensure the components would not fail under maximum load. An example for this the mass-minimisation strategy is the optimised centre-piece shown in Figure 5. The use of the in-house printer means design iterations are in the order of minutes or hours, and replacement components for repairs can be reproduced quickly.

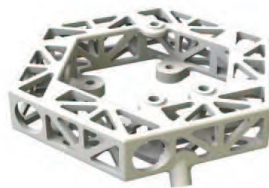


Figure 5: CAD model of the hexrotor centre piece showing the mass minimisation efforts

The vehicle's landing gear consists of three tripods consisting of carbon fibre struts and a foam ball. The landing gear struts were sized to ensure sufficient rotor ground clearance at a landing up to a 45° pitch or roll attitude. The chosen arrangement provided an easily replaceable landing gear as well as a desired structural failure point, minimising damage to the vehicle, with minimal mass penalty.

2.4 Indoor positioning solutions

Closed loop position and attitude control requires absolute measurement of position and/or attitude. In the absence of GPS for indoor testing, two relatively low cost positioning systems have been considered. The first is based on the use of time of flight measurements from ultrasonic beacons and the second is an optical based system using augmented reality markers.

For the ultrasonic solution a master unit equipped with multiple receivers is mounted on the vehicle. Each of the external slave units (currently 3) is commanded via the radio modem to emit a pulse. The master unit then calculates time of flight and therefore distance to slave unit for each of the vehicle receivers. This provides both position and attitude to the flight controller via the SPI bus.

In the visual system, Augmented Reality (AR) markers are used as an absolute metric for location detection. This approach consists of the use of a standard low cost camera and custom software developed to identify and localize specially designed AR markers.

The camera is initially calibrated offline using an existing MATLAB toolbox [7]. Algorithms from this toolbox extract the camera's intrinsic parameters including: focal length, principle point, skew, distortion and pixel error. These parameters are used for software configuration allowing the camera to be used as a projective sensor. In order to detect the markers a modified version of [9] is used to robustly detect edge pixels and then collect them to create line structures using RANSAC. A coarse grid system is used to obtain real-time performance.

The AR markers were specially designed to be identified within this UAV environment and operate in colour space rather than gray scale as is more usual. Once the markers have been detected the corner points were used to create a series of points that, when combined with the camera properties result in the position of the camera relative to the markers. The software was developed in C++, using OpenCV [8] for platform independence and for common image processing functions.

The main issue with the current AR approach is that this system only works with a single camera, however to get adequate coverage of a test area multiple cameras will be needed. There are also issues to be resolved with errors due to the non-uniform lighting of the AR markers.

Marker Size	1.5m	2m	2.5m	3m
8cm	100%	80%	20%	0%
10cm	100%	100%	50%	20%
12cm	100%	100%	100%	100%

Table 2: Optical positioning system performance against distance of camera from vehicle (percent of successfully recognized markers)

Although this method is currently limited by the effective detection range, with the right control of the camera the use of AR markers can be used as part of a complete solution for UAV localization within a controlled test area.

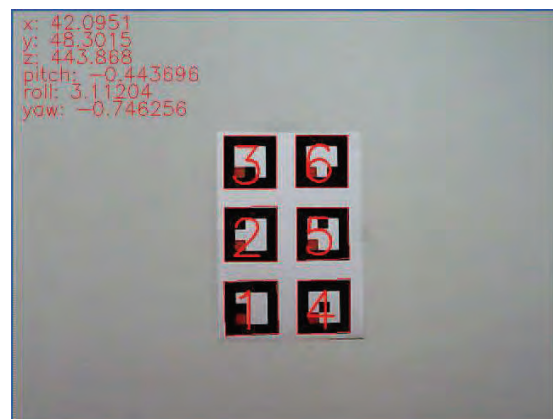


Figure 6: Example of the AR markers

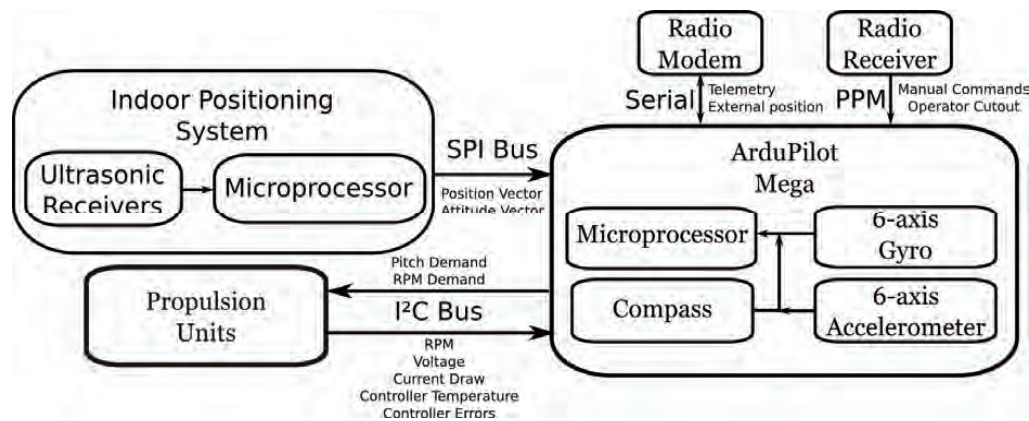


Figure 7: Avionics system block diagram highlighting major components

2.5 Avionics solution

In order to reduce costs and reduce development time, the onboard electronics package is based on the use of open-source hardware and open source software.

Stabilization and guidance is controlled using an open-source autopilot unit called ArduPilot Mega [10]. The unit comprises of:

- ATmega 1280 microprocessor
- 3-axis gyros and 3 axis accelerometers
- Absolute pressure sensor
- On-board datalogging

The design is compatible with devices from the Arduino project [11], which focuses on simple to use devices for prototyping embedded applications. ArduPilot uses the same software ecosystem but with a form factor more suitable for micro-avionics use.

Control software is based upon code from both the ArduPilot and AeroQuad projects [11]; this reduces development time by providing proven libraries for common autopilot tasks to which the unique control algorithms required for this project are being implemented.

Digi XBee 2.4 Pro wireless modems provide bi-directional telemetry and manual control is via a standard Spektrum 2.4GHz 12 channel transmitter and a 2.4GHz DSM2 receiver to minimize interference issues. A decision was made to use BL-CTL V1.2 speed controllers from the Mikrokopter [12] project. These units provide open source access to controller algorithms, allow the flight controller to query each propulsion unit about current draw, rotational velocity and temperature and are also approximately 25-35% lighter than similar specification commercial speed controllers.

Power is supplied by a commercially available 3-cell lithium-polymer battery, and is distributed to all speed controllers and avionics via a common power bus.

3 CONTROL STRATEGIES

3.1 Bi-state control principle

Because of the novel actuation architecture of this vehicle standard quadrotor control algorithms are not suitable. Instead, the vehicle uses what is referred to as a bi-state control scheme: In the translational state the vehicle rotational axes are fixed relative to earth axis and the vehicle is commanded to translate in the Earth fixed x, y and z directions. In the rotational state, the vehicle position is fixed and control inputs are used to change the orientation of the vehicle. Control inputs may be rotations (or rates) about either the vehicle body axes or the earth fixed axes. Rotation and position are persisted across state changes allowing the operator to rotate the vehicle into the correct orientation before translating toward the region of interest (within the limits of actuator saturation).

3.2 Static mapping control strategy

To reduce the thrust time constant of the propulsion system the proposed mode of operation is to run the propulsion system at a constant base rotational speed and achieve control through variable-pitch alone.

Full state feedback control for the vehicle has previously been synthesised and presented in [6]. This section is included in order to outline some of the implementation issues encountered.

The decoupling of force and torque inherent in the design of the vehicle imply a decoupling between translation and attitude. Therefore, the vehicle effectively has two controllers, which independently calculate the required force and torque vectors for the vehicle. A mapping function then translates these vectors into a pitch command for each propulsion unit.

The controllers are implementation specific; example controllers derived using classical theory can be found in [6] and these controllers form the basis of the ones used on board the current vehicle. The mapping function however is of interest as it is specific only to the general vehicle configuration.

For a required force and torque response, assuming static conditions such as around hover, a requirement vector $[u_1 \ u_2 \ u_3 \ u_4 \ u_5 \ u_6]^T$ can be defined using

$$(1) \begin{pmatrix} u_{1ot} \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} I & I \\ I & -I \end{pmatrix} C V \begin{pmatrix} P^{-1} R^T & 0 \\ 0 & Q^{-1} \end{pmatrix} \begin{pmatrix} f_{0_1} \\ f_{0_2} \\ f_{0_3} \\ t_{b_1} \\ t_{b_2} \\ t_{b_3} \end{pmatrix}$$

$$P = \hat{k} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad Q = \frac{\hat{k}l}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 1 \\ 0 & -1 & 1 \\ 1 & -1 & 0 \end{pmatrix}$$

$$S = k_T \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

Where :

f_0 = Force demand vector in earth axes (N)

t_b = Torque demand vector in body axes (Nm)

$\hat{k}$ = Coefficient relating pitch to force (N/rad)

l = Rotor arm length (m)

k_T = Coefficient relating pitch to torque (Nm/rad²)

R^T = Rotational matrix mapping between body and earth axes

The required pitch angles for each rotor, δ_i , are calculated by substituting the requirements vector $[u_1 \ u_2 \ u_3 \ u_4 \ u_5 \ u_6]^T$ into Equation 2. $[u_4 \ u_5 \ u_6]$ are then substituted in Equation 3 to obtain the complete pitch requirement vector.

$$(2) \begin{pmatrix} \delta_4 \\ \delta_5 \\ \delta_6 \end{pmatrix} = \begin{pmatrix} u_4 \\ u_5 \\ u_6 \end{pmatrix} + \frac{1}{2} Q^{-1} S \begin{pmatrix} (u_1 + u_4)^2 \\ (u_2 + u_5)^2 \\ (u_3 + u_6)^2 \end{pmatrix} \times$$

$$\left[I + Q^{-1} S \begin{pmatrix} u_1 + u_4 & 0 & 0 \\ 0 & u_2 + u_5 & 0 \\ 0 & 0 & u_3 + u_6 \end{pmatrix} \right]^{-1}$$

$$(3) \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix} = \begin{pmatrix} u_1 + u_4 \\ u_2 + u_5 \\ u_3 + u_6 \end{pmatrix} - \begin{pmatrix} \delta_4 \\ \delta_5 \\ \delta_6 \end{pmatrix}$$

Manual control of the vehicle by the operator can be implemented by using a standard RC controller to control either f_0 or t_b via appropriate scaling and damping. The

uncontrolled vector is simply set to 0 (the force input vector includes a gravitational compensation term).

Both control & mapping functions are significantly more computationally expensive than those of a traditional quadrotor. Implementation of the controllers and mapping function in the ArduPilot controller involves translation of the required algorithms into the Wiring language used by Arduino compatible embedded controllers such as the ArduPilot Mega.

This procedure is somewhat complicated by the lack of native matrix support in Wiring itself. Furthermore, the ATmega 1280 lacks a floating-point unit and therefore floating-point arithmetic must be calculated in software. The gyro and accelerometer sensor filtering already makes heavy use of floating-point calculations, restricting CPU cycles available for other parts of the flight control software. Whilst this is not an issue with the current classical controllers, implementation of more computationally expensive controllers may require the control & mapping algorithms to be ported to utilize fixed-point or even integer arithmetic. This should be possible with little or no loss of control quality and would yield significant performance improvement.

3.3 Hardware in the loop simulation

Implementation of the new EVP system for the hexrotor vehicle requires development of a new flight control system in terms of flight control hardware and software. For the purposes of initial ground testing a hardware-in-the-loop simulation approach has been adopted. This allows testing of the physical embedded flight controller by generating virtual IMU sensor data and feeding them into the controller under test. The generation of virtual data is done by a Simulink model of the vehicle. The controller response is then determined either by direct measurement of the control response signal or by measuring the physical response of the system in terms of the collective pitch angle and the rotational speed of the rotor blades.

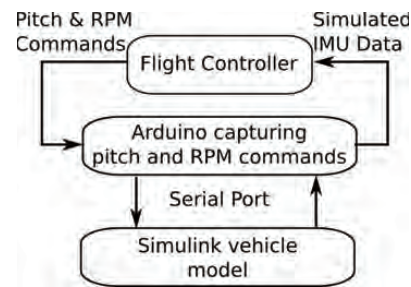


Figure 8: Proposed hardware in the loop setup

Virtual quantities from the simulation model are sent over a serial communication channel to an electrical emulator, which is also based on the Arduino platform. This emulator then transforms the simulated digital data into a proportional analogue voltage signal which is, in turn, fed as input to the flight controller.

This simulation approach allows verification of the basic functionalities of the microcontroller in response to the

simulated environment. This method is suitable for testing operation-critical systems such as flight controllers without the risk of damaging the vehicle. Additional response data (e.g. overall thrust, moments, etc.) will be obtained by placing the integrated vehicle on an appropriate six component force balance.

4 SYSTEM MODELLING

4.1 Modelling concept

A simulation environment for the hexrotor vehicle was created as tool for preliminary design and controller validation. The model is implemented in a modular Simulink system and draws on a range of standalone simulation blocks which could be changed to lower/higher fidelity models as required. The simulation model also includes guidance, visualisation blocks and pilot interface blocks, so that operations can be simulated and visualised up to a mission level.

4.2 Airframe model

The airframe itself is modelled as rigid body with the mass and inertial properties obtained directly from a CAD model. The relatively complex nature of the vehicle geometry means that it is convenient to define both a conventional body axes coordinate frame with x pointing forwards, y to starboard and z down and an additional body fixed frame aligned with the rotor planes as shown in Figure 2. Suitable rotation matrices were employed to resolve earth axes velocities and the local forces on the rotor disks.

Most published work on multi-rotor vehicles, including our previous work [6], treats fuselage aerodynamics as insignificant near hover. To improve the fidelity of the simulation at non hover conditions the airframe aerodynamics are modelled using axial and normal forces as functions of body attitude. The model was calibrated using wind tunnel data from tests on similar-sized multi-rotor fuselage designs at a range of practical body attitudes.

4.3 Rotor aerodynamics modelling

The aerodynamic modelling of the rotors was one of the key modelling challenges since the unconventional arrangement of the rotors lead to rotor disk angle of attack which are not typically found on aircraft propellers or helicopter rotors under normal operating conditions. The simulation had to deliver 6-dof forces/moments and the required rotor power for an initially unknown rotor load and at all velocity vectors of practical interest. Based on these requirements and the need to obtain rapid solutions with modest computing resources, a numerical blade element model was developed. Further details of the model and the wind tunnel validation experiments were published in [13, 14] only a brief summary is presented here.

The inputs to the blade element aerodynamics function are the current velocity vector onto the rotor disk, the rotational speed and the collective pitch angle. Aerofoil chord, pitch and section are treated as functions of the rotor radius and synthetic airfoil data is used to represent the nonlinear lift and drag coefficients at the local blade element Reynolds number for a $\pm 180^\circ$ angle of attack range. A first order flapping model is used to approximate the tip path

plane for a hingeless rotor system.

The induced velocity is approximated by combining a radial inflow variation with a 1st order harmonic inflow model for forward flight [15] or a Mangler & Squire model [16] for advance ratios greater than 0.1. The induced velocity model is used in an iteration loop based on the local velocity vector and loading of every blade element. The iteration loop is stabilised using an over relaxation scheme based on the blade element loading.

The local element loading is resolved into disk forces and moments and fed-back into the vehicle simulation environment. Since the current method is that it does not correctly predict vortex ring state conditions an acceptable vehicle velocity envelope had to be mapped out, so that none of the individual rotors would be subject to vortex ring state conditions. It is also noted that interference effects between rotors are ignored and this assumption is tested experimentally.

4.4 Electric systems modelling

The brushless motor is modelled using a standard 1st order DC motor model, such as presented by Drela [17] and in previous quadrotor design projects [18], and combining it with slight modifications to account for the particular nature of brushless motors. Based on simple parameters from motor datasheets or measurements the electric power consumption, current draw and electric efficiency can be estimated.

A first order model form [18] for lithium-polymer batteries is used with a focus on the battery voltage drop and losses for a given load. It is also used to ensure practical limits on the battery discharge in the simulation model.

4.5 Rotor-motor dynamics

The dynamics of the propulsion system can be roughly broken down into two effects: the rotor aerodynamics and the inertia of the rotor-motor system that is affecting the time constant for changes in rotational speed.

Variable-speed rotor systems can experience a significant delay in the rotational speed response which is largely driven by the inertia of the system [1]. By using a variable-speed design this effect can be avoided, but it is appreciated that further work is required to establish the dynamics of the rpm governor control loops.

For variable-pitch rotors there is a time delay between a pitch actuation and the corresponding thrust change. This delay can be modelled using the time constant of the dynamic inflow ODE [16]

$$(4) \quad \tau_\lambda = \frac{0.849}{4\lambda_i \Omega}$$

For a typical hexrotor hover rpm of 8000 ($\Omega = 420$ rad/s), and a mean hover inflow ratio ($\bar{\lambda}_i = 0.086$) this yields a time constant of 3 ms. This delay is 1-2 orders of magnitude small than on full sized helicopters and was considered insignificant for the purpose of this study.

4.6 Experimental validation

Experiments were carried out to evaluate the rotor performance and establish the magnitude of rotor interference effects.

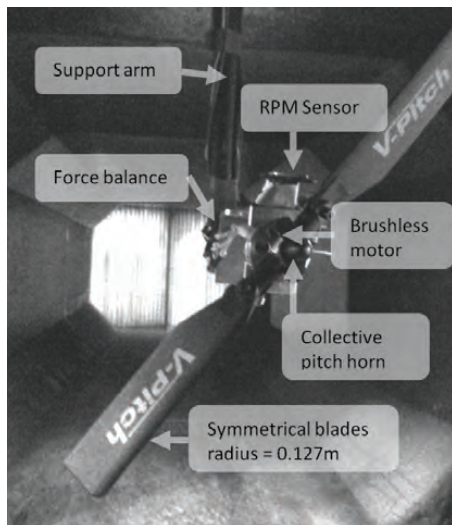


Figure 9: Variable pitch wind tunnel test rig with force balance and rpm sensor. Rotor diameter = 254 mm

The rotor performance was measured on a static testrig and in windtunnel tests a rotor similar to the rotors used on the final design. The test rotors had a radius of 0.127 m, a solidity of 0.099 and symmetrical, untwisted with a constant chord between 40-100% radius. The rotor was tested in an open-circuit blow down low speed tunnel with a 0.9x11.m test section and a turbulence level of 0.5%. A 6-axis force balance was fitted between the motor and the support arm and an overhead yaw control mechanism was used to allow for a 360° rotation. The assembly was equipped with a Hall effect sensor for the rotational speed and pressure transducers for the tunnel velocity.

Real-time data acquisition equipment was used to set demands for collective pitch and rpm and for taking simultaneous measurements of all sensors. The data was post-processed and expressed in nondimensional force coefficient and advance ratios which could be used to partly validate the blade element method over the range of flow conditions of practical interest. Further details of the testing and experiments can be found in [13, 14].

The rotor-rotor interference of the proposed hexrotor rotor arrangement was studied using a geometrically similar fixed-pitch hexrotor mounted on a 6-component force balance. Forces and moments were demanded through the static mapping relationships described in 3.2 and the linearity and off-axes response was studied. In hover interference was found to have negligible effects on net forces and moments [6], although no judgement can be made on interaction effects in forward flight.

5 RESULTS

5.1 Variable pitch operating mode and performance

For an electric variable pitch system the thrust demand can be met with a range of collective pitch and rotational speed combinations. Using simulations and experimental validation an operating mode was defined to satisfy the key constraint of hovering on two rotors whilst optimising the

combined aerodynamic and motor efficiency for typical hover conditions.

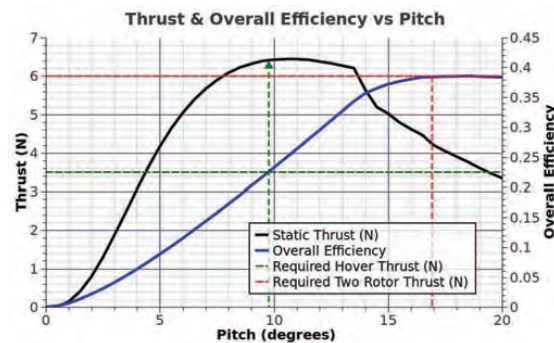


Figure 10: Hover performance validation and efficiency of a single EVP unit

Figure 10 shows single rotor thrust and combined efficiency against collective pitch. It demonstrates the suitability of the proposed EVP system, because the worst case design constraint of lifting the vehicle on two rotors ($T=6\text{N}$) can be satisfied if the system is operated at a sufficiently high base rpm of 12500. This rotational speed also maximises the system efficiency around typical hover conditions (Pitch = 9.5° , $T=3.5\text{N}$) for which the weight support is provided by all six rotors. Around these operating conditions thrust responds linearly to small changes in collective pitch.

The vehicle endurance at the proposed operating conditions was studied as a function of vehicle mass. Based on the availability of suitable batteries and the trade-off between additional battery capacity and increased power demand the maximum endurance for the current design is estimated around 6-7 minutes with a 3.5Ah 3-cell lithium polymer battery. Whilst this endurance is considerable lower than on similarly sized planar rotorcraft vehicles, this is a necessary trade-off resulting from the unique capabilities of the hexrotor design.

5.2 Current state of the project

Based on the method and experiments presented in this paper a prototype of the vehicle has been designed using commercially available parts and open-source electronics solutions. The flight capable version of the prototype, as of August 2011, can be seen in Figure 11.

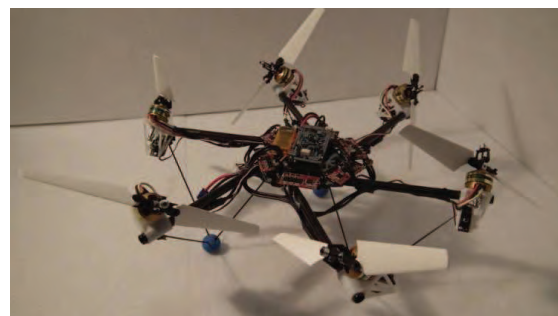


Figure 11: Flight-capable prototype of the variable-pitch hexrotor with assembled EVP units and an Ardupilot mega

Further static and free flight testing of the hexrotor is planned for September 2011.

The take off mass of the prototype is 1.2kg. The mass breakdown in Figure 12 is quite different from traditional coplanar vehicles due to the high propulsion group and low airframe mass fraction.

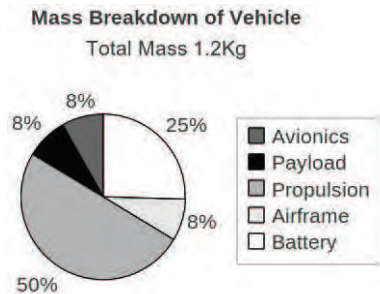


Figure 12: Vehicle mass breakdown

6 CONCLUSIONS

The engineering design for a novel hexrotor vehicle with variable-pitch rotors has been introduced which has the ability to point the thrust vector independent of the body attitude. The key contribution of this paper is to present an engineering and simulation approach that enables the design of such a vehicle with commercially available components and largely open source based software solutions.

It is shown that the key design constraint that arises from the hexrotor concept is the need to provide sufficient thrust with two rotors to lift the entire vehicle, so that thrust can be vectored at any attitude. This requirement has been met by designing an airframe with a mass fraction of around 8% of the all up weight and the selection of an electric variable pitch propulsion system based on a maximum propulsion unit thrust to weight ratio rather than efficiency.

Simulation and testing strategies for the unique rotor arrangement were presented and the feasibility of the propulsion unit static thrust response was demonstrated through ground testing and simulation.

ACKNOWLEDGMENTS

This work was supported by a grant from University of Manchester Intellectual Property Company (UMIP). The authors would like to thank Andrew Kennaugh for support with the experimental work and Ivan Pankratov, Chuan Wang and Christophe Le Brun for their hard work on this project. The original fixed-pitch hexrotor vehicle was developed by Phil Geoghegan and Matt Pilmoor.

REFERENCES

1. Pounds, P., Mahony, R., Corke, P.I., *Design of a Static Thruster for Microair Vehicle Rotorcraft*. Journal of Aerospace Engineering, 2009. 22(1): p. 85-94.
2. Bouabdallah, S., P. Murrieri, and R. Siegwart, *Towards autonomous indoor micro VTOL*. Autonomous Robots, 2005. 18(2): p. 171-183.
3. Hoffmann, G.M., Huang, H., Waslander, S.L., Tomlin, C.J., *Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment*, in *AIAA Guidance, Navigation and Control Conference and Exhibit, 20-23 August 2007*. 2007, AIAA: Hilton Head, South Carolina.
4. Salazar, S., Romero, H., Lozano, R., Castillo P., *Modeling and Real-Time Stabilization of an Aircraft Having Eight Rotors*. Journal of Intelligent and Robotic Systems, 2009. 54: p. 455-470.
5. Crowther, W.J., Lanzon, A., Pilmoor, M., Geoghegan, P., *Rotary Wing Vehicle*, U.K. Patent GB2462452A, Editor. 2008: UK.
6. Crowther, W.J., Lanzon, A., Maya-Gonzalez, M., Langkamp, D., *Kinematic Analysis and Control Design for a Non Planar Multirotor Vehicle* Journal of Guidance, Control, and Dynamics, 2011. **accepted**.
7. Bouguet, J.-Y., *Camera calibration toolbox for Matlab*. 2001.
8. Bradski, G., *The OpenCV Library*. Dr. Dobb's Journal of Software Tools, 2000.
9. Hirzer, M., *Marker Detection for Augmented Reality Applications in Austria Seminar Project*. 2008, Inst. for Computer Graphics and Vision Graz University of Technology: Graz.
10. Anderson, C. *ArduPilot Project*. Available from: <http://diydrones.com/profiles/blogs/ardupilot-main-page>.
11. Banzi, M., et al. *Arduino Project*. Available from: <http://arduino.cc/>.
12. HiSystems GmbH, *Mikrokopter*. Available from: <http://www.mikrokopter.de>.
13. Langkamp, D., Crowther W.J., *A low order rotor aerodynamics model for UAVs in wind*, in *2010 RAeS Aerodynamics Conference - Applied Aerodynamics: Capabilities and Future Requirements*. 2010, RAeS: Bristol.
14. Langkamp, D., Crowther W.J., *The role of collective pitch in multi rotor UAV aerodynamics*, in *36th European Rotorcraft Forum*. 2010: Paris.
15. Chen, R.T.N., *A Survey of Nonuniform Inflow Models for Rotorcraft Flight Dynamics and Control Applications*, in *NASA Technical Memorandum 102219*. 1989, Ames Research Centre.
16. Leishman, J.G., *Principles of Helicopter Aerodynamics 2nd edition*. 2006, Cambridge: Cambridge Aerospace Series.
17. Drela, M. *Qprop theory document*. 2007; Available from: <http://web.mit.edu/drela/Public/web/qprop/>.
18. Stepaniak, M.J., *A Quadrotor Sensor Platform*, in *Russ College of Engineering and Technology*. 2008, Ohio University.

Closing the gap between simulation and reality in the sensor and motion models of an autonomous AR.Drone

Arnoud Visser, Nick Dijkshoorn, Martijn van der Veen and Robrecht Jurriaans*

ABSTRACT

This article describes a method to develop an advanced navigation capability for the standard platform of the IMAV indoor competition: the Parrot AR.Drone. Our development is partly based on simulation, which requires both a realistic sensor and motion model. This article describes how a visual map of the indoor environment can be made, including the effect of sensor noise. In addition, validation results for the motion model are presented. On this basis, it should be possible to learn elevation maps, optimal paths on this visual map and to autonomously avoid obstacles based on optical flow.

Keywords: Quadrotor, visual SLAM, monocular vision, SURF features, noise models

1 INTRODUCTION

Small quadrotors with on-board stabilization which can be bought off-the-shelf, like the Parrot AR.Drone, make it possible to shift the research from basic control of the platform towards applications that make use of their versatile scouting capabilities. Possible applications are surveillance, inspection and search & rescue. Still, the limited sensor suite and the fast movements make it quite a challenge to fully automate the navigation for such platform. One of the prerequisites for autonomous navigation is to capability to make a map of the environment.

2 RELATED WORK

Our approach was inspired by Steder *et al.* [1], who presented a system that allows aerial vehicles to acquire visual maps of large environments using comparable setup with an inertia sensor and low-quality camera pointing downward. In their approach the inertia sensor was used to estimate a number of parameters in the spatial relation between two camera poses, which reduces the dimensionality of the pose to be estimated. Equivalent with our approach, Steder uses Speeded-Up Robust Features (SURF) [2] that are invariant with respect to rotation and scale. By matching features between different images, one can estimate the relative motion of the

camera, and thus, construct the graph that serves as input to the TORO-based network optimizer [3].

Prior to 2011 no standard platform existed in the Micro Air Vehicle competition¹, which allowed to participants to construct optimal sensor configurations. For instance, the team from MIT [4] installed a Hokuyo laser scanner on their UAV and where able apply many techniques developed for ground robotics on their flying robot. Another example is the approach of the team from Canterbury [5], where an omnidirectional camera was mounted on the platform, which was used to acquire an heading estimation from the optical flow. At the latest edition of the Micro Air Vehicle competition² an Parrot AR.Drone was extended with an infrared camera to be able to follow people [6]. Because the AR.Drone is a quite novel development, the number of studies based on this platform is limited. A recent publication is from Cornell University [7], where an AR.Drone is used to automatically navigate corridors and staircases based on visual clues.

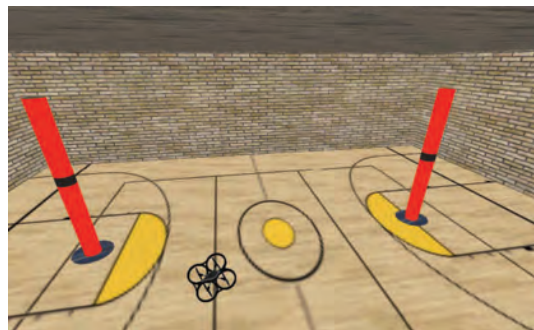


Figure 1: 3D model of a gym with realistic ground and wall textures which represents the Micro Air Vehicle pylon challenge.

As indicated in [8]; an accurate simulation of a quadrotor is a valuable asset, which allows safe and efficient development of control algorithms. Additionally, it gives direct access to ground truth values and allows to design repeatable experiments. To be used in such a way, not only the models of the actuators and sensors have to be validated. Another key feature is that the simulation environment is equipped with an editor which allows detailed modifications to the simulated

*Intelligent Systems Lab Amsterdam, Universiteit van Amsterdam

¹<http://www.emav2009.org>, <http://www.imav2010.org>

²<http://www.springimav2011.com>

environment so that the details needed for the experiment can be easily added as indicated in Figure 1. The environment selected is USARsim [9], which allows physical realistic simulations and a versatile environment editor.

3 METHODS

3.1 Map stitching

Our approach uses image stitching to build a visual map of an indoor environment. The frames from the AR.Drone's low-resolution down-looking camera are aligned together and drawn on a canvas object. A set of matches between two camera frames is used to estimate a homographic model for the apparent image motion. This model can be composed with the estimated motion of the rest of images in order to build a mosaic. The sonar sensor is used to detect obstacles and mask these obstacles in the camera image to make the image stitching more robust.

The first camera frame $\mathbf{I}_0$ is added at the center of the canvas object without any image transformation. This image determines the scale of the map, because consecutive images are related to this image. Each next image is related to the previous image by calculating a local perspective transformation $\mathbf{H}_{(i-1)i}$. This perspective transformation $\mathbf{H}_{(i-1)i}$ is used to transform image $\mathbf{I}_i$ such that it matches the scale, orientation and translation of image $\mathbf{I}_{i-1}$.

$\mathbf{H}_{(i-1)i}$ is computed by matching point features from two consecutive images. A certain degree of image overlap is required in order to find features that are present in both images. We use Speeded-Up Robust Features (SURF) [2] that are invariant with respect to rotation and scale. Each feature is represented by a descriptor vector and its position, orientation, and scale in the image. Each features from image $\mathbf{I}_i$ is matched with a feature from image $\mathbf{I}_{i-1}$ that has the shortest Euclidean distance.

The local perspective transformation $\mathbf{H}_{(i-1)i}$ is calculated by minimizing the back-projection using a least-squares algorithm. However, if not all of the point pairs fit the rigid perspective transformation (i.e. there are some outliers), this initial estimate will be poor. We use Random Sample Consensus (RANSAC) [10] to filter the set of matches in order to detect and eliminate erroneous matches. RANSAC tries different random subsets of four corresponding point pairs. It estimates the homography matrix using this subset and then computes the quality of the computed homography by counting the number of inliers.

Now that image $\mathbf{I}_i$ is related to image $\mathbf{I}_{i-1}$, it can be related to the first image $\mathbf{I}_0$ (canvas) by multiplying the local perspective transformation with the local perspective transformations from all previous matched frames.

$$\hat{\mathbf{H}}_{j0} = \prod_{k=j}^1 \mathbf{H}_{(k-1)k} \quad (1)$$

The resulting transformation $\hat{\mathbf{H}}_{j0}$ is used to warp image $\mathbf{I}_j$.

Finally, the warped image $\mathbf{I}_j^w$ is added to the canvas.

The local transformations between consecutive images can be composed to obtain the global transformation of the current frame, but local errors lead to a progressive drift in the transformation of future frames. One faulty local transformation can have disastrous impact on the image stitching. Therefore, we included several methods to detect faulty local transformations. Once a faulty local transformation is detected, the corresponding frame is dropped.

The first method to detect faulty transformations is counting the percentage of outliers. If the percentage exceeds threshold γ_1 the frame is being dropped (ignored). All feature points from image $\mathbf{I}_j$ are transformed with local transformation $\mathbf{H}_{(i-1)i}$. Then, the Euclidean distance between each transformed point from image $\mathbf{I}_i$ and the matched point from image $\mathbf{I}_{i-1}$ is measured. If the distance is larger than threshold ϵ the point is marked as outlier.

Another method to detect faulty transformations is by calculating the relative change of the global transformation when a new local transformation $\mathbf{H}_{(i-1)i}$ is added. Large relative changes indicate unlikely local transformations, because consecutive frames are likely to have only small differences in translation, rotation and scaling. If the relative change introduced by a frame's local transformation exceeds threshold γ_2 the frame is being dropped (ignored). The relative change $\delta\mathbf{H}$ introduced by frame $\mathbf{I}_j$ and its local transformation $\mathbf{H}_{(i-1)i}$ is calculated as following:

$$\delta\mathbf{H} = \hat{\mathbf{H}}_{(i-1)0} \cdot \hat{\mathbf{H}}_{i0} \quad (2)$$

where $\cdot /$ is a per-element division. The frame is dropped if $\langle \|\delta\mathbf{H}\| \rangle > \gamma_2$.

Processing a single frame and merging it into the visual map requires approximately 150ms. The AR.Drone's framerate is fixed at 15fps, which is too high to process each single frame in realtime. In order to achieve realtime map stitching, frames that are receiving while another frame is still being process, are dropped. However, this reduces the vehicle's maximum speed that provides enough overlap between consecutive frames. For example, when flying at 1m altitude, the camera (64 degree FOV) perceives 1.24m floor. The maximum horizontal speed to achieve 50% overlap at 15fps is 9.3m/s, which is nearly twice the actual maximum speed of the AR.Drone (5m/s). With reduced framerate (6.67fps instead of 15 fps), the maximum horizontal speed is 4.13m/s, 83% of the actual maximum speed.

3.1.1 Obstacle masking

The map stitching method just described assumes that the terrain is approximately flat. However, this assumption does not hold when flying at low altitude. The parallax effect results in faulty local transformations and errors in the map.

A method to prevent the parallax effect is by removing obstacles from the camera frames, such that a flat terrain re-

mains. The sonar sensor is used to create an elevation map. This elevation map is being generated simultaneously with the visual map and has the same scale and size. When image I_i is received, the corresponding piece from the elevation map is extracted by transforming the elevation map with $\hat{H}_{i0}^{-1}$. Now, the extracted elevation map has the same size and scale as the received image. All pixels from image I_i with an elevation greater than γ_3 are masked and not being used for feature detection. The method is currently under study, details will be published elsewhere.

3.1.2 Inertia

The motion was in the previous sections purely estimated on visual clues only. The AR.Drone is equipped with a number of additional sensors which give constants updates about the motion. The AR.Drone's inertial sensor data (body acceleration and attitude) can be used to estimate the current position. To get a robust estimate an Extended Kalman Filter is applied. The state vector comprises a position vector p^W , velocity vector v^W , acceleration vector a^W and attitude vector q^W .

$$x = [p^W v^W a^W q^W] \quad (3)$$

The resulting position estimate can be used as input for the map stitching algorithm. This method is applied in one experiment to study if the map stitching algorithm could benefit from this additional information.

3.2 Simulation model

The AR.Drone is a stabilized system (see Figure 2). When no control signals are given the quadrotor hovers on the same location, which is accomplished by a feedback loop which uses the sonar (for altitude) and the bottom camera (for horizontal position). The simulation makes use of this assumption. When no control signal is given, the AR.Drone stays at the same location. When a control signal for an longitudinal or lateral velocity is given, it calculates the force needed to reach that velocity (and assuming that the drag force D_b increases linearly with the velocity). When the control signal stops, the drag force D_b slows the quadrotor down until it hovers again. The USARSim quadrotor model uses the Karma physics engine (part of the Unreal Engine [11]) to simulate the force and torque acting on the aircraft. Yet, only the overall trust is calculated, the differential trust is not used. When moving in the horizontal plane, a real quadrotor changes its angle of attack (which is the defined as the angle between direction of motion e_V and the body frame e_N [12]). The Karma physics engine does not need this angle to calculate the resulting horizontal movement. Yet, this angle of attack has direct consequences for the viewing directions of the sensors, so the roll and the pitch should be adjusted in correspondence with horizontal movements.

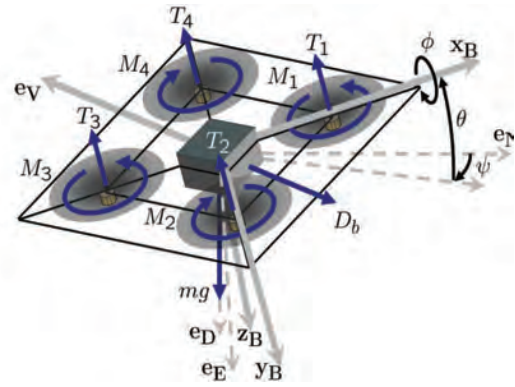


Figure 2: Free body diagram of a quadrotor helicopter (Courtesy Hoffman *et al.* [13]). Note that a right-handed orthogonal coordinate system is used with the z -axis pointing down. Each of the 4 motors has a trust T_i and momentum M_i . Together the motors should generate sufficient vertical trust to stay airborne, which is indicated by the gravity force mg in the direction e_D . Differential trust between the motors can provide roll ϕ and pitch θ torques, which lead to an angle of attack α . This can result in fast movements of the helicopter (e.g. in the horizontal plane) in the direction e_V which a resulting drag force D_b .

Control signals for vertical and rotational movements (around the z -axis) are calculated in the same manner. For vertical movements not only the drag force D_b is taken into account. In this case also the gravitational force mg is included in the equation. Rotations around the z -axis stop quite quickly when no control signal is given. For this rotational movement a 20x larger drag force D_r is used to model the additional inertia.



Figure 3: 3D model of the Parrot AR.Drone. This is a simplified model, based on the highly detailed model provided by Parrot SA.

The result is a simulation model (see Figure 3), which maneuvers close to the actual AR.Drone. Both the simulated and real system have the same dimensions $(0.525, 0.515, 0.115)m$. The principal elements of inertia are calculated correspondingly to $(0.0241, 0.0232, 0.0451)kg \cdot m^2$, assuming a homogeneous distribution of the mass.

4 RESULTS

4.1 Map stitching

Three types of experiments have been carried out to evaluate the map stitching approach. Each experiment is performed with the AR.Drone and the simulated AR.Drone using USARSim.

The first experiment measures the performance of the method without obstacle masking. Four large puzzle pieces are lead out on the floor within a square of $1.6m \times 1.6m$. These puzzle pieces are used as landmarks that are easily recognizable in the map. Each landmark has a label (A, B, C, etc). The distance between the centers of the landmarks is $1.3m$, except for the distance B-D ($\sqrt{1.3^2 * 1.3^2} = 1.84m$). Smaller puzzle pieces are lead out inside the square to provide enough texture for feature detection.

The realtime map stitching method is performed on the floor as described above. The distances between the landmarks inside the generated map (red lines) are compared to the know ground truth to compute the error of the stitched map.

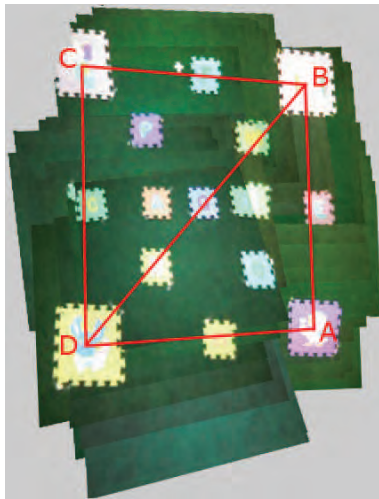


Figure 4: Map created by the map stitching method. Camera images are taken by the AR.Drone flying at approximately 0.85m.

The results of this experiment can be found in Table 1 and Figures 4 (real AR.Drone) and 5 (simulated AR.Drone). Both for the simulated and real AR.Drone a visual map is created with enough quality for human navigation purposes. The visual map created by the simulated AR.Drone contains fewer errors than the map of the real AR.Drone. This is not an intended result. Care has been taken to reproduce the circumstances in simulation as good as possible; the camera images in simulation are post-processed (decreased saturation, increased brightness, down-sampled resolution) to mimic the real images as close as possible. The difference between real and simulated visual map could also be explained by

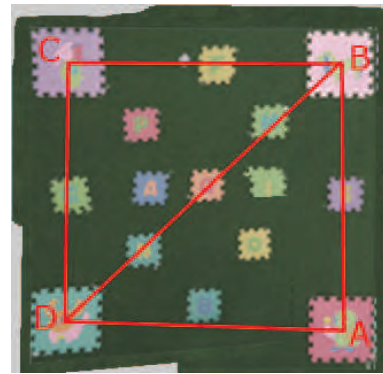


Figure 5: Map created by the map stitching method. Camera images are taken by the simulated AR.Drone flying at approximately 0.80m.

landmarks	A-B	B-C	C-D	D-A	B-D
AR.Drone					
mean error (m)	0.385	0.146	0.608	0.156	0.445
error (%)	29.6	11.2	46.8	12.0	24.1
USARSim simulator					
mean error (m)	0.019	0.047	0.026	0.075	0.028
error (%)	1.46	3.62	2.00	5.77	2.15

Table 1: Accuracy of the realtime stitched map by measuring the distance between landmarks. The maps are creating using the AR.Drone (Figure 4) and the USARSim simulator (Figure 5) .

smoother movements between the frames in simulation. Yet, also here care has been taken to mimic the dynamics of the AR.Drone as close as possible (as described in Section 4.2). Visual inspection of the video stream shows that there are equivalent changes in the movements between frames. Our hypothesis is that the remaining difference between simulation and reality are due to the effect of automatic white balancing of the real camera. In the next experiment this hypothesis will be further studied.

A second experiment is performed to study our hypothesis that the real AR.Drone produces less accurate maps caused by the automatic white balancing of the camera. The post-processing step from the images in simulation is extended with an additional filter, which changes the brightness randomly (including a non-linear gamma correction). With this additional error source the map of the simulated AR.drone (Figure 6) close resembles the map of the real AR.Drone (Figure 4). Also the quantitative comparison of Table 1 and 2 show now that the errors in simulation increased from maximal $7.5cm$ to $25.4cm$, which means an improvement in realism. The maximum error in the real visual map is still a bit larger ($60.8cm$), but this can be attributed to a scaling error (a systematic error in the attitude measurement of the sonar sensor).

Closer inspection shows that the white balance variations reduce the stability of the detected features, i.e., less of the same features are found between consecutive frames. This finding is supported by a statistical measure: the average Euclidean distance of features that are matched across consecutive frames. The post-processing increases the average feature distance from $22.1px$ to $32.9px$. This resembles the average feature distance of the real AR.Drone ($32.7px$).

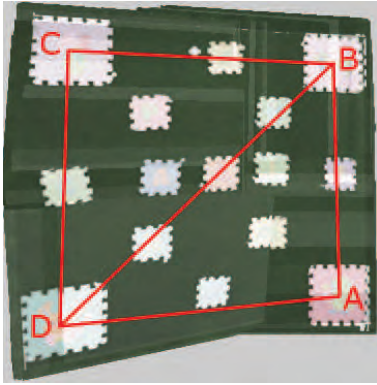


Figure 6: Map created by the map stitching method. Camera images are taken by the simulated AR.Drone that mimics the real AR.Drone's white balance variations.

landmarks	A-B	B-C	C-D	D-A	B-D
USARSim simulator (white balance variations)					
mean error (m)	0.031	0.181	0.215	0.254	0.190
error (%)	2.21	12.93	15.36	18.14	10.27

Table 2: Accuracy of the realtime stitched map by measuring the distance between landmarks. The map is creating using the USARSim simulator (Figure 6) with post-processing of the camera images to increase realism.

The floorplan from the first and second experiment is repeated 3 times in both directions. Now, the texture on the floor covers $4.8m \times 4.8m$. The AR.Drone flew in a 8-shape above the floor to capture an intermediate loop (point A and E from Figure 7). This experiment is performed on the simulated AR.Drone to show how this method scales up under favorable conditions (without additional white balance noise).

The third experiment shows the limit of the current approach without loop-closure or information from other sources (inertia measurements and controls). Figure 7 and Table 3 reveal the accumulative error propagation of the current stitching method, which was not clearly visible from first experiment. The error at point A-E can only be reduced with a global optimization routine. In section 6 an indication is given how this accumulative error propagation can be battled.

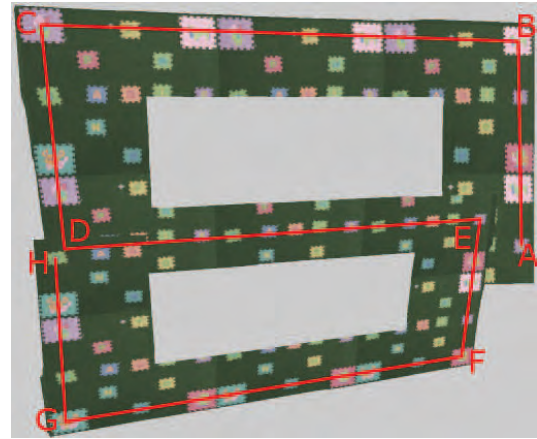


Figure 7: Map created by the map stitching method. Camera images are taken by the simulated AR.Drone flying at approximately 0.80m.

landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m)	0.220	0.87	0.579	0.220	0.523
error (%)	10.48	19.33	27.57	4.89	24.90
landmarks	F-G	G-H	H-A	B-G	
mean error (m)	0.011	0.244	0.788	0.14	
error (%)	0.24	11.62	17.51	2.20	

Table 3: Accuracy of the realtime stitched map (Figure 7) by measuring the distance between landmarks. The map is creating using the USARSim simulator.

To show the effect of additional information, the last experiment is repeated with data from the AR.Drone's inertial sensor. The information from this sensor (body acceleration and attitude) is used in an Extended Kalman Filter to estimate the current position, as described in Section 3.1.2.

landmarks	A-B	B-C	C-D	D-E	E-F
mean error (m)	0.029	0.689	0.049	0.565	0.013
error (%)	1.38	15.31	2.33	12.56	0.62
landmarks	F-G	G-H	H-A	B-G	
mean error (m)	0.596	0.080	0.720	0.243	
error (%)	13.24	3.81	16.0	3.83	

Table 4: Accuracy of the EKF-based stitched map (Figure 8) by measuring the distance between landmarks. The map is creating using the USARSim simulator.

Figure 8 clearly shows that the AR.Drone flew a 8-shaped trajectory. Also the quantitative comparison showed that the relative error drops from maximum 27.57% in Table 3 is reduced to 16.0% in Table 4.

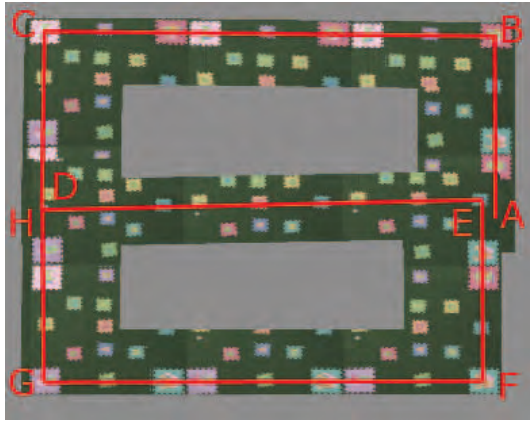


Figure 8: Map created by an Extended Kalman Filter (EKF). Camera images are taken by the simulated AR.Drone flying at approximately 0.80m.

4.2 Validation of the simulation model

To evaluate the USARSim quadrotor model, a set of maneuvers is flown with the actual AR.Drone and simulated AR.Drone. The differences between the maneuvers are studied in detail. To enable multiple repetitions of the same maneuver it is described as a set of time points (milliseconds since initialization) each coupled to a movement command. We wrote wrappers for the AR.Drone programming interface and for USARSim interface which read these scripts and output a control signal, using the system clock to manage the timing independently from the game engine and the AR.Drone hardware. Orientation, altitude and horizontal speed are recorded at a frequency of 200Hz during the maneuvers. These are gathered through the AR.Drone's internal sensors and the build-in algorithms, which are also used by the controller to operate the drone. The filtered output of the MEMS gyroscope is used for estimating orientation. The filtered output of the ultrasound distance sensor is used for estimating altitude. The optical flow algorithm using the bottom camera is used for estimating the horizontal (linear and lateral) speeds. The simulator has equivalent sensors. In addition, simulation can provide ground-truth data. Also for the real maneuvers an attempt was made to generate ground truth via an external reference system; the movements were recorded with a synchronized video system consisting of with four firewire cameras, capturing images at 20 frames per second at a resolution of 1024 x 768 pixels. The position of the AR.Drone in each frame has been annotated by hand.

Corresponding to NIST guidelines [14] a set of experiments of increasing complexity was performed. For the AR.Drone four different experiments were designed. The first experiment is a simple hover, in which the drone tries to maintain its position (both horizontal and vertical). The second experiment is linear movement, where the drone actuates a single movement command. The third experiment

is a small horizontal square. The last experiment is a small vertical square.

4.2.1 Hovering

Quadrotors have hovering abilities just like a helicopter. The stability in maintaining a hover depends on environmental factors (wind, underground, aerodynamic interactions) and control software. If no noise model is explicitly added, the USARSim model performs a perfect hover; when no control signal is given the horizontal speeds are zero and the altitude stays exactly the same.

For the AR.Drone, this is a good zero-order model. One of the commercial features of the AR.Drone is its ease of operation. As part of this feature it maintains a stable hover when given no other commands, which is accomplished by a visual feedback loop. So, the hovering experiment is performed indoors with an underground chosen to have enough texture for the optical-flow motion estimation algorithm.

As experiment the AR.Drone maintains a hover 35 seconds. This experiment was repeated 10 times, collecting 60k movement samples for a total of 350 seconds. Over all samples the mean absolute error in horizontal velocity (the Euclidean norm of the velocity vector) is $0.0422m/s$ with a sample variance of $0.0012m^2/s^2$. From the samples we obtain the distribution of the linear and lateral velocity components.

From the velocity logs the position of the AR.Drone during the 35 second flight was calculated. The mean absolute error of the horizontal position is $0.0707m$ with a sample variance of $0.0012m^2$.

4.2.2 Horizontal movement

In this experiment the drone is flown in a straight line. It is given a control pulse with a constant signal for 5 different time periods: 0.1s, 1s, 2s, 3s, and 5s. Each pulse is followed by a null signal for enough time for the drone to make a full stop and a negative pulse of the same magnitude for the same period, resulting in a back and forth movement. In Figure 9 the red line shows the control signal, the blue line the response of the AR.Drone. The experiment was repeated for 5 different speeds. The control signal s specifies the pitch of the drone as a factor (between 0 and 1) of the maximum absolute tilt θ_{max} which was set to the default value³. The trails were performed with the values of 0.05, 0.10, 0.15, 0.20, 0.25 for the control signal s .

Robots in USARSim are controlled with a standardized interface, which uses SI units. A robot in USARSim expects a DRIVE command with a speed in m/s and not the AR.Drone native signal s . Thus in order to fly comparable trials the relation between the drone's angle of attack α and the corresponding velocity v has to be investigated. When flying

³ARDrone firmware (1.3.3)

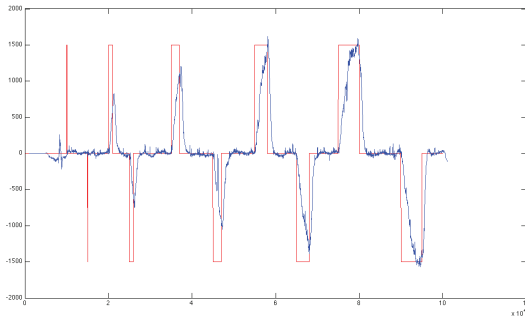


Figure 9: Response of the real AR.Drone on a number of pulses with a amplitude of $s = 0.15$.

straight forwards, the angle of attack α is equivalent with the pitch θ . In order to do this the samples from the logs where the drone has achieved maximum velocity has to be selected. Closer inspection of the velocity logs show that in each trial there is still constant increase of velocity for the first three pulses. For the last two pulses there is obvious plateauing, which indicates that the last two seconds of the five-second pulses is a good indication for the maximum velocity. Therefore the velocity at those last two seconds was used to compute mean absolute speeds $\bar{v}$, which are combined with the mean absolute pitch $\bar{\theta}$ as measured by the MEMS gyroscope. The estimates for $\bar{v}$ and pitch $\bar{\theta}$ are presented in Table 5 together with their standard deviation. Extrapolating the mean pitch $\bar{\theta} \simeq 7.5^\circ$ at control value $s = 0.25$ to the maximum control signal gives an indication of the drone's maximum pitch $\theta_{max} \simeq 30^\circ$ value. For typical usage, the angle of attack never exceeds 12° degrees.

	Control signal s				
	0.05	0.10	0.15	0.20	0.25
$\bar{v}$ (m/s)	0.4044	0.6284	1.4427	1.7587	2.2094
σ_v (m/s)	0.096	0.226	0.070	0.126	0.165
$\bar{\theta}$ (deg)	1.4654	2.9025	4.1227	5.7457	7.4496
σ_θ (deg)	0.455	0.593	0.482	0.552	0.921

Table 5: Averaged velocity $\bar{v}$ measured at the end of a 5 seconds pulse of the control signal s , including the corresponding pitch $\bar{\theta}$ as measured by the gyroscope.

To convert the drone's control signal s to USARSim commands v a least-squares fit through the points of Table 5 is made for the linear function $v = c \cdot \theta$, which gives us $c = 0.2967$. Equation 4 gives the final conversion of a control signal s to a velocity v in m/s given the drone's maximum pitch θ_{max} in degrees.

$$v = 0.2967 \cdot s \cdot \theta_{max} \quad (4)$$

The USARSim model has a parameter P_θ for calculating

the angle in radian given the velocity, which is the value $\hat{P}_\theta = 0.057$, as used in subsequent simulations.

The next experiment checks the acceleration of the real and simulated AR.Drone. First we give an estimate of how quickly the drone's controller changes its pitch to match the commanded pitch and how well it can keep it. For this we select all samples from 100ms after the start of the 2s, 3s, and 5s pulses till the first sample at which the commanded pitch has been reached. This corresponds to the time-span between which the drone has started to act on the change in the control signal until it reaches the commanded pitch. The result is illustrated in Figure 10.

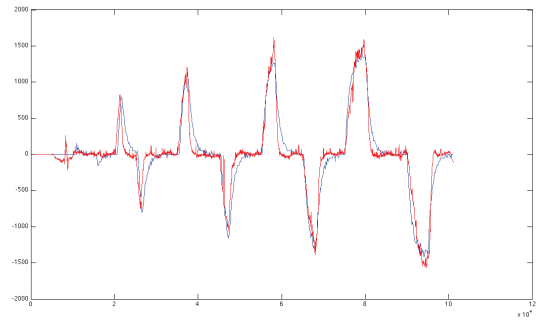


Figure 10: Response of the real (red) and simulated AR.Drone (blue) on the same pulses as shown in Figure 9.

As one can see, the acceleration has for the real and simulated AR.Drone nearly the same slope. The deceleration of the simulated AR.Drone is slightly slower. In the real AR.Drone the feedbackloop based on the optical flow of the ground camera actively decelerates the system. Overall, the dynamic behavior of the simulator closely resembles the dynamic behavior of the real system. Additionally, tests with more complex maneuvers (horizontal and vertical square) have been recorded, but unfortunately not yet analyzed in detail.

5 CONCLUSION

The current map stitching method is able to map small areas visually with sufficient quality for human navigation purposes. Both the AR.Drone and USARSim simulator can be used as source for the mapping algorithm. The visual map created by the simulated AR.Drone contains fewer errors than the map of the real AR.Drone. An experiment showed that the difference can be explained by the effect of automatic white balancing of the real camera.

The validation effort of the hovering and the forward movement shows that the dynamic behavior of the simulated AR.Drone closely resembles the dynamic behavior of the real AR.Drone. Further improvement would require to include the characteristics of the Parrot's proprietary controller into the simulation model.

6 FUTURE WORK

Future work will use visual slam to build the map and estimate the position of the robot. Such methods can handle loop-closures events when places are revisited and corrects for drift. Other future work is the integration of graphic framework to display a visual elevation map similar to [1].

A further improvement would be to include the images of the high-resolution front camera into the process. Those images can be used to calculate the optical flow from monocular stereo vision, which serves as the basis for both creating a disparity map of the environment. The disparity map can be used to detect obstacles ahead, which can be used in autonomous navigation. When combined with a time to contact method, the measurements can be used for a crude 3D reconstruction of the environment and another source for estimating the ego-motion.

Once the visual map exists, this map can be extended with range and bearing information towards landmarks (as the pylons from the IMAV challenge). On close distance the bearing information could be derived directly from perception. By modeling this as an attractive force, the correct heading could be spread over the map by value iteration. In addition, obstacles visible in the disparity map of the front camera could be used as basis for a repulsive force. Both forces could be combined to a force field that guides a robot on the map. The availability of a realistic simulation model will add in the training of these machine-learning approaches.

ACKNOWLEDGEMENTS

We like to thank Parrot S.A. for providing an AR.Drone for the competition. This research is partly funded by the EuroStars project 'SmartINSIDE'. We like to thank Carsten van Weelden for his experiments to validate the motion model of the AR.Drone.

REFERENCES

- [1] B. Steder, G. Grisetti, C. Stachniss, and W. Burgard. Visual SLAM for flying vehicles. *Robotics, IEEE Transactions on*, 24(5):1088–1093, 2008.
- [2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346 – 359, 2008.
- [3] G. Grisetti, S. Grzonka, C. Stachniss, P. Pfaff, and W. Burgard. Efficient estimation of accurate maximum likelihood maps in 3d. pages 3472–3478, San Diego, CA (USA), 2007.
- [4] Abraham Bachrach, Ruijie He, and Nicholas Roy. Autonomous flight in unknown indoor environments. *International Journal of Micro Air Vehicles*, 1(4):217–228, December 2009.
- [5] John Stowers, Andrew Bainbridge-Smith, Michael Hayes, and Steven Mills. Optical flow for heading estimation of a quadrotor helicopter. *International Journal of Micro Air Vehicles*, 1(4):229–239, December 2009.
- [6] E. Jovanov, M. Milosevic, R. Tilly, M. Truex, and C. Jones. Autonomous Personnel Tracking Help AR.Drone. In *International Micro Air Vehicle Conference*, May 2011.
- [7] Cooper Bills, Joyce Chen, and Ashutosh Saxena. Autonomous mav flight in indoor environments using single image perspective cues. In *International Conference on Robotics and Automation (ICRA)*, 2011.
- [8] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar. The grasp multiple micro-uav testbed. *Robotics Automation Magazine, IEEE*, 17(3):56 –65, sept. 2010.
- [9] Workshop on robots, games, and research: Success stories in usarsim. In Stephen Balakirky, Stefano Carpin, and Mike Lewis, editors, *Proceedings of the International Conference on Intelligent Robots and Systems (IROS 2009)*. IEEE, October 2009.
- [10] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [11] Stefano Carpin, Jijun Wang, Michael Lewis, Andreas Birk, and Adam Jacoff. High fidelity tools for rescue robotics: Results and perspectives. In Ansgar Breidenfeld, Adam Jacoff, Itsuki Noda, and Yasutake Takahashi, editors, *RoboCup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Computer Science*, pages 301–311. Springer Berlin / Heidelberg, 2006.
- [12] T. Yechout, S. Morris, D. Bossert, and W. Hallgren. *Introduction to Aircraft Flight Mechanics: Performance, Static Stability, Dynamic Stability, and Classical Feedback Control*. American Institute of Aeronautics and Astronautics, Reston, VA, 2003.
- [13] Gabriel M. Hoffmann, Haomiao Huang, Steven L. Wasl, and Claire J. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *Proc. of the AIAA Guidance, Navigation and Control Conference*, 2007.
- [14] Adam Jacoff, Elena Messina, Hui-Min Huang, Ann Virts, and Anthony Downs. Standard Test Methods for Response Robots. ASTM International Committee on Homeland Security Applications, January 2010. subcommittee E54.08.01.

PLASE : A Novel Planar Surface Extraction Method for the Autonomous Navigation of Micro-Air Vehicle

Rafid Siddiqui*, Mohammad Havaei*, Siamak Khatibi*, Craig Lindley*

*Blekinge Institute of Technology, 37179 Karlskrona Sweden.

Abstract— A planar surface extraction method is proposed for the indoor navigation of a micro-air vehicle (MAV). The algorithm finds planar clusters from the unorganized pointclouds. This is achieved by implementing a novel approach that first segments the data points into clusters and then each cluster is estimated for its planarity. The method is tested on indoor point cloud data obtained by 3D PrimeSense based sensor. In order to validate the algorithm, a simulated model containing a set of planes has been constructed, with noise injected into the model. The results of the empirical evaluation suggest that the method performs well even in the presence of the noise and non-planar objects, suggesting that the method will be a viable one for use in MAV navigation in the presence of noisy sensor data.

I. INTRODUCTION

THE Efficient extraction of surfaces from the environment is a fundamental requirement for robust robotic perception, although this is a challenging task and an active research area. It is important to extract only those parts of the environment that are most relevant in the application domain, so the data can be processed effectively and efficiently. Selective data extraction can also result in improvement of the overall robustness of the algorithms used in various crucial areas, including robotic navigation, manipulation, object recognition, and scene analysis, etc. The success or failure of the particular solution in these areas is dependent on the application domain as well as the robotic architecture due to the different demands posed by each domain [1] and the challenges posed by the nature of the maneuverability of the robot. In case of navigation of MAVs (Micro Aerial Vehicles) the challenges are quite different from those faced by ground robots, since MAVs require faster decision making with minimal onboard resources. From this perspective, it is important to retrieve only the most important information from the raw data coming from sensors, and to process only the information that is most crucial to the application domain.

In general, navigation solutions can be divided into two categories: (a) Map-based navigation and (b) Model-based navigation. The former has been applied in ground robotics for the past decade and is robust when computational power is not an issue or decision making is not required so frequently. The approach is to build a map of the environment and update it continuously. The resulting maps are good for path planning and for supporting cognitive abilities of robots, but they require lot of computation and hence slower decision making since the refresh rate of a map is dependent on the scan rates of sensors. Recently, applying such methods for navigation has been a trend in flying robots. Keeping in view the context, we are only discussing the three map based techniques which differ from

each other such that they can sub-divide the area. One technique describes a distributed approach while second one describes a conventional SLAM based approach using both laser and stereo vision sensors. The last one gives a semi-map based technique using occupancy grids.

In [2] a decentralized approach to map building is proposed in which one node utilizes a shared map from other nodes. It was observed that this method works well unless one node has a higher error rate than the others, in which case the error rate is reduced for other nodes while it increases for the node itself. A map based technique for GPS denied indoor environments was employed in [3] using a stereo vision camera system and a laser scanner. Obstacle avoidance in such systems is vulnerable since it is dependent on the scan rate of the laser sensor. A good utilization of maps for navigation is achieved by building occupancy grids for obstacle avoidance. In this context a Multi Volume Occupancy grid (MVOG) is an efficient method which divides the 3D space into occupied and vacant space [4].

In contrast to map-based navigation methods, model-based methods utilize primitive shape models that are common in the environment, or grab the structure that is generated from motion. In 2D visual navigation, a lot of work has been performed using optical flow for navigation of MAVs because of its simplicity and resemblance to biological vision systems of insects [5, 6]. In [7] a hybrid approach has been proposed that combines 2D video information from a camera and position information from a laser scanner. The Structure From Motion (SFM) was obtained from video camera images and the information was fused with GPS and inertial sensors in order to improve the accuracy of navigation. The extraction of most primitive features (e.g. plane, cone, sphere, and torus) is another efficient method that can be used for robust navigation of MAVs [8].

In the work reported in this paper, we intend to extract planar structures from the environment, since these are the most common ingredient in the man-made structures. Surfaces are extracted from the point cloud by segmenting it into clusters using Normalized Cuts [9] with an improved association function. The NCut is chosen as a segmentation technique for its broad applicability in the related context [10-11] because of being based on graph partitioning approach which is one of the robust methods used in image segmentation. The complete detail of the algorithm is described in the next section. The resulted planes will be used in ongoing work for the estimation of MAV location in the point cloud of an indoor scene.

This article is organized as follows: Section II provides details of the proposed algorithm. In Section III the architecture of the target MAV platform is described. Experimental results are presented in section IV and section V presents our conclusions.

II. METHODOLOGY

In this section, the methodology used for extracting the planar surfaces from the environment is described. First there is a brief description of NCut and then the improved weighting function is presented that is used in the study. The algorithm is given in the later part of the section.

A. Normalized Cuts

The segmentation of a point cloud can be modeled as a graph partitioning problem. Each point in 3D space is modeled as a node V_i and an edge E_{ij} is formed by connecting two nodes in a weighted undirected graph $G(V,E)$. The weight for each edge is the function of the distance from one point to the other point in a pair with small distance resulting in larger weights. This represents a function of similarity between the nodes/points. We want to segment the point cloud into disjoint set of points or clusters $V_1, V_2, V_3, \dots, V_m$ such that the similarity of nodes in V_i is maximized and similarity across the nodes of V_j is minimized. According to the NC (Normalized Cut) algorithm, the optimal bipartition of a graph into two subgroups A and B is obtained by minimizing the Ncut value as follows:

$$(1) \quad Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)}$$

where $cut(A,B) = \sum_{(u \in A, v \in B)} w(u,v)$ represent the dissimilarity between A and B and $w(i,j)$ is the similarity between the node i and j . $assoc(A,V)$ is the total connection from nodes of A to all the other nodes and $assoc(B,V)$ is the total connection from nodes of B to all other nodes. Let x be an $N=|V|$ dimensional vector for a partition that divides graph V into two sets A and B such that $x_i = 1$ if i is in A and -1 otherwise. Let $d_i = \sum_j w(i,j)$ be the total connections from node i to all other nodes. According to [9] an approximate discrete solution to minimize $Ncut(A,B)$ can be obtained by solving the following equation:

$$(2) \quad \arg \min_x Ncut(x) = \min_y \frac{y^T(D-W)y}{y^T D y}$$

Where $D = \text{diag}(d_1, d_2, d_3, \dots, d_n)$, $d_i = \sum_j w(i,j)$, $W = [w_{ij}]$, and $y = (1+x) - (1-x) \frac{\sum_{x_i > 0} d_i}{\sum_{x_i < 0} d_i}$. Using $y \in \mathbb{R}$ (2)

can be solved by the general Eigen value system:

$$(3) \quad (D - W)y = \lambda D y$$

B. Planar Dissimilarity

The core of NCut is the representation of the association among the points. This is done using a weighting function

describing the similarity/dissimilarity of the data. As mentioned in [9], this function is application specific; hence it needs to be modified in order to work on the target point cloud data. The association function in its original form representing the weights for each edge is given by [9]:

$$(4) \quad W = e^{-\left(\frac{d_k}{\sigma}\right)^2}$$

where d_k is the distance from node i to node j and σ is the scale factor taken as 5-10 percent of the maximum distance in the original work. After analyzing the histograms of distances for some points it becomes clear that 'w' needs to give more weight to the closer points than the points that are far away. This requires a modified association function with adaptive for each node. Sample histograms showing the distances from a point to all other points are depicted in the figure 1.

In order to get better clustering for 3d pointcloud, an updated weighting function ' w_p ' is used which can emphasize closer points more than ones that are far away. This improves the effect of minimization in the Ncut technique. The new weighting function with median μ taken as the center of gravity of each node is given in the (5).

$$(5) \quad W_p = e^{\left(\frac{-\mu(2d_k - \mu)}{\sigma_p^2}\right)}$$

where $p = k$ with ' $k=0.25$ ' is used in this study. This makes adaptive for each node, which helps in reshaping the weighting function according to the input data.

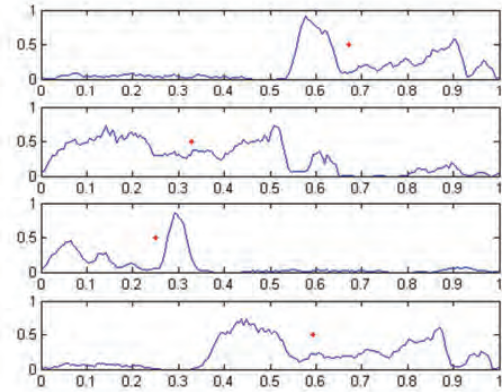


Figure 1: Normalized histograms of some sample graph nodes with respective center of gravities.

The normalized histogram of distances between one node to the other nodes can be seen as indicator of node's association to the other nodes. The center of gravity of the normalized histogram also indicates the center of point clouds mass seen from a node position. The desired weight function would emphasize the importance of the nodes belonging to a cluster and also would separate the outliers nodes. On the other hand the range of distances for a given node to other nodes varies due to the change in the respective neighborhood. For such a variation in the data nodes, a fixed function as was being used earlier, can fail to

distinguish between smooth edges thus resulting in incorrect segmentation of the interconnected or occluded objects in the scene.

The main improvement of the weight function lies in the scale invariant property. It can be observed from the figure 1, that the range of a distances for a given node varies due to the change in the respective neighborhood. However, the weighting function itself is scalable function which is adaptively scaled depending on the center of gravity. For example, 'wp' would generate more amplified response for node 1 than it would generate for node 3 of the data. This results in an overall segmentation that is more sensitive to the variation in the small changes by enhancing the effect of minimization in the NCut which is dependent on the individual weights.

C. Surface Estimation

In this section we describe the process of estimating the planarity of each cluster. For 'K' clusters obtained by NCut we identify those clusters that have a planar tendency. In order to achieve this, Least-Square Plane (LSP) fit is applied to the point normals which are calculated for each point by taking 'K' nearest neighbors and finding a surface normal made by the point with its neighbors. Suppose the normal of the LSP is $\vec{P} = (p_x, p_y, p_z)$ and the error associated with plane fitting is $E = \sqrt{\sum_{i=1}^N d_i^2 / N}$ where 'N' is the number of points in the cluster and d_i is the distance of the point $p(x_i, y_i, z_i)$ and the plane. The minimization of the error can then be done by using Singular Value Decomposition (SVD) of the following matrix 'S':

$$(6) \quad S = NN^T$$

where $N = (\mathbf{x} - \mu_x, \mathbf{y} - \mu_y, \mathbf{z} - \mu_z)$ with $\mathbf{x} = (x_i)^T, \mathbf{y} = (y_i)^T, \mathbf{z} = (z_i)^T$ being the x, y and z coordinates of the point normal of the data and $\mu = (\mu_x, \mu_y, \mu_z)^T$ is the centroid of the data normals. The eigenvectors with Eigen values $\lambda_1, \lambda_2, \dots, \lambda_N$, are then calculated. The normal vector $\vec{P}$ can be equated with the Eigenvector with the minimum eigenvalue and the plane fit error can be equated with $\sqrt{\lambda_{min}/N}$. The value of E gives a fast estimate of whether a given cluster is planar or not. If $E \leq$ then the cluster is considered to be planar, where is taken sufficiently close to zero.

D. Post Processing

The planar clusters can have some part of the noise that passes through the plane extraction threshold. As it can be seen in the figure 5(c). These scattered noisy points in the planar clusters should be removed so that they might not affect later processing. In order to remove these points, a local variance based pruning method is implied. Each point is classified as data or noise based on its spread from the 'k' nearest neighbors. The result of this pruning method is shown in the figure 5 (e).

E. Planar Surface Extraction Procedure

In this section we describe our novel PLASE (PLANar Surface Extractor) method. In order to extract planar clusters from the point cloud, the cloud is segmented using

Normalized Cuts [9] with a new weighting function as described in the previous section. Each cluster is processed for identification of its planarity nature and candidate clusters are obtained. This planar data is filtered for possible noise, which then would provide the input to RANSAC (RANDOM Sample Consensus) that would fit a planar model within the selected clusters although application of RANSAC is not being done in this study.

- Set $k=2$
- While(true)
 - Extract 'k' clusters C_i for $i=1,2,\dots,k$
 - For each C_i form a set of planar segments $P = \{C_1, C_2, \dots, C_m\}$ such that $E_i <$ where $m \leq k$.
 - Apply post processing filtration for each $P_j, j=1,2,\dots,m$ in order to remove the noisy data points.
 - $\Delta s = s_{k+1} - s_k$ where $s_k = \sum \text{var}(C_i)$
 - If $(\Delta s \leq t)$ terminate else $k = k+1$

III. SYSTEM SPECIFICATION

The proposed algorithm is designed to be used on a MAV platform that has been customized as part of the study. The MAV system consists of 6-rotor copter with obstacle avoidance and vision processing capabilities. The architecture of the system is depicted in figure 2. A PrimeSense 3D camera has been mounted on the MAV to provide point cloud data. An on-board CPU with 1.3GHz processing speed is responsible for the vision processing tasks. Lower level obstacle avoidance is performed by a microcontroller that is responsible for generation of the control signals to the MAV's flight controller. The microcontroller uses input from 6 ultrasonic sensors chained together to generate the appropriate control (Yaw, Pitch, Roll) commands to change the state of the MAV.

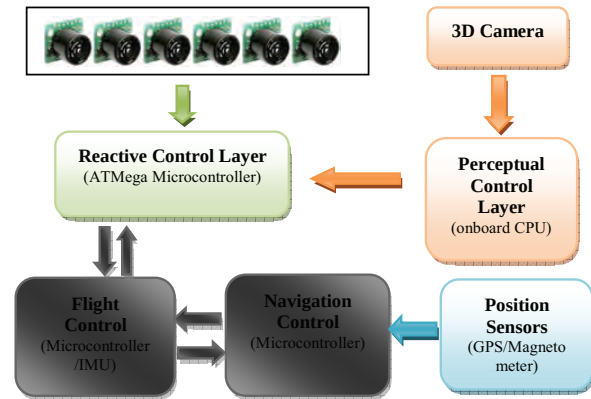


Figure 2: Architecture of the MAV platform with sensor integration and decision making layers.

The reactive control layer is a PID controller which is responsible for the low level decision making for the absolute obstacle avoidance. It also has a higher priority over perceptual control layer. The auto pilot commands are generated as the result of visual algorithm and are sent to the reactive layer which makes decision of forwarding

these commands to the flight controller by observing the signals received from the distance sensors. In case of near obstacle, reactive control generates its own pilot commands to keep the vehicle at a safe margin.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The proposed algorithm is applied on a set of point cloud data obtained from a 3D PrimeSense camera and the results are visualized. The results of clustering with original and improved weighing functions are depicted in the figure 3 with each cluster in distinct color. These clusters are then tested for their planarity nature using the method explained in the section II and the results can be visualized in the last row of the figure 3. The clustering result has been improved by the use of W_p which enhances the cuts in the graph of the nodes. Adaptivity in the calculation helps to better represent the locality of each node and hence ensures the relative cluster spread. This also makes the method invariant to changes in data due to view point change or sensor noise, although it has been observed that the association function is not invariant across different scales of the data. In this study, each point cloud is down sampled to the same degree and hence every point cloud has the same data size.

In order to validate the algorithm, a simulation model containing a set of planes and noise has been constructed. The planar point cloud model data is injected with Gaussian noise and some random structures to mimic the natural scene as shown in figure 4. This point cloud model is used to generate various indoor scenes containing different numbers of planes and noisy objects, and the performance of the algorithm is measured. The scenes created from the model along with their respective results can be visualized in the figure 5.

For each P_j $j=1,2,...,m$ cluster that results from the application of the algorithm, an error rate ϵ_j is measured that reflects the clustering performance of the proposed method over multiple scenes. Each cluster of the ground truth is assigned to a class that is most frequent in the resultant cluster. Then the error rate of this assignment is measured by counting the number of correctly assigned points:

$$(7) \quad \epsilon(\Omega, \mathbb{C}) = 1 - \left[\frac{1}{n} \sum_k \max_j |\omega_k \cap c_j| \right]$$

where $\Omega = \{\omega_1, \omega_2, \omega_3, \dots, \omega_k\}$ is the set of clusters and $\mathbb{C} = \{c_1, c_2, c_3, \dots, c_j\}$ is the set of classes.

The error for every scene is calculated 10 times with different spread of the noise each time and the average error for each scene is taken. The evaluation results of each scene with different weighting functions are depicted in the table 1. This reflects the average error rate over multiple runs of the same scene with varying noise () density of the scene. The effect of sigma on error rate for each scene is depicted in the Appendix A1.

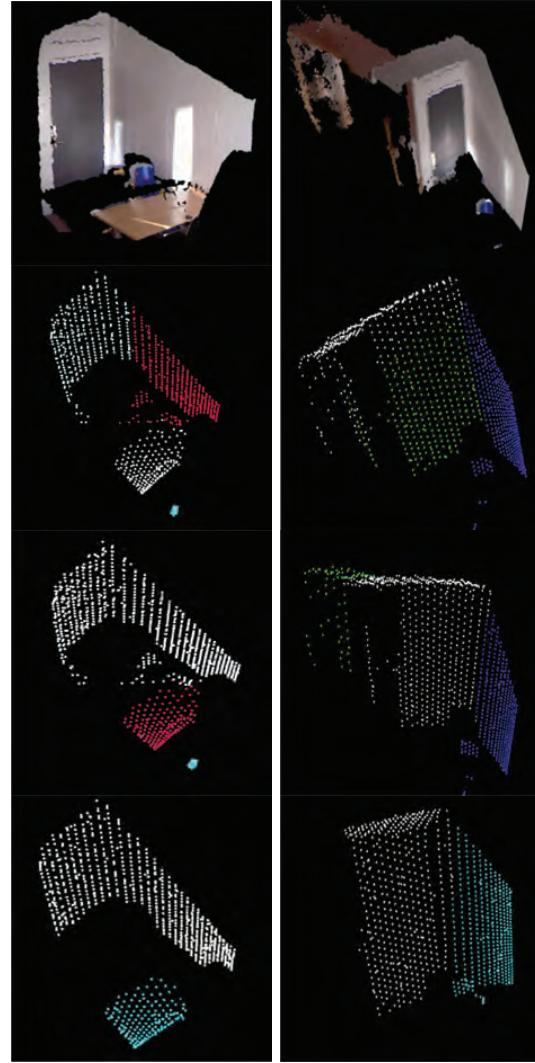


Figure 3: First Row: High resolution point cloud of original scenes. Second Row: clustering using 'W' Third Row: Clustering using ' W_p ' and Fourth Row: The clusters from the third row are then tested for their planarity nature to obtain planar clusters as mentioned in section IIC.

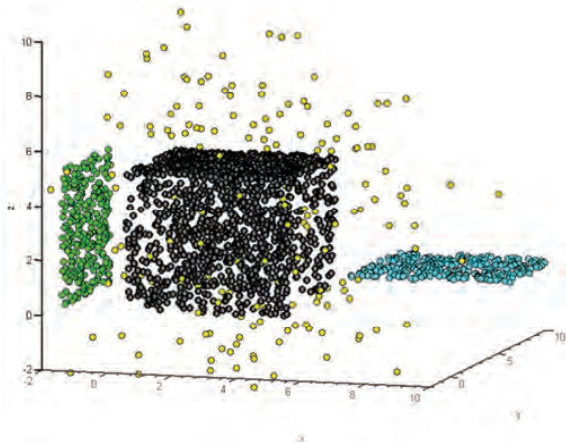


Figure 4: A simulation model of the environment.

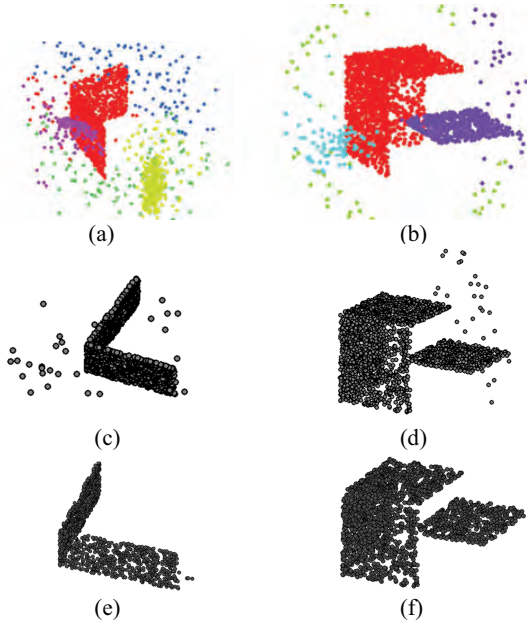


Figure 5: (a)-(b) clustering using simulated scenes. (c)-(d) Plane extraction result on different scenes, (e)-(f) Result of pruning..

The precision and recall are widely used performance metrics for evaluating the correctness of a detection algorithm. They can be considered as an extended version of accuracy that is a simple metric which computes the fraction of instances for which correct result is obtained. In order to compute the precision and recall of the proposed system, labels for a given point is divided into two classes: planes and, non-planes or noise. Recall is then computed as the fraction of correctly classified points as planes among all points that actually belong to the planar class. While precision is the fraction of correctly classified planar points among those that the algorithm believes to belong to the planar class. The relationship of precision and recall can be observed in the figure 6. A high recall means that we haven't missed any plane in the scene but we might have unwanted points in the planar clusters. High precision means that everything detected was planar but we might not have covered all the planar points. Formally, the precision for a given class is the ratio of number of true positive to the total number of elements labeled as belonging to the planar class. Recall in this context can be defined as the ratio of true positives to the total number of elements that actually belong to planar class. So, the precision and recall of the system can be written as following:

$$(8) \quad P = \frac{\tau^+}{\tau^+ + \psi^+}, R = \frac{\tau^+}{\tau^+ + \psi^-}$$

where τ^+ is the number of planar points that are correctly classified as planes and ψ^+ is the number of non-planar points that are classified as planes. Whereas the ψ^- is the number of planar points that are classified as non-planes.

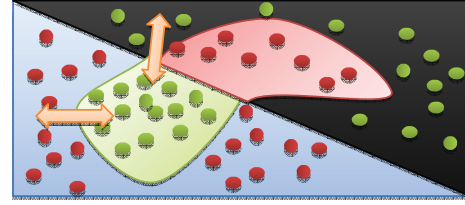


Figure 6: Plane: blue triangle, Noise: grey triangle, τ^+ : green polygon in blue triangle, τ^- :red polygon in grey triangle, ψ^+ : green points in grey triangle, ψ^- :red points in blue triangle, Precision: vertical arrow, Recall: horizontal arrow.

A measure that combines the precision and recall is called harmonic mean or the F-measure. We used F-score as a combined measure in order to calculate the plane extraction performance of the algorithm. The general F_α measure can be given as following:

$$(9) \quad F_\alpha = \frac{(1+\alpha^2).P.R}{\alpha^2.P+R}$$

Where α is the weight of precision to recall. F_1 is the balanced F-score because precision and recall are equally weighted.

Table 1: Performance of the algorithm using surfaces from simulated scenes. Clustering error is calculated as according to equation 7 and planar cluster estimation column depicts the F-score.

Scene	Avg Clustering Error ($\sum \epsilon / n$)		PLASE(PLAnar Surface Extractor) $F_\alpha = \frac{(1+\alpha^2).P.R}{\alpha^2.P+R}$		
	$(D - W_p)y = \lambda Dy$	$(D - W^-)y = \lambda Dy$	$F_{0.5}$	F_1	F_2
1	0.25	0.44	0.93	0.87	0.82
2	0.12	0.46	0.86	0.86	0.87
3	0.03	0.38	0.97	0.95	0.94
4	0.10	0.25	0.40	0.36	0.30
5	0.05	0.37	0.85	0.80	0.77

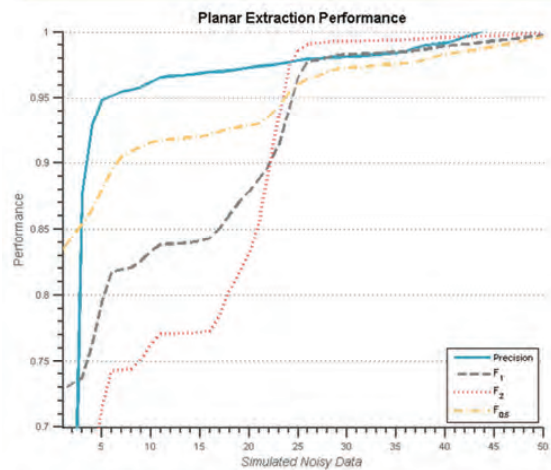


Figure 7: Performance of planar surface extraction. Horizontal axis represents the individual instances of the data for each scene with varying noise and Vertical axis represents the classification performance.

The results indicate the tendency of the algorithm for successfully finding planes in the environment even in the presence of noise and other non-planar objects. As it can be observed in the figure 7 which depicts the performance of the proposed method on the simulated data over multiple runs of the five scenes, that for most of the data instances, the performance (F-score) of the method remains above ninety percent. The proposed plane extraction method is also able to classify the scenes consisting of planar surfaces with good precision rate, as can be observed in the figure 7.

V. CONCLUSIONS AND FUTURE WORK

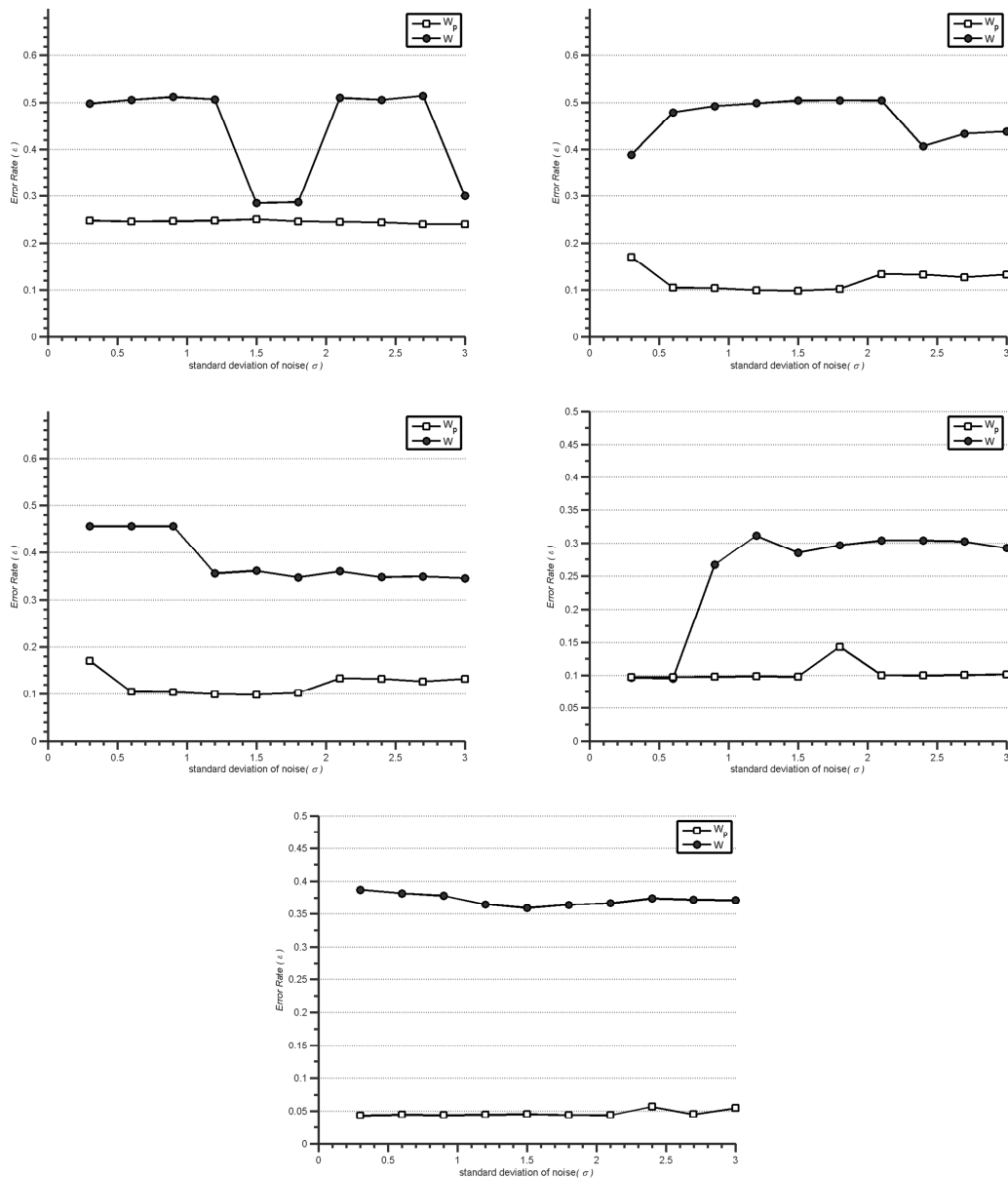
In this study we proposed a novel plane extraction method (PLASE) that can be used to localize a MAV in an indoor environment. The results suggest that this method performs well even in the presence of noise and non-planar objects, and that it should be effective for indoor navigation of a MAV. The performance of the algorithm is dependent on the success of the individual steps, which means that wrongly segmented data can lead to bad planes. This can be a potential problem that could be overcome by using adaptive clustering in NCut. Due to time constraints, a simple iterative cluster number estimation method was used; this will be improved in future work.

The method is implemented in MATLAB and is not real time; the method will be re-implemented in C and for the MAV platform as part of the future work.

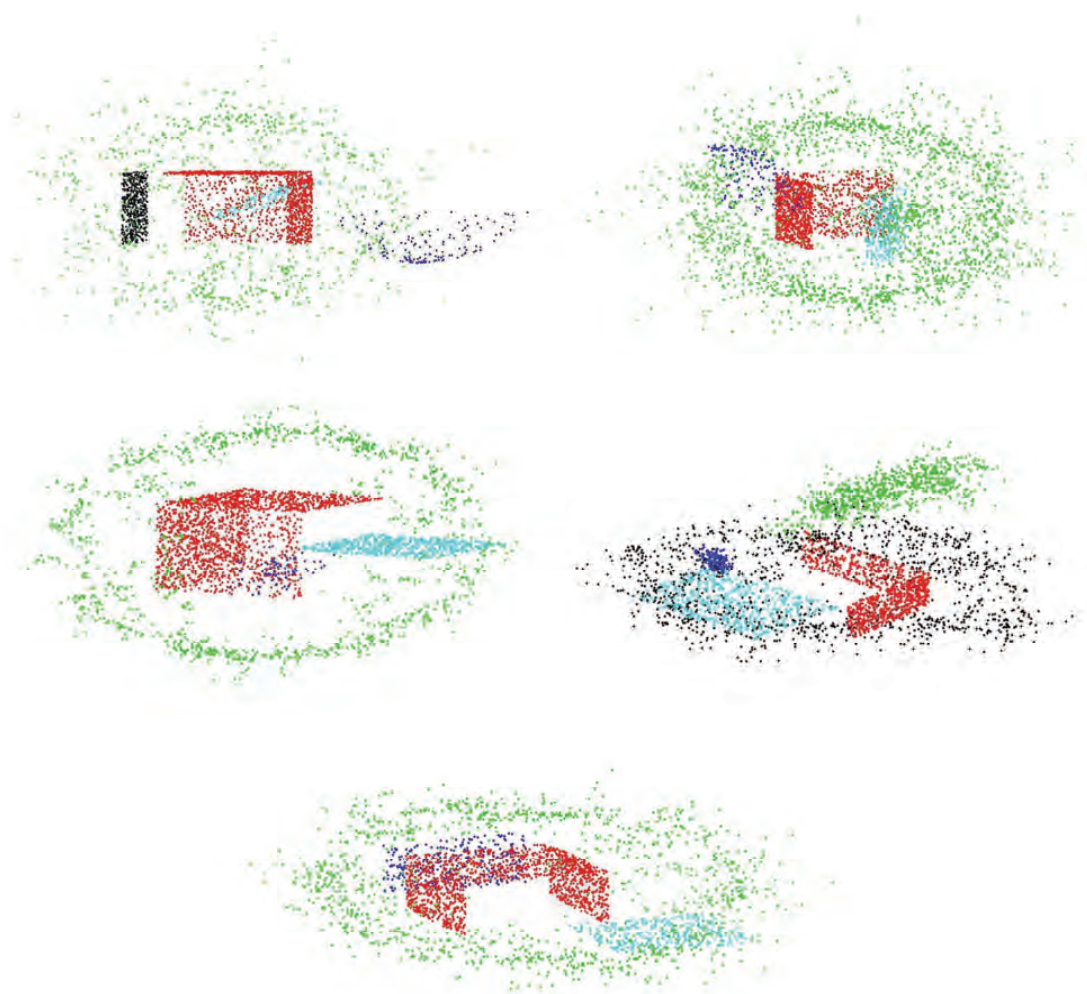
REFERENCES

- [1] D. Kragic and M. Vincze, "Vision for Robotics," *Foundations and Trends in Robotics*, vol. 1, 2010, pp. 1-78.
- [2] J. Kim, L.O. Lee, E. Nettleton, and S. Sukkarieh, "Decentralized approach to unmanned aerial vehicle navigation: Without the use of the global positioning system and preloaded maps," Professional Engineering Publishing Ltd., 2004, pp. 399-416.
- [3] M. Achtelik, A. Bachrach, R. He, S. Prentice, and N. Roy, "Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments," *Unmanned Systems Technology XI, 14-17 April 2009*, USA: SPIE - The International Society for Optical Engineering, 2009, p. 733219 (10 pp.).
- [4] I. Dryanovski and W. Morris, Jizhong Xiao, "Multi-volume occupancy grids: An efficient probabilistic 3D mapping model for micro aerial vehicles," *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010), 18-22 Oct. 2010*, Piscataway, NJ, USA: IEEE, 2010, pp. 1553-9.
- [5] F. Kendoul, Z. Yu, and K. Nonami, "Guidance and nonlinear control system for autonomous flight of minirotorcraft unmanned aerial vehicles," *Journal of Field Robotics*, vol. 27, 2010, pp. 311-334.
- [6] A. Beyeler, J.-C. Zufferey, and D. Floreano, "Vision-based control of near-obstacle flight," Van Godewijkstraat 30, Dordrecht, 3311 GZ, Netherlands: Springer Netherlands, 2009, pp. 201-219.
- [7] T. Kanade, O. Amidi, and Q. Ke, "Real-time and 3D vision for autonomous small and micro air vehicles," *2004 43rd IEEE Conference on Decision and Control (CDC), 14-17 Dec. 2004*, Piscataway, NJ, USA: IEEE, 2004, pp. 1655-62.
- [8] G.-P.M. Hegde, Cang Ye, "Extraction of planar features from Swissranger SR-3000 range images by a clustering method using normalized cuts," *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009), 11-15 Oct. 2009*, Piscataway, NJ, USA: IEEE, 2009, pp. 4034-9.
- [9] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, 2000, pp. 888-905.
- [10] C. Cigla and A. A. Alatan, "Object segmentation in multi-view video via color, depth and motion cues," in *2008 IEEE International Conference on Image Processing, ICIP 2008, October 12, 2008 - October 15, 2008*, San Diego, CA, United states, 2008, pp. 2724-2727.
- [11] J. G. Rogers and H. I. Christensen, "Normalized graph cuts for visual SLAM," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009), 11-15 Oct. 2009*, Piscataway, NJ, USA, 2009, pp. 918-23.

APPENDIX



A1: A comparison of error rates of clustering performance using weight functions (W, W_p) for five scenes while each scene has been repeated ten times with varying noise in the simulation model.



A2: Simulated scenes depicting the construction of planar models containing Gaussian noise with $\sigma=2.1$ and some random non-planar objects.

BioMAV: bio-inspired intelligence for autonomous flight

Paul K. Gerke^a, Jurriaan Langevoort^a, Sjoerd Lagarde^a, Laurie Bax^a, Tijl Grootswagers^a, Robert-Jan Drenth^a, Vincent Sliker^a, Louis Vuurpijl^{a,*}, Pim Haselager^{a,b}, Ida Sprinkhuizen-Kuyper^{a,b}, Martijn van Otterlo^a and Guido de Croon^c

^a Department of Artificial Intelligence, Radboud University Nijmegen

^b Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour

^c European Space Agency, Noordwijk, The Netherlands

ABSTRACT

This paper aims to contribute to research on biologically inspired micro air vehicles in two ways: (i) it explores a novel repertoire of behavioral modules which can be controlled through finite state machines (FSM) and (ii) elementary movement detectors (EMD) are combined with a center/surround edge detection algorithm to yield improved edge information used for object detection. Both methods will be assessed in the context of the IMAV 2011 pylon challenge.

1 INTRODUCTION

Autonomous flight of ever smaller Micro Air Vehicles (MAVs) poses a major challenge to the field of artificial intelligence. Recent successes in autonomous flight of MAVs have been obtained on quad rotor platforms, able to carry and power miniaturized laser scanners [1, 2]. The laser scanner allows the use of well-established *simultaneous localization and mapping* (SLAM) algorithms for navigation.

For reasons of energy efficiency and a greater potential for miniaturization, there is an increasing body of research on autonomous flight of MAVs using small camera equipment. Currently, there are two main approaches to this challenge. The first approach aims to apply visual SLAM to the MAV's environment, tackling both navigation and obstacle avoidance at the same time [3, 4, 5]. A disadvantage of this technique is the computational complexity of the algorithms involved, leading to a requirement of processing power unlikely to be available on light-weight MAVs in the order of a few grams. In addition, there are still problems concerning *drift* [5].

The second approach is more biologically inspired. Typically, camera images are used to calculate the optic flow field [6, 7, 8, 9, 10]. This field is then mapped to actions that allow the MAV to avoid obstacles or navigate corridors. The biological approach is computationally more efficient. However, (artificial) optic flow requires the presence of strong texture in the environment. In addition, the biological approach typically only provides solutions for obstacle avoidance, not for more difficult behavioral tasks such as navigation.

In this article, we present a novel biologically inspired approach to the autonomous flight of MAVs. It extends existing biologically inspired methods in two ways.

The first extension concerns the behavioral capability of biologically inspired MAVs. Most work on biologically inspired autonomous flight assumes the MAV to be flying straight forward, while being corrected by the optic flow in order to avoid obstacles or navigate corridors. We aim to develop a more elaborate behavioral repertoire inspired by the interaction between modules in flying insects [11, 12, 13, 14]. These modules, understood as relatively fixed, simple functional behaviors (reflexes) based on underlying dedicated neural structures and pathways, can, through their interaction, produce quite complex adaptive behavior. Different modules may work in parallel or in sequence, under the influence of regulatory sensory inputs. We were particularly inspired by the interaction between two modules, leading to the visual reflexes [15] of object fixation and expansion avoidance, together producing a relatively straight trajectory towards a salient object without hitting it. The interaction between the modules will be set by tuning the parameters governing their interaction through artificial evolution (see, e.g. [16]). The result of our research is a control structure that will represent a small step beyond reactive control structures, in virtue of its biological grounding.

The second extension involves vision, and in particular, going beyond the sole use of optic flow. Recent studies on the visual behavior and cognition of fruit flies [17] suggest that flies actively navigate in their environment on the basis of visual appearance cues as well: they prefer to move towards long vertical structures and avoid small (predator-like) structures. In [18], a visual appearance cue was used successfully to complement optic flow, allowing the flapping wing MAV DelFly II to avoid obstacles in a wider array of environments. While tests on real flies are typically done in visually simplified environments (led arrays / black-and-white flight arenas), normally flies and real MAVs have to fly in the visually complex real world. Recognizing a long vertical structure in real-world camera images is a challenging problem. Yet, evidence suggests that flying insects like honeybees use edge detection to guide their landing behavior [19] as well as for the recognition of shapes [20]. In this paper, we will introduce a computationally efficient edge detection algorithm that combines detected edges with motion information provided by elementary movement detectors (EMDs) [21], in order to identify and track behaviorally relevant objects over time.

Both extensions to existing biologically inspired methods will be tested in context of the indoor pylon challenge, one

*contacting author: l.vuurpijl@ai.ru.nl

of the indoor competitions at the International Micro Air Vehicle conference and competitions 2011 (IMAV 2011). Our flight platform is the Parrot AR.Drone quadricopter [22], to which, because of our biological perspective, we will refer to as BioMAV. The pylon challenge requires the MAV to fly as many 8-shapes as possible around two poles in the environment. Although not in itself a challenge faced by biological systems, it is suitable to investigate the two extensions. The behavioral complexity of flying 8-shapes is higher than that of following a corridor or avoiding obstacles, making it suitable for the first extension. Moreover, vertical structures are important affordances to flying animals, making the detection of large poles suitable for investigating the second extension.

Since our hardware is fixed, we have to adapt and fine-tune our biologically inspired algorithms for a robust autonomous flight of our Drone. To avoid too many crashes and to be able to use evolutionary algorithms we have built a simulator for the Drone. An iterative strategy of testing in simulation and on the real platform will be used to obtain the best result in the real world.

The remainder of the article is organized as follows. The simulation platform is described in Section 2. In Section 3, promising results of our first explorations of simulated behaviour of our BioMAV in the pylon challenge are presented. In Section 4, we will present our new vision module, which combines the results of center/surround edge detection with dynamic information provided by elementary motion detectors. As discussed in Section 4, our first results with this combined approach are very promising. Finally, in Section 5 we conclude and outline our next research steps.

2 PLATFORM AND SIMULATOR

In this section we will first present our physical and simulated environments.

2.1 Tests on a real platform

The Parrot AR.Drone, a model-sized helicopter with four rotors [22], utilizes an embedded linux-based operating system to automatically stabilize itself when flying. The operating system allows to control the flight of the drone using abstract control commands, like “takeoff”, set the drone-pitch to let it accelerate in a certain direction, or “hover” which lets the drone automatically cancel its movement with respect to the ground. Table 1 shows a complete overview of the available commands. These commands provide an abstraction layer from physical aspects and implementation details which are required for maintaining a stable drone flight. These commands allow us to focus on the development of biologically inspired vision processing modules and high-level behavioral control layers.

2.2 Simulation and ground control

The final program that enables the drone to fly 8-shaped figures around two poles will consist of three modules (see Figure 1 for the architecture): The ground control software is

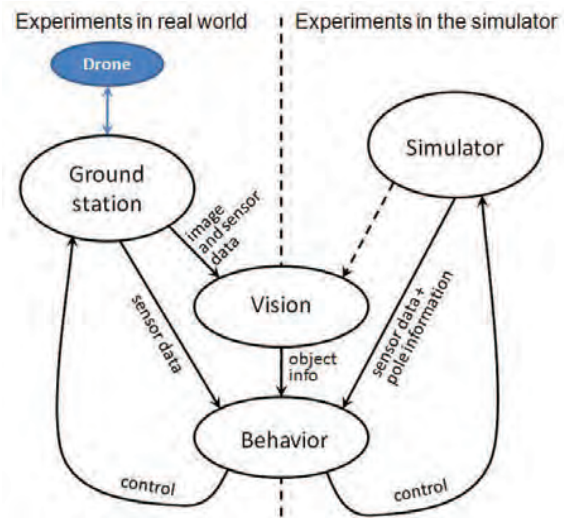


Figure 1: Architecture of our BioMAV platform

written in C++ and it directly communicates with the drone. It communicates sensory data and control commands to a Java environment. The vision module is a Java program which uses a biologically inspired algorithm to detect poles in front camera images. The positions of the detected poles are translated into relative heading and an approximated distance (see Section 4). These data are given to the third module which manages the behavior of the drone. The biologically inspired behavior module uses sensory data and the extracted data about visible poles to navigate the drone in 8-shaped figures around the visible poles (see Section 3).

The ground control software is an application programming interface that allows to send commands to the drone using Java code. Since we will not need fine-grained height control, the ground station allows to set the drone height rather than the height-speed. The other commands available for controlling the drone are modeled after the original control commands mentioned in Table 1.

We are developing the simulator in such a way that the behavior and vision modules can be developed and tested independently. In the final implementation, the vision module will provide information about poles that are visible to the drone. In the controlled world of the simulator we can calculate this information without using the vision module, thereby allowing us to test the behavior module on its own. Likewise, image data can be generated using the simulator to test the vision module. The simulator allows us to test algorithms for the drone without using the actual drone itself. The advantages of using a simulator are that the test cycles are faster and the tests inside the simulator do not endanger the available hardware. These advantages allow us, in a later step, to use the simulator as part of a program which iteratively tunes parameters of the behavior module to optimize the drone’s flight behavior. The disadvantage of using a simulator is the

Command	Effect
Hover	Lets the helicopter hover over a certain point at a constant height. For this the helicopter makes use of the bottom camera to calculate the optic flow and nullify its ground speed. Furthermore, it uses its height sensor (ultrasound sonar) to maintain a stable height above the ground.
Pitch	Makes the helicopter pitch until it reaches a given angle to control forward or backward movement. The helicopter thereby maintains a stable height. If desired this command can be combined with the <i>yaw</i> , <i>roll</i> , or <i>lift</i> commands.
Yaw	Makes the helicopter yaw around its vertical axis at a given speed. The helicopter thereby maintains a stable height. If desired this command can be combined with the <i>pitch</i> , <i>roll</i> , or <i>lift</i> commands.
Roll	Makes the helicopter roll until it reaches a given angle to control sideward movement. The helicopter thereby maintains a stable height. If desired this command can be combined with the <i>pitch</i> , <i>yaw</i> , or <i>lift</i> commands.
Lift	Makes the helicopter ascent upwards or descent downwards at a given speed. If desired this command can be combined with the <i>pitch</i> , <i>yaw</i> , or <i>roll</i> commands to let the helicopter change its height during these commands.
Takeoff	Lets the helicopter start and ascent to a standard height (about 1 meter).
Land	Lets the helicopter land on the ground.
Emergency mode	Cuts the power to all helicopter motors. Only used for safety purposes.

Table 1: Available commands for controlling the Parrot AR.Drone [22]

reality gap. We will revisit this problem in Section 3.

The simulator generates visual input corresponding to the images that the camera of the drone would generate. The simulator can provide in addition all other information the drone itself would produce. The drone's control commands are modeled after the commands that the ground station provides. We use OpenGL¹ to render the three-dimensional environment for the drone. An integrated rigid body simulation based on the OpenDynamicsEngine library² provides us with collision detection algorithms and Newtonian mechanics for simulated bodies. The reactions of the drone to the given set of control commands are specified using a script written in the scripting language Lua³. The simulated reactions are a simplification of the real behavior. The drone-behavior Lua script allows for easy adaption of the drone's behavior to make the drone react more realistically in future simulations. One of the avenues we are exploring is to train the parameters of a neural network or a Kalman filter to map the current state and commands to the resulting behavior of the drone. The reasons for this approach are that (a) the system identification of the individual components of the Parrot platform is difficult with the standard firmware, and that (b) some physical parameters may be hard to identify in the first place (like the effects of the bending of the rotor blades during flight maneuvers).

Using a second simulator Lua script, the simulated world can be modified. In this script, the layout of the simulated world can be determined by creating walls, placing poles, or setting the initial position of the drone. The future goal is to expose a Lua interface to Java so that arbitrary Lua scripts can

be executed from Java. Thereby the Lua scripts can control the simulated world or the drone inside it. We plan to release the simulator as an Open Source project after the competition.

3 BEHAVIOR

Many behaviors can be accomplished by means of feedforward, reactive controllers [23, 24]. Indeed, in the field of evolutionary robotics, most studies focus on the evolution of feedforward neural networks for successful control of the robot [25, 26]. When moving towards more complex behaviors, recurrent neural networks of various kinds are used [24, 27, 28, 29, 30, 31]. These systems are hard to analyze and to correct in cases where the displayed behavior differs from the behavior in simulations or when the performance in operational settings decreases compared to the training conditions. Our ongoing research explores Finite State Machines (FSM) [32] as an alternative framework for the design of complex behaviors. FSM are mathematically well-understood and have the additional advantage that they are easier to understand by a human designer. FSMs can combine different behavioral modules to achieve the required behavior. In the current article, the IMAV2011 pylon challenge is used as a vehicle to introduce our approach of exploring different simulated behaviors implemented through FSMs.

An abstract FSM schema as used in the BioMAV drone, containing *states* and *transitions*, is depicted in Figure 2. Rounded boxes represent states and arrows represent state transitions. A state describes, at a general level, the type of behavior displayed by the drone. Transitions describe the conditions that need to be met in order to move from one state to another. Only one state can be active at a time.

¹<http://www.opengl.org>

²<http://www.ode.org/>

³<http://www.lua.org>

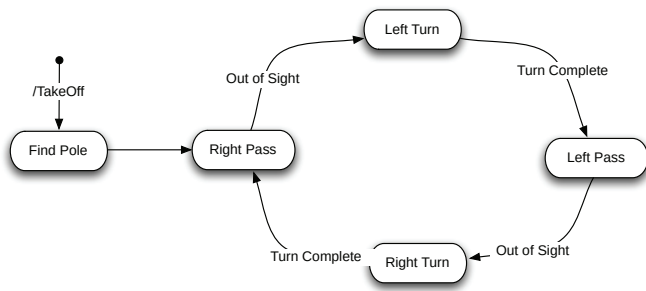


Figure 2: FSM schema for 8-shaped flight.

After take off, the drone enters the ‘Find Pole’ state. Upon finding a pole, the transition to the ‘Right Pass’ state is made. In this state the drone flies towards the right of a pole. When the pole is out of sight, the ‘Left Turn’ state will become active. Here the drone will make a left turn around the pole, completing the first half of the 8-shaped figure. There is no final state; the drone will continue to fly 8-shaped figures.

Each state of the FSM executes three ‘sub-behaviors’: *entry behavior*, *state behavior*, and *exit behavior*. Entering and exiting a state results in executing, respectively, the corresponding entry or exit behaviors. In the current implementation no entry behaviors are specified. All states have an exit behavior that makes the drone hover at its current location as to stabilize the vision input. The state behavior makes the drone execute a specific sequence of commands, e.g. processing of sensory data. A state can have multiple transitions associated with it, each described by its own set of criteria (triggers, guards) that need to be met for the transition to become active. Each transition can also be equipped with a behavior that will be executed when the transition is active. In the case of the ‘Out of sight’ transition, for example, the transition behavior is that the drone will fly a bit further forward in order to make a safe turn.

3.1 First FSM flight simulations: timers and gyroscope

The behaviors of the first version of the FSM were based on timers. Timing parameters determined the amount of turn time to fly around a pole and the time to fly in a straight line. Our simulation results without noise factors such as drift and vision failures showed the feasibility of our approach. To test the performance of the drone in a more realistic setting, Gaussian noise was added to the horizontal and vertical speeds ($\mathcal{N}(0, 0.10)$ in m/s) and to the initial position ($\mathcal{N}(0, 0.7)$) of the drone.

Figure 3 shows a simulation where the drone flies in an unstable, jagged trajectory. The first part of the 8-shaped figure is completed without problems. On the second turn, however, the drone loses its path and misses the next pole.

To explore our approach in more realistic noisy conditions, the use of simulated gyroscope readings was evaluated as an alternative method. These readings provide feedback on the turning angle when performing a “Left Turn” or “Right

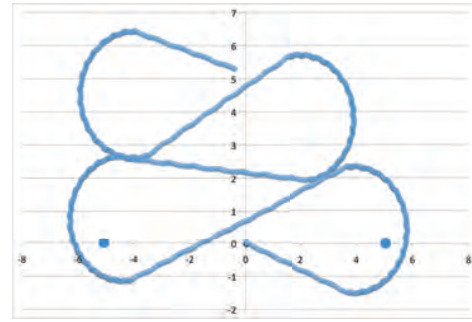


Figure 3: Simulated trajectory of the drone, using additive Gaussian noise on the horizontal/vertical speeds and the starting position of the drone. Behavioral modules are based on timers. The drone starts at position (0,0) and the poles are located at (-5,0) and (5,0).

Turn”. The results as displayed in Figure 4 show that using gyroscopes is a more robust method than using timers.

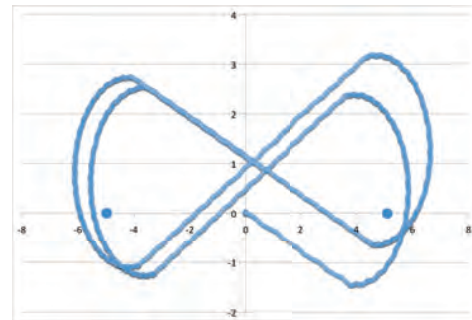


Figure 4: Simulated trajectory of the drone, using additive Gaussian noise on the horizontal/vertical speeds and the starting position of the drone. Control of behavior is based on gyroscope readings.

3.2 FSM flight: closed-loop control with simulated vision

A second, more complex, FSM was used to incorporate simulated vision (see Figure 5).

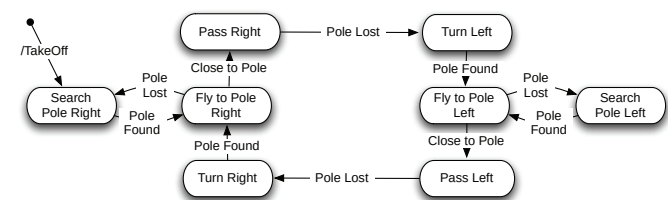


Figure 5: The structure of the improved FSM.

The simulator acts as a stand-in vision module, by providing (i) the location of and (ii) the distance to the pole with random noise and frame-loss. The position of the pylon was used to determine the angle of approach relative to the drone. Based on this information, the new FSM controls three transitions, which each use a different threshold

to determine whether one of these conditions hold: “Pole Found”, “Close To Pole” and “Pole Lost”. Each setting of these three thresholds results in a certain flight behavior (trajectory) which can be assessed quantitatively by comparing the generated trajectory to the pre-determined 8-shaped figures. The required thresholds are set through an evolutionary algorithm (EA) which uses these quantitative measures as fitness value. Again, several noise factors are added to explore the robustness of the FSM when transferred to the real world.

In Figure 6 the new flight paths of the drone are depicted. Because the behavior is now more sensor driven, the drone is able to correct its path and repeatedly fly in an 8-figure around the poles. Even when the drone lost the pole, it is able to correct its path by including an extra loop in its flight.

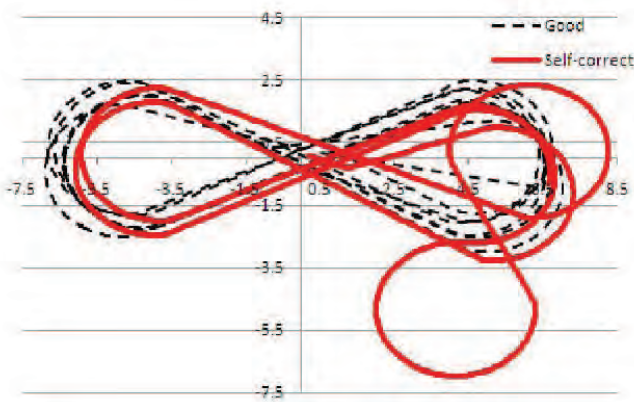


Figure 6: Simulated trajectory of the drone, using the sensory driven FSM. The trajectories are from two separate simulations: one (black, dashed) in which everything went smoothly and one (red) in which the drone corrected its path by flying an extra loop.

It can be concluded that using FSMs for controlling the drone's behavior provides us more insights in which parameters are important for executing a certain behavior. Moreover, the use of a genetic algorithm for searching for suitable parameter settings shows promising results.

3.3 From simulated experiments to real flight

The fitness function used for the EA mentioned above employs a quantitative measure which is based on the difference between the required trajectory and observed *simulated* behavior. Many algorithms for trajectory matching can be employed for this purpose, e.g., borrowed from the field of gesture recognition [33]. Using the high-resolution tracking equipment from our virtual reality lab ⁴, the real flight behaviors of our BioMav can be recorded. We have created a setup that allows for an evaluation whether behaviors of the drone in simulations can be compared to trajectories recorded during real flight. Please consider Figure 7 for an example.

⁴See <http://www.rivierlab.nl>

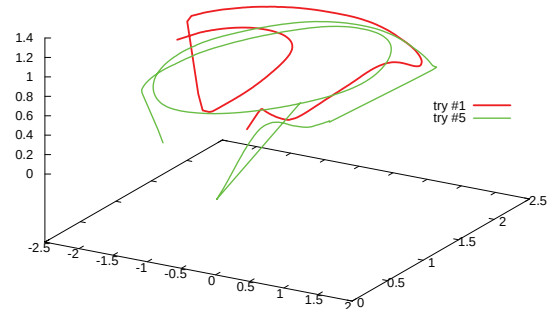


Figure 7: Picture of the drone with rigid body attached (top picture). The bottom picture shows the 3-dimensional flight patterns of two short tries in which the drone attempted to fly in circles.

To record the three-dimensional orientation and position of our BioMAV, a light-weight rigid body is attached on top of the drone. This has no significant impact on the drone's flying behavior and can be tracked by the tracking equipment of the VR lab. The bottom picture from Figure 7 shows the recorded flight trajectories of the drone while trying to complete circles. The drone is controlled via the ground station software which was described in Section 2.2. We are currently using this set up to improve our FSM models for the IMAV2011 pylon challenge.

4 VISION

It is well known that natural vision systems fuse different information sources to obtain more robust and accurate estimates concerning objects in the environment. In our ongoing research, appearance and motion cues are explored for extracting information about the color, location, movement and shape of objects in the environment [34, 35, 36]. In the IMAV2011 pylon challenge, the main task for vision is to detect poles and return the distance and angle between pylon and drone. Traditionally, in computer vision, this task is achieved by image segmentation, which transforms a raw image representation in sets of detected objects. Many approaches for image segmentation have been proposed, which all use representations of texture, color, and shape to some extent. Since the quality of the contours of detected objects relies heavily on the detection of relevant boundaries, or edges, in the image, edge detection is a prominent step in image segmentation. We propose to combine edge detection with

motion information provided by elementary movement detectors (EMDs) [21]. EMDs use spatially separated inputs with a certain delay in time to produce a measure for the motion in a specific direction. The use of EMDs is especially useful in UAVs, since the flying task induces temporal and motion effects (which are known to cause the detection of spurious edges), ensuring that there is always activation from the EMDs. During the competition, we will apply an object recognition method on the resulting edge images to detect the size and location of the pole in each image. We will consider available techniques such as the generalized Hough transform for this purpose. The resulting information will subsequently be used by the FSM described in Section 3 to generate appropriate control commands based on the angle and distance to detected poles.

4.1 EMD-based filtering of spurious edges

To generate motion information, we have used the EMD implementation of Zhang et al. [37]. This implementation is less dependent on differences in contrast and color. The idea behind the use of EMDs to improve the image segmentation, is that borders of relevant objects will produce higher EMD responses. Rotational movements of the drone will lead to edge enhancements by the EMDs. Translational movements of the drone will lead to larger apparent motion of the objects closer to the drone, which are typically more relevant. This holds in particular for a flying platform that is constantly moving. The constant self motion will provide a steady and reliable source of information about the objects in the scene.

Although very useful, the EMDs rarely give a complete picture, because they rely on differences in contrast to activate. Therefore the EMDs alone are not enough to successfully segment the image and to provide the drone with sufficient information about the world. However, EMDs do provide an abundance of additional information to the more traditional approach.

Next to the resulting EMD information sketched above, our algorithm uses the edges (as provided by a simple center/surround edge detection algorithm) as a basis on which to work. Edge detection assumes a difference in contrast and color, where the boundaries of objects can in general be distinguished from the background. However, edges may also be detected in image samples containing rough textures or areas with uneven terrain. These edges are known as spurious edges. The EMDs are less dependent on contrast differences for their activity and depend more on movement. As can be observed in the bottom-left picture from Figure 8, this produces clean edges of objects in the direction of the movement. By combining the edge and EMD information, noise can be removed from the edge information, producing a clean image of the relevant edges.

4.2 Results of combined edge detection

Figure 8 shows a natural scene (data recorded in the virtual reality lab), the EMD information of this scene and both

the edge information and the combined EMD and edge information. As can be observed, the use of EMD information can be used to filter out much of the noisy edges. Large planes such as floors, walls and ceilings are prone to produce noisy data. Yet if they are far away, they have very little observed movement, which allows us to confidently remove many superfluous edges. If they are close by, such as the floor can be, fine texture is typically smoothed out by motion blur, again leading to low EMD activity.

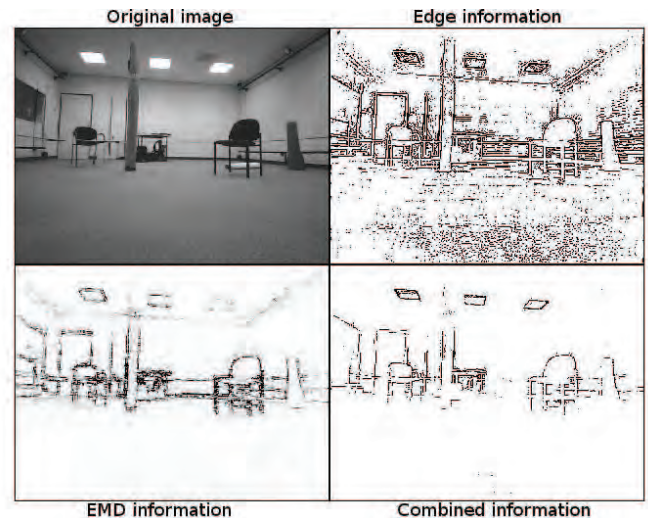


Figure 8: Example of using EMD for removing spurious edges.

To quantify the improvement of this combined edge detection method, the difference between edge detection with and without the use of the EMD information is computed for each pair of pixels in both edge images. The two methods are compared to manually segmented images of realistic input captured by the frontal camera of the drone in our VR lab. Contours in the segmented groundtruth images are manually produced and subsequently blurred by a small Gaussian filter to allow for small variations in the location of edges. In this way 10 images have been processed. Due to the Gaussian brush width, a relatively large number of pixels was marked as “edge”: on average 24% of the pixels are labeled as positive. Table 2 shows the confusion matrices of both algorithms averaged over the images (expressed in as ratios).

	Plain edge detection		Combined with EMD	
	true	false	true	false
positive	0.96	0.86	0.67	0.30
negative	0.14	0.04	0.70	0.33

Table 2: Confusion matrices of plain edge detection and of its combination with EMD.

When considering Table 2, the effect of filtering spurious edges becomes apparent. The combined technique improves

both the false positives (where the edge detection erroneously detects edges) and the true negatives (where the edge detection correctly detects no edge). However, also the number of true positive edges decreases because of the filtering. In Figure 8 this effect is in particular visible at the top of the pole. While plain edge detection correctly classifies 34% of all pixels, the combination with the EMD leads to correct classification of 69% of the pixels. Our current efforts are targeted at the implementation of suitable object detection techniques which operate on the resulting improved edge images. Within the context of the pylon challenge, we will further explore the trade-off between removing spurious edges and losing parts of the target object.

5 CONCLUSIONS

In this paper, we have introduced our ongoing research on biologically inspired MAVs (BioMAVs). Two new approaches have been discussed. For the FSM-based control of behavioral modules, the feasibility of our approach was assessed and promising results have been obtained. Furthermore, the novel edge detection algorithm which exploits motion information provided by EMDs, has shown to yield much cleaner edge images than when using traditional edge detection techniques. Our main conclusions and next steps are described below.

The FSM controller structure was inspired by the presence of behavioral modules in natural systems. As outlined in Section 3, finite state machines offer a suitable framework for implementing such modules for controlling different behaviors. Results from various simulations show that the explored FSM architectures provide a mechanism for the execution of the pylon challenge. Furthermore, using this approach, the designer is able to evaluate different settings and locate and modify defects. Three experiments were presented in this paper. All experiments were run in the simulation environment which is presented in Section 2. Timer information was used as a first naive approach, which mainly assessed the feasibility of our approach. Simulations were run on gyroscope information in several noise settings, resulting in successful 8-shaped flight patterns. Furthermore, we developed an evolutionary algorithm to optimize the drone control on the basis of simulated vision. Based on closed-loop control, using the perceived distance and angle of approach to the drone, the resulting FSM was able to exhibit the required 8-flight behavior. It was demonstrated that the drone is able to correct its flight when missing a pole.

For the required object detection and tracking, we are still in the process of finalizing our vision module. We have argued that the detection of relevant edges is important in biologically-inspired vision. The results of the vision module indicate that edge detection is helped by combining motion cues with information provided by edge detection. This improved edge detection will most probably lead to fewer spurious segments in the subsequent image segmentation and ob-

ject detection processing steps.

Within the context of our BioMAV project, we will gradually shift from performing simulations to real flight control. Through controlled experimental studies in our virtual reality (VR) lab, the gap between simulation and reality can be explored. As we have shown in Section 3.3, the VR lab enables us to employ high-precision tracking equipment to record actual flying behavior of our BioMAV. The results of these experiments are expected to yield two important contributions to our work. On the short term, we will be able to systematically calibrate the drone behavior according to the environmental and task conditions of the IMAV2011 indoor pylon challenge. On the longer term, we hope to improve our understanding of issues that cause the reality gap, by establishing a proper calibration between simulation algorithms and parameters, corresponding simulated behavior, and the real flight capabilities of our BioMAV. For an ongoing report of our progress and achievements, the reader is invited to visit our BioMAV website at <http://www.biomav.nl>.

REFERENCES

- [1] A. Bachrach, R. He, and N. Roy. Autonomous flight in unstructured and unknown indoor environments. In *European Micro Air Vehicle conf. and competitions, EMACV 2009*, The Netherlands, 2009.
- [2] S. Shen, N. Michael, and V. Kumar. 3D estimation and control for autonomous flight with constrained computation. In *ICRA*, 2011. In press.
- [3] A. Davison and D. Murray. Simultaneous localisation and map-building using active vision. *IEEE PAMI*, 2002.
- [4] K. Celik, S.J. Chung, and A. Somani. Mono-vision corner slam for indoor navigation. In *(EIT 2008)*, pages 343–348, 2008.
- [5] M. Blösch, S. Weiss, D. Scaramuzza, and R. Siegwart. Vision based MAV navigation in unknown and unstructured environments. In *IEEE Int. conf. on Robotics and Automation*, 2010.
- [6] N. Franceschini, J.M. Pichon, C. Blanes, and J.M. Brady. From insect vision to robot vision. *Philosophical Trans.: Biological Sciences*, 337(1281):283–294, 1992.
- [7] F. Iida. Goal-directed navigation of an autonomous flying robot using biologically inspired cheap vision. In *(ISR 2001)*, 2001.
- [8] A. Beyeler, J-C. Zufferey, and D. Floreano. 3d vision-based navigation for indoor microflyers. In *ICRA 2007, Italy*, pages 1336–1341, 2007.
- [9] F. Ruffier and N.H. Franceschini. Aerial robot piloted in steep relief by optic flow sensors. In *(IROS 2008)*, pages 1266–1273, 2008.

- [10] A.M. Hyslop and J.S. Humbert. Autonomous navigation in three-dimensional urban environments using wide-field integration of optic flow. *AIAA Guidance, Control, and Dynamics*, pages 147–159, 2010.
- [11] E.M. Leise. Modular construction of nervous systems: a basic principle of design for invertebrates and vertebrates. *Brain Research Reviews*, 15:1–23, 1990.
- [12] M.A. Frye and M.H. Dickinson. Fly flight : A model for the neural control of complex behavior. *Neuron*, 32:385–388, 2001.
- [13] D. Floreano, J.-C. Zufferey, M.V. Srinivasan, and C. Ellington. *Flying insects and robots*. Springer, Berlin, 2011.
- [14] D. Floreano and J.-C. Zufferey. Insect vision: A few tricks to regulate flight altitude. *Current Biology*, 20(19):R847–R849, 2010.
- [15] M.B. Reiser and M.H. Dickinson. Drosophila fly straight by fixating objects in the face of expanding optic flow. *The J. of Exp. Biology*, 213:1771–1781, 2010.
- [16] P. Petrovic. Evolving behavior coordination for mobile robots using distributed finite-state automata. In H. Iba, editor, *Frontiers in evolutionary robotics*. Viena: I-Tech Education and Publishing, 2008.
- [17] G. Maimon, A.D. Straw, and M.H. Dickinson. A simple vision-based algorithm for decision making in flying drosophila. *Current Biology*, 18(6):464–470, 2008.
- [18] G.C.H.E. de Croon, E. de Weerd, C. de Wagter, B.D.W. Remes, and R. Ruijsink. The appearance variation cue for obstacle avoidance. In *ROBIO*, 2010.
- [19] M. Lehrer, M.V. Srinivasan, and S.W. Zhang. Visual edge detection in the honeybee and its chromatic properties. *Proc. of the Royal Society B: Biological Sciences*, 238(1293):321–330, 1990.
- [20] Natalie Hempel de Ibarra and Martin Giurfa. Discrimination of closed coloured shapes by honeybees requires only contrast to the long wavelength receptor type. *Animal Behaviour*, 66(5):903 – 910, 2003.
- [21] W. Reichardt. Evaluation of optical motion information by movement detectors. *J. Comp. Phys. A*, 161:533–547, 1987.
- [22] S. Piskorski. *AR.Drone Developer Guide SDK1.5*, 2010.
- [23] S. Nolfi. Power and limits of reactive agents. *Neuro-computing*, 42:119–145, 2002.
- [24] M. van Dartel, I. Sprinkhuizen-Kuyper, E. Postma, and J. van den Herik. Reactive agents and perceptual ambiguity. *Adaptive Behavior*, 13(3):227–242, 2005.
- [25] F. Ruini and A. Cangelosi. Extending the evolutionary robotics approach to flying machines: An application to mav teams. *Neural Networks*, 22:812–821, 2009.
- [26] N. Bredeche, E. Haasdijk, and A.E. Eiben. On-line, on-board evolution of robot controllers. In *Proc. of the 9th int. conf. on Artificial evolution*, pages 110–121, 2010.
- [27] K. Stanley and R. Miikkulainen. Efficient reinforcement learning through evolving neural network topologies. In *Proc. of the Genetic and Evolutionary Computation Conf.*, 2002.
- [28] C.F. Juang. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. *IEEE Trans. on Systems, Man, and Cybernetics*, 34(2):997–1006, 2004.
- [29] Stefano Nolfi and Davide Marocco. Evolving robots able to integrate sensory-motor information over time. *Theory in Biosciences*, 120:287–310, 2001.
- [30] R.D. Beer. On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509, 1995.
- [31] F. A. Gers and J. Schmidhuber. LSTM recurrent networks learn simple context free and context sensitive languages. *IEEE Trans. on Neural Networks*, 12(6):1333–1340, 2001.
- [32] V. Sklyarov. Hierarchical finite state machines and their use for digital control. *IEEE Trans. on Very Large Scale Integration Systems*, 7:222–228, 1999.
- [33] D. Willems, R. Niels, M. van Gerven, and L. Vuurpijl. Iconic and multi-stroke gesture recognition. *Pattern Recognition*, 42(12):3303–3312, 2009.
- [34] Reid R. Harrison. *An Analog VLSI Motion Sensor Based on the Fly Visual System*. PhD thesis, California Institute of Technology, Pasadena, California, 2000.
- [35] Reid R. Harrison and Christof Koch. A robust analog vlsi motion sensor based on the visual system of the fly. *Autonomous Robotics*, 7:211–224, 1999.
- [36] L.F. Tammero. The influence of visual landscape on the free flight behavior of the fruit fly drosophila melanogaster. *J. of Exp. Biology*, 205:327–343, 2002.
- [37] Tianguang Zhang, Haiyan Wu, Alexanderr Borst, Kolja Kühnlenz, and Martin Buss. An FPGA implementation of insect-inspired motion detector for high-speed vision systems. In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 335–340, 2008.

Practical Aspects of Trirotor MAV Development

Andrzej Ryś, Roman Czyba, Grzegorz Szafrński

Silesian University of Technology, 16 Akademicka St., 44-100 Gliwice, Poland

ABSTRACT

This paper outlines the mechanical architecture, hardware components, and software structure for an unmanned micro aerial vehicle (MAV) trirotor. It focuses on the practical approach to the design and assembly of an aerial platform with three rotors. Furthermore, an embedded microcontroller-based navigation and control system is proposed to provide efficient angular stabilization and stable hover during the different flight conditions. The design, analysis and the validation tests have been undertaken on the experimental aerial platform.

1 INTRODUCTION

Within the last years, rapid development of small and powerful microcontrollers, access to cheap and relatively accurate inertial sensors, reliable wireless links and finally efficient control systems allowed to advance some formerly existing aerodynamic configurations that were very difficult to control by human pilots only, like quadrotors. The most popular multi rotor configuration is undoubtedly quadrotor. It became widespread thanks to its mechanical simplicity and easily understandable dynamics. However, for some applications a quadrotor might be not particularly suitable due to relatively high mass caused by four motors, which reduces the power efficiency. Therefore, the purpose of this project was to design and assemble a vehicle with the number of motors reduced to three and capable of stable hover flight. This is an multidisciplinary problem with various requirements for mechanics, hardware and software. There are multiple publications on UAVs navigation and control [1],[2] based on commercially available platforms, there is also a number of papers focusing on theoretical considerations [3], [4]. These publications do not provide comprehensive overview of the UAV assembling from concept to flight.

The main goal was to design a simple, light and durable airframe carrying embedded measurement and control system, sufficient for first tests and flexible enough to adopt easily to future development without major changes. The main intention was to decrease the number of motors to three. It is obvious that such a modification introduces certain changes to the general system dynamics and require various modifications in the mechanical design. The issue that needs to be dealt with is the unbalanced torque acting on the trirotor caused by uneven number of motors. Alterations in the mechanical design proposed in this paper introduce a tilt mechanism for one of the motors. The tilt mechanism allows

adjusting orientation of one of the motors providing a method of compensation of excessive torque as well as control of the yaw angle. The trirotor flight principle can be explained following[5]:

$$(1.1) \quad = (f_2 - f_1)l_1$$

$$(1.2) \quad = f_3 l_2 \cos \theta - m_3 g l_2 + (f_2 + f_1)l_1 \sin \theta - 2m_{12}g \sin \theta$$

$$(1.3) \quad = f_3 l_2 \sin \theta$$

where:

ϕ, θ, ψ – generalized moments

f_1, f_2, f_3 – forces generated by motors

l_1 – distance from the motors 1 and 2 to the x-axis

l_2 – distance from the motor 3 to the center of gravity

θ – tilt angle of the motor 3

ψ – angle between side booms and axis y

m_{12}, m_3 – motors masses

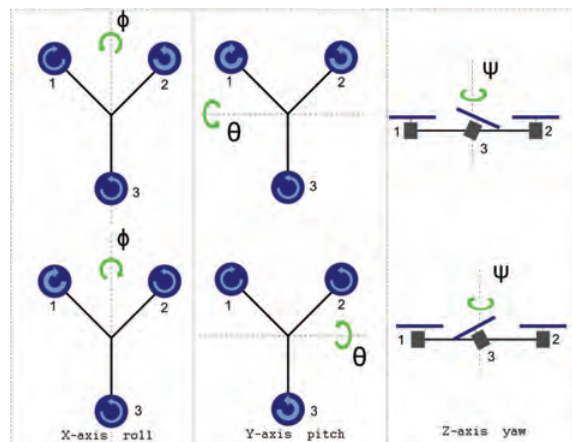


Figure 1: Principles of flight.

If all rotors are spinning with the same angular velocity, the total torque causes rotation along yaw axis. Therefore, the motors 3 is tilted which decomposes the generated thrust into lift force and a component acting in opposite direction to the unbalanced torque. The tilt angle can be adjusted resulting in rotation along yaw axis. Roll angle changes are achieved by generating unbalanced forces by motors 1 and 2. Rotation along y-axis is caused by unequal forces produced by motor 3 and sum of motors 1 and 2.

This paper presents a short overview of four stages of the practical implementation of a trirotor. The second chapter

Email address: andrzej.rys@gmail.com, roman.czyba@polsl.pl, grzegorz.szafranski@polsl.pl.

introduces mechanical design and dedicated flight computer. Measurement system including data filtering is described in the third chapter. Chapter 4 brings in the navigation equations and is followed by control system design in chapter 5. Tests results are presented in chapter 6.

2 MECHANICAL DESIGN

2.1 Airframe

The frame is the key element of an UAV. It should be possibly durable, stiff and light as well as affordable within project budget. Therefore, the materials selected to use were of the possibly finest quality limited only by price. A crucial factor for frame design is the material strength. However, the word “strength” have different meaning for different materials. For metals, strength means yield strength, for composites it is the tensile failure strength. The materials selection for the airframe was based on [6] which analyses their strength relative to cost and density. The materials chosen for the project were carbon fibre reinforced polymer (CFRP), aluminium, and polyamide PA 66. CFRP was used for side booms while the combination of aluminium and PA 66 were used for motors holders, tilt mechanism and central part. The airframe design in 3D CAD is shown in figure 2 and 3.

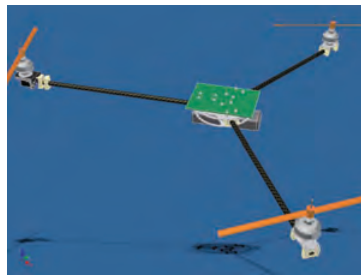


Figure 2: 3D tricopter design.

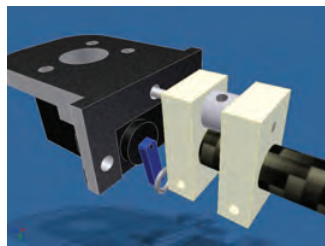


Figure 3: Tilt mechanism with analogue servo.



Figure 4: Assembled tricopter.

2.2 Components

The components used for assembly originate from RC models. Battery, servo, motors and propellers were adopted without changes. The brushless motors selected for the design require electronic speed controllers (ESC). These controllers are available to buy as RC models equipment, unfortunately, in this configuration they are not particularly useful for such a demanding task as stabilization of a rotorcraft. The main limitation is the standard RC model interface, so called pulse position modulation PPM, that operates with frequency as low as 50Hz. Therefore, a dedicated ESC with I2C or UART needs to be used in order to obtain a higher sample rate of the control signals. Some changes were made, so that the device after conversion could be controlled via I2C interface with up to 100kHz transmission frequency and is compatible with 2, 3 and 4-cell Li-Pol batteries (7,4V – 14,8V). The conversion process was performed according to the information from [7] where the new firmware originates from as well.

2.3 Hardware

A dedicated, microcontroller-based flight computer has been developed. The central unit of the flight computer is an Atmel ATmega 128 which performs data collection, solves navigation equations, reads and interprets orders from RC receiver, runs stabilization algorithms and controls the motors and servo. ATmega microprocessor was chosen due to number of available communication protocols (SPI, I2C, USART), low power consumption, and uncomplicated programming. The designed circuits and PCB can be used for both trirotor and quadrotor aerial platforms. It has four sockets for I2C bus and two PWM outputs. It's intended to work with ADIS16400 but due to flexible architecture can also communicate with other sensors types.

3 MEASUREMENT SYSTEM

3.1 Sensors array

Analog Devices ADIS16400 was selected as the main measurement unit. It is an integrated device containing triaxial MEMS gyroscope, triaxial MEMS accelerometer, triaxial magnetometer and auxiliary analog – digital converter. In the small (23 x 23 x 23 mm) and light, 16g, package there are also embedded temperature and voltage sensors allowing precise biasing of readings. Communication with the instrument is performed using SPI interface. The system provides fully autonomous operation and data collection after as little as 220ms start – up time and 4ms sleep mode recovery time. According to the datasheet, calibration of sensitivity, bias and axial alignment was performed for +25°C. The system provides also a Barlett-window FIR digital filtering.

3.2 Filtering

Due to the MEMS sensors nature, a noise in output signal is unavoidable. Additionally, running motors generate vibrations of wide spectrum. These two sources of noise can be modeled as white noise sequence. The white noise can be removed from sensors readings only by describing its statistical characteristics and applying appropriate filtering [8]. Digital linear, time-invariant (LTI) filters can be defined

by a difference equation of the following form:

$$(3.1) \quad \sum_{i=1}^M a_i y[n-i] = \sum_{j=1}^N b_j x[n-j]$$

where,

a_i – feedback coefficients

b_j – feed forward coefficients

The characteristics of input signals must be known to design a proper filter. The measurements, sampled at 200Hz, were collected from sensors placed on trirotor while motors were running to provide possibly close approximation of real flight conditions. A perfect overview of signals components was obtained using a high resolution PSD (Power Spectral Diagram) estimate[8].

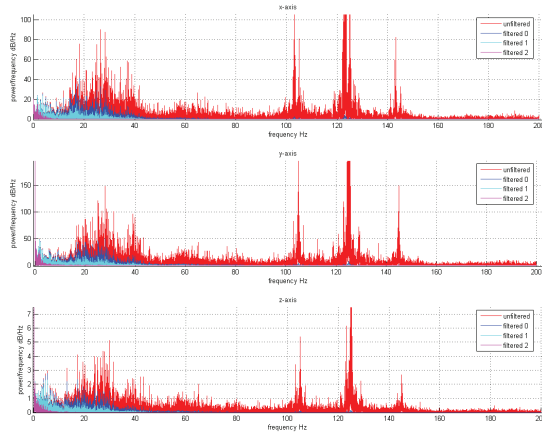


Figure 5: PSD estimate of gyroscopes outputs.

	FILTER 0 $c = 20\text{Hz}$		FILTER 1 $c = 10\text{Hz}$		FILTER 2 $c = 5\text{Hz}$	
	a	b	a	b	a	b
1	1.0000	0.1367	1.0000	0.0730	1.0000	0.0155
2	-0.7265	0.1367	0.0730	0.0730	-0.9691	0.0155

Table 1: Filters cut-off frequencies and coefficients.

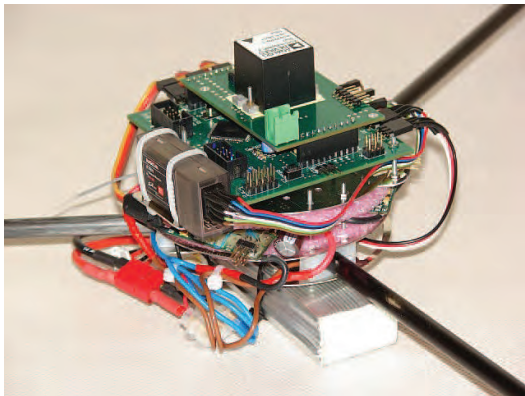


Figure 6: Central part of the trirotor with dedicated electronics, AHRS ADIS16400, RC receiver and battery.

Figure 5 shows PSD of measurements collected during the experiment. There are two main frequency intervals containing most of the noise components. The three peaks

between 100Hz and 150Hz are caused by vibrating three side booms, motors and propellers. The exact location of these peaks is dependent on motors rotational velocity. Noise in this frequency range is not particularly harmful for the navigation because it can be very easily filtered out by a low-pass filter of relatively high cut off frequency. The noise components located between 10Hz and 50Hz have much larger negative impact on the navigation system. Filtering performed with a low-pass filter of cut off frequency below 10Hz slows down the system significantly, on the other hand too high cut off frequency causes the navigation system collect remarkable noise, so the SNR (signal-to-noise ratio) decreases meaningfully.

Excessive tests of various filters determined that first order Butterworth IIR filters with cut-off frequency in range 10-20Hz are most suitable for the trirotor. They perfectly filter out the signal components of frequency 80-160Hz caused by rotating motors and significantly reduce low frequency noise appearing between 20Hz and 40Hz. What is important, the delay introduced by filters of these order is acceptable by navigation and control algorithms.

4 NAVIGATION

During the research on INS, an efficient, novel algorithm for MARG system was found in [9]. It was a perfect solution due to very low computational load, quaternions internal implementation and high accuracy comparable with KF (Kalman Filter) solutions. The algorithm has been modified in order to provide higher usability and performance. During the first run the algorithm saves the initial orientation which will be a basis for further calculation.

4.1 Orientation

The algorithm firstly calculates the orientation using gyroscopes readings - ${}^S_E \mathbf{q}_{,k}$. At this stage there is no correction for drift so the error is accumulating. At the second stage it calculates the orientation using vector measurements from accelerometers and magnetometers. The original algorithm from [9] uses a combination of accelerometers and magnetometers readings, however the modification introduced in this research assumes that a distinctive triple of measurements is provided for any orientation in space by both triaxial accelerometers and triaxial magnetometers. Therefore, the optimization problem from [9] can be extended eventually to:

$$(4.1) \quad {}^S_E \mathbf{q}_{(v,k+1)} = {}^S_E \mathbf{q}_k - k \frac{\nabla f({}^S_E \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}, {}^E \mathbf{b}, {}^S \mathbf{m})}{\|\nabla f({}^S_E \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}, {}^E \mathbf{b}, {}^S \mathbf{m})\|},$$

$$k = 0, 1, 2, \dots, n$$

where:

${}^S_E \mathbf{q}_{(v,k+1)}$ orientation calculated from vectors measurements

${}^S_E \mathbf{q}_k$ last attitude estimation

S, E – sensor and Earth frame
k – no. of iteration

and

$$(4.2) \quad f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}, {}^E \mathbf{b}, {}^S \mathbf{m}) = \left[\frac{f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a})}{f({}_E^S \mathbf{q}, {}^E \mathbf{b}, {}^S \mathbf{m})} \right]$$

where:

$$(4.3) \quad f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}) = \begin{bmatrix} 2g_x(\frac{1}{2} - q_3^2 - q_4^2) + \\ 2g_x(q_2q_3 - q_1q_4) + \\ 2g_x(q_1q_3 + q_2q_4) + \\ +2g_y(q_1q_4 + q_2q_3) + 2g_z(q_2q_4 - q_1q_3) - a_x \\ +2g_y(\frac{1}{2} - q_2^2 - q_4^2) + 2g_z(q_1q_2 + q_3q_4) - a_y \\ +2g_y(q_3q_4 - q_1q_2) + 2g_z(\frac{1}{2} - q_2^2 - q_3^2) - a_z \end{bmatrix}$$

$$(4.4) \quad f({}_E^S \mathbf{q}, {}^E \mathbf{b}, {}^S \mathbf{m}) = \begin{bmatrix} 2b_x(\frac{1}{2} - q_3^2 - q_4^2) + \\ 2b_x(q_2q_3 - q_1q_4) + \\ 2b_x(q_1q_3 + q_2q_4) + \\ +2b_y(q_1q_4 + q_2q_3) + 2b_z(q_2q_4 - q_1q_3) - m_x \\ +2b_y(\frac{1}{2} - q_2^2 - q_4^2) + 2b_z(q_1q_2 + q_3q_4) - m_y \\ +2b_y(q_3q_4 - q_1q_2) + 2b_z(\frac{1}{2} - q_2^2 - q_3^2) - m_z \end{bmatrix}$$

where:

${}^S \mathbf{a} = [0 \ a_x \ a_y \ a_z]^T$ accelerometers normalized readings

${}^S \mathbf{m} = [0 \ m_x \ m_y \ m_z]^T$ magnetometers normalized readings

${}^S \mathbf{g} = [0 \ g_x \ g_y \ g_z]^T$ gravity field initial orientation vector

${}^S \mathbf{b} = [0 \ b_x \ b_y \ b_z]^T$ magnetic field initial orientation vector

which claims

$$(4.5) \quad \nabla f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}, {}^E \mathbf{b}, {}^S \mathbf{m}) = \\ = J^T({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}) f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}) + \\ = \begin{cases} J^T({}_E^S \mathbf{q}, {}^E \mathbf{g}) f({}_E^S \mathbf{q}, {}^E \mathbf{g}, {}^S \mathbf{a}) \\ J^T({}_E^S \mathbf{q}, {}^E \mathbf{b}) f({}_E^S \mathbf{q}, {}^E \mathbf{b}, {}^S \mathbf{m}) \end{cases}$$

where:

$$(4.6) \quad J^T({}_E^S \mathbf{q}, {}^E \mathbf{g}) = \\ = \begin{bmatrix} 2g_yq_4 - 2g_zq_3 & 2g_yq_3 + 2g_zq_4 \\ -2g_xq_4 + 2g_zq_2 & 2g_xq_3 - 4g_yq_2 + 2g_zq_1 \\ 2g_xq_3 - 2g_yq_2 & 2g_xq_4 - 4g_zq_2 - 2g_yq_1 \\ -4g_xq_3 + 2g_yq_2 - 2g_zq_1 & -4g_xq_4 + 2g_yq_1 + 2g_zq_2 \\ 2g_zq_4 + 2g_xq_2 & -4g_yq_4 + 2g_zq_3 - 2g_xq_1 \\ 2g_yq_4 - 4g_zq_3 + 2g_xq_1 & 2g_yq_3 + 2g_xq_2 \end{bmatrix}$$

and

$$(4.7) \quad J^T({}_E^S \mathbf{q}, {}^E \mathbf{b}) = \\ = \begin{bmatrix} 2b_yq_4 - 2b_zq_3 & 2b_yq_3 + 2b_zq_4 \\ -2b_xq_4 + 2b_zq_2 & 2b_xq_3 - 4b_yq_2 + 2b_zq_1 \\ 2b_xq_3 - 2b_yq_2 & 2b_xq_4 - 4b_zq_2 - 2b_yq_1 \\ -4b_xq_3 + 2b_yq_2 - 2b_zq_1 & -4b_xq_4 + 2b_yq_1 + 2b_zq_2 \\ 2b_zq_4 + 2b_xq_2 & -4b_yq_4 + 2b_zq_3 - 2b_xq_1 \\ 2b_yq_4 - 4b_zq_3 + 2b_xq_1 & 2b_yq_3 + 2b_xq_2 \end{bmatrix}$$

The orientations resulting from accelerometers and magnetometers readings are averaged and denoted as ${}^S \mathbf{q}_{v,k}$.

4.2 Fusion

There are two independent sources of updates for new attitude estimation: ${}^S \mathbf{q}_{,k}$ and ${}^S \mathbf{q}_{v,k}$. Fusion of the two estimates was performed by a complementary filter:

$$(4.8) \quad {}^S \mathbf{q}_k = (1 - \beta) {}^S \mathbf{q}_{,k} + \beta {}^S \mathbf{q}_{v,k}, 0 \leq \beta \leq 1$$

The complementary filter after derivation performed in [9] has a following form:

$$(4.9) \quad {}^S \dot{\mathbf{q}}_k = {}^S \dot{\mathbf{q}}_{,k} - \beta {}^S \dot{\mathbf{q}}_{v,k}$$

Which is equal to:

$$(4.10) \quad {}^S \dot{\mathbf{q}}_k = {}^S \dot{\mathbf{q}}_{,k} - \frac{\nabla f}{\|\nabla f\|}$$

And leads to:

$$(4.11) \quad {}^S \mathbf{q}_k = {}^S \mathbf{q}_{(k-1)} + {}^S \dot{\mathbf{q}}_k \Delta t$$

The parameter β was found during the experiments, providing fast algorithm reaction for rapid changes along with good drift correction in long run.

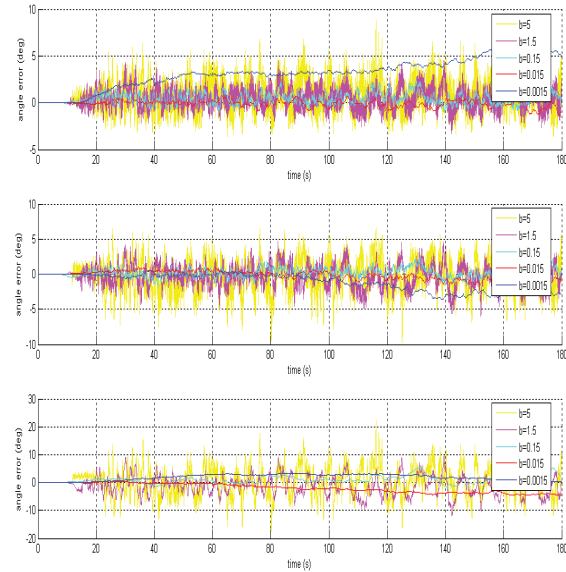


Figure 7: Influence of parameter β on performance of navigation algorithm.

Tests were performed while ADIS16400 was mounted on trirotor frame, motors were running and filtering was applied. The measure used to evaluate algorithm performance was root mean square error (RMSE). It was calculated over the interval 0-180s for each axis separately. Eventually, the arithmetic mean of these values was computed and was presented in the table 2:

	RMSE			
	x-axis	y-axis	z-axis	mean
0.0015	12.5147	2.5062	4.6179	6.5463
0.015	0.1301	0.3954	6.2707	2.2654
0.15	0.6223	0.4473	2.6729	1.2475
1.5	1.3319	2.3426	12.1701	5.2815
5	3.0773	4.5904	25.1942	10.9540

Table 2: RMSE of navigation algorithm with various values of parameter β .

The parameter β determines how big is the influence of vector measurements on the final result. For small values of β , the navigation performance was affected by gyros drift that could not be compensated by absolute vector measurements. For large values of β , the drift correction overshoots the exact solution causing the optimization algorithm operate far from the optimal solution. The tests have shown that the value of β should be close to 0,15 in order to provide highest navigation system performance.

5 CONTROL SYSTEM

System state is estimated according to the measurements obtained from a MARG sensors array. As it was already shown, ADIS16400 provides information about current angular velocity, linear acceleration and magnetic field orientation, with sample rate up to 819,2Hz. The raw readings are processed by a navigation algorithm which outputs the information about current UAV orientation.

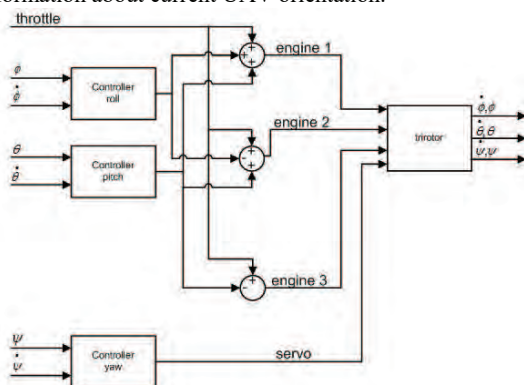


Figure 8: General system structure.

It is important to note, that the raw measurements, especially angular velocities, are of much faster dynamics than the attitude orientation returned by navigation algorithm. The reason is that they are measured directly by sensors, while the navigation algorithm introduces some extra dynamics.

The objective of the control algorithm for trirotor was stabilization of angles, in other words, ability to track and maintain given roll, pitch and yaw angles. The control system was decoupled and as a result each of the axes has a separate control algorithm. Three control algorithms return separate inputs for three motors and servomechanism which need to be combined together with throttle signal in a mixer. The general system structure is presented in the figure 8.

According to the research on existing control systems for UAVs and investigation of measurements characteristics a PID cascade was chosen for realization of trirotor control algorithm. An example of the control system of roll angle is presented in the figure 9.

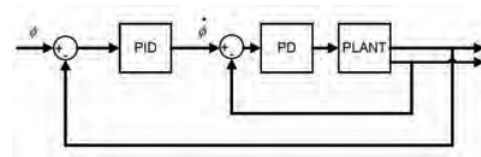


Figure 9: Control schema for roll stabilization.

The structure of controller chosen for the internal loop is of type B with filtered derivative [10]. The type B structure was selected in order to avoid undesired overreactions on changes in setpoint. Moreover, a filter in the derivative term provides additional control quality improvement. This loop guarantees quick reaction on disturbances in the system and eases tuning of the inherently unstable system. The outer loop, similarly to the inner one, is of type B with filtered derivative, however, there are some modifications introduced. The first modification concerns the way the setpoint changes are applied to the system. The main concern behind it is to reduce the overshoot when the setpoint varies rapidly. One method to avoid this problem would be to use type C PID controller. Although it removes the overshoot, it might be too slow for aerial vehicle, because the desired value is being tracked only by the integral term. The method that was implemented in the trirotor algorithm is a low-pass filter which reduces the speed of change of the setpoint to 10deg/s which is slow enough to avoid peak reaction of proportional term, and on the other hand, fast enough to provide high UAV performance. The second algorithm enhancement is a double anti-windup protection system. It limits the maximal value of error that supplies the integral part and defines lower and upper bound for the integral term value. The third enhancement of the algorithm is a filter in derivative term.

6 TESTS RESULTS

The tests were performed in order to examine whether all components are working properly and stabilization of trirotor attitude is possible. The ability to maintain a given setpoint for at least 60 seconds was tested for each axis separately using a specialized stand that limited the number of degrees of freedom to one. The tuning process followed cascade systems tuning rules i.e. the inner loop was tuned to have possibly high proportional and derivative gain. These setting provided good rejection of disturbances appearing in the inner loop.

Secondly, the outer loop having slower dynamics was tuned so, that the trirotor was able to maintain a given setpoint. Tests results were satisfactory, the trirotor was able to hold a given setpoint for unlimited time. The tests results are shown in figure 10.

Once the trirotor was capable of maintaining a given, constant setpoint it was tested against step changes in setpoints. The gains required slight modifications in order to prevent a overshoot to occur and provide damping of oscillations. The tests have proven that the trirotor is properly tracking changes in setpoints in all axes. The results are presented in the figure 11.

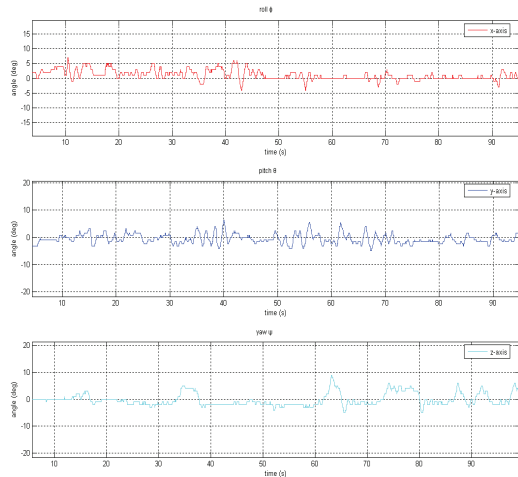


Figure 10: Euler angles during the hover flight.

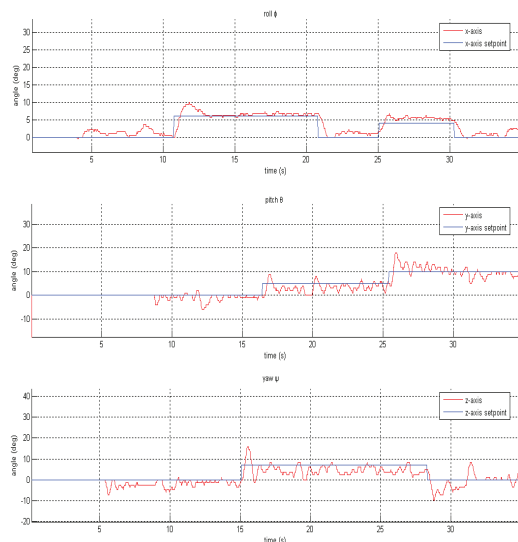


Figure 11: Tracking of the Euler angles desired values.

7 CONCLUSIONS

In this paper we have presented the whole trirotor system, beginning with the mechanical construction and then through the electronics till the control algorithm implementation at the last stage. The trirotor aerial robot can make possible of plenty of potential applications for unmanned aerial vehicles.

An approach to hardware implementation of the navigation algorithm, as well as control algorithm has been described. The navigation algorithm based on accelerometers, gyroscopes and magnetometers combined with the cascade control structure provides good stabilization during the different flight conditions. The trirotor is able to maintain given orientation as well as track the changes of the Euler angles. The practical realization of the attitude stabilization system is an important step in the development process of more advanced functionality of the autonomous flying vehicles. First attempts to outdoor flights were very promising and achieved results are satisfactory.

ACKNOWLEDGEMENTS

This work has been granted from funds for science in 2010-2012 as a development project No. OR00011811.

REFERENCES

- [1] Immanuel Ashokaraj, Antonios Tsourdos, Peter M. G. Silson & Brian A. White. A Robust Approach to Multiple Sensor Based Navigation for an Aerial Robot, *Proceedings of the 2006 IEEE/RSJ, International Conference on Intelligent Robots and Systems, Beijing, China*, pages 3533-3538.
- [2] Samir Bouabdallah, Andk Noth and Roland Siegwart. PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor. *Proceedings of 2004 IEEE/RSJ International Conference On Intelligent Robots and Systems, Sendai, Japan*, volume 3, page 2451
- [3] Holger Voos. Nonlinear State-Dependent Riccati Equation Control of a Quadrotor UAV, *Proceedings of the 2006 IEEE International Conference on Control Applications, Munich, Germany*, page 2547
- [4] Tarek Madani and Abdelaziz Benallegue. Backstepping Control for a Quadrotor Helicopter, *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China*, page 3255
- [5] Salazar S., Lozano R., Escareño J. Stabilization and nonlinear control for a novel trirotor mini-aircraft, *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain*, pages 2612-2617
- [6] Ashby, Michael F. Materials Selection in Mechanical Design (3rd Edition), Butterworth-Heinemann 2005
- [7] <http://www.rcgroups.com/forums/showthread.php?t=766589&page=21>, last access on May 15th, 2011
- [8] Liu Rui Hua, Liang Rongqiang, Zhang Lei. Filtering Algorithm Research on MEMS Gyroscope Data, *2008 International Conference on Computer Science and Software Engineering, Wuhan, Hubei*, pages 186-189
- [9] Magdwick Sebastian O.H., Harrison Andrew J.L., Vaidyanathan Ravi. An efficient orientation filter for IMUs and MARG sensor arrays. April 9, 2010, unpublished
- [10] Kiam Heong Ang, Gregory Chong, Yun Li. PID Control System Analysis, Design, and Technology. July, 2005, IEEE Transaction on Control Systems Technology, vol. 13 no.4.

Different Approaches of PID Control UAV Type Quadrotor

G. Szafranski, R. Czyba

Silesian University of Technology, Akademicka St 16, Gliwice, Poland

ABSTRACT

In this paper we focus on the different control strategies for the unmanned aerial vehicles (UAV). The control task is formulated as an angular stabilization of the quadrotor platform, and also as a tracking problem of chosen state variables. The PID algorithm has been considered in three structures in respect of the optimal control signal applied to the actuators. For better performance of quadrotor during the hover mode the cascade control system has been proposed.

The experiment results for the platform orientation control with different PID controller architectures are presented, and confirm the effectiveness of the proposed method and theoretical expectations.

1 INTRODUCTION

Unmanned aerial platforms are not a new invention. They were first introduced during the World War I, but not until recently have been flown autonomously. Among the several kinds of mini and micro unmanned aerial vehicles (MUAVs), quadrotors are probably the most common. This platform can occur in one of two configurations, "plus" or "cross" shape, and has been widely developed by many Universities such as MIT or Stanford/Berkeley, and commercial companies Draganflyer, X3D-BL, Xaircraft [6]. The great maneuverability and possible small size of this platform make it suitable for indoor use, as well as for outdoor applications. Such aerial platform has several application domains [4], [7]: safety, natural risk management, environmental protection, management of the large infrastructures, agriculture and film production. This aerial vehicle is highly maneuverable, has the potential to hover and to take-off, fly, and land in small areas, and has a simple control mechanism. However, a quadrotor is unstable and impossible to fly in full open loop system. The dynamics of a flying vehicle is more complex than the ground robots, so that even the hovering becomes a non-trivial task. Thus, control of a nonlinear plant is a problem of both practical and theoretical interest.

Improved performance expected from the new generation of VTOL vehicles is possible through derivation and implementation of specific control techniques incorporating limitations related to sensors and actuators. The well-known approach to decoupling problem solution based on the Non-linear Inverse Dynamics (NID) method may be used if the parameters of the plant model and external disturbances are exactly known. Usually, incomplete information about systems in real practical tasks take place. In this case adaptive control methods or control systems with sliding mode [3], [4] may be used for solving such control problem.

A way of the algorithmic solution of this problem under condition of incomplete information about varying parameters of the plant and unknown external disturbances is the application of the Dynamic Contraction Method (DCM) [14] applied in [10]. But the most problems of those approaches in real applications are: high order of the controller equations and influence of measurement noise for a control quality. Approximations of higher derivatives amplify the measurement noise and cause abrupt changes of control signal. Therefore in this paper the different structures of PID controllers, which can reduced the adverse effects are considered.

The main aim of this research effort is to examine the effectiveness of a designed attitude control system for the quadrotor in the cascade control system with different types of PID controllers [2],[5].

The paper is organized as follows. First, a mathematical description of the quadrotor nonlinear model is introduced. The next part presents a general structure of a cascade control system, and investigation of three types of PID controllers with modified loop structure. This section includes the schemes and description of PID controller type A, B, C. The next chapter shows the results of simulations in two sections: first – inner loop control with all types PID controllers; second – performed in the cascade control system. Finally, the conclusions are briefly discussed in the last chapter.

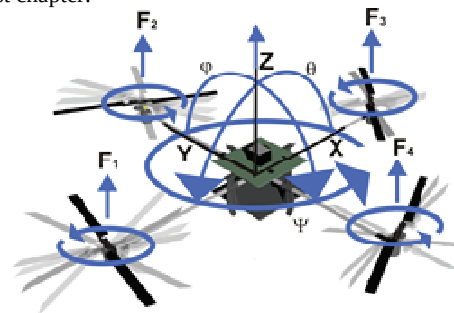


Figure 1: Quad-thrust aerial vehicle.

2 QUADROTOR MODEL

Described below the quadrotor model based on the self-modified version of a commercial Draganflyer platform. The aerial vehicle consists of a rigid cross frame equipped with four rotors as shown in Fig. 1.

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 longitudinal motion result

Email address: grzegorz.szafranski@polsl.pl, roman.czyba@polsl.pl.

from 1 and 3 propeller's speed conversely modified. Yaw rotation – as a result from the difference in the counter-torque between each pair of propellers [8].

2.1 Rigid Body Model

The quadrotor is a six degrees of freedom system defined with twelve states. The following state and control vectors are adopted:

$$(1) \quad X = [\ddot{x}, \ddot{y}, \ddot{z}, \dot{x}, \dot{y}, \dot{z}, x, y, z]^T$$

$$(2) \quad U = [u_1, u_2, u_3, u_4]^T$$

where: u_i - control input of motor,
 $i = 1, 2, 3, 4$ - motor number.

Six out of twelve states govern the attitude of the system (Fig.2). These include the Euler angles (ϕ, θ, ψ) and angular rates around the three orthogonal body axes. The other six states determine the position (x, y, z) and linear velocities of the center of mass of the quadrotor with respect to a fixed reference frame.

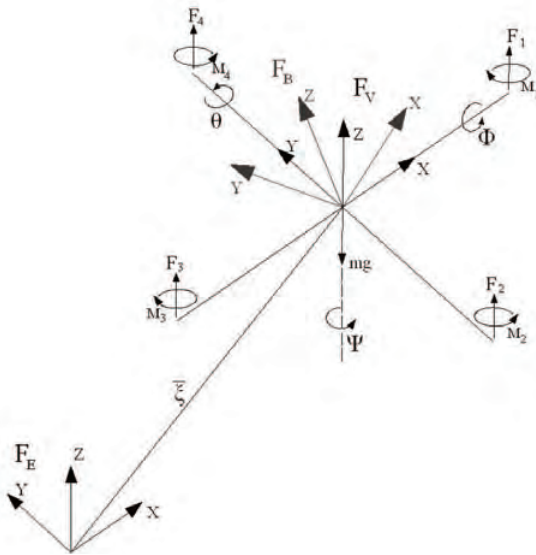


Figure 2: Quad-thrust aerial vehicle.

Using the Lagrangian, and the general form of the equations of motion in Lagrange method [1], [8], [12], [13]:

$$(3) \quad L = T_K - V$$

$$(4) \quad F = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q}$$

where: L is Lagrangian, T_K is kinetic energy, V is potential energy, $q = [x, y, z, \phi, \theta, \psi]^T$ is a vector of generalized coordinates, $F = (F_E, T)$ are a generalized forces F_E and moments T applied to the quadrotor due to the control inputs.

For translational motion the Lagrange equation has a form:

$$(5) \quad F_E = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x}$$

where: $[x, y, z]^T$ - position coordinates,

$$F_E = \begin{bmatrix} \sin(\phi) \\ -\sin(\phi)\cos(\theta) \\ \cos(\phi)\cos(\theta) \end{bmatrix} \cdot f_g$$

$$f_g = F_1 + F_2 + F_3 + F_4$$

$$F_i = b\Omega_i^2$$

Ω_i - rotor speed

b - thrust factor

Accordingly, the Lagrange equation for rotary motion is following:

$$(6) \quad T = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi}$$

where:

$[\phi, \theta, \psi]^T$ - Euler angles

$$T = [T_\phi, T_\theta, T_\psi]^T$$

$$T_\phi = bl(\Omega_4^2 - \Omega_2^2) - J_r \dot{\psi}(\Omega_1 + \Omega_3 - \Omega_2 - \Omega_4)$$

$$T_\theta = bl(\Omega_3^2 - \Omega_1^2) + J_r \dot{\psi}(\Omega_1 + \Omega_3 - \Omega_2 - \Omega_4)$$

$$T_\psi = d(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2)$$

l - distance between propeller center and CoG

J_r - rotor inertia

d - drag factor

Above torques equations (T_ϕ, T_θ, T_ψ) consist of the action of the thrust forces difference of each pair, and from the gyroscopic effect.

Finally the quadrotor dynamic model with x, y, z , motions as a consequence of a pitch, roll and yaw rotations is as follows:

$$(7) \quad \ddot{x} = \frac{1}{I_{xx}} \left(-\dot{\psi}^2 (I_{xx} - I_{zz}) s(\phi) c(\theta) - \dot{\phi} \dot{\psi} I_{zz} c(\phi) + T_\phi \right)$$

$$(8) \quad \ddot{y} = \frac{1}{I_{yy} (1 + s^2(\phi))} \left(-\ddot{\psi} I_{zz} s(\phi) - \dot{\phi} \dot{\psi} c(\phi) s(\theta) \cdot (2I_{zz} - 2I_{yy}) - \dot{\phi} \dot{\psi} I_{zz} c(\phi) + T_\psi \right)$$

$$(9) \quad \ddot{z} = \frac{1}{I_{zz}} \left(-\ddot{\psi} I_{zz} s(\phi) + T_\psi \right)$$

$$(10) \quad \ddot{\phi} = \frac{f_g}{m} s(\phi)$$

$$(11) \quad \ddot{\theta} = -\frac{f_g}{m} c(\phi) s(\theta)$$

$$(12) \quad \ddot{\psi} = \frac{f_g}{m} c(\phi) c(\theta) - g$$

where:

s and c are abbreviations of 'sin' and 'cos',

I_{xx}, I_{yy}, I_{zz} - inertia moments.

2.2 Propulsion system

The dynamics of the propulsion system consists of a DC motor and propeller. Motor model can be considered as a first order differential equation (13) because of a very low inductance.

$$(13) \quad J_r \frac{d\Omega_m}{dt} = \frac{k_m}{R} u - \frac{k_m k_e}{R} \Omega_m - I$$

where: u - motor input, R - motor resistance, k_e - motor electrical constant, Ω_m - motor angular speed, J_r - rotor inertia, k_m - torque constant, I - motor load.

The torque produced by motor is converted by propeller to the thrust force [9], [11]. The relationship between the angular velocity and the thrust is given in the following form:

$$(14) \quad F_T = C_T D^4 \cdot n_p^2$$

The thrust coefficient C_T is a propeller parameter and it primarily depends on the λ ratio given as:

$$(15) \quad \lambda = \frac{V}{n_p D}$$

where: V - air speed, n_p - propeller velocity in revolutions per second, D - propeller diameter, ρ is the air density.

Finally we obtain propulsion system modeled as a series connection of a linear first order dynamic element and static nonlinear second degree polynomial (Fig. 3).

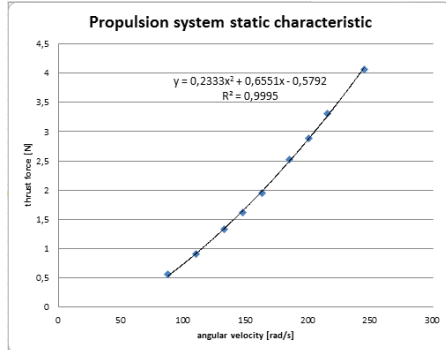


Figure 3: Static characteristic of propulsion system.

3 CONTROL SCHEME

In control applications, the rejection of external disturbances and performance improvement is a major concern. In order to fulfill such requirements, the implementation of a cascade control system can be considered. Basically, in a cascade control schema the plant has one input and two or more outputs [2]. Indeed, this requires an additional sensor to be employed so that the fast dynamics could be measured.

The primary controller and the primary dynamics are components of the outer loop. The inner loop is also a part of the outer loop, since the primary controller calculates the set point for the secondary controller loop. Furthermore the inner loop represents the fast dynamics, whereas the outer

should be significantly slower (with respect to the inner loop). This assumption allows to restrain interaction that can occur between them and improve stability characteristics. Therefore a higher gain in the inner loop can be adopted. An additional advantage is, that the plant nonlinearities are handled by the controller in the inner loop and they do not have meaningful influence on the outer loop [5].

In this paper the cascade control structure is proposed, as a solution to control task formulated as an angular stabilization. The angular velocities of the rotating platform are additional measurements that can be used in the inner loop. In this case, there is no need to assemble any extra sensors, thus the AHRS (Attitude and Heading Reference Signal) provides not only angles but also other raw data, such as accelerations, angular velocities and gravitational field strength. The outer loop is based on the Euler angles, the measurements are calculated from the combination of the accelerometers, gyroscopes and magnetometer. The cascade control loop for the quadrotor vehicle is shown in Fig. 4.

3.1 Inner and Outer PID controllers

In both loops three types of PID controllers are considered.

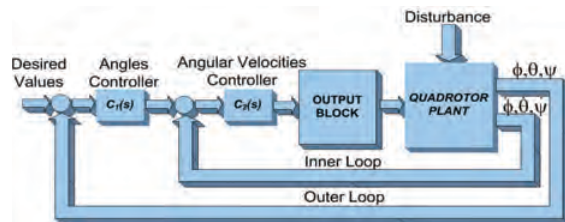


Figure 4: Cascade control system for quadrotor.

3.2 PID Controller – type A

In control theory the ideal PID controller in parallel structure is represented in the continuous time domain as follows:

$$(16) \quad u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

where: K_p - proportional gain,

K_i - integral gain,

K_d - derivative gain.

A block diagram that illustrates given controller structure is shown in the Fig. 5.

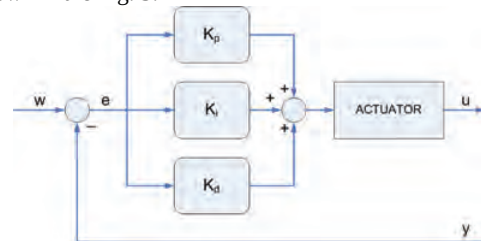


Figure 5: PID Controller – type A.

The problem with conventional PID controllers is their reaction to a step change in the input signal which produces an impulse function in the controller action. There are two sources of the violent controller reaction, the proportional term and derivative term. Therefore, there are two PID

controller structures that can avoid this issue. In literature exists different names [2], [5]: type B and type C; derivative-of-output controller and set-point-on-I-only controller; PI-D and I-PD controllers. The main idea of the modified structures is to move either the derivative part or both derivative and proportional part from the main path to the feedback path. Therefore, they are not directly subjected by jump of set value, while their influence on the control reaction is preserved, since the change in set point will be still transferred by the remaining terms.

3.3 PID Controller – type B

It is more suitable in practical implementation to use "derivative of output controller form". The equation of type B controller is following:

$$(17) \quad u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau - K_d \frac{dy(t)}{dt}$$

A block diagram that illustrates given controller structure is shown in Fig. 6.

If PI-D structure is used, discontinuity in $r(t)$ will be still transferred through proportional into control signal, but it will not have so strong effect as if it was amplified by derivative element.

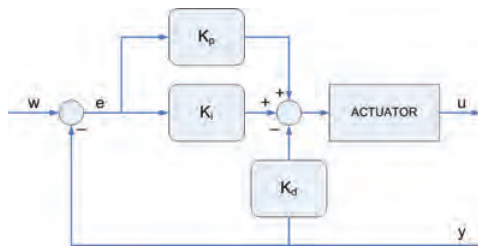


Figure 6: PID Controller – type B.

3.4 PID Controller – type C

This structure is not so often as PI-D structure, but it has certain advantages. Control law for this structure is given as:

$$(18) \quad u(t) = -K_p y(t) + K_i \int_0^t e(\tau) d\tau - K_d \frac{dy(t)}{dt}$$

Block diagram for type C controller is shown in Fig. 7.

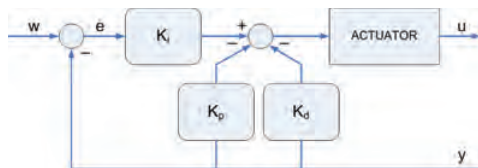


Figure 7: PID Controller – type C.

With this structure transfer of reference value discontinuities to control signal is completely avoided. Control signal has less sharp changes than with other structures.

4 SIMULATIONS RESULTS

In this section, we present the results of simulations which were conducted to evaluate the performance of the designed attitude control system in the cascade structure with different types of PID controllers. The presented simulations consisted in transition with predefined dynamics

from one steady-state flight to another. In the design process we consider three types of PID controllers (type A, B, C) optimizing the parameters in view of the assumed reference model. To evaluate the quality of control it was taken into account the tracking of the reference model, and in particular the realizability of the control in practical aspects.

The entire MIMO control system consists of three cascade control channels with two PID controllers each of them. Feedback data for the regulators are six variables: Euler angles α, β, γ (outer loop) and angular velocities $\dot{\alpha}, \dot{\beta}, \dot{\gamma}$ (inner loop). Control signals are motors inputs: u_1, u_2, u_3, u_4 . The general control task is stated as a tracking problem for the following variables:

$$(19) \quad \lim_{t \rightarrow \infty} [\alpha(t) - \alpha_0(t)] = 0$$

$$\lim_{t \rightarrow \infty} [\beta(t) - \beta_0(t)] = 0$$

$$\lim_{t \rightarrow \infty} [\gamma(t) - \gamma_0(t)] = 0$$

where $\alpha_0(t), \beta_0(t), \gamma_0(t)$ are the desired values of the considered variables.

In view of the complexity and multidimensionality of the considered problem only the results in pitch control channel are presented.

The tuning of the cascade controller parameters were made in two steps. First, inner loop controller was tuned based on the assumed reference model. The desired dynamics is determined by a following transfer function:

$$(20) \quad K_{ref,i}(s) = \frac{1}{sT_i + 1}$$

where $T_i = 0.25$ [s].

At this stage of designing we consider three structures of PID controller: type A, type B (PI-D), and type C (I-PD).

In this case the accuracy requirements for the system are formulated in a form of two performance indices related to the time responses of the system. Therefore, there was introduced the following quadratic integral index for the tracking performance:

$$(21) \quad I_{tr} = \int_0^T [w(t) - y(t)]^2 dt$$

Second index determines the effort of control signal and is defined as follows:

$$(22) \quad I_u = \int_0^T u^2(t) dt$$

Numerical results are shown in Table 1.

Structure	Value
Control Index	
Type A	1399.365
Type B	1.8674
Type C	4.8704
Tracking Index	
Type A	0.90128
Type B	0.77048
Type C	1.0

Tabel1: Performance Indices.

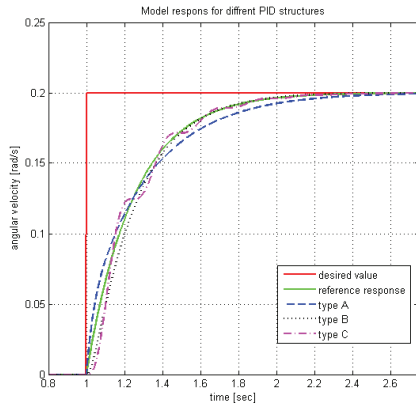
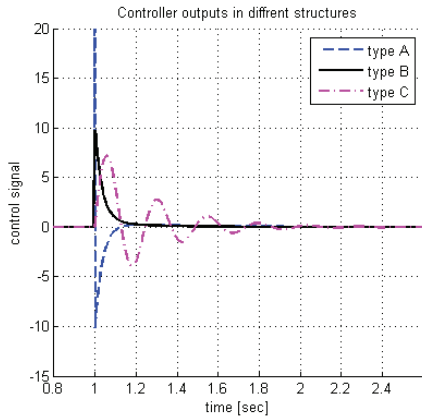


Figure 8: Time history of angular velocity

Figure 9: Time history of control signal u

Remark 1: The relative order of the inner loop with PID controller is equal one.

Remark 2: Based on the remark 1 the reference model is provided by the first order inertia system (20).

Remark 3: Gradient descent method allows to tune the controller parameters, and obtain the satisfactory reference model tracking results in all structures (Table1).

Remark 4: However, the index of control signal effort in particular types of PID controllers indicates significant differences.

Remark 5: In the terms of practical implementation the type A seems to be not acceptable (value=2047). On the basis of the presented findings, the most common structure is type B, therefore this one will be used in the next step, as the best possible solution.

In the second step the outer loop controller was tuned based on the assumed following reference model:

$$(23) \quad K_{ref,O}(s) = \frac{1}{s^2 + 2\zeta s + 1}$$

where: $\zeta = 0.4$ - undamped resonance period,
 $\zeta = 1$ - relative damping factor.

Remark 6: In respect of the slower outer loop dynamics the reference model was determined as a second order differential equation (23).

Remark 7: Referring to remark no. 5, the advantages of

the type B PID controller has been confirmed in cascade control system.

Remark 8: In case of PI controller the architectures type A and B are equivalent.

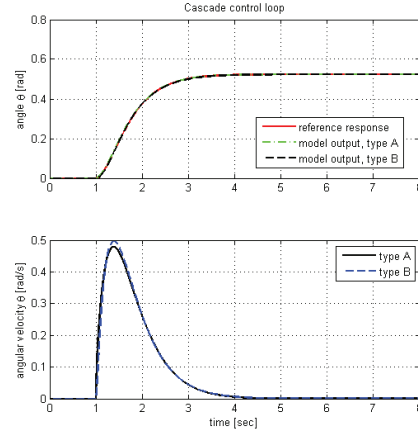


Figure 10: Time history of pitch angle and angular velocities.

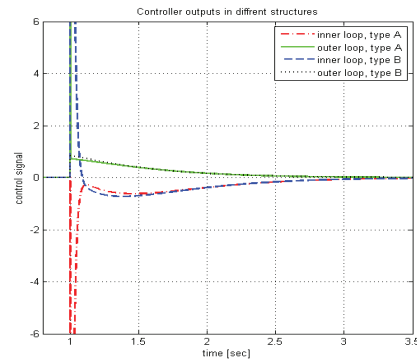


Figure 11: Controllers signals.

5 CONCLUSION

In this paper, the different approaches to the problem of attitude control of a quadrotor were considered. The main goal of this research is the evaluation of different types of PID algorithm in practical aspects of control systems design. Three architectures were presented and examined with respect to the best performance. All of the reviewed architectures of the controllers resulted in almost the same model output response time but significantly different control signals. Taking into consideration the proposed control effort index, type B architecture is the most comprehensive choice.

Assumed reference models provides time separation between fast and slow dynamics in the cascade system. The application of a cascade control structure gives the possibility to adapt the simple PID algorithm for controlling complex systems, such as vertical take-off and landing platform. Proposed approach is an alternative solution to the advanced control algorithms but it requires additional sensor, which provides measurement for the inner control loop. However, from the point of unmanned aerial vehicle,

such as quadrotor, the cascade control architecture can be implemented without any extra sensing elements.

The conducted simulations and analysis proved the ability, of the designed cascade structure, to control the orientation platform angles, and provide the promising fundamentals for practical experiments with a physical plant.

ACKNOWLEDGMENT

This work has been granted from funds for science in 2010-2012 as a development project No. OR00011811.

REFERENCES

- [1] A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadrotor aircraft", *IEEE Trans. on Control Systems Technology*, vol. 14, no. 3, 2006, pp. 562-571.
- [2] A. Visioli, "Practical PID Control", Springer-Verlag, London 2006
- [3] H. Bouadi and M. Tadjine, "Nonlinear observer design and sliding mode control of four rotors helicopter", *Int. Jour. of Mathematical, Physical and Engineering Sciences*, vol. 1, no.2, pp. 115-120.
- [4] H. Bouadi, M. Bouchoucha, and M. Tadjine, "Sliding mode control based on backstepping approach for an UAV type-Quadrotor", *Int. Jour. of Applied Mathematics and Computer Sciences*, vol. 4, no. 1, pp. 12-17.
- [5] H. L. Wade, "Basic and Advanced Regulatory Control: System Design and application", ISA, United States of America, 2004.
- [6] K. P. Valavanis, *Advances in Unmanned Aerial Vehicles*. The Netherlands: Springer-Verlag, 2007.
- [7] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, D. Nakzawa, 2010. *Autonomous Flying Robots*. Springer, 1st edition.
- [8] P. Castillo, R. Lozano, and A. E. Dzul, *Modelling and Control of Mini-flying Machines*. London: Springer-Verlag, 2005, ch. 3.
- [9] P. Lindahl, E. Moog and S.R. Shaw, 2009. *Simulation, Design and Validation of a UAV SOFC Propulsion System*. Aerospace Conference, pp.1-8, Big Sky, MT.
- [10] R. Czyba, 2009. *Attitude Stabilization of an Indoor Quadrotor*. Proc. Of European Micro Aerial Vehicle Conference and Competition, EMAV.
- [11] R. Goel, S.M. Shah, N.K. Gupta, N. Ananthkrishnan, 2009. Modeling, Simulation and Flight of an Autonomous Quadrotor. *Proceedings of ICEAE*.
- [12] S. Bouabdallah, "Design and control of quadrotors with application to autonomous flying", Ph.D. dissertation, School of Computer and Communication Sciences, Lausanne, 2007.
- [13] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor", *Proc. of Int. Conf. On Intelligent Robots and Systems, Japan, 2004*.
- [14] V. D. Yurkevich, *Design of Nonlinear Control Systems with the Highest Derivative in Feedback*. World Scientific Publishing, 2004.

Equilibrium Transition Study for a Hybrid MAV

M. Itasse, and J.M. Moschetta
ISAE, Toulouse, France
Y. Ameho, and R. Carr
ONERA, Toulouse, France

ABSTRACT

Wind tunnel testing was performed on a VTOL aircraft in order to characterize longitudinal flight behavior during a transition between vertical and horizontal flight modes. Trim values for airspeed, pitch, motor speed and elevator position were determined. Data was collected by independently varying the trim parameters, and stability and control derivatives were identified as functions of the trim pitch angle. A linear fractional representation model was then proposed, along with several methods to improve longitudinal control of the aircraft.

1 INTRODUCTION

Modern urban reconnaissance missions dictate the need for a Micro Air Vehicle (MAV) platform capable of performing a complex mission: rapid and efficient ingress to a target location followed by slow loiter for quality image capture. This mission must be performed despite the presence of typical urban obstacles, possible lack of GPS, and erratic wind-gusting. This may be achieved using a tilt-body fixed-wing vehicle which combines the speed, range, and gust-hardiness of a fixed wing with the loiter and precision capability of a rotorcraft vehicle.

The MAVion, shown in Figure 1, has been created for this purpose. It combines a fixed-wing airframe with tandem counter-rotating rotors without resorting to a hollow shaft system as in the case of a coaxial rotor in tractor position. The direction of motor rotation has been chosen to counter wing tip vortices. Also, a tandem-rotor configuration has the advantage of providing an extra degree of freedom to control the vehicle along the yaw axis. From an aerodynamic perspective, it also allows for a larger wing area within the propeller slipstream, yielding higher aerodynamic flap efficiency and a better aerodynamic performance due to a higher aspect ratio.



Figure 1: The MAVion, a tandem-rotor tilt-body configuration

At the 2009 International MAV competition held in Pensacola, Florida, a team of students from l'Institut Supérieur de l'Aéronautique et de l'Espace (ISAE) demonstrated the horizontal flight capabilities of the MAVion in the outdoor navigation competition. At the 2010 meeting in Braunschweig Germany a vertical version of the aircraft was flown in the indoor competition. Development of the MAVion has continued and the aircraft is now capable of performing fully autonomous transitions between vertical and horizontal flight modes. A wheeled hybrid version of the aircraft has also been developed to enable the MAVion to roll along the ground, walls, or ceilings, increasing its usefulness in indoor settings [1,2].

There are many ways to transition between horizontal and vertical flight modes [3]. The intent of the current study is to explore equilibrium transition, defined here as maintaining the steady state forces and moments near to zero. This results in an energy efficient change between flight modes with no gain or loss of altitude. Wind tunnel tests were performed to determine the throttle setting, the elevator deflection and the angle of attack for a given velocity during an equilibrium transition.

2 EXPERIMENTAL SETUP AND PROCEDURE

2.1 Experimental Apparatus

Experimental studies on MAVs necessitate a low Reynolds number wind tunnel with low turbulence intensity. The SabRe wind tunnel located at ISAE has been designed for this purpose [4]. As shown in Figure 2, SabRe is a closed-loop wind tunnel.

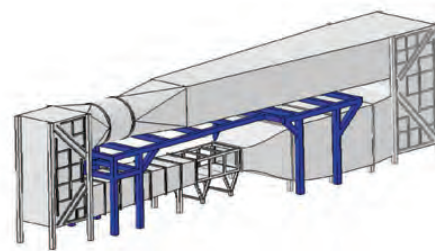


Figure 2: SabRe wind tunnel.

The test section is 2400mm long with a square cross-sectional area of 1200 x 800mm with a contraction ratio of 9. In order to produce a stable and uniform flow a pitch controlled fan was implemented along with a series of honeycomb grids and screen to split and damp vertical structure. By changing the fan speed and the pitch angle, turbulence intensity can be optimized. For velocities ranging from 2 to 15m.s⁻¹, typical for

a VTOL MAV, the turbulence intensity was found to be lower than 0.1%.

A new 3-component aerodynamic balance devoted to MAV studies was used to measure lift, drag and pitching moment. The balance depicted in Figure 3 was designed to enable measurements at angles of attack between 0 and 90°. Two identical load cells were mounted on the parallelogram balance, giving uncoupled measurements of lift and drag.

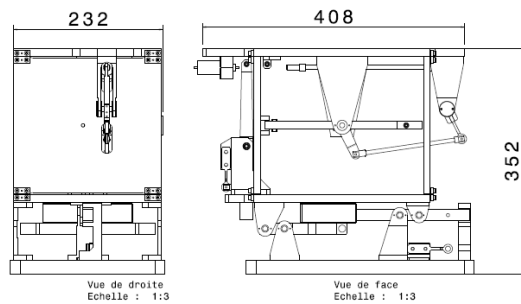


Figure 3: Monnin Balance

The model is attached to the balance by three struts. The two forward supports are fixed at 10.5cm from the leading edge while the rear, attached above the trailing edge, is raised and lowered to modify the angle of attack. This system is demonstrated in Figure 4. A final force sensor is mounted to the rear strut enabling a calculation of the moment about the center of gravity.



Figure 4: Angle of attack is modified by raising or lowering the rear support

The model contains a core of expanded polypropylene surrounded by a shell of fibreglass. An onboard control and telemetry system was developed to reduce the need for physical cables. The speed controllers, which communicate by I2C, have been programmed to return the rotation period of each motor. This information is used in a proportional-integrator (PI) loop to precisely control rotational velocity. This enables the operator to maintain a motor rotation speed despite changes in wind velocity or angle of attack. Model geometry is displayed in Table 1, while Figure 5 displays general information about the data acquisition system.

Wingspan	360mm
Chord	220mm
Wing thick.	20 mm
Airfoil	MH 45
Wing Area	792cm ²
Aspect Ratio	1.64
Aileron Area	216cm ²
Motors	Pulso - Brushless 2203/52
Propellers	GWS - EP 6030(R)-3

Table 1: Model information

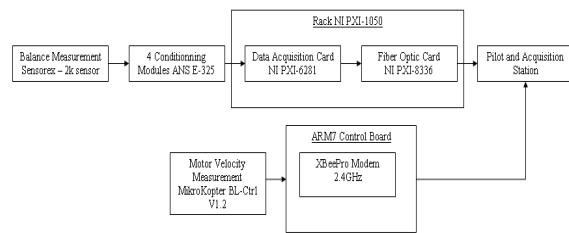


Figure 5: Acquisition System

2.2 Procedure

The balance was mounted in the tunnel without the model to first determine drag created by the struts at varying velocities. These effects were subtracted at real-time from the drag measurements on the model. The balance was recalibrated before each series of tests and gravity effects were eliminated.

Because the test section is closed, large pitching angles may cause the flow to impinge upon the lower vane surface. To evaluate this phenomenon, the model was oriented at zero pitch with zero upstream velocity. The throttle was then set to produce a 3.2N drag force, equivalent to weight of the aircraft, with a slight elevator deflection to remove lift generated by the model asymmetry. This configuration simulates vertical flight with no wall impingement. Then, the model was rotated to 30 and 60 degrees without modifying throttle and elevator deflection. Figure 6 demonstrates that for the range of angle of attack used in this study, ground effect does not observably modify the measured forces. Other forces and moments were not measurably affected by the possible flow impingement on the lower wall.

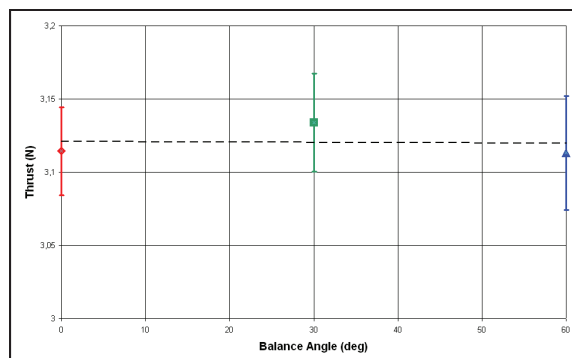


Figure 6: Loiter configuration with various balance angle

It is also important to note that as the model is rotated a larger surface area ($\times 10$) is presented to the flow resulting in increased blockage. This effect is further complicated by the presence of the rotating propellers. A blockage model is not

presented here, but future experimental and computational studies are planned to identify corrections to dynamic pressure, lift and drag [5].

The control variables used in the experiment are elevator angle of deflection, motor rotational speed, and upstream velocity. Angle of attack was held fixed to eliminate hysteresis effects caused by vibration of the model while rotating. Tunnel velocity, elevator deflection, and motor speed were then modified to achieve equilibrium, as described on the Figure 7, that is:

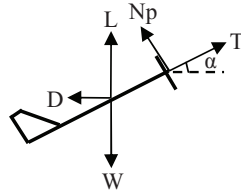


Figure 7: Forces applied to the model

$$(1) \begin{aligned} L_T &= T \sin \alpha + N_p \cos \alpha + L = W \\ D_T &= T \cos \alpha - N_p \sin \alpha - D = 0 \\ M_y &= 0 \end{aligned}$$

Above, L is lift, W is weight of the vehicle, D is drag, T is thrust, N_p is lateral force generated by the propeller and M_y is the moment calculated about the center of gravity. The total lift and drag, L_T and D_T , represent a summation of all forces in the inertial frame x and z directions, respectively. During testing, velocity was modified until the total lift was equal to the weight. Next, the throttle was changed so the total drag was balanced to zero. Then, the flaps were deflected until the moment was eliminated. However, modifying any of the three control variables resulted in a change in all three measurements. Therefore, multiple iterations of the process were necessary. Once the equilibrium point was located, the reference or trim velocity was recorded. Additional data were taken by independently modifying the velocity, motor speed, and elevator position from the equilibrium point. Four additional data points were recorded for each variable as described in Table 2. This data will be used to establish near-equilibrium aerodynamic coefficients and control effectiveness.

	Angle of attack (deg)	Velocity (m/s)	Rotational speed (rad/s)	Aileron deflection (deg)
Equilibrium Point	Fixed	Fixed	Fixed	Fixed
Upstream velocity sweep	Fixed	± 0.3 ± 0.6	Fixed	Fixed
Rotational speed sweep	Fixed	Fixed	± 30 ± 60	Fixed
Elevator deflection sweep	Fixed	Fixed	Fixed	± 4 ± 8

Table 2: Equilibrium investigation

3 WIND TUNNEL CAMPAIGN RESULTS

3.1 Equilibrium Transition

Velocity, elevator deflection, and motor speed values corresponding to equilibrium were found for angles of attack varying between 15 – 65 degrees. Below 20 degrees or above 8 m/s, the required motor velocity began to sharply increase. This is accompanied by an increase in required

current. Although the aircraft is able to fly higher velocities, it does so at the expense of flight time. The maximum flight speed was not discovered during testing.

At the other end of the spectrum, above 65 degrees, the flow speed of the wind-tunnel descended below 2 m/s, where the quality of the flow is not appropriate for highly sensitive measurements. The equilibrium curves are displayed in Figures 8 – 10.

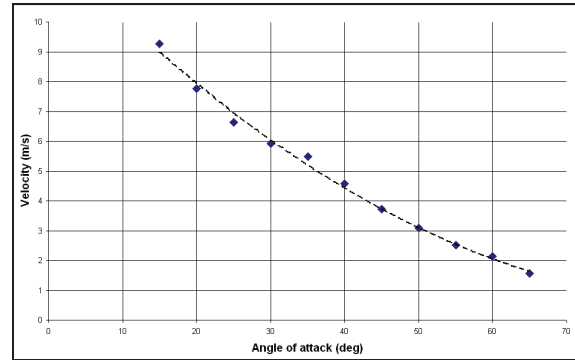


Figure 8: Equilibrium curve Wind-Tunnel Velocity (m/s) for a specified Angle of Attack (deg)

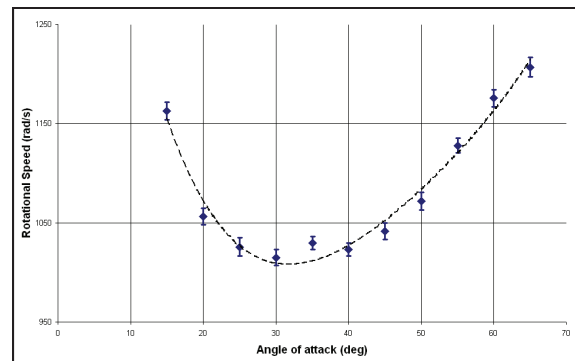


Figure 9: Equilibrium curve model motor speed (rad/s) for a specified Angle of Attack (deg)

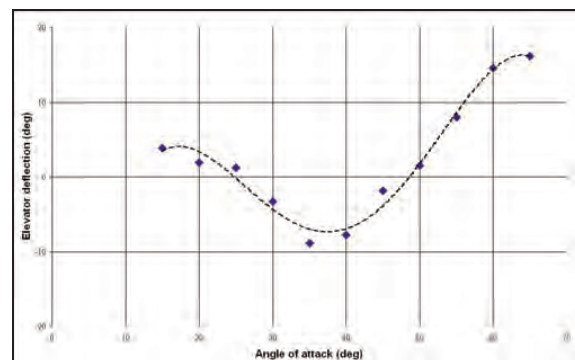


Figure 10: Equilibrium curve elevator deflection (deg) for a specified Angle of Attack (deg)

These plots represent the trim condition of the aircraft for various angles of attack. Henceforth equilibrium or trim values are represented with the $_0$ subscript.

Without motors the aircraft would normally stall at 20 degrees angle of attack. However, due to the momentum injected into the flow by the propwash, stall is never

encountered. However, between 20 – 40 degrees angle of attack, corresponding to 6 – 4 m/s, the required propeller rotation speed reaches a minimum. This the point at which the least power is required of the motors to maintain equilibrium flight. The cruise speed of the aircraft should be set between 4 – 6 m/s to increase endurance for long flights. This is also the region where the momentum induced by the propwash is lowest; therefore the danger of stalling the aircraft is likely highest. The controller chosen for the aircraft should avoid sudden decreases in motor speed in this region.

3.2 Near Equilibrium Aerodynamic Model

Once an equilibrium point was found for each angle of attack, data was then taken for near-equilibrium conditions by modifying the velocity, flaps, and motor speed independently. This data was used to explore the sensitivity of the lift, drag, and pitching moment to the various parameter changes. For example, the change in lift due to a change in elevator angle is shown in Figure 11 for several test points. After nondimensionalizing, the slope of this line is the control derivative, C_{LTe} . Note the T subscript indicates total lift as defined by Equation 1. Figure 12 shows the flap control derivative relationship with angle of attack, along with an appropriate 2nd order polynomial fit.

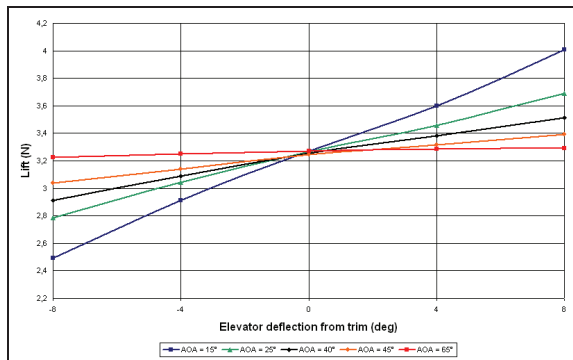


Figure 11: Lift force produced for a given elevator angle (deg) at various angles of attack (deg)

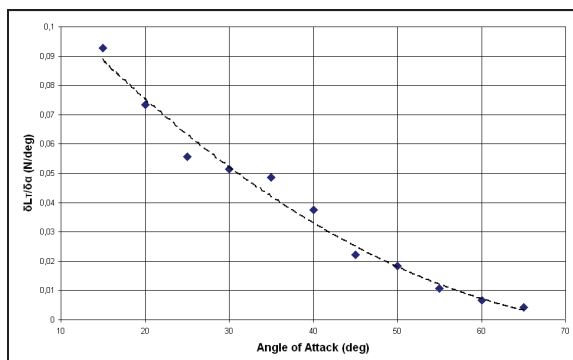


Figure 12: Lift-flap control derivative ($C_{LT\alpha}$) vs angle of attack (deg)

Control derivatives for flap-drag and flap-pitching moment, C_{DTe} and C_{mTe} , were also measured. The effect of changing motor rotation speed, ω (rad/s), can also be expressed as control derivatives: $C_{LT\omega}$, $C_{DT\omega}$, and $C_{mT\omega}$. Variations in coefficients due to changes in velocity, C_{LTV} , C_{DTV} , C_{mTV} and are known as stability derivatives. During analysis, it was observed that a velocity squared term was

also necessary to model the data: C_{LTV2} , C_{DTV2} and C_{mTV2} . This tendency is likely a result of the complicated interaction between propwash and upstream flow. These twelve derivatives enable calculation of lift, drag, and pitching moment for a given angle of attack, velocity, elevator deflection, and motor speed as demonstrated in Equation 2.

$$\begin{aligned} C_{LT} &= C_{LT0} + C_{LT}(\alpha)(\alpha - \alpha_0) + C_{LT\epsilon}(\epsilon)(\epsilon - \epsilon_0) + C_{LT}(\omega)(\omega - \omega_0) \\ &+ C_{LTV}(\omega)(V - V_0) + C_{LTV2}(\omega)(V^2 - V_0^2) \\ (2) \quad C_{DT} &= C_{DT0} + C_{DT}(\alpha)(\alpha - \alpha_0) + C_{DT\epsilon}(\epsilon)(\epsilon - \epsilon_0) + C_{DT}(\omega)(\omega - \omega_0) \\ &+ C_{DTV}(\omega)(V - V_0) + C_{DTV2}(\omega)(V^2 - V_0^2) \\ C_{mT} &= C_{mT0} + C_{mT}(\alpha)(\alpha - \alpha_0) + C_{mT\epsilon}(\epsilon)(\epsilon - \epsilon_0) + C_{mT}(\omega)(\omega - \omega_0) \\ &+ C_{mTV}(\omega)(V - V_0) + C_{mTV2}(\omega)(V^2 - V_0^2) \end{aligned}$$

Note that these stability and control derivatives are not defined by a change from zero, but by the perturbations from the trim condition. Resulting force and moment calculations are only valid in regions near equilibrium. Also because at equilibrium the drag and moment are zero, there are no C_{DT0} or C_{mT0} terms. The derivatives and the zero lift coefficient C_{LT0} were determined as polynomial functions of angle of attack. The polynomial coefficients for these functions are listed in Appendix A.

4 CONTROL DURING EQUILIBRIUM TRANSITION

The equilibrium data discovered during the campaign can be immediately applied to the current PID control of the aircraft. Knowing motor speed as a function of theta (Figure 9) allows definition of the trim value for throttle as a function of θ . This relationship can be added directly to the PID altitude loop used to determine throttle setting. Similarly, the velocity hold calculation benefits from addition of the trim pitch angle as a function of the desired velocity. By further exploring the data obtained from perturbations from equilibrium, however, an entirely different controller may be designed.

Classical control theory uses a linear time invariant (LTI) model to synthesize static or dynamic controllers. It is usually represented by either its state-space form or its transfer function. The relation between both representations is shown by (3).

$$(3) \quad \begin{cases} \dot{x} = Ax + Bu \\ y = Cx + u \end{cases}, \quad G(s) = D + C(sI - A)^{-1}B$$

However, this representation is only valid in the neighborhood of an equilibrium point. The transition from hover to cruise and vice versa quickly traverses a set of equilibrium points. A representation of the system that enables proof of stability over the whole transition process is desired. Linear Parameter Varying (LPV) models are a suitable choice. Equation (4) shows such a representation where $\theta(t)$ is a time-varying vector. When some states of the system are in $\theta(t)$ the model is called quasi-LPV.

$$(4) \quad \begin{cases} \dot{x} = A(\theta(t))x + B(\theta(t))u \\ y = C(\theta(t))x + D(\theta(t))u \end{cases}$$

It is easy to see the changing dynamics of the MAVion during transition using the pitch angle as a parameter. As a beginning, well-established control methodologies can be used. First stability and performance analysis (μ -analysis [7]) of previously designed controllers guarantees the behavior of the system along the trajectory. Robust synthesis (H_∞ synthesis) provides tools to optimize a fixed linear dynamic controller such that the closed loop reaches the performance objective. Using a fixed controller to ensure the performance is an ambitious task.

Many aerospace control problems must deal with changing dynamics with operating point. Thanks to some knowledge of the operating point, a fixed controller can be interpolated: this is known as gain-scheduling. For sufficient slow variation, gain scheduling was shown practically efficient for more than thirty years and later proven theoretically under some hypotheses [8].

Linear Parameter Varying (LPV) controller synthesis methodologies [10] attempt to design a controller explicitly using the information about the current operating point, as described Figure 13. It gives stronger stability results than gain-scheduling.

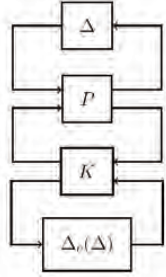


Figure 13: LPV Control Closed Loop, top half LFR, bottom half controller

In order to implement an LPV controller it is necessary to reformulate the model shown in Equation 4 as a Linear Fractional Representation (LFR), pictured shown as the top two blocks in Figure 13.

4.1 Linear Fractional Representation (LFR) Model

The linear fractional representation (LFR) formalism is a theoretical tool that enables one to describe any rational function in some parameters in terms of the feedback interconnection as shown in Figure 14. The variables expressing the rational dependence are grouped in Δ . The reference documentation of the LFRT [6], a Matlab Toolbox dedicated to the manipulation of such representations, gives an extensive description of this formalism and how to build LFR models.

Any rational function $K(\delta_1, \delta_2, \dots, \delta_N)$ admits a Linear Fractional Representation:

$$(5) \quad \begin{cases} z = G_{11}w + G_{12}u \\ y = G_{21}w + G_{22}u \end{cases}, \quad w = \Delta z, \quad G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$$

with $\Delta = \text{diag}(\Delta_1 I_{d_1}, \Delta_2 I_{d_2}, \dots, \Delta_N I_{d_N})$

such that $y = K(\Delta_1, \Delta_2, \dots, \Delta_N)u$. For the present case, K would correspond to the coefficients in Equation 2, while there is only δ_1 representing angle of attack.

The transfer between y and u is written $\mathcal{F}_u(G, \Delta)$.

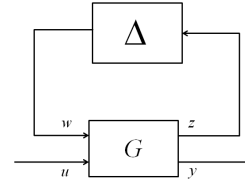


Figure 14: LFT Interconnection

In modern control theory Linear Fractional Transformation is used to separate any time varying or static non-linearities from a nominal linear system. Uncertainties, time-varying parameters or non-linearities complicate any control problem and thus are pulled out in the Δ block. The LFR consists of a nominal LTI system interconnected with an operator Δ modeling the uncertainties or other behaviors. The nominal system is described as a LTI system with exogenous inputs and outputs along with the control inputs and measurement outputs. The exogenous outputs are multiplied by an uncertainty block to be then fed back to the inputs. For a more detailed development of how equation 3 relates to Figure 14, the reader may wish to reference Appendix C.

To show how one can transform a polynomial into an LFR, take the lift trim coefficient (6) as an example. Look for the LFT such that:

$$(6) \quad y = C_{L0}(\alpha)u = F_u(G, \Delta)u, \quad \text{with } \Delta = \alpha I$$

Use Hörner factorization and identify the LFR interconnection signals.

$$(7) \quad \begin{aligned} C_{L0}(\alpha) &= a^4 + b^3 + c^2 + d + e \\ &= e + (d + (c + (b + \underbrace{\alpha}_{z_1}))) \end{aligned}$$

This results in the following LFR described by Figure 15:

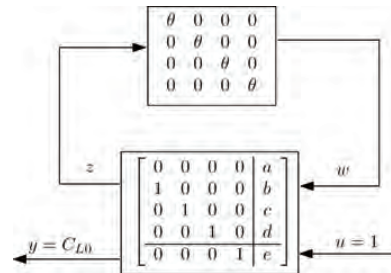


Figure 15: LFR Interconnection

The LFR set has a convenient property: stability through several algebraic operations such as multiplication, addition and inversion. One can thus build the system dynamic by manipulating LFR of the coefficients. Further details on the link between equations (4) and (5) are provided in Appendix C.

4.2 Linearization

It is desirable to form a linear model of the form shown in Equation 3. Taking the state and control vectors to be as follows:

$$(8) \quad \begin{aligned} x &= [q \quad V \quad V_z]^T \\ u &= [e \quad \omega]^T \end{aligned}$$

Here, θ is the pitch angle. Assuming there is very small perturbation in wind, this is equal to the angle of attack presented previously. q is the pitching rate in radians per second, V is the aircraft inertial frame horizontal velocity and V_z is the vertical velocity also in the inertial frame. The control variables are e , elevator in radians and ω , motor rotational velocity in radians per second. The rate vector can be defined in terms of the aerodynamic coefficients:

$$(9) \quad \dot{x} = \begin{bmatrix} q \\ \frac{1}{J_{yy}} \left(\frac{1}{2} S_{ref} L_{ref} V^2 C_{mT} \right) \\ \frac{1}{m} \left(\frac{1}{2} S_{ref} V^2 C_{DT} \right) \\ \frac{1}{m} \left(\frac{1}{2} S_{ref} V^2 C_{LT} \right) \end{bmatrix}$$

Equation 10 represents a non-linear model of the behavior of the aircraft, however, the aerodynamic coefficients are only identified near equilibrium suggesting a linearized model is needed. The rate vector for x can be linearized as:

$$(10) \quad \dot{x} = \begin{bmatrix} \frac{\partial \dot{q}}{\partial (\partial)} + \frac{\partial \dot{q}}{\partial e} (\partial e) + \frac{\partial \dot{q}}{\partial V} (\partial V) + \frac{\partial \dot{q}}{\partial V_z} (\partial V_z) \\ \frac{\partial \dot{V}}{\partial (\partial)} + \frac{\partial \dot{V}}{\partial e} (\partial e) + \frac{\partial \dot{V}}{\partial V} (\partial V) + \frac{\partial \dot{V}}{\partial V_z} (\partial V_z) \\ \frac{\partial \dot{V}_z}{\partial (\partial)} + \frac{\partial \dot{V}_z}{\partial e} (\partial e) + \frac{\partial \dot{V}_z}{\partial V} (\partial V) + \frac{\partial \dot{V}_z}{\partial V_z} (\partial V_z) \end{bmatrix}$$

The derivatives are then separable into the A and B matrices found in Equation 12-13:

$$(11) \quad A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial \dot{q}}{\partial (\partial)} & 0 & \frac{\partial \dot{q}}{\partial V} & 0 \\ \frac{\partial \dot{V}}{\partial (\partial)} & 0 & \frac{\partial \dot{V}}{\partial V} & 0 \\ \frac{\partial \dot{V}_z}{\partial (\partial)} & 0 & \frac{\partial \dot{V}_z}{\partial V} & 0 \end{bmatrix}$$

$$(12) \quad B = \begin{bmatrix} 0 & 0 \\ \frac{\partial \dot{q}}{\partial e} & \frac{\partial \dot{q}}{\partial \omega} \\ \frac{\partial \dot{V}}{\partial e} & \frac{\partial \dot{V}}{\partial \omega} \\ \frac{\partial \dot{V}_z}{\partial e} & \frac{\partial \dot{V}_z}{\partial \omega} \end{bmatrix}$$

Combining this linearization to the relationships in Equation 10, one can define expressions for the above

derivatives. These are presented in Appendix B. The first derivative is provided as an example:

$$(13) \quad \frac{\partial \dot{q}}{\partial (\partial)} = \frac{1}{2J_{yy}} S_{ref} L_{ref} V_0^2 \begin{bmatrix} -C_{m\dot{e}} \frac{\partial e_0}{\partial (\partial)} - C_{m\dot{V}} \frac{\partial V_0}{\partial (\partial)} \\ -C_{m\dot{V}_z} \frac{\partial V_{z0}}{\partial (\partial)} - 2C_{m\dot{V}_z} V_0 \frac{\partial V_0}{\partial (\partial)} \end{bmatrix}$$

The twelve derivatives contained in **A** and **B** vary as functions of pitch. As each coefficient and equilibrium parameter is a fourth order polynomial, the derivative is an nineteenth order polynomial and the Δ block of the associated LFR will be of equal size. The dimension of the Δ block is critical to the success of the numerical procedures used for controller synthesis. Additionally, this model, though nonlinear in the parameter, is a family of linearization and in such remains valid only if the states and inputs stay near the equilibrium trajectory.

5 CONCLUSION AND FUTURE WORKS

A wind tunnel campaign was performed to reveal the trim state of a fixed wing VTOL MAV during the transition between horizontal and vertical flight modes. Further exploration of perturbations around the trim state yielded a linearized model whose stability and control derivatives are functions of angle of attack. The model has been expressed in Linear Fractional Representation as a first step for use in adaptive control methods such as LPV.

Future aerodynamic testing will investigate the specific nature of the flow. Propwash impingement on the wing, swirl effects, and stall are areas requiring more detailed studies. Numerical simulations are also in progress to validate the trim model and evaluate the blockage effect in the wind-tunnel test section. A structured C-grid with an actuator disk is used to model the aircraft and the propellers as shown Figure 16.

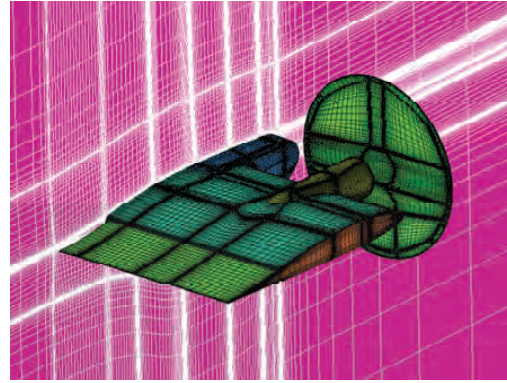


Figure 16: Structured C-grid around the MAVion

The end goal of the aerodynamic testing and controls development presented here is to produce aircraft uniquely capable of vertical and horizontal flight modes. Several prototypes have already been constructed and outfitted with autopilots developed at ISAE. There are two varieties of the aircraft in development. The first is equipped with GPS for outdoor navigation, and has demonstrated fully autonomous transitions between vertical and horizontal flight. It is able to drop a small payload and auto take-off and land.

The second version, destined for mostly indoor flight, has carbon fibre wheels attached which are used to roll along the

ground, walls, and ceilings guided by an attached camera using First Person Vision (FPV). Both versions feature quaternion based PID controllers for stability. Future work will utilize the model that has been developed here to implement modern adaptive controllers to improve the performance of the aircraft as it transitions between vertical and horizontal flight modes.

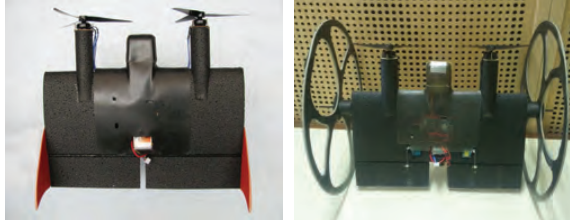


Figure 17: Two MAVion prototypes. At left is the outdoor GPS equipped version. At right is the wheeled version for indoor vision based flight.

ACKNOWLEDGMENT

The present work has been partially supported by the Pôle de recherche et d'enseignement supérieur (PRES) of the University of Toulouse, the European Office for Aerospace Research and Development (EOARD), the Direction Générale de l'Armement (DGA) and the US Air Force. The authors would like to express their gratitude for the contribution of Sylvain Masseboeuf in conducting the wind tunnel experiments and to thank the other members of the ISAE for their involvement to this work: Rémi Chanton, Xavier Foulquier, Serge Gérard, Marc Grellet, Guy Mirabel, Patrick Morel and Serge Moretti. For their helpful discussions and contributions to the present study, the authors would also like to thank the students: Roméo Brogna, Hugo Martin, Mathieu Melgrani and Amelie Peyret.

REFERENCES

- [1] C. Thypopoulos, B. Bataillé, J.M. Moschetta. A Fixed Wing Micro Air Vehicle for Transition Flight: Aerodynamic and Propulsion Study. *IMAV Conference, Pensacola, Florida USA*, July 2009.
- [2] R. Carr, J.M. Moschetta, C. Thypopoulos, G. Mehta. *IMAV Conference 2010, Braunschweig, Germany*, July 2010.
- [3] R.H. Stone and G. Clarke. Optimization of Transition Manoeuvres for a Tail-Sitter Unmanned Air Vehicle (UAV). *Australian International Aerospace Congress, Canberra, Australia*, 2001
- [4] S. Masseboeuf. Manuel d'Utilisation de la Soufflerie Sabre. *Internal publication for ISAE, Toulouse, France*, November 2010.
- [5] A. Krynytzky. Classical Corrections for Closed Test Sections. In B.F.R. Ewald, editors, Advisory Group for Aerospace Research & Development, *Darmstadt University of Technology, Federal Republic of Germany*, volume 336, pages 2-13 - 2-35. Canada Communication Group Inc., 1998.
- [6] J. Magni, User Manual of the Linear Fractional Representation Toolbox Version 2.0. *Onera DCSD*
- [7] John C. Doyle; Structured uncertainty in control system design, *Decision and Control, 1985 24th IEEE Conference on*, vol.24, no., pp.260-265, Dec. 1985
- [8] J. S. Shamma, Analysis and design of gain scheduled control systems, Phd Thesis 1988.
- [9] C. W. Scherer, LPV control and full block multipliers, *Automatica*, vol. 37, no. 3, pp. 361-375, 2001.
- [10] P. Apkarian and P. Gahinet, A convex characterization of gain-scheduled H infinity controllers, *IEEE Transactions on Automatic Control*, vol. 40, pp. 853-864, may 1995.

APPENDIX A: POLYNOMIAL COEFFICIENTS FOR FUNCTIONS OF PITCH ANGLE

The polynomials are represented as follows:

$$V_0(\cdot) = a^4 + b^3 + c^2 + d + e$$

with in radians.

	a	b	c	d	e
V_0	5,97E+01	-1,63E+02	1,63E+02	-7,82E+01	2,13E+01
e_0	-1,07E+01	2,96E+01	-2,71E+01	9,43	1,03
ω_0	4,67E+06	-9,96E+03	1,10E+04	-5,38E+03	1,97E+03
C_{LTV}	-2,07E+01	9,96E+01	-1,07E+02	4,54E+01	-5,58
C_{LTe}	3,12E+01	-7,03E+01	5,54E+01	-2,04E+01	9,55E-01
C_{LTW}	5,20E-03	4,16E-02	-5,89E-02	2,91E-02	-4,00E-03
C_{LTV}	-8,57E+02	1,82E+03	-1,40E+03	4,51E+02	-5,14E+01
C_{LTV2}	2,09E+02	-4,58E+02	3,60E+02	-1,19E+02	1,38E+01
C_{DTe}	2,45E+01	-8,85E+01	8,88E+01	-3,69E+01	4,64
C_{DW}	3,57E-02	-1,03E-01	9,51E-02	-3,72E-02	4,30E-03
C_{DTV}	-2,60E+02	7,25E+02	-6,64E+02	2,47E+02	-3,08E+01
C_{DTV2}	3,40E+01	-1,03E+02	9,88E+01	-3,77E+01	4,84
C_{mTe}	-1,70E+01	3,32E+01	-2,51E+01	8,15	-3,83E-01
C_{mTW}	-9,00E-05	-1,64E-02	2,25E-02	-1,05E-02	1,40E-03
C_{mTV}	-1,60E+02	4,61E+02	-4,33E+02	1,63E+02	-2,06E+01
C_{mTV2}	1,53E+01	-5,02E+01	5,02E+01	-1,95E+01	2,53

APPENDIX B: STABILITY AND CONTROL DERIVATIVES

$$\frac{\partial \dot{q}}{\partial} = \frac{1}{2J_{yy}} S_{ref} L_{ref} V_0^2 \begin{bmatrix} -C_{m_e} \frac{\partial e_0}{\partial} - C_{m_e} \frac{\partial \omega_0}{\partial} \\ -C_{mV} \frac{\partial V_0}{\partial} - 2C_{mV2} V_0 \frac{\partial V_0}{\partial} \end{bmatrix}$$

$$\frac{\partial \dot{q}}{\partial V} = \frac{1}{2J_{yy}} S_{ref} L_{ref} V_0^2 [C_{mTV} + 2V_0 C_{mTV2}]$$

$$\frac{\partial \dot{q}}{\partial e} = \left(\frac{1}{2J_{yy}} S_{ref} L_{ref} V_0^2 \right) C_{mTe}(\cdot)$$

$$\frac{\partial \dot{q}}{\partial} = \left(\frac{1}{2J_{yy}} S_{ref} L_{ref} V_0^2 \right) C_{mT}(\cdot)$$

$$\frac{\partial \dot{V}}{\partial} = \frac{1}{2m} S_{ref} V_0^2 \begin{bmatrix} \frac{\partial C_{DT0}}{\partial} - C_{DT_e} \frac{\partial e_0}{\partial} - C_{DT_e} \frac{\partial \omega_0}{\partial} \\ -C_{DTV} \frac{\partial V_0}{\partial} - 2C_{DTV2} V_0 \frac{\partial V_0}{\partial} \end{bmatrix}$$

$$\frac{\partial \dot{V}}{\partial V} = \frac{1}{2m} S_{ref} V_0^2 [C_{DTV} + 2V_0 C_{DTV2}]$$

$$\frac{\partial \dot{V}}{\partial e} = \left(\frac{1}{2m} S_{ref} V_0^2 \right) C_{DTTe}(\cdot)$$

$$\frac{\partial \dot{V}}{\partial} = \left(\frac{1}{2m} S_{ref} V_0^2 \right) C_{DT}(\cdot)$$

$$\frac{\partial \dot{V}_z}{\partial} = \frac{1}{2m} S_{ref} V_0^2 \begin{bmatrix} \frac{\partial C_{LTV}}{\partial} - C_{LTV_e} \frac{\partial e_0}{\partial} - C_{LTV_e} \frac{\partial \omega_0}{\partial} \\ -C_{LTV} \frac{\partial V_0}{\partial} - 2C_{LTV2} V_0 \frac{\partial V_0}{\partial} \end{bmatrix}$$

$$\frac{\partial \dot{V}_z}{\partial V} = \frac{1}{2m} S_{ref} V_0^2 [C_{LTV} + 2V_0 C_{LTV2}]$$

$$\frac{\partial \dot{V}_z}{\partial e} = \left(\frac{1}{2m} S_{ref} V_0^2 \right) C_{LTVTe}(\cdot)$$

$$\frac{\partial \dot{V}_z}{\partial} = \left(\frac{1}{2m} S_{ref} V_0^2 \right) C_{LTV}(\cdot)$$

APPENDIX C : LINEAR FRACTIONAL REPRESENTATION

Any rational function $K(\delta_1, \delta_2, \dots, \delta_N)$ admits a Linear Fractional Representation:

$$(C.1) \quad \begin{cases} z = G_{11}w + G_{12}u \\ y = G_{21}w + G_{22}u \end{cases}, \quad w = \Delta z, \quad G = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}$$

$$\Delta = \begin{bmatrix} \Delta_1 I_{d_1} & 0 & \dots & 0 \\ 0 & \Delta_2 I_{d_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Delta_N I_{d_N} \end{bmatrix}$$

such that $y = K(\Delta_1, \Delta_2, \dots, \Delta_N)u$.

The transfer between y and u is written $\mathcal{F}_u(G, \Delta)$.

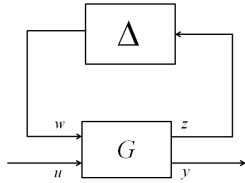


Figure C1 : LFT Interconnection

Equation (C.1) represents the most general case of a LFR. The Δ block contains only static parameters but using the integrator operator, $1/s$, dynamical system can be represented. The state space representation of a system can then be seen as the LFR of $G(s)$ with $\Delta = 1/sI_n$, and n the number of states of the system.

$$(C.2) \quad \begin{cases} \dot{x} = Ax + Bu \\ y = Cx + u \end{cases}, \quad G(s) = D + C(sI - A)^{-1}B$$

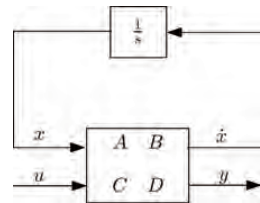


Figure C2 : Dynamic system

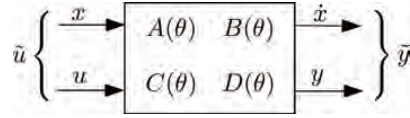
The state-space matrices may depend on a parameter vector as described by Equation (C.4).

$$(C.4) \quad \begin{cases} \dot{x} = A(\theta)x + B(\theta)u \\ y = C(\theta)x + D(\theta)u \end{cases}$$

Provided that this dependence is rational, an LFR can be derived. First, the following notations are introduced:

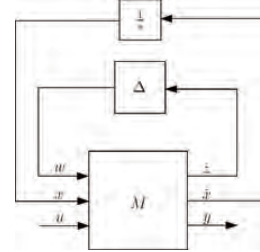
$$(C.5) \quad \tilde{y} = \begin{bmatrix} \dot{x} & y \end{bmatrix}^T, \quad \tilde{u} = \begin{bmatrix} x & u \end{bmatrix}^T,$$

$$K = \begin{bmatrix} A(\theta) & B(\theta) \\ C(\theta) & D(\theta) \end{bmatrix}$$



K admits a LFR representation such that $\tilde{y} = F_u(G, \Delta)\tilde{u}$. G is usually decomposed as follows :

$$(C.6) \quad \begin{bmatrix} z \\ \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} D_{11} & C_1 & D_{12} \\ B_1 & \tilde{A} & B_2 \\ D_{21} & C_2 & D_{22} \end{bmatrix} \begin{bmatrix} w \\ x \\ u \end{bmatrix}$$



After a suitable vector reordering, this formulation leads to the traditional notation for a LFR state-space system found in the literature:

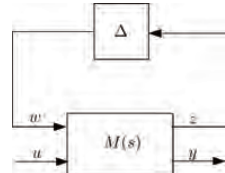
$$(C.7) \quad \begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} \tilde{A} & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

The original state-space matrices can be expressed using the new variables by closing the loop with the Δ block. The state space matrices of equation (4) then become:

$$(C.8) \quad \begin{aligned} A(\theta) &= \tilde{A} + B_1(I - \Delta D_{11})^{-1}\Delta C_1 \\ B(\theta) &= B_2 + B_1(I - \Delta D_{11})^{-1}\Delta D_{12} \\ C(\theta) &= C_2 + D_{21}(I - \Delta D_{11})^{-1}\Delta C_1 \\ D(\theta) &= D_{22} + D_{21}(I - \Delta D_{11})^{-1}\Delta D_{12} \end{aligned}$$

In turn, if the loop is closed around the integrator operator, a transfer function can be derived:

$$(C.9) \quad \begin{aligned} G_{11}(s) &= C_1(sI - \tilde{A})^{-1}B_1 + D_{11} \\ G_{12}(s) &= C_1(sI - \tilde{A})^{-1}B_2 + D_{12} \\ G_{21}(s) &= C_2(sI - \tilde{A})^{-1}B_1 + D_{21} \\ G_{22}(s) &= C_2(sI - \tilde{A})^{-1}B_2 + D_{22} \end{aligned}$$



Vortex-lift modeling provides reliable force predictions for flapping-wing micro air vehicles

W. Thielicke^{1,2}, A.B. Kesel¹ and E.J. Stamhuis^{1,2}

¹Biomimetics-Innovation-Centre, University of Applied Sciences Bremen, Germany

²Ocean Ecosystems, University of Groningen, The Netherlands

ABSTRACT

Vertical and horizontal force of a flapping-wing micro air vehicle (MAV) has been measured in slow-speed forward flight using a force balance. Detailed information on kinematics was used to estimate forces using a blade-element analysis. Input variables for this analysis are lift and drag coefficients. These coefficients are usually derived from steady-state measurements of a wing in translational flow. Previous studies on insect flight have shown that this method underestimates forces in flapping flight, mainly because it cannot account for additional lift created by unsteady phenomena. We therefore derived lift and drag coefficients using a concept for delta-wings with stably attached leading edge vortices. Resulting lift coefficients appeared to be a factor of 2.5 higher than steady-flow coefficients, and match the results from previous (numerical) studies on instantaneous lift coefficients in flapping flight. The present study confirms that a blade-element analysis using force coefficients derived from steady-state wind tunnel measurements underestimates vertical force by a factor of two. The equivalent analysis, using "vortex-lift" enhanced coefficients from a delta-wing analogue, yields very good agreement with force balance measurements, and hence seems to be a good approximation for lift-enhancing flow phenomena when modeling flapping flight.

1 INTRODUCTION

The desire to understand the aerodynamics of flapping flight in insects, birds and bats has been the motivation of many studies in the past. Early attempts applied the blade-element theory (BET), a theory often used to estimate thrust and torque of revolving propellers, to explain forces required during sustained insect hovering flight [1]. The basis of this theory is a "quasi-steady" approach that assumes the instantaneous forces of a flapping wing to be identical to the forces of the same wing under steady motion with identical angle of attack and velocity [1]. The idea of the BET is to divide the wings into small elements along the wing span. For each element, the effective angle of attack as well as the instantaneous flow velocity is derived from detailed time-resolved information on the kinematics of the flapping wing. The forces created by each element can be calculated when lift and drag coefficients of the wing sections are known. Usually, these coefficients are derived from static force measurements of a series of angles of attack of the airfoils under steady-flow conditions in a

wind tunnel. However, the application of the BET appeared to seriously underestimate the forces observed in flapping insect flight [1]-[3]. By studying the flow around flapping robotic insect wings, Ellington *et al.* [4] identified an explanation for this discrepancy. In a scaled model of a hovering hawkmoth, they observed large vortices on top of the wings increasing the circulation and therefore the aerodynamic forces. These leading edge vortices (LEVs) remain stably attached to the wing and contribute substantially to lift throughout the full downstroke by increasing the amount of bound circulation of the wing.

Subsequent studies identified LEVs in other insects, robotic flapping-wing devices, hovering birds and slow-flying bats (e.g. [5]-[9]). Lentink *et al.* [10] suggest that LEVs are a universal and efficient high lift mechanism for slow flapping flight over a quite large range of animal sizes.

The amplifying effect of these vortices on the lift and drag coefficients during wing flapping [6] is not present when determining lift coefficient (C_L) and drag coefficient (C_D) from steady-flow force measurements in a wind tunnel. Hence, forces calculated with a blade-element analysis underestimate forces of flapping wings. Although the discovery of LEVs in insect flight substantially contributed to understanding the mechanics of flapping flight, these vortices were well known to aircraft designers before they were found in nature: During relatively slow flight, delta-wing aircrafts like the Concorde largely rely on lift created by additional circulation of stable leading edge vortices (e.g. [11]). The sharp leading edges of the wings of such aircrafts induce flow separation, a feature that can also be found on insect and bird wings (e.g. [12]). In delta-wing aircrafts, vortices are stabilized by the wing sweep which allows for a spanwise flow parallel to the swept leading edge, convecting vorticity to the wing tip and preventing the LEV to grow and detach [13]. Although the stabilization mechanisms for the LEVs in delta-wing aircraft and flapping insect wings are probably not exactly the same (e.g. [10], [14]), the flow phenomena and the aerodynamic effects of these vortices are analogous [4]. Lift coefficients for delta wings with attached vortical flow on top of the wing range from 4 to 6 [13], which is substantially

Email address: William.Thielicke@hs-bremen.de

higher than the lift coefficients of conventional wings.

Polhamus [11] introduces a concept to predict lift coefficients of sharp-edge delta wings (up to an aspect ratio of 4) based on the combination of potential-flow lift and vortex lift. His theory includes a simple trigonometric relationship between the lift (respectively drag) coefficient and geometric angle of attack. The concept was verified by wind-tunnel measurements of sharp-edge, highly swept wings and provides a very good prediction of total lift [15] which may find wider application than for swept wings only.

In the present study, we measured lift and drag of a simple flapping-wing MAV. The MAV is equipped with bio-inspired wings which have a sharp leading edge at the outer 2/3 of the wing and a round leading edge close to the wing base. Classical lift and drag coefficients are obtained from steady-flow measurements in a wind tunnel. Three-dimensional flow patterns around the same type of wing during flapping were analyzed in an earlier study, showing a prominent and stable leading edge vortex that developed immediately at the beginning of the downstroke [16]. We use a blade-element analysis to estimate aerodynamic forces, by generating a set of force coefficients using the trigonometric relationship proposed for delta-wings [11] to account for the additional circulation generated by LEVs. The results of the blade-element analysis using steady-flow force coefficients and force coefficients from a delta-wing analogue are compared to aerodynamic force measurements at the MAV.

2 MATERIALS AND METHODS

2.1 MAV

The wings of the MAV (see Figure 1) are modeled from 3 mm closed-cell extruded polystyrene foam sheet (DEPRON®). The planform is inspired by the wings of swiftlets (*Collocalia linchi*) with some camber at the base and a sharp leading edge at the outer part of the wing (see Figure 1). The total wing span (tip-to-tip) is 0.33 m with an average chord length of 40 mm and an aspect ratio of 8.3. The wings are mostly rigid showing only some aeroelastic bending near the tip at higher flapping frequencies, similar to the wings of swifts and swiftlets [16]-[18].



Figure 1: Flapping-wing MAV mounted on the force balance

The wings each have two rotational degrees of freedom (shoulder joint: up and down wing excursion; and longitudinal joint: pro- / supination parallel to the spanwise axis, allowing the wings to change geometric angle of attack (α_{geo} , see Figure 5) and are driven by a single small geared DC motor. Flapping frequency (0-9 Hz) was set by altering the voltage of a power supply. The specific arrangement of linkage elements makes the wings supinate during upstroke and pronate during downstroke, resulting in very similar kinematics as in an earlier study that focused on flow measurements [16]. The change of geometric angle of attack and excursion throughout wing beat cycle is shown exemplary for two different situations in Figure 2.

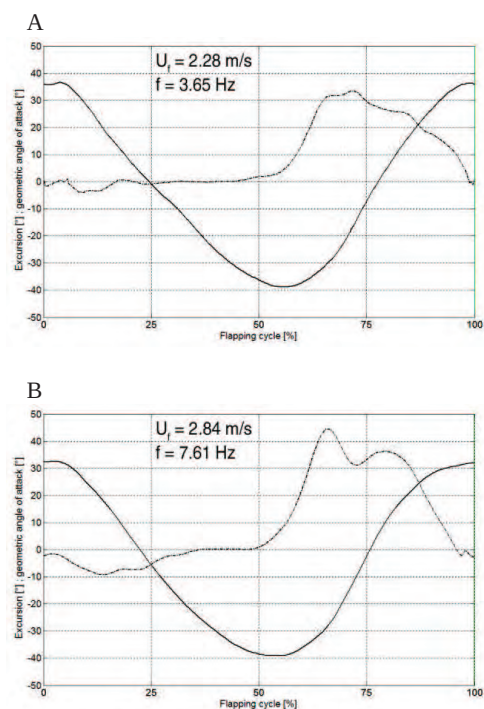


Figure 2: Wing excursion (solid line) and geometric angle of attack (angle between U_f and wing chord; dashed line). (A) Wing kinematics for a flapping frequency of 3.65 Hz in 2.28 m/s flow (B) Wing kinematics for a flapping frequency of 7.61 Hz in 2.84 m/s flow. The kinematics change at increasing flapping frequency and free flow velocity due to increasing aerodynamic and inertial load and some elasticity in the mechanical design

The stroke plane was set to 90° in relation to the free flow. To mimic slow-flight conditions, flow velocities between 2.28 m/s and 2.84 m/s were tested in an open jet low speed wind tunnel (test section diameter = 0.45 m; $u_{\max} = 14$ m/s). The Reynolds number (Re), a measure for the importance of inertial vs. viscous forces, is calculated as

$$(1) \quad Re = \frac{\sqrt{\bar{v}_{tip}^2 + U_f^2} \cdot \bar{c}}{\nu},$$

where $\bar{v}_{tip}$ = mean vertical tip velocity; U_f = free flow velocity; $\bar{c}$ = mean chord; ν = kinematic viscosity

Measurements were done for Re between $8 \cdot 10^3$ and $1.3 \cdot 10^4$. The advance ratio J, given by

$$(2) \quad J = \frac{U_f}{|\bar{v}_{tip}|}$$

is a measure for forward flight speed vs. wing tip velocity in flapping flight. It ranges from 0.6 to 1.7 for the parameters tested in flapping flight, here.

2.2 Flapping flight force measurements

Vertical (F_V , “lift”) and horizontal (F_H , “thrust”) force of the MAV was recorded with a 2-axes force balance (for details see [19]), sampled at 1200 Hz, digitized with an analogue-to-digital converter and processed with MATLAB and Excel. Instantaneous forces of eighteen successive full flapping cycles were recorded for each setup. Eight flapping frequencies between 3.5 and 9 Hz were tested for three flow velocities (2.28; 2.57; 2.84 m/s). Forces were integrated over the wing beat cycle to derive mean horizontal ($\bar{F}_H$) and mean vertical force ($\bar{F}_V$). The mean vertical force coefficient is derived by

$$(3) \quad \bar{C}_V = \frac{2\bar{F}_V}{\rho U_f^2 A},$$

where ρ = density; A = total wing area [20]

2.3 Lift and drag coefficients

Steady-state lift and drag coefficients (subsequently denominated “steady” coefficients) were obtained from measurements of lift and drag of the isolated wings in the same wind tunnel ($Re = 1.4 \cdot 10^4$). Forces were sampled for geometric angles of attack between -45° and 65° (step size 1°, $n = 3$). C_L and C_D were derived via

$$(4) \quad C_L = \frac{2F_L}{\rho U_f^2 A}; \text{ respectively } C_D = \frac{2F_D}{\rho U_f^2 A},$$

where F_L = lift; F_D = drag

Maximum lift coefficient $C_{L,\max}$ is 1.01 ± 0.01 at 11° geometric angle of attack (see Figure 3). For the blade-element analysis, coefficients were stored in a lookup table, non-integer values were determined via linear interpolation.

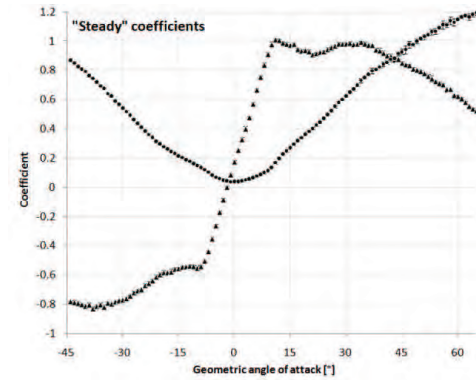


Figure 3: “Steady coefficients”. Lift (triangles) and drag (circles) coefficient of the wings in steady-flow for geometric angles of attack between -45° and 65°

An additional set of lift and drag coefficients was created following [11] as explained in short earlier in this paper (subsequently denominated “vortex-lift” coefficients). For a delta-wing with stable leading edge vortices, total lift coefficient can be approximated using

$$(5) \quad C_L = K_p \sin \alpha \cos^2 \alpha + K_v \cos \alpha \sin^2 \alpha * \frac{\alpha}{|\alpha|} + C_{L_0},$$

where α = angle of attack; K_p = constant of proportionality in potential-flow lift term; K_v = constant of proportionality in vortex lift term; C_{L_0} = lift coefficient of the MAV wings at 0° geometric angle of attack

Polhamus [11] calculated K_p and K_v for aspect ratios up to 4 using a modified Multhopp lifting-surface theory ($K_p = 3.35$; $K_v = 3.45$). Drag coefficient due to lift is given as

$$(6) \quad \Delta C_D = C_L \tan \alpha \quad [15]$$

Total drag coefficient can be approximated as

$$(7) \quad C_D = \Delta C_D + C_{D_0},$$

where C_{D_0} = drag coefficient of the MAV wings at 0° geometric angle of attack

Lift and drag coefficients derived with Equation 5 and 7 are plotted in Figure 4.

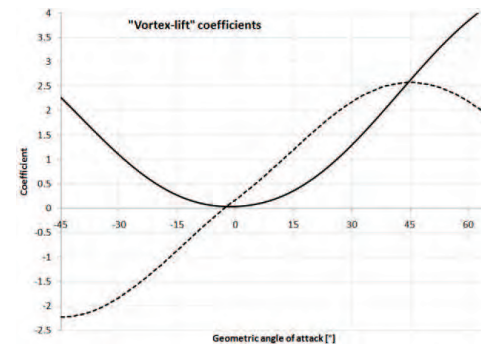


Figure 4: “Vortex-lift” coefficients. Lift (dashed line) and drag (solid line) coefficients for a wing with attached LEVs for geometric angles of attack between -45° and 65°

2.4 Blade-element analysis

We used a blade-element analysis to predict $\overline{F_V}$ of the flapping-wing MAV using data derived from the wing kinematics and the two different sets of force coefficients (“steady” and “vortex-lift” coefficients). The wing planform was digitized and divided into 496 elements in span wise direction. Lift L_r and drag D_r of each element at distance r from the wing base (see Figure 5 for nomenclature) is calculated using the equation

$$(8) \quad L_r(t) = \frac{1}{2} \rho v_r(t)^2 A_r C_L(\alpha_{eff}),$$

where v_r = effective velocity at r ; A_r = area of wing element r ; α_{eff} = effective angle of attack

and

$$(9) \quad D_r(t) = \frac{1}{2} \rho v_r(t)^2 A_r C_D(\alpha_{eff})$$

Effective velocity was calculated as

$$(10) \quad v_r(t) = \sqrt{(r\omega(t))^2 + U_f^2},$$

where r = radial distance of the wing element to the base; ω = angular velocity (derived from kinematics)

C_L and C_D depend on the effective angle of attack (α_{eff}) of the blade element which is calculated following

$$(11) \quad \alpha_{eff,r}(t) = \alpha_{geo}(t) - \alpha_{ind}(t),$$

where α_{geo} = geometric angle of attack (derived from kinematics); α_{ind} = induced angle of attack = $\text{atan}\left(\frac{r\omega(t)}{U_f}\right)$

L_r and D_r were integrated for all wing elements and resolved into horizontal (F_H) and vertical (F_V) force components:

$$(12) \quad F_V(t) = \cos(\alpha_{ind}) * L_r(t) + \sin(\alpha_{ind}) * D_r(t)$$

$$(13) \quad F_H(t) = \sin(\alpha_{ind}) * L_r(t) - \cos(\alpha_{ind}) * D_r(t)$$

As the stroke plane of the MAV was set to 90° with respect to U_f , the component of F_V supporting the weight of the MAV changes with angular position of the wing only. Close to the upper or lower turning point of the wings, F_V contributes less than when wings are at mid-down or -upstroke. This is accounted for using

$$(14) \quad F_{V,net}(t) = \cos(\Theta(t)) * F_V,$$

where Θ = excursion angle of the wing

Integrating instantaneous forces over one wing beat cycle for two wings yields mean vertical force ($\overline{F_V}$) and mean horizontal force ($\overline{F_H}$) for two sets of force coefficients, and is compared with the results from mean force measurements at the MAV.

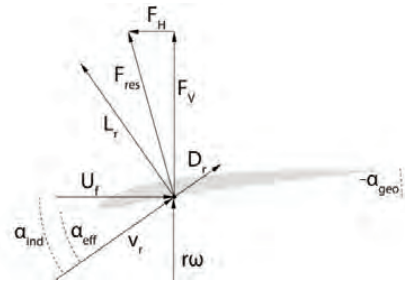


Figure 5: Forces and velocities on a blade-element: α_{eff} = effective angle of attack; α_{geo} = geometric angle of attack; α_{ind} = induced angle of attack; U_f = free flow velocity; v_r = effective velocity; r = radial distance of wing element; ω = angular velocity of the wing; L_r = lift; D_r = drag; F_{res} = resulting force; F_H = horizontal force component; F_V = vertical force component

3 RESULTS

3.1 Flapping-wing MAV force measurements

In flapping flight, the MAV creates a force perpendicular to U_f (F_V , “lift”) and a force parallel to U_f (F_H , “thrust”). Mean vertical force ($\overline{F_V}$) and mean horizontal force ($\overline{F_H}$) both increase with flapping frequency (Figure 6). $\overline{F_V}$ is always positive for the setups that were tested; increasing U_f also increases maximal $\overline{F_V}$ measured (see Figure 6A). Mean horizontal force is a measure for net thrust. $\overline{F_H}$ is generally lower for high free flow velocities due to increased drag of the whole MAV system, but flapping frequencies > 8 Hz result in net thrust for all flow velocities under test (see Figure 6B). The mean vertical force coefficient increases substantially with decreasing advance ratio compared to the maximum steady-flow lift coefficient ($C_{L,max} = 1.01 \pm 0.01$) for all but two measurements (see Figure 7).

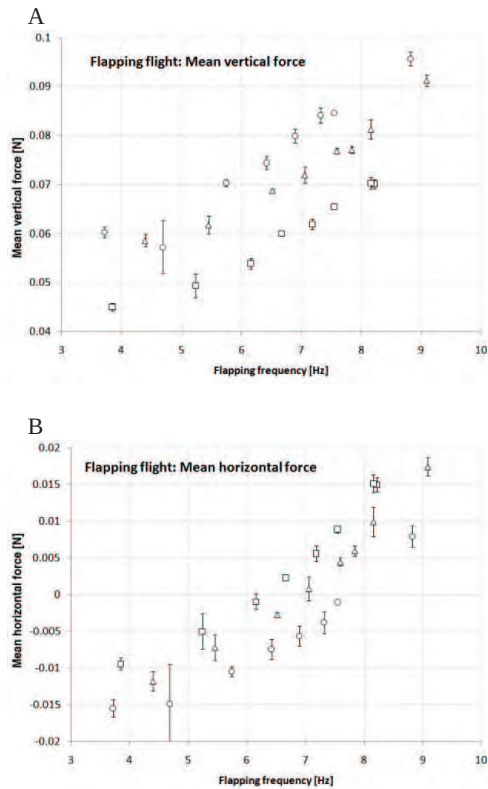


Figure 6: Mean forces of the flapping wings for different U_f (squares = 2.28 m/s ; triangles = 2.57 m/s ; circles = 2.84 m/s) (A) Mean vertical force ($\overline{F_v}$) increases with flapping frequency. (B) Mean horizontal force ($\overline{F_H}$) increases with flapping frequency and becomes positive for high flapping frequencies. The MAV creates "net thrust".

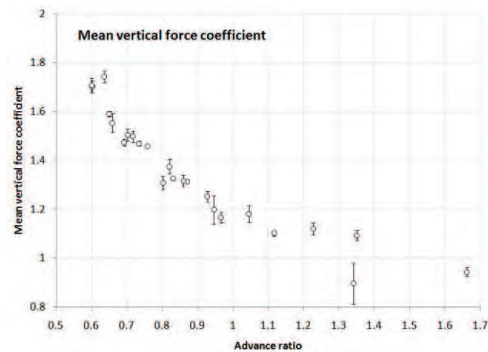


Figure 7: Mean vertical force coefficient ($\overline{C_v}$) vs. advance ratio. $\overline{C_v}$ peaks at about 1.7 for low advance ratio.

3.2 Blade-element analysis

Mean vertical force derived from the blade-element analysis using "steady" C_L and C_D reveals a large defect in force (see Figure 8). For flapping frequencies above 6 Hz, $\overline{F_v}$ is underestimated by the blade-element approach by a factor of more than two. The defect is found in all free flow velocities. The slope of $\overline{F_v}$ vs. frequency calculated via "steady" coefficients is very small, increasing flapping frequencies hardly produce additional $\overline{F_v}$. The defect is smaller for flapping frequencies

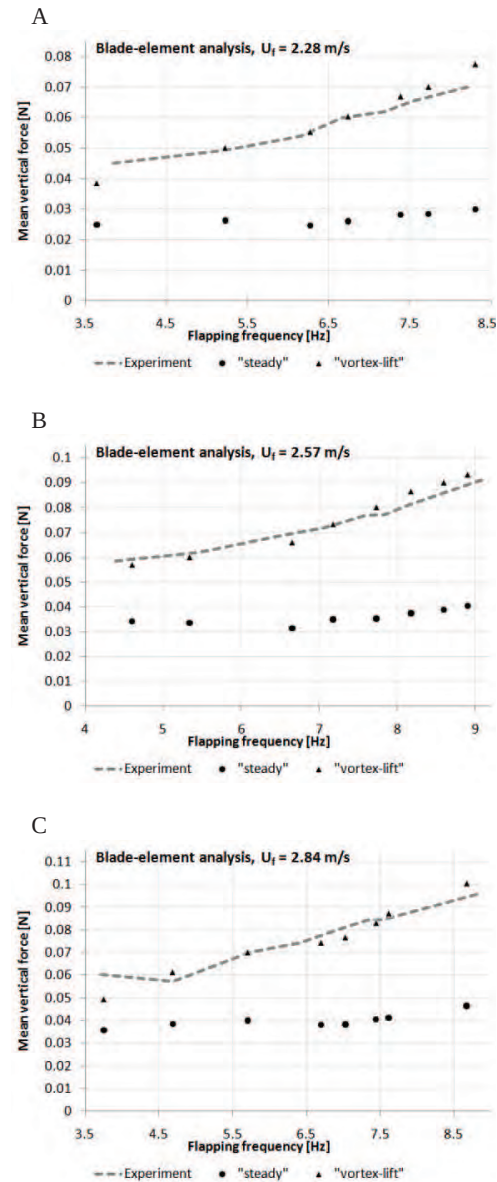


Figure 8: Results of the blade-element analysis with "steady" (circles) and "vortex-lift" (triangles) force coefficients compared to force balance measurements (dashed line). (A) $U_f = 2.28$ m/s (B) $U_f = 2.57$ m/s (C) $U_f = 2.84$ m/s. In all cases, "steady" coefficients underestimate mean vertical force, whereas "vortex-lift" coefficients show a good agreement.

< 6 Hz (see Figure 8). In contrast, the results of the blade-element analysis using "vortex-lift" coefficients are very similar to experimental results. The difference to experimental data is maximally 12% (see Figure 9), excluding the two lowest flapping frequencies, which were recorded very close to the resonant frequency of the balance system and are therefore probably not reliable. The mean difference in $\overline{F_v}$ of "vortex-lift" coefficients is $1.4\% \pm 4.8\%$; "steady" coefficients result in a mean difference of $50.0\% \pm 4.8\%$.

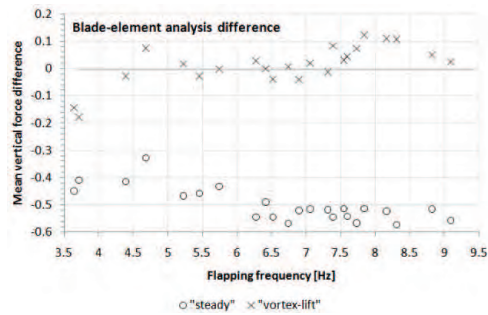


Figure 9: Deviation of experimentally determined mean vertical force compared to results from the blade-element analysis. $\bar{F}_V$ of the balance measurements was subtracted from the corresponding result of the blade-element analyses for “steady” and “vortex-lift” coefficients. The result is expressed as fraction of experimentally determined $\bar{F}_V$, data for different free flow velocities was pooled. “Steady” coefficients underestimate $\bar{F}_V$ by a factor of two, “vortex-lift” coefficients deviate by maximally 12 % (excluding measurements at frequencies close to the resonant frequency of the balance system), and on average by $1.4 \% \pm 4.8 \%$.

4 DISCUSSION

4.1 Micro air vehicle

Vertical and horizontal force of a flapping-wing MAV was determined by means of a force balance. The wings create on average enough vertical and horizontal force to keep a small, fully equipped MAV airborne. Mean vertical force coefficient is inversely related to advance ratio. This is due to the fact that advance ratio decreases with increasing flapping frequency. The increase in flapping frequency causes an increase in the flow velocity over the wing and at the same time increases the effective angle of attack. These all contribute to an increase in aerodynamic force. The relation between mean vertical force coefficient and advance ratio as well as the magnitude of $\bar{C}_V$ is very similar to the results reported by Kim *et al.* [20]. That study evaluated lift forces of a flapping wing MAV of a size similar to ours but with flexible foil wings, where airfoil camber could be changed using macro-fiber composite actuators. The performance of our MAV design in generating vertical force thus seems to be reliable.

4.2 Blade-element analysis using “steady” coefficients

Using data derived from kinematics, we applied a blade-element analysis to calculate forces using two different sets of force coefficients. Lift and drag coefficients from steady-flow measurements of the MAV’s wings applied to the blade-element theory underestimate mean vertical force by a factor of two. Previous studies using a similar method report comparable results: The “quasi-steady” approach has been applied to insects (e.g. [1]-[3]) and also to slow-speed flapping flight of cockatiels [21], where the wings are exposed to large effective angles of attack. However, in all cases the magnitude of

aerodynamic forces observed could not be explained with “quasi-steady” assumptions.

This discrepancy can be related to the effective angle of attack (α_{eff}) during the beat cycle, in particular close to the wing tip (see Figure 10). Our measurements of “steady” coefficients show that $C_{L,max}$ peaks at 11° geometric angle of attack; at higher angles of attack the lift decreases, as the wing stalls in a steady-flow environment. Hence, high flapping frequencies with relatively large α_{eff} will increasingly seriously underestimate C_L . The fact that in the blade-element model the vertical force still increases at increasing α_{eff} is because the wing drag starts to contribute to the vertical force with $\sin(\alpha_{ind})C_D$ (see Equation 12).

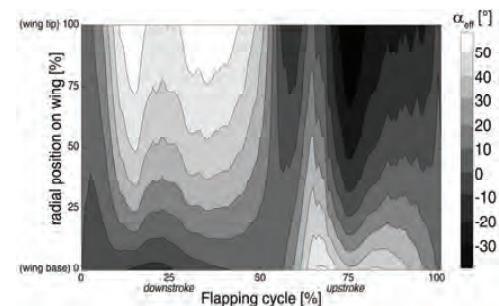


Figure 10: Effective angle of attack as a function of span wise position and flapping cycle. During downstroke and close to the wing tip, the effective angle of attack reaches 50° (see colour bar at the right)

4.3 Blade-element analysis using “vortex-lift” coefficients

Several studies prove the existence of leading edge vortices in flapping flight and the ability of stably attached vortices to augment lift (e.g. [4], [6], [8], [9], [14]). Stamhuis *et al.* [16] have shown that LEVs instantly developed on the same type of wing that was flapping with very similar kinematics. An appropriate concept to model C_L and C_D including additional lift created by LEVs was introduced by Polhamus [11]. Using this concept, we model $C_{L,max}$ to be 2.5; a value much higher than $C_{L,max}$ under steady-flow conditions. Lift coefficients that are much higher than C_L under steady-flow conditions seem to be typical for flapping and pitching airfoils. A numerical study on rapidly pitching airfoils ($Re = 1700$) reveals instantaneous lift coefficients of 2.4 to 3.2 [23]. Similar lift coefficients were reported in a CFD simulation of fruit fly wings ($Re < 1800$), and the presence of a stable LEV is made responsible for increasing C_L up to a value of 3.2 at mid-downstroke of the insect wing [24]. Modelling C_L with a concept that accounts for the additional lift of attached leading-edge vortices hence seems to be a good approximation of aerodynamic phenomena in flapping flight. This is also supported by Dickson *et al.* [25] who conclude that a

“quasi-steady” aerodynamic model may explain the force balance of a hovering insect when appropriate force coefficients are used.

4.4 Mean horizontal force

Comparing the mean horizontal force ($\overline{F_H}$) from wind tunnel measurements and blade-element analysis does not seem to be feasible, due to limitations of our blade-element model. In contrast to the balance measurements, which measure $\overline{F_H}$ of the entire MAV, the blade-element analysis only accounts for forces created by the wings. The present model does not account for the extra drag of the chassis, neither for any interference drag between the chassis and the flapping wings. Additionally, limitations in our force balance equipment required that the strut for attaching the MAV to the balance lies inside the wake of a wing. The wings accelerate the air periodically and the drag of the strut increases with velocity squared, increasing the overall drag measured by the force balance. Time-resolved measurements of the fluid velocity in the wake, together with detailed assumptions on the magnitude of the interference drag and improvements on the design of the setup could circumvent this limitation in future studies.

5 CONCLUSION

The aim of this study was to check the feasibility of extending a relatively simple blade-element approach to include additional lift-enhancing aerodynamic effects. A concept initially postulated for sharp-edge delta wings provides data on C_L and C_D under the presence of leading edge vortices. The resulting maximal lift coefficient is a factor of 2.5 greater than typical steady-flow coefficients, and agrees well with data reported in earlier studies on flapping flight. The key requirement for the applicability of the “vortex-lift” approach is the presence of a stable LEV. As Lentink and Dickinson [10] suggest, LEVs in flapping flight are stabilized by the centripetal and Coriolis acceleration. As these accelerations are relatively independent of the Reynolds number [10], it is likely, that the “vortex-lift” approach is not limited to a small bandwidth of flapping wing devices, as long as the advance ratio is low and wing geometry and kinematics create sufficient centripetal and Coriolis accelerations to stabilize the LEV.

We believe that the approach presented in this study might be an appropriate tool to assess and predict forces of flapping-wing flyers and MAVs that operate at low advance ratio and potentially benefit from increased lift due to leading edge vortices.

REFERENCES

- [1] C. P. Ellington. The Aerodynamics of Hovering Insect Flight. I. The Quasi-Steady Analysis. *Philosophical Transactions of the Royal Society of London. Series B*, 305: 1-15, 1984.
- [2] A. R. Ennos. The Kinematics and Aerodynamics of the Free Flight of some Diptera. *The Journal of Experimental Biology*, 142: 49-85, 1989.
- [3] J. M. Zanker and K. G. Gotz. The Wing Beat of *Drosophila melanogaster*. II. Dynamics. *Philosophical Transactions of the Royal Society of London. B*, 327: 19-44, 1990.
- [4] C. P. Ellington, C. van den Berg, A. P. Willmott and A. L. R. Thomas. Leading-edge vortices in insect flight. *Nature*, 384: 626-630, 1996.
- [5] J. Usherwood and C. P. Ellington. The aerodynamics of revolving wings. I. Model hawkmoth wings. *The Journal of Experimental Biology*, 205: 1547-1564, 2002.
- [6] R. J. Bomphrey, N. J. Lawson, N. J. Harding, G. K. Taylor and A. L. R. Thomas. The aerodynamics of *Manduca sexta*: digital particle image velocimetry analysis of the leading-edge vortex. *The Journal of Experimental Biology*, 208: 1079-1094, 2005.
- [7] D. R. Warrick, B. W. Tobalske, and D. R. Powers. Aerodynamics of the hovering hummingbird. *Nature*, 435: 1094-1097, 2005.
- [8] T. Y. Hubel and C. Tropea. The importance of leading edge vortices under simplified flapping flight conditions at the size scale of birds. *The Journal of Experimental Biology*, 213: 1930-1939, 2010.
- [9] F. T. Muijres, L. C. Johansson, R. Barfield, M. Wolf, G. R. Spedding and A. Hedenstroem. Leading-Edge Vortex Improves Lift in Slow-Flying Bats. *Science*, 319: 1250-1253, 2008.
- [10] D. Lentink and M. H. Dickinson. Rotational accelerations stabilize leading edge vortices on revolving fly wings. *The Journal of Experimental Biology*, 212: 2705-2719, 2009.
- [11] E. C. Polhamus. A concept of the vortex lift of sharp-edge delta wings based on a leading-edge-suction analogy. *NASA TN D-3767*, 1966.
- [12] J. J. Videler, E. J. Stamhuis and G. D. E. Povel. Leading-Edge Vortex Lifts Swifts. *Science*, 306: 1960-1962, 2004.
- [13] J. Wu, A. Vakili and J. Wu. Review of the physics of enhancing vortex lift by unsteady excitation. *Progress in Aerospace Sciences, Elsevier*, 28: 73-131, 1991.
- [14] J. M. Birch and M. H. Dickinson. Spanwise flow and the attachment of the leading-edge vortex on insect wings. *Nature*, 412: 729-733, 2001.
- [15] E. C. Polhamus. Application of the leading-edge-suction analogy of vortex lift to the drag due to lift of sharp-edge delta wings. *NASA TN D-4739*, 1968.
- [16] E. J. Stamhuis, W. Thielicke, I. Ros and J. J. Videler. Unsteady aerodynamics essential during low speed flapping flight in bird. *Submitted for publication*, 2011.
- [17] D. B. O. Savile. The Flight Mechanism of Swifts and Hummingbirds. *The Auk*, 67: 499-504, 1950.
- [18] P. Henningson, G. R. Spedding and A. Hedenstrom. Vortex wake and flight kinematics of a swift in cruising flight in a wind tunnel. *The Journal of Experimental Biology*, 211: 717-730, 2008.
- [19] A. B. Kesel. Aerodynamic characteristics of dragonfly wing sections compared with technical aerofoils. *The Journal of Experimental Biology*, 203: 3125-3135, 2000.
- [20] D.-K. Kim, J.-H. Han and K.-J. Kwon. Wind tunnel tests for a flapping wing model with a changeable camber using macro-fiber composite actuators. *Smart Materials and Structures*, 18: 024008, 2009.

- [21] T. Hedrick, B. Tobalske and A. Biewener. Estimates of circulation and gait change based on a three-dimensional kinematic analysis of flight in cockatiels (*Nymphicus hollandicus*) and ringed turtle-doves (*Streptopelia risoria*). *The Journal of Experimental Biology*, 205: 1389-1409, 2002.
- [22] J. Kweon and H. Choi. Sectional lift coefficient of a flapping wing in hovering motion. *Physics of Fluids*, 22: 071703, 2010.
- [23] H. Liu and K. Kawachi. A Numerical Study of Insect Flight. *Journal of Computational Physics*, 146: 124-156, 1998.
- [24] J. Wu and M. Sun. Unsteady aerodynamic forces of a flapping wing. *The Journal of Experimental Biology*, 207: 1137-1150, 2004.
- [25] W. B. Dickson and M. H. Dickinson. The effect of advance ratio on the aerodynamics of revolving wings. *The Journal of Experimental Biology*, 207: 4269-4281, 2004.

Aerodynamic analysis of the wing flexibility and the clap-and-peel motion of the hovering DelFly II

T. Gillebaart*, A.H. van Zuijlen, and H. Bijl

Delft University of Technology, 2629 HS Delft, The Netherlands

ABSTRACT

A Micro Aerial Vehicle developed at the Delft University of Technology (DelFly II) uses flapping wings to produce the forces needed for forward flight and hovering. The DelFly II is inspired by the dragonfly with two flexible wings making a 'clap-and-peel' motion at each side of the body. In this paper the influence of wing flexibility and the 'clap-and-peel' motion (one versus two wings) on the upward (lifting) force in hover is studied. CFD simulations based on the ALE Navier-Stokes equations together with radial basis function mesh interpolation are used. For the 'clap-and-peel' motion an immersed symmetry plane is introduced into the computational domain. The flexible shapes of the wing are obtained from experiments. Preliminary results are shown for a flexible and an equivalent rigid wing. These results show that the use of a flexible wing increases the upward force for both the single wing motion and the 'clap-and-peel' motion. The 'clap-and-peel' motion increases the upward force per wing for both the flexible and rigid wing when compared to the single wing making the same motion. From the preliminary results it can be concluded that flexibility enhances the production of the upward force.

1 INTRODUCTION

Aerodynamic characteristics of flapping flight have been extensively studied during the past years (e.g. [1]). First only the main physical phenomena were investigated, such as the importance of the leading edge vortex [2]. More recently also the influence of flexible deforming wings has been the subject of these studies [3], for which CFD simulations have been increasingly used as an analytical tool. In [4, 5] a more fundamental study on flexibility showed the potential benefits of flexible wings over rigid wings. In these studies it is concluded that wing flexibility can be tailored such that the forces produced are higher than with a rigidly moving wing. In 2005 a flapping MAV was developed at the TU Delft, named the DelFly [6], which has been used to study flapping

flight. In [7] a first analysis of the aerodynamics of the hovering DelFly II has been made by use of PIV experiments. After this a study on the aerodynamics of different wing designs using again PIV experiments has been performed [8]. At the same time [9] studied several aspects of flapping wings using CFD simulations with moving meshes.

In this study the influence of wing flexibility and the 'clap-and-peel' motion on the upward (lifting) force in hover conditions is investigated using two-dimensional CFD simulations. To determine the influence of wing flexibility a wing with imposed flexibility and an equivalent rigid wing are simulated. Chord wise wing deformation of the DelFly II wing has been measured at several span wise locations [8]. These deformed wing shapes are used in this study to simulate the flexibility of the wing. In this study the novel approach is taken by enforcing the wing shapes instead of using full fluid-structure interaction. The interaction from the fluid to the structure is absent, which eliminates the necessity of a structural model of a complex 3D wing translated to 2D. However due to this there is no damping in the acceleration imposed from the wing on the fluid.

Earlier studies showed that the 'clap-and-peel' motion greatly enhances the upward force in flapping flight [10, 11]. In this study the influence of the 'clap-and-peel' motion on the upward force produced is investigated by comparing results for a single wing and a wing close to a mirror plane to model the presence of a second wing. The Reynolds number is relevant for MAV's (around 7000) and the imposed flexibility is based on a real wing of the DelFly II. For the 'clap-and-peel' motion a newly developed method is used based on a immersed symmetry plane together with mesh topology adjustments.

2 METHODS

The method to simulate the flapping wing is by solving the Arbitrary Lagrangian-Eulerian form of the unsteady, incompressible, laminar Navier-Stokes equations in OpenFOAM, which uses finite volume discretisation [12]. For the mesh movement radial basis function (RBF) mesh interpolation is used [13]. An immersed boundary method is used to impose symmetry plane conditions in the flow in case of 'clap-and-peel' motion.

2.1 2D incompressible flow simulations

Fluid flows are described by the Navier-Stokes equations. In case of small flapping wings, the Reynolds and Mach number may be small enough that the unsteady, incompressible, laminar Navier-Stokes equations can be used [9, 14, 15]. For

*Email address(es): t.gillebaart@student.tudelft.nl

now, only two-dimensional simulations are performed. Since hovering is considered close attention should be given to the boundary conditions, because vortices shed by the wing are not convected downstream but can also leave the domain through the sides. Because of the zero velocity as far field condition, in and out flow can occur at the same place at different times. To achieve this the pressure and the velocity are coupled at the boundary by using a total pressure boundary condition. The total pressure at the boundary is set to a constant value and depending on the flow velocity the static pressure is set. The direction of the flow at the boundary is set based on the pressure difference at the boundary. When the pressure difference at the boundary indicates an outward flux this is coupled to the velocity by imposing an outward flow and vice versa. In this way the pressure and velocity can vary at the boundary in time and also cope with vortices leaving the domain. Other settings for OpenFOAM can be found in Table 1.

Setting	Values
p solver	Algebraic multi-grid
p tolerance	$1e^{-7}$
U solver	BiCGStab
U tolerance	$1e^{-9}$
PISO loops	2
Non-orthog. cor.	1 (only for double wing)
Courant nr.	0.4
$\frac{\partial}{\partial t}$	Backward differencing formula (2^{nd})
∇	Gaussian (2^{nd})
$\nabla \cdot$	Gaussian Gamma differencing (2^{nd})
∇^2	Gaussian linear corrected (2^{nd})
Mesh	59906 cells
Coarse mesh	15314 cells

Table 1: Settings for OpenFOAM.

To simulate the moving and deforming wing the mesh should deform accordingly. To accomplish this the Arbitrary Lagrangian-Eulerian (ALE) Navier-Stokes equations are used.

2.2 ALE Navier-Stokes with OpenFOAM

Simulating the fluid flow around flapping wings is done by using the Arbitrary Lagrangian-Eulerian Navier-Stokes equations [16]. The Eulerian description is still used for the fluid, but the moving mesh is described using the Lagrangian method. In Equation (1) the momentum equation of the unsteady, incompressible, laminar ALE Navier-Stokes is shown.

$$\frac{\partial}{\partial t} \int_{V_C} \mathbf{u} dV + \oint_{S_C} \mathbf{n} \cdot (\mathbf{u} - \mathbf{u}_{mesh}) \mathbf{u} dS - \int_{V_C} \nabla \cdot (\nu \nabla \mathbf{u}) dV$$

$$= \int_{V_C} \frac{\nabla p}{\rho} dV \quad (1)$$

Two terms in these equations are different from the standard momentum equation. Firstly the temporal term, which is different because the derivative cannot be placed inside the integral as the volume does change over time. Secondly in the convective term the mesh velocity can be found. Due to the movement of the faces an extra flux is created through the faces. Both these terms are caused by the movement of the mesh through the fluid and the deformation of the mesh cells. A (detailed) description on how this is used to simulate moving body aerodynamics can be found in [9]. For a more (detailed) description on the implementation in OpenFOAM, please see [17]. The reason for using this method is clearly explained in [18], which shows that as long as remeshing is not needed the ALE Navier-Stokes should be used to simulate moving and deforming body aerodynamics. Since only the movement of the wing is known the movement of the rest of the mesh should be interpolated based on the wing movement. This is done with radial basis function (RBF) mesh interpolation [13].

2.3 RBF mesh interpolation

Besides the equations to model the fluid, also an algorithm is needed to deform the mesh based on the wing movement. In [13] it is shown that RBF mesh interpolation is a promising mesh movement algorithm when considering the mesh quality and the simplicity of the implementation. Especially in rotation the RBF mesh interpolation performs better than alternative methods available in OpenFOAM [9]. In the flapping wing kinematics typically large rotations are present and therefore the RBF mesh interpolation is chosen as method for deforming the mesh based on the movement of the wing. RBF mesh interpolation is based on smooth interpolation from known displacements of the control points to the rest of the field using radial basis functions. In this study the control points are chosen such that the simulated wing is deformed as the wing shapes from the experiments. For the basis function evaluations only the euclidean distance between each control point and each internal point is needed. So no grid connectivity is needed, which makes this method easy to implement. RBF mesh interpolation provides a robust and easy method to deform the mesh smoothly when only the movement of the body is known. For more details on this method see [13].

2.4 Mesh quality in time

During the complete flapping period the mesh should preserve its quality to ensure reliable computations. By means of skewness and non-orthogonality the mesh quality is measured during a full period to determine which control points should be used. Since the Delfly II wing, which is made of a thin flexible foil, has a very small thickness a 'zero-thickness' wing is used in the simulations. In CFD simulations using OpenFOAM a 'zero-thickness' wing can be modelled by creating two rows of faces connected to the same points, but with

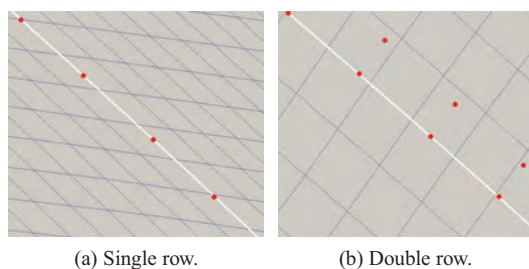


Figure 1: Resulting deformed meshes for a single row and two rows of control points.

there normals opposite to each other. The control points are directly related to the wing shape. Since the wing has a 'zero-thickness' a single row of points can describe the movement of the wing. However as can be seen in Figure 1a, which displays the mesh close to the wing (white line) and the used control points (red dots), using just a single row of control points does not provide the best results in terms of mesh quality as the cells become skewed. Only the translation of each point is captured by the RBF mesh interpolation when using a single row of control points. By introducing a second row of control points close to the wing and perpendicular to the local wing shape, the rotation of the wing is interpolated to the rest of the mesh. This ensures that the cells close to the wing stay perpendicular to the wing, while the rotation is smoothly reduced when moving further away from the wing. This can be seen in Figure 1b. Figure 2 shows the mesh skewness and non-orthogonality for one full flapping period for both a single row of control points and two rows of control points.

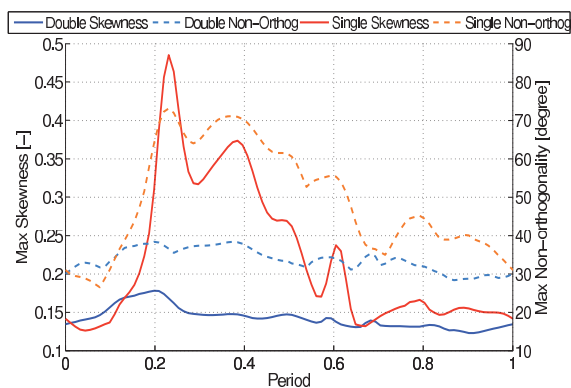


Figure 2: Mesh quality characteristics for single and double row of points.

Non-orthogonality is measured by the angle between the cell centres line (between two cell centres) and the normal of the intermediate face. The degree of skewness is measured as the distance between the face centre and the cross point between the face tangent and the cell centres line. This distance is then normalised with the magnitude of the cell centres line. A schematic explanation can be found in [9]. In

OpenFOAM meshes are considered of bad quality when the non-orthogonality is larger than 70 degrees or skewness exceeds 4.0. Considering these two criteria it can clearly be seen that two rows of control points are needed to ensure a good mesh quality during the complete flapping motion. Therefore in all simulations two rows of control points are used. It is clear that RBF mesh interpolation can provide a robust mesh deformation while preserving the original mesh quality.

3 WING SHAPES

The wing shapes are extracted from experiments at specific points in time during a full flapping period. Splines are used to construct the wing shapes in space from the PIV images for specific points in time. Fourier series are used to interpolate the shape in time.

3.1 The PIV results

From the PIV experiments the chord wise shape at different span-wise locations of the Delfly II wing is obtained [8]. The flapping frequency is 11 Hz, which is characteristic for the Delfly II. At 71% of the span (with a chord of 7.4 cm) 50 shapes in time have been recorded for one full period. Spline interpolation is used to interpolate the wing shape in space for each specific point in time. In Figure 3 the resulting splines, representing the wing shapes, are shown for the out-stroke or 'peel' phase.

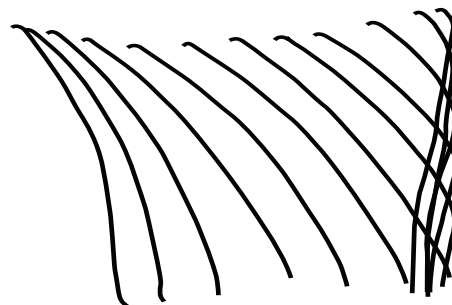


Figure 3: Deforming wing shape for out-stroke obtained from experiments.

3.2 Interpolation in space

For RBF mesh interpolation discrete points in space are needed to function as control points. These discrete points are obtained from the spline interpolations in space. From the continuous description using splines any number of discrete points can be chosen along the wing. Sixty uniformly distributed points along the wing are used as control points for the RBF mesh interpolation. Next step is to interpolate these points in time.

3.3 Interpolation in time

A requirement for the interpolation in time is that the acceleration (second derivative of the interpolation in time) is continuous. If this requirement is not met numerical oscillations will occur in the simulation due to the temporal term in

the Navier-Stokes equation. The temporal term is indirectly related to the pressure gradient. Due to the high accelerations at the wing a high pressure gradient will be present at the wing causing an immediate increase in pressure and thus in the forces. For the interpolation in time Fourier series are used, which ensures that the second derivative is continuous and that the function is periodic. For each coordinate from each control point a Fourier series interpolation is performed in time. In total 120 Fourier series interpolation in time are needed for the 60 points in space. However how many Fourier modes must be taken to interpolate in time? Increasing the number of modes will increase the accuracy of the interpolation through the measured data points, but as the measured data points contain scatter from measurements errors, it will also increase the dominating frequency in the acceleration. High frequencies in the acceleration induce non-physical oscillations into the simulation, which should be prevented at all times.

3.4 Influence of shape interpolation

To determine how many modes must be used both the position error and the influence of the acceleration frequency is studied. In Figure 4 the position error with respect to the control points obtained from the experiments is shown for the interpolation in time by means of the maximum normalized maximum error and the maximum normalized 2-norm error. For the normalisation the stroke amplitude is used, which is equal to 8 cm. The 2-norm error is also normalised by the number of points.

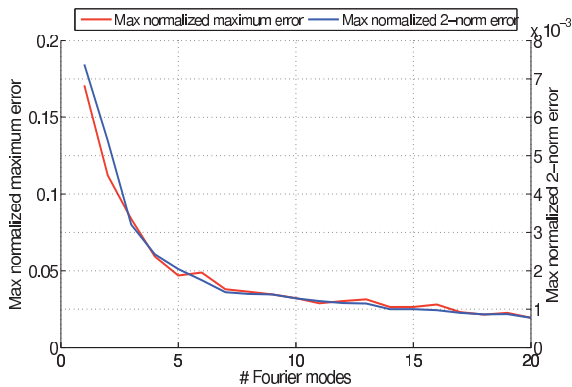


Figure 4: Maximum normalised maximum interpolation error vs. # Fourier modes.

It is clear that both the errors decrease when the number of Fourier modes is increased. When more than 7 modes are used the errors seem to level off and thus using 7 modes for interpolation can be considered sufficient in terms of position error. However using higher modes has a significant influence on the acceleration of the wing.

In Figure 5 the acceleration in the x-direction of the leading edge is shown for 4, 7 and 10 Fourier modes together with the acceleration of the control point obtained by finite differenc-

ing. The dominating frequency in the acceleration is equal to the highest mode used in the Fourier interpolation. In Figure 6 the periodic average of the upward force for 4, 7 and 10 Fourier modes can be seen. This periodic average is obtained by averaging the force over period 6 till 44.

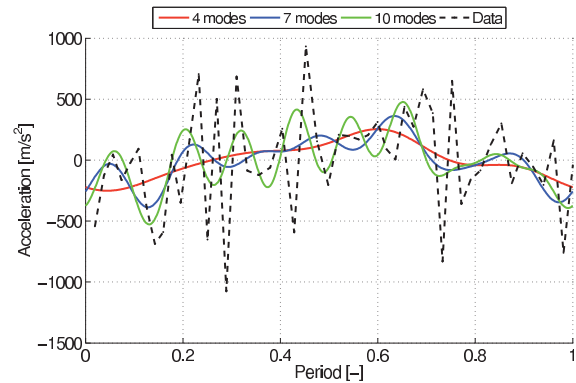


Figure 5: Leading edge acceleration for different number of Fourier modes.

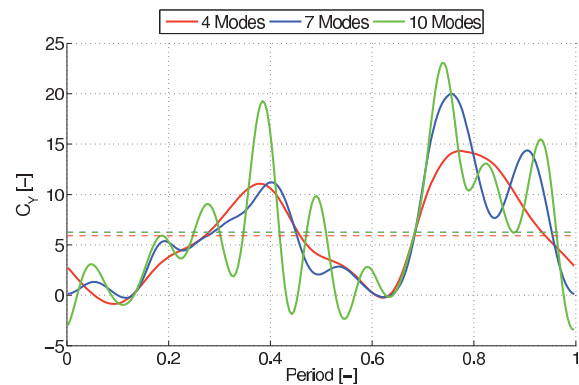


Figure 6: Upward force for different number of Fourier modes.

The dominating frequency in the upward force is equal to the dominating frequency in the acceleration of the wing, which is determined by the highest Fourier mode. Earlier studies (e.g [2]) showed that vortex development and shedding and not the frequency of the wing's acceleration is dominating the force variation. Using the higher Fourier modes for interpolation introduces non-physical oscillations in the pressure field and consequently in the forces. To prevent this a DelFly II like wing is simulated by using 4 Fourier modes. This still ensures that general shape of the flexible wing is captured, while the introduction of high frequency oscillations is limited. Further investigation into the selection of the number of Fourier modes is done at a later stage by carefully determining which oscillations are caused by the acceleration and which by the vortex behaviour.

4 FLEXIBLE VS. RIGID

The influence of the wing flexibility is studied comparing results of the flexible wing with an equivalent rigid wing. Both the upward force and the vortex development and shedding is compared.

4.1 Equivalent rigid wing

To determine the influence of the deforming shape an equivalent rigid wing is needed. No Delfly II experiments were conducted with rigid wings. Therefore an equivalent rigid wing is constructed based on the leading edge and trailing edge movement of the flexible wing. The leading edge motion of the rigid wing is equal to that of the flexible wing. Drawing a line from the leading edge to the trailing edge of the flexible wing gives the direction which is used as pitch angle. With these two parameters (leading edge location and pitching angle in time) and the constant wing length (7.4 cm) the equivalent rigid wing can be modelled during the complete flapping period. In Figure 7 both the flexible and the equivalent rigid wing are shown for different moments in time during the out-stroke.

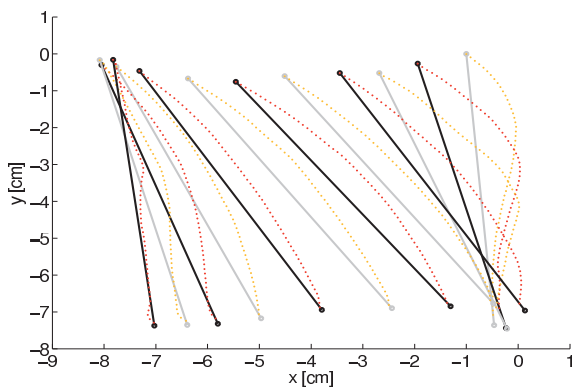


Figure 7: Flexible and equivalent rigid wing.

4.2 Results

For the rigid and flexible wing the forces are calculated for several periods. In total 44 periods are simulated for each calculation. A periodic average is calculated from periods 8 till 44. In Figure 8 the periodic average of the upward force is shown for the three wings. The dashed line is the average force of the period.

Comparing the two wings it can be seen that the average force is higher for the flexible wing. For a large part this is caused by the dip in force for the rigid wing at the start of the out-stroke (around 22% of the flapping cycle). In Figure 9 the vorticity field of the flexible wing is shown at 22% of the period. When this is compared to Figure 10, which is the vorticity field for the rigid wing at the same time, a clear difference is found.

For the flexible wing the leading edge vortex is being formed and the trailing edge vortex is relatively small, caus-

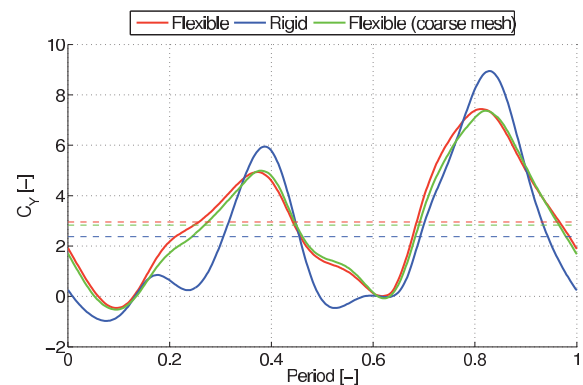


Figure 8: Periodical average of the upward force for the three wings.

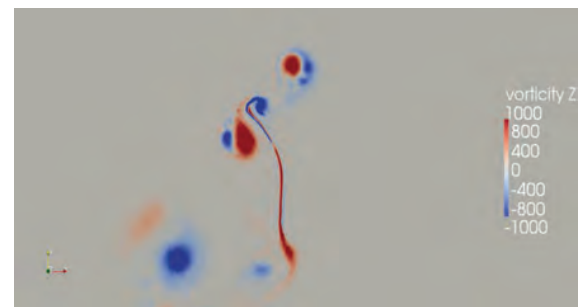


Figure 9: Vorticity field for flexible wing at 22% of the period.

ing a positive upward force. A different situation is found for the rigid pitching wing. Here the first leading edge vortex is already shed due to the interaction with the leading edge vortex of the in-stroke, while the trailing edge vortex is still attached and already larger than the one of the flexible wing. Due to this difference the flexible wing produces a significant amount of lift, while the rigid wing has a dip in the upward force. At a later stage (around 40% of the period) the flexible wing starts to lose lift due to the developed trailing edge vortex and shedding of the leading edge vortex. Here the rigid wing performs better due to the shedding of the trailing edge vortex and the development of a 'new' leading edge vortex. Here the complex behaviour of interacting vortices leads to the difference in forces. The shape of the wing does have a significant influence on how the vortices develop during the flapping cycle.

5 'CLAP-AND-PEEL' MOTION

How the 'clap-and-peel' motion influences the upward force production and the vortex development and shedding, is studied by comparing the results of two wings clapping and peeling together with the results obtained in the previous paragraph using a single wing. The wing shapes of the Delfly obtained from the experiments are recorded during the 'clap-

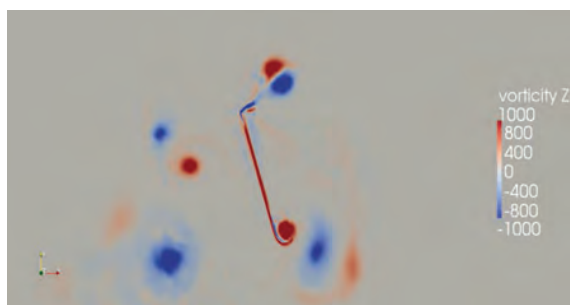


Figure 10: Vorticity field for the rigid wing at 22% of the period.

and-peel' motion and can thus be used in this simulation. In Figure 11 the 'clap-and-peel' motion is illustrated by means of the wing shapes obtained from the experiments.

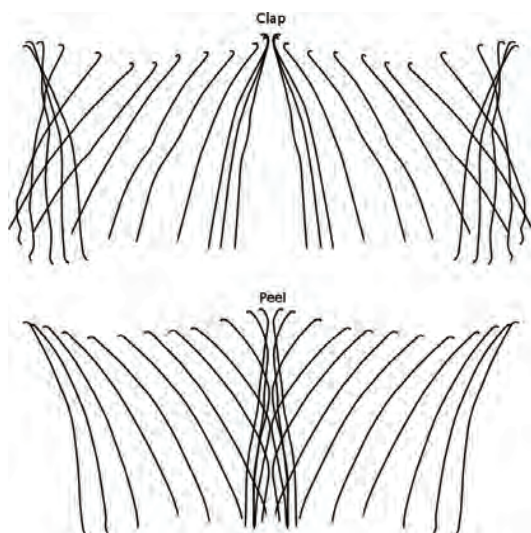


Figure 11: 'Clap-and-peel' motion.

By adding a 'second' wing making the mirrored motion the 'clap-and-peel' is created as can be seen in Figure 11. This can again be done for both the rigid wing and the flexible wing. However the ALE Navier-Stokes method presented earlier has difficulties when two bodies are moving towards each other. Especially when they almost touch each other the cells become very small, which is undesirable. Due to this the mesh quality cannot be preserved any more. Because the 'clap-and-peel' motion of the two wings is symmetric, a symmetry plane together with a single wing performing this motion models the 'clap-and-peel' motion. To impose the symmetry conditions on the correct location an immersed symmetry plane (based on the immersed boundary method) together with mesh topology adjustments near this symmetry plane is used.

5.1 Immersed symmetry plane

Several approaches are available to implement the immersed boundary method [19, 20]. The basis (cell identification) for this method used for this study is developed by Zeljko Tukovic, University of Zagreb, Croatia. A method based on the cut-cell finite volume approach as explained in [20] is implemented. The immersed symmetry plane cuts through a certain set of cells. By changing the shape of these cells an actual finite volume boundary is created in the mesh. In this way the boundary conditions, which are zero gradient for the pressure and slip for the velocity, are enforced at the correct location. Also the boundary conditions are implemented implicitly in the equations in the same way as for a normal mesh boundary. Each time step this procedure is repeated, because the mesh moves through the immersed symmetry plane. This method is chosen because the mesh quality is not compromised in most of the mesh because the ALE approach with RBF mesh interpolation can still be used. Only the symmetry boundary cells are temporarily changed in shape, which might lead to non-orthogonal cells. In the preliminary results a gap of 3.5% of the chord is used between the wing and the symmetry plane, but the gap can be as small as a single mesh cell. Preliminary results are generated using this method. Further development and validation of this method will be done at a later stage.

5.2 Preliminary results

For both the rigid and the flexible wing results are obtained in terms of the periodic upward force averaged over 38 periods and the vorticity fields. For these simulations only the coarse mesh is used to reduce the computational time, since parallel computing is not yet available. The maximum difference between the two meshes in the periodic averaged upward force for a single wing is 5%. Therefore the coarse mesh can be used for these simulations.

In Figure 12 the periodic average upward force is shown for four cases: the flexible single wing, 'double' flexible wing, rigid single wing and the 'double' rigid wing. The forces shown here are per wing and the dashed line is the integrated force during the single average cycle.

From these results it can be seen that there is a significant difference between the single wing and double wing results. For both the rigid and flexible wing using the 'clap-and-peel' motion increases the upward force produced per wing. Moreover the difference between the rigid and the flexible wing increases when the 'clap-and-peel' motion is applied. For the rigid wing the increase compared to the single wing is approximately 30%, while for the flexible wing it is close to 40%. 'Clap-and-peel' with the flexible wings results in 30% more lift per wing compared to the rigid 'clap-and-peel'. As seen in the single wing cases the development of the leading edge vortices and trailing edge vortices during the out-stroke are of vital importance for the difference in force production. At 30% of the period (in the middle of the out-stroke) the

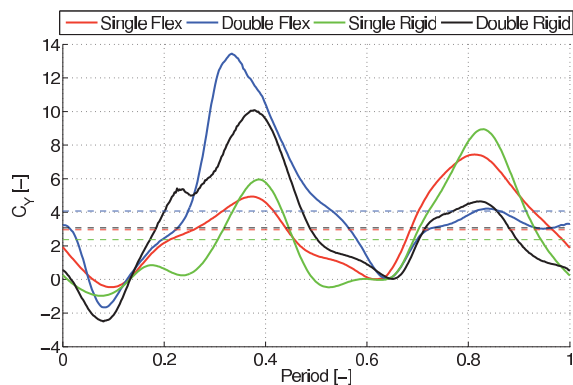


Figure 12: Periodic average upward force for four cases.

difference between the rigid and flexible wing can clearly be seen in case of the 'clap-and-peel' motion. In Figure 13 the vorticity plot of the flexible wing is shown, while in Figure 14 the vorticity of the rigid wing is displayed.

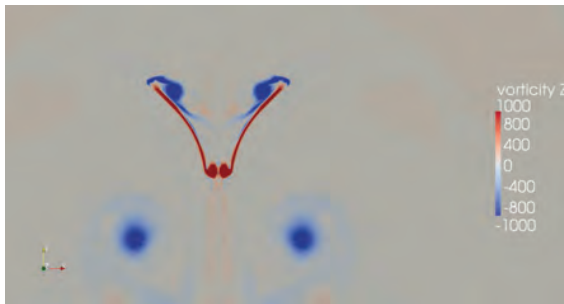


Figure 13: Vorticity field for the flexible wing at 30% of the period with immersed symmetry plane.

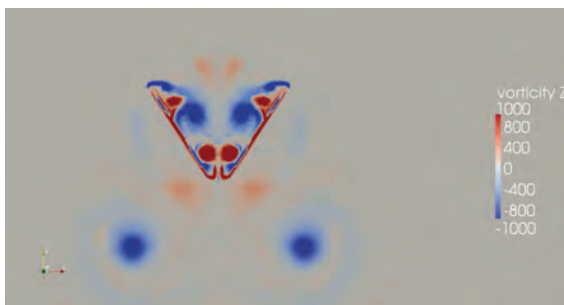


Figure 14: Vorticity field for the rigid wing at 30% of the period with immersed symmetry plane.

These plots show that for the rigid wing the leading edge vortex is already shed at this stage, while for the flexible wing this large leading edge vortex is still attached. At a later time in the period the trailing edge vortex will be shed earlier for the flexible wing compared to the rigid wing. These two phenomena explain the higher upward force for the flexible wing. When compared to the single wing vorticity plots

(even though they are at different times in the flapping cycle) the main difference is the size and the strength of the leading edge vortex. Due to the strong suction over the leading edge during the 'peel' movement a larger leading edge vortex is created when performing the 'clap-and-peel' motion.

6 CONCLUSIONS AND FUTURE WORK

In this paper the influence of wing flexibility and the 'clap-and-peel' motion on the upward force in hover is studied. A single wing, both rigid and flexible, has been simulated using the ALE approach with RBF mesh deformation. The clap-and-peel motion is simulated using an immersed symmetry plane together with the ALE approach for both the flexible and rigid wing. From the preliminary results the following conclusions can be drawn.

- The method of ALE Navier-Stokes together with RBF mesh interpolation provide good mesh quality during the complete flapping cycle when two rows of control points are used.
- Interpolation of the DeFly II wing shapes influences the frequency of the resulting forces. This is caused by the acceleration of the wing which is dominated by the highest frequency.
- Both the single wing simulations and the 'clap-and-peel' simulations show an increased upward force for the flexible wing compared to the rigid wing. This is mainly caused by early vortex shedding of the leading edge vortex during the out-stroke for the rigid wing.
- For both the flexible and rigid wing the 'clap-and-peel' motion increases the upward force per wing significantly due to the increased size and strength of the leading edge vortex during the out-stroke.

Even though several preliminary conclusions can already be drawn from these results, further studies are needed. The following topics are considered for future analysis.

- Validation of the immersed symmetry plane method by comparing to other methods.
- Simulating different equivalent rigid wings to compare to the flexible wing. It is well known that the pitch angle is important in flapping wings.

Due to the absence of fluid-structure interaction no damping is present in the movement and deformation of the wing. Two-dimensional simulations are known for the earlier shedding of vortices compared to three-dimensional simulations. Therefore a complete three-dimensional fluid-structure simulation together with an experimental study would create an even better insight in the physics. However this study showed which methods can be used for the simulation of the 'clap-and-peel' motion and gives a good insight into the physics of 'clap-and-peel' with flexible and rigid wings.

ACKNOWLEDGEMENTS

Dr. Zeljko Tukovic, University of Zagreb, Croatia for providing a immersed symmetry plane code for OpenFOAM. Hrvoje Jasak for providing help in setting up OpenFOAM and suggesting OpenFOAM settings.

REFERENCES

- [1] S. Heathcote and I. Gursul. Flexible flapping airfoil propulsion at low reynolds numbers. *AIAA Journal*, 45:1066–1079, 2007.
- [2] C.P. Ellington, C. van den Berg, A.P. Willmott, and A.L.R. Thomas. Leading-edge vortices in insect flight. *Nature*, 384:626–630, 1996.
- [3] S. Heathcote and I. Gursul. Flexible flapping airfoil propulsion at low reynolds numbers. *AIAA Journal*, 45:1066–1079, 2007.
- [4] H. Aono, S.K. Chimakurthi, P. Wu, E. Sllstrm, B.K. Stanford, C.E.S. Cesnik, P. Ifju, L. Ukeiley, and W. Shyy. A computational and experimental study of flexible flapping wing aerodynamics. In *48th AIAA Aerospace Sciences Meeting*, Orlando, Florida, January 2010. AIAA 2010-554.
- [5] Hikaruo Aono, Chang kwon Kang, Carlos E.S. Cesnik, and Wei Shyy. A numerical framework for isotropic and anisotropic flexible flapping wing aerodynamics and aeroelasticity. In *28th AIAA Applied Aerodynamics Conference*, Chicago, Illinois, July 2010. AIAA 2010-5082.
- [6] G.C.H.E. de Croon, K.M.E. de Clercq, R. Ruijsink, B. Remes, and C. de Wagter. Design, aerodynamics and vision-based control of the delfy. *International Journal of Micro Air Vehicles*, 1:71–97, 2009.
- [7] K.M.E. de Clercq, R. de Kat, B.W. van Remes, B. Oudheusden, and H. Bijl. Flow visualization and force measurements on a hovering flapping-wing mav 'delfy ii'. In *39th AIAA Fluid Dynamics Conference*, San Antonio, Texas, June 2009. AIAA 2009-4035.
- [8] M. Groen, B. Bruggeman, B. Remes, R. Ruijsink, B. van Oudheusden, and H. Bijl. Improving flight performance of the flapping wing mav delfy ii, 2010. International Micro Air Vehicle conference and competitions.
- [9] F. Bos. *Numerical simulations of flapping foil and wing aerodynamics*. PhD thesis, Technical University Delft, 2010.
- [10] L. Miller and C.S. Peskin. A computational fluid dynamics of 'clap and fling' in the smallest insects. *The Journal of Experimental Biology*, 208:195–212, 2005.
- [11] L.A. Miller and C.S. Peskin. Flexible clap and fling in tiny insect flight. *The journal of experimental biology*, 212:3076–3090, 2009.
- [12] H. Jasak. *Error analysis and estimation in the finite volume method with applications to fluid flow*. PhD thesis, Imperial College, 1996.
- [13] de A. Boer, van der M.S. Schoot, and H. Bijl. Mesh deformation based on radial basis function interpolation. *Computers and Structures*, 85:784–795, 2007.
- [14] C.P. Ellington. The novel aerodynamics of insect flight: applications to micro-air-vehicles. *The Journal of Experimental Biology*, 202:3439–3448, 1999.
- [15] W. Shyy, H. Aono, S.K. Chimakurthi, Trizila P., C.-K. Kang, C.E.S. Cesnik, and H. Liu. Recent progress in flapping wing aerodynamics and aeroelasticity. *Progress in Aerospace Sciences*, In Press:1–44, 2010.
- [16] J. Donea, S. Giuliani, and J.P. Halleux. An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions. *Computer methods in applied mechanics and engineering*, 33:689–723, 1982.
- [17] H. Jasak and Z. Tukovic. Automatic mesh motion for the unstructured finite volume method. *Transactions of FAMENA*, 30:1–30, 2007.
- [18] R. van Loon, P.D. Anderson, F.N. van de Vosse, and S.J. Sherwin. Comparison of various fluid-structure interaction methods for deformable bodies. *Computer and Structures*, 85:833–843, 2007.
- [19] H. Bandringa. Immersed boundary methods. Master's thesis, University of Groningen, 9700 AK, Groningen, August 2010.
- [20] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37:239–261, 2005.

Aerodynamics Study of Fixed-Wing MAV: Wind Tunnel and Flight Test

C. Thipyopas and N. Intaratap

Department of Aerospace Engineering, Faculty of Engineering, Kasetsart University
50 Phaholyothin Rd., Chatuchak, Bangkok 10900, Thailand

ABSTRACT

Obtaining accurate aerodynamic characteristics of in-flight Micro Air Vehicles (MAVs) was viewed as difficult, due to the nature of very low Reynolds number, 3D complex flow, and strong influence from propulsion slipstream. This paper presents the study of the tailless, fixed-wing MAV, KuMAV-001, performed at Kasetsart University. The team investigated different analysis and testing methods to determine the aerodynamics characteristics of this MAV. The Vortex Lattice Method was introduced in the conceptual design phase and helped with the evaluation of the 3D effects for winglet configurations. The wind tunnel tests with main wing and fully configured MAV were conducted for powered and unpowered models. The influence of propulsion-induced flows on C_L , C_D , and $C_M(cg)$ was investigated during the wind tunnel testing. Verification of the performance results are to be completed with flight test data in the future.

1 INTRODUCTION

Micro Air Vehicles have been of interest for more than 10 years. Many concepts have been realized and successfully flown for both military and civilian applications. This recent development has been made possible thanks to the progress in miniature size control systems. Originally aimed for outdoor missions, the MAV design in the last few years has also been tested for indoor flights. ISAE-Supaero initially incorporated an indoor mission to the international MAV flight competition in September 2007.

A fixed-wing MAV is suitable for outdoor missions due to its high forward flight efficiency. This was successfully demonstrated by many fixed-wing MAVs, such as the BlackWidow, a flexible wing platform of the University of Florida and the MAVs from the University of Arizona. On the other hand, a rotorcraft MAV provides good hovering and low speed flight required in indoor missions. There have been various unconventional rotorcraft concepts: tri-rotor, quad-rotor, coaxial rotor as well as flapping-wing type MAVs under continuing development. The famous DelFly [1] has outstandingly presented complex aerostucture mechanism for miniature flying vehicles.

Even with these achievements, more investigation and sophisticated test equipments are still necessary for understanding this difficult subject.

Currently, multi-mission capabilities are of great interest. A new rotorcraft design has been studied and developed for a multi-mission UAV [2]. Many rotorcraft MAVs extend their task for outdoor missions and did reasonably well. In

the last few years, the rotorcraft concept has won over the fixed-wing configuration in both outdoor and indoor competition. Arguably, this accomplishment may have been the result of the limitation of the competition field size. Hence, the fixed-wing configuration could remain an apt option for outdoor assignments, particularly in high turbulence and unsteady flows [3].

Multi-mission MAVs have also been studied in fixed-wing configurations, particularly at the University of Arizona [4] and at the Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-Supaero) [5]. In Thailand, the market survey of UAV applications indicated that a fixed-wing configuration is more fitting to domestic requirements, which include real-time observations of forest, traffic, electrical lines and pipes. The Department of Aerospace Engineering (AE), Kasetsart University (KU) initially started working on a fixed-wing MAV design in 2009 with a primary focus on aerodynamics.

Aerodynamic characteristics are significant to the improvement of MAVs' performance and capacity. At very critically low Reynolds numbers, MAVs' flight efficiency is comparatively poor. Strong three-dimensional flow of extremely low aspect ratio wings introduces more difficulty to the prediction of their aerodynamic characteristics. Furthermore, there are strong flow interactions between propulsion system, wing and airframe. Many studies have been focusing on the effects of propulsion slipstream on the aerodynamics of MAV wings. Longitudinal aerodynamic characteristics affected by propellers have already been extensively investigated [6,7]. Favorable or unfavorable results of the interaction highly depend on the installation angle of the propellers. As displayed in MITE MAV, the strong wingtip vortex of the very low aspect ratio wing is suddenly reduced when a propeller was placed at wingtip. Shkhaevy *et al.* [8] empirically predicted thrust requirement for hovering and level flight of their first VTOL MAV by considering additional drag induced by propulsion. In this design, a wing is divided into two parts; the central part submerged in the propeller-induced slipstream and the external part influenced by freestream only.

At present, the numerical methods are not sufficiently reliable to determine accurate aerodynamic coefficients and often require wind tunnel testing for validation. However, there are very few experimental data of low Reynolds number MAVs unlike larger air vehicles. Both force and speed are relatively small and difficult to measure. To obtain correct aerodynamic characteristics, extremely developed facilities, such as those at the University of

Email address: fengcpt@ku.ac.th

Florida [9] and the ISAE [10], are necessary. The wind tunnel tests are certainly helpful for the study of low Reynolds number problems as seen in many publications. However, such experimental work has not always yielded results corresponding to real-flight characteristics. The study of Watkins [11] exhibited large variations in the real environment flow where highly turbulent and unstable flow was observed. With no conclusion on the best method to predict aerodynamics characteristics of MAV, the wind tunnel test results then should be compared with other methods. Flight test is one solution used by Ostler [12] to study aerodynamic characteristics of a flying wing in leveled flight. Longitudinal aerodynamic coefficients were calculated from flight data.

The present paper describe methods for obtaining the longitudinal aerodynamic characteristics of the tractor-propulsive fixed-wing KU-MAV001. These methods include a simple Vortex Lattice Method, a small-scale wind tunnel test, a flying-scale wind tunnel test with and without propeller, and a flight test.

2 KU-MAV-001: CONCEPTUAL DESIGN AND PROTOTYPE

The team at Kasetsart University started working with a fixed-wing MAV. Due to a lack of experience and research equipments for low Reynolds number aerodynamic testing, the monoplane wing studied by the Arizona State University (ASU) [13] was selected as a baseline model to compare results. The KU design consists of propulsive tractor and elevon configuration as is most often found in MAVs. A relatively larger size wing with a span of 50 cm was selected since local commercial components and experienced pilot were not available for smaller size air vehicle. Estimated mass is around 500 grams. As a result, the model of ASU is scaled up 3 times to obtain a wing span of 45 cm. A fuselage was added to a thick airfoil E212 cross section to allow for practical center of gravity (CG) arrangement. The mission requirement was set with a flight speed of 7 to 15 m/s and minimum endurance of 20 minutes. Finally, the Paparazzi System developed by ENAC team [http://paparazzi.enac.fr/wiki/Main_Page] was chosen for the autopilot version. A summary of design configuration is mentioned in Table 1.

Table 1: KUMAV-001 design configuration

Parameter		Parameter	
Airfoil	E212	Max.Thickness	10.55%c
Root chord	45 cm	Tip chord	39 cm
Aspect ratio	1.07	Span	45 cm
LE Swept	17deg	Dihedral	0 deg
Aerodynamic Center	22% Cr = 10cm LE _r	Center of gravity	7.5% = 7cm LE _r
Control surface	20% LE _r = 9cm	Flight Speed	7-15m/s

The CG location and the size of the control surfaces were achieved using the Vortex Lattice Method. The *Tornado* program [<http://www.redhammer.se/tornado/>] gave the aerodynamic center at 10cm from the leading edge on the wing root chord (LE_r). The CG position was then marked forward at 7cm (static margin of 7.5%). The control surfaces chord length of 20% of the root chord was selected

to minimize drag at a cruise speed of 10 m/s and with no power.

By this point, all components and their weights were known except for the propulsion system. The propulsion set was estimated from the *Qprop* program [<http://web.mit.edu/drela/Public/web/qprop/>]. Various on-shelf propellers with diameters of 15.2 to 22.9cm were calculated and compared. The two main design constrains for the propulsion system were being the most efficient at the cruise speed (10m/s) and able to reach maximum speed of 15m/s with a suitable rotational speed. Hovering ability was also considered in the calculation but finally not taken into account. Four best candidates ended up being the APC 6x4, 7x4, 7x5, and 8x4. Their efficiencies were on the order of 0.43-0.55. The thrust results from Qprop were in good agreement with the static test results, but torque was under predicted. Comparison plots are presented in Appendix A.

Finally, an RC version of KuMAV-001 illustrated in Figure 1 was fabricated by hotwire-cut foam. The total weight was 550 grams (including a three cells lipo-battery has capacity of 1300 mAh).



Figure 1: KuMAV-001.

3 MODELS AND EQUIPMENTS

3.1 Models

This section provides detailed descriptions of the two wind tunnel models investigated in this study. Each model has its own studying objectives.

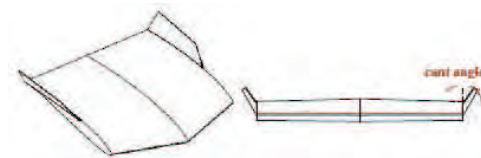
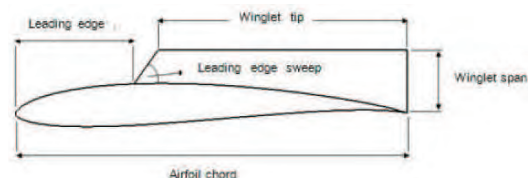


Figure 2: Wind tunnel model for winglet study.

3.1a) *Half-scale wing model*: This model was designed to study the effect of winglets on wing performance. The lateral characteristics were observed by a 6-components balance. The main wing was made of foam covered by composite material for rigidity. The planform dimensions were half scale of those shown in Table 1. Different winglet parameters, including winglet span, cant angle, leading edge sweep angle, and leading edge location as illustrated in

Figure 2, were designed by the Vortex Lattice Method and evaluated by wind tunnel testing.

3.1b) *Full scale and flying model*: This model was primarily used for flight testing. Longitudinal characteristics were determined by wind tunnel testing using a 3-component balance and were then compared to flight data. The model is the same one as presented in Figure 1 of section 2.

3.2 Force Measurement Systems (for Wind Tunnel Test)

As described in section 3.1, there were two force balance setups used in this study: one for the half-scale model and one for the full-scale model. Both force balance were not designed for very low Reynolds number testing, yet they were the only equipment available in the Department.

3.2a) *6-component balance*: The lateral effects of winglets were of primary interest for this study. The aerodynamic characteristics of the half-scale model were measured using a balance equipped with an ATI 6-component Force/Torque sensor (DALTA model) at its base, as illustrated in Figure 3. The measurement ranges were $\pm 330\text{N}$ for axial and side force with $\pm 990\text{N}$ for normal force. The moment capacity was $\pm 30\text{Nm}$. The balance was installed on a turn table under the test section so as to vary the yaw angle.

A two-strut mounting system was used to provide support for the model as well as a mechanism for changing the angle of attack. The model weight was carefully observed and subtracted from the test results.

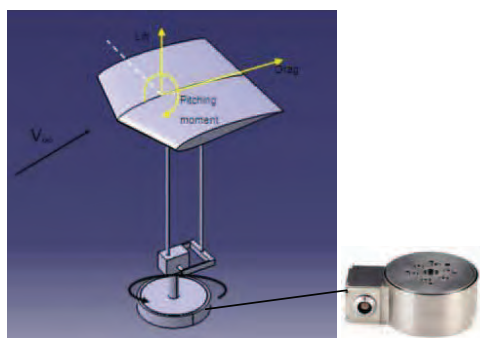


Figure 3: 6-component balance and set up of the half-scale model.

3.2b) *3-component balance*: To simulate level flight conditions, the flying model was supported by a single strut at the CG position. This location was selected to minimize any additional error arising from the force-moment translation. Lift, drag, and pitching moment were obtained directly by a pyramid balance outside the test section. The balance and angle of attack adjustment system using the turn table are shown in Figure 4. The maximum capacities were $\pm 500\text{N}$ and $\pm 50\text{ Nm}$ for force and moment respectively. The accuracies of the balance were 1N and 0.5Nm .

3.3 Closed-Loop Wind Tunnel

Kasetsart closed-loop wind tunnel has a test section of $1\text{m} \times 1\text{m} \times 3\text{m}$ ($W \times H \times L$). A contraction ratio of 4 results the maximum speed of 60 m/s generated by a 2m -diameter fan

with maximum power consumption of 75kW . The highest speed presented in this study was only 15m/s . The layout of the wind tunnel is displayed in Appendix B. The current equipment of the facility did not allow for measurements of the turbulence intensity. Wind speed was measured by a Pitot-static tube installed in front of the model and a digital manometer with an accuracy of 5Pa .

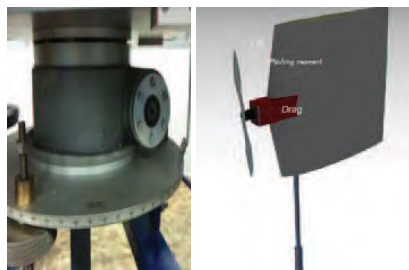


Figure 4: 3-component balance and test set up of the full-scale model.

3.4 Propulsive Measurement

Due to the uncertainties in the results obtained from Q_{prop} , the propulsion sets were tested for both static and dynamic performances. An in-house propulsive test bench, using 6N -load cell, was designed to measure thrust and torque. Future improvements of the test bench will include the measurement of other force and moment components will for characterizing propeller performance at various angles of attack. The schematic of the test bench and equipments are presented in Figure 5. The set up included a DC power supply of 30A and 15V max capacity, RC commercial brushless motor speed counter, RC transmitter, digital volt meters, and a National Instruments NI A/D 24 bits converter. The motor was rated at 11.5V and powered by three battery packs. The measurement parameters included the speed of the propeller (RPM), current (Amp), voltage (Volt), torque (N.m), and thrust (N). Thrust and torque were obtained by averaging 500 samples measured at 500Hz .

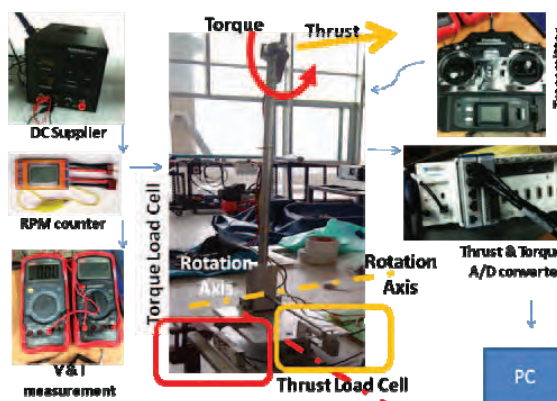


Figure 5: Propulsive Measurement System.

4 METHODOLOGY

4.1 Preliminary Calculation by Vortex Lattice Method

The preliminary estimations of the aerodynamic characteristics for the KuMAV-001 were obtained from the Tornado code. Although the code was not designed for very

low Re flow, the results were acceptable for conceptual design of aerodynamic center, control surface efficiency, and other aerodynamic derivatives at small angle of attacks. The results of these estimations were used for the design of the first prototype mentioned in the section 2.

The code was also used to study the effects of winglets. It gave the preliminary analysis of the winglet configurations, illustrated in Figure 2, used to minimize the number of wind tunnel models.

4.2 Wind Tunnel Test Low Reynolds Number Wing

After successfully obtaining the prototype design from the Vortex lattice Method, the half-scale wing model was tested in the wind tunnel to characterize the effects of winglets on the main wing longitudinal and lateral behaviors. Three winglet models detailed in Table 2 were studied.

The first winglet model was based on the characteristics designed and simulated by ASU. Another two winglet models were selected from Tornado's results. The 2nd winglet model (s60_c60) gave the highest maximum lift-to-drag ratio (L/D) while the 3rd winglet model (s60_c30) had the best lateral stability. The wing model with the winglet s60_c60 is presented in Figure 6.

Table 2: Half Scale Test Model.

Winglet shape	Model			
	No Winglet	8B	S60_c60	S60_c30
Airfoil	-	Flat plate	Flat plate	Flat plate
LE-Swept angle	-	26.6 deg	60 deg	60 deg
LE-Location	-	25.2% c	25.2% c	25.2% c
Span	-	16.9% c	16.9% c	16.9% c
Cant angle (from vertical)	-	0 deg	60 deg	30 deg



Figure 6: Half Scale wing model with winglet s60_c60.

4.3 KuMAV-001RC's Flight Test

As previously mentioned in section 2, the first RC-prototype of the KuMAV001 was built and tested. The first flight was performed with the APC8x6 propeller. Due to limited experience in flying a very low aspect ratio wing model, the flight test did not initially achieve all the objectives. The torque from the propeller was too high for our pilot to control. Winglets were consequently added to damp the rolling motion. The winglets had a 10cm span with a length of 30cm.

The second attempt completed with an endurance of 10 minutes. The third flight was tested by changing the propeller to the APC7x4 model without winglet. This model flew but unstable roll was still an issue.

The addition of the smaller winglets to the same APC7x4

propeller resulted in the first fully successful flight of the KuMAV-001RC.

4.4 Finding Propulsive Characteristics

To evaluate the propulsion performances, the static and dynamic thrusts of a set of propellers (APC8x4, APC7x5, APC7x4 and APC6x4) were measured using the propulsive test bench mentioned in section 3.4.

Thrust and torque of motor-propeller were recorded at different motor speeds, controlled by a radio transmitter sending the PWM signal to the propulsive system. Other parameters measured included motor speed, voltage and current.

Dynamic thrust of selected propeller was also investigated in the wind tunnel at the freestream, velocities of 5, 7.5, and 14.5 m/s. Although propulsive characteristics would change when a propeller was at incidence [14], the current set up has only been studied at zero degree AoA (the propeller disc was normal to freestream flow). The impact of the propeller incidence should be small as long as the angle of attack is less than 20 degrees. The picture of the dynamic thrust test bench used in the wind tunnel is illustrated in Figure 7a.

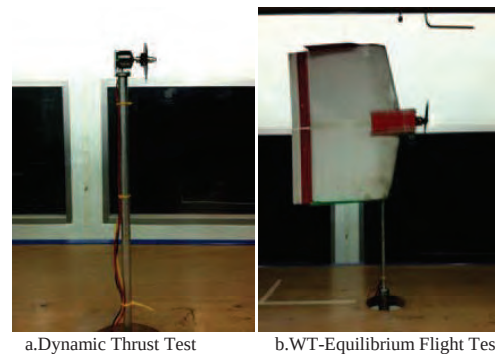


Figure 7: Wind tunnel test of propeller and full scale model

4.5 Equilibrium Test of Powered MAV: Wind Tunnel

Due to the difficulty and low accuracy of the measurement system for the in-flight test, a wind tunnel experiment was conducted at wind speeds of 5, 7.5, and 14.5 m/s to validate the results. The wind tunnel tests were performed with the real flight RC-model equipped with the APC7x4 propeller as presented in Fig. 7b. Both configurations (the models with and without winglets) were investigated. The equilibrium points (or trim point) for level flight were run at the same wind speeds of 5, 7.5, and 14.5m/s. The throttle, AoA, and elevon setting were adjusted in order to reach zero pitching moment and minimum drag at a fixed wind speed.

The drag correction from the strut effects was performed in real time during the measurements. One of the advantages of the wind tunnel tests is that they generated several data points for different lift configuration while for the flight test, the weight of the model was rather constant or hardly modified (primarily by changing the battery size). In this test, the lift forces of 300g to 500g were produced in the wind tunnel. All necessary parameters were collected including lift, motor speed, motor electrical data, AoA, and approximated elevon deflection angle.

4.6 Traditional Wind Tunnel Test on Unpowered MAV

The wind tunnel testing of the unpowered full-scale model was carried out at the same speeds of 5, 7.5, and 14.5 m/s. At each wind speed, a full polar curve with AoA between 0 and 36deg were produced. Also, the effects of the control surfaces on the aerodynamic characteristics were carefully observed at full deflection using an RC transmitter control.

4.7 Equilibrium Test of Powered MAV: In-Flight

The second flight model was fabricated with the same material and method as the RC model. At this point, the test-flight is scheduled to include a Paparazzi system (PZ), with the normal TBOG system. Due to some shipping delay for the PZ system, the test-flight phase is still in progress. However, the PZ system was successfully tested inside a 1m-span conventional airplane model as illustrated in Figure 8.

The procedure consists in flying the KuMAV-001AT at constant level through two points A and B separated by a distance of 500m and observing the data during the flight. At each speed, 10 flights are performed and results are consequently averaged to determine the longitudinal aerodynamic characteristics. Now that the PZ system has been validated, the flight test is expected to be completed soon.

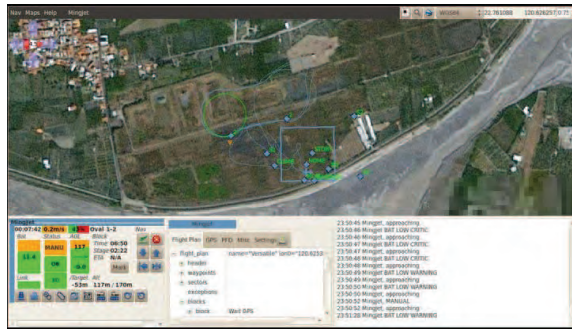


Figure 8: Flight trajectory of Paparazzi system done by conventional plane.

4.8 Result and Data Correction

Two corrections were required for the wind tunnel test data. The first one was a wind tunnel correction due to wall effect. The standard blockage and wall correction of Barlow et al. [15] was applied to the primary data, with a particular focus on the change in AoA. For the equilibrium test in the wind tunnel, the propulsive force or thrust (T) was subtracted out to determine the exact wing aerodynamic characteristics. The equilibrium level flight is detailed in Equations 1-3 from which the C_L , C_D , and C_M of MAV can be calculated.

$$(1) \quad \sum F_x = 0: T \cos(\alpha) + D = 0$$

$$(2) \quad \sum F_z = 0: T \sin(\alpha) + L + mg = 0$$

$$(3) \quad \sum M_y = 0: Tz + M_{cg} = 0$$

In Eq. (1-3), L , D , M , m , g and α are respectively the lift, drag, pitching moment, mass, gravitational acceleration and AoA. Since the thrust vector was aligned with the wing chord line, the distance z is equal to zero.

5 RESULTS AND COMPARISON

5.1 Comparison with ASU model

The lift coefficients for the half-scale model without winglet are comparable to the numerical values produced by ASU (Figure 9). Both results have similar lift-curve slopes. The wind tunnel results appear to have some non-linearity in the lift curve slope possibly due to some uncertainty in the angle of attack.

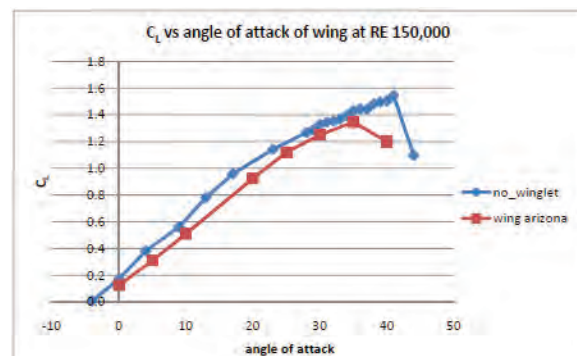


Figure 9: Comparison of CFD and Wind Tunnel Test.

5.2 Effect of winglet

The effects of the winglets on the half-scale model performance and stability were studied in the wind tunnel at Reynolds numbers between 150,000 and 250,000. Longitudinal static stability does not appear to be influenced by the winglets. Figure 10 shows however that the static lateral stability is clearly affected. The most stable model is the 8B model due to its cant angle of 0 deg and to the fact that the vertical projection of the winglet area behind the aerodynamic center is greater than for other models.

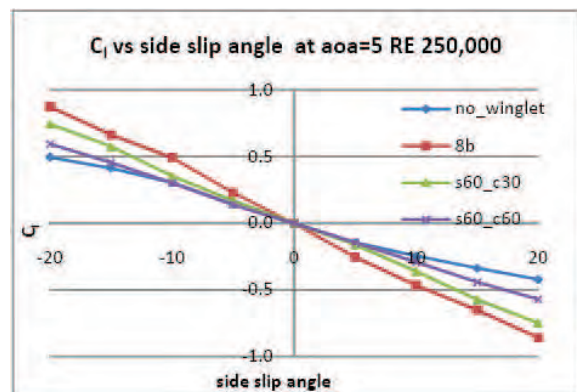


Figure 10: Effect of winglet on lateral static stability.

Winglets also have a strong effect on L/D. They are usually found to increase the lift curve slope and drag. Considering the L/D plot shown in Figure 11, the winglets do not appear to improve the wing aerodynamic performance. On the contrary, they seem to reduce the L/D of the MAV wing.

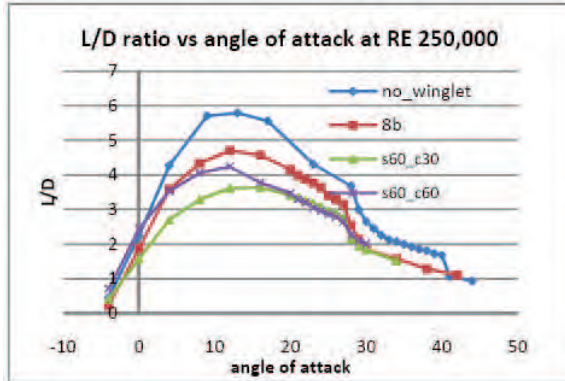


Figure 11: Effect of winglet on L/D.

5.3 Thrust results

Propulsive force measured by load cell is presented in Figures 12 and 13. Thrust is plotted against the electric input power consumption in order to determine the most efficient propulsive system. The APC 6x4 and 7x4 propellers produce the best efficiencies. Comparison between these two propellers reveals that the APC 7x4 is able to provide higher maximum thrust. Since this propeller has also already been tested in the RC flight test, it was selected for the next step of the project.

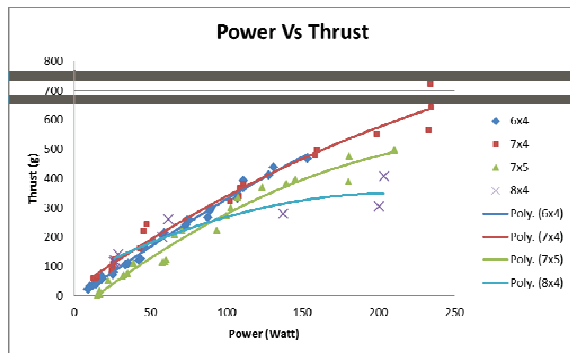


Figure 12: Static Thrust of Propeller APCs.

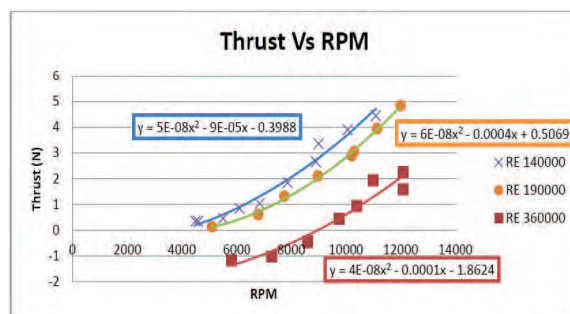


Figure 13: Dynamic Thrust of Propeller APC 7x4

The dynamic thrust results are used in the thrust prediction of the powered MAV model. Dynamic thrusts for the APC 7x4 propeller at wind speeds of 5.5, 7.5, and 14.5 m/s are plotted as a function of propeller speed. In Figure 13, dynamic thrust is plotted for Reynolds numbers of 140000, 190000, and 360000, corresponding to the speeds of 5.5, 7.5, and 14.5m/s respectively. The data was fitted with polynomial functions to extract the variation of the dynamic thrust with propeller speed. Such analysis were also performed as function of the input current but is not presented here. It will be however used to present with the flight test results.

5.4 Full Scale Result

This project also plans to test full-scale flight model in both by wind tunnel and free flight environments. Due to some delay in obtaining the flight control system, the data presented in the current study pertains only to the wind tunnel test. The flight test results will be published at a later date.

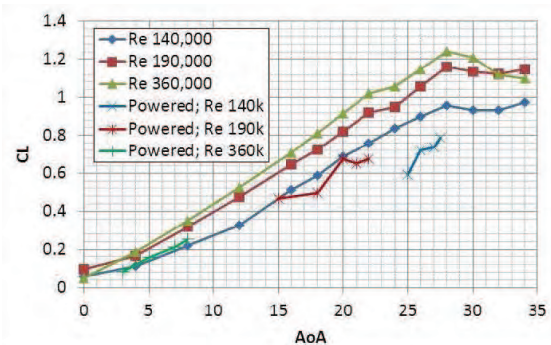


Figure 14: Lift Coefficient of Full Scale Model.

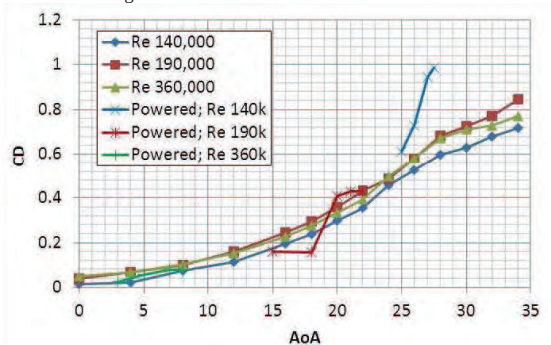


Figure 15: Drag Coefficient of Full Scale Model.

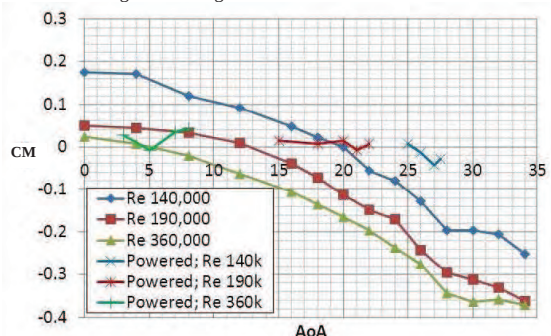


Figure 16: Pitching Moment Coefficient at C.G. of Full Scale Model.

The results for lift, drag, and pitching moment of the model without winglets are presented in Figures 14-16. C_L , C_D and $C_M(cg)$ of powered and unpowered models are computed and plotted. Firstly, the Reynolds number effects are clearly revealed by the results of unpowered model as the lift curve slope increases with Re . Stall angle is consistently at about 28 degrees. Comparing the lift coefficient of the powered and unpowered models at high Re and low AoA shows that the C_L of the powered model is slightly lower than that of the unpowered model. The difference increases with decreasing Re and increasing AoA. These results clearly highlight the effects of propulsive induced flow on the main wing aerodynamic characteristics.

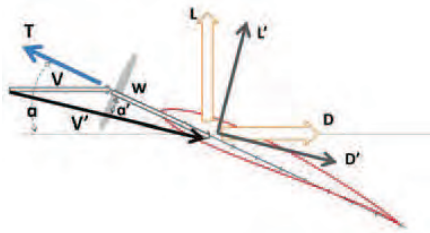


Figure 17: Schematic of Propulsive Induced Flow Vector.

Following the nomenclature presented in Figure 17 [16], an MAV is assumed to be flying from right to left with a speed V and at the AoA of α deg. Without propeller, lift and drag are represented by the thick orange arrows. When the propeller is introduced, thrust T is then produced (tractor propeller in this configuration). According to momentum theory, induced velocity w is obtained by Equation 4.

$$(4) \quad w = \sqrt{[(2T)/(\rho A)] + [V \cos(\alpha)]^2}$$

Therefore, the resultant velocity vector acting on the main wing is V' and the local AoA is α' as illustrated. There are 2 main effects of propulsive-induced flow which directly impact the aerodynamic characteristics: the reduction of local AoA (from α to α') and the increase of the local speed (from V to V') resulting in the modified local lift and drag L' and D' respectively. Finally, the lift and drag force on the MAV must be a projection vector product of L' and D' to $(X$ and $Y)$, the freestream aerodynamic axes with an angle of $(\alpha - \alpha')$. The larger local velocity enhances the aerodynamic coefficients while the reduction of the local AoA degrades them. However, according to Figure 14, C_L s decreased due to the effects of propulsive-induced flow as a result of the reduction in the local AoA rather than the increase of the local velocity.

Drag coefficient results shown in Figure 15 have no clear tendency except at the lowest Re . For high Re , the C_D of the powered model is just slightly less than that found in the unpowered model, but it is higher at the lowest Re . At the high speed, the enhanced velocity behind the propeller is negligible compared to that at low speed. Therefore it appears that the increment of drag force from induced speed is low.

Consider now a regular curve of C_D vs AoA. C_D changes very little at very low AoA (parabola curve). Thus, the

reduction of local AoA induced by the propeller does not present in the C_D curves. On the other hand, at high AoA and low flight speed, the propeller flow strongly affects C_D on both AoA and local velocity account which may compensate each other. However, the sum of drag also includes the component of L' as the local AoA changes. Consequently, additional ('propulsive' induced) drag is presented on the model. The net effect is the same for the pitching moment presented in Figure 16. Hence, more accurate measurements of the force and local flow speed behind the propeller are necessary to obtain a clear picture of the flow behavior at each flight speed. Hopefully, the flight test results will provide an insight and confirm the wind tunnel data.

6 CONCLUSIONS

This paper presents the development of Micro Air Vehicles (MAVs) at the Department of Aerospace Engineering, the Faculty of Engineering, Kasetsart University, Bangkok, Thailand. The 45cm-span fixed-wing MAV named KuMAV-001 was designed and studied. The wing model was based on the study of the Arizona State University (ASU).

Aerodynamic characteristics of KuMAV-001 were investigated through several methods including an estimation by the Vortex Lattice Method from *Tornado* code, half-scale and full-scale wind tunnel tests, powered model wind tunnel tests, and data from flight test.

New experimental facility was designed and constructed to examine the propulsive characteristics. The *Qprop* code was used to validate the measurements. Dynamic thrust of various propellers was also measured in a 1m×1m test section wind tunnel. The effects of different winglets were evaluated by half-scale wind tunnel testing using 6-component force balance to evaluate lateral characteristics as well as longitudinal characteristics. The results were then compared with that provided by ASU.

A full scale model was fabricated and tested in both free flight and in a wind tunnel. In the wind tunnel test, the measurement for the full scale model was performed using a 3-component longitudinal force balance. The test was conducted to simulate level flight conditions; pitching moment = 0 and result drag force = 0. An unpowered model was also built and tested for conventional aerodynamic data.

Due to the propulsive induced flow effects on the central part of model, local angle of attack and velocity were modified. This phenomenon explains the difference in the longitudinal aerodynamic characteristics between powered and unpowered models. A method by McCormick on the propulsive induced effects was offered as an explanation for the difference between the wind tunnel results of powered and unpowered models. However some of this difference may also be the result of errors introduced by the accuracy of facility and measurement system.

7 ON-GOING EFFORTS

MAV activity is just starting at Kasetsart University. But the success of the current study highly motivates team members and students. Students suddenly got an opportunity to apply what they had learned from their classes while discovering an entertaining new field. The

MAV study is continuing since it is very attractive and a field of great potential in Thailand.

The work is now focused on the development of new multi-mission platforms aimed at using new MAV in forest and city environment. VTOL and low speed flight capacity are very necessary for such areas. New propulsion system with better capabilities for forward and hovering flights will be researched as well as the study on optimizing propeller-wing interaction. Active morphing wing with propulsive interaction will be another interesting option to improve flight performance. A new sensor for non-GPS environment will be studied as inter-departmental research projects. The Department of Aerospace Engineering at Kasetsart University will welcome any cooperation in MAV research and education.

ACKNOWLEDGMENT

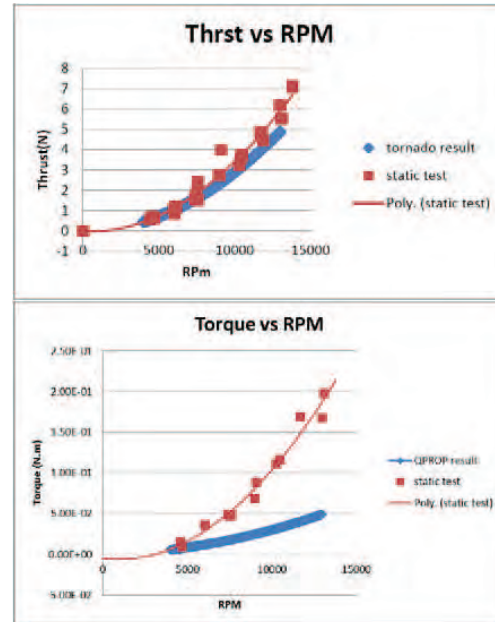
The authors would like to thank the Faculty of Engineering, and the Department of Aerospace Engineering who financially supported this work. This work could not be completed without the long hours students spent on the models and many tests. Finally, thank you to the Paparazzi team for their good job on this open source system.

REFERENCES

- [1] G. Mark, B. Bart, R. Bart, R. Rick, O. Bas van, and B. Hester, "Improving flight performance of the flapping wing MAV DeFly II", Proceedings of IMAV 2010, Braunschweig, Germany, July 6-9, 2010.
- [2] G. Grondin, C. Thipyopas, and J.M. Moschetta, "Aerodynamic Analysis of a Multi-Mission Short Shrouded Coaxial UAV: Part III – CFD for Hovering Flight", 28th AIAA Applied Aerodynamics Conference, Chicago, Illinois, June 28-1, 2010.
- [3] M. Abdulrahim, S. Watkins, R. Segal, M. Marino, J. Sheridan, "Dynamic Sensitivity to Atmospheric Turbulence of Unmanned Air Vehicles with Varying Configuration," Journal of Aircraft 2010, Vol. 47, no. 6, p. 1873-1883.
- [4] J.M. Moschetta, B. Bataillé, C. Thipyopas, and S. Shkarayev, "On-Fixed-Wing Micro Air Vehicles with Hovering Capacities", 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan 7-10, 2008.
- [5] R. Carr, J.M. Moschetta, G. Mehta, and C. Thipyopas, "A Tilt-Body Fixed-Wing Micro Air Vehicle for Autonomous Transition Flight," Proceedings of IMAV 2010, Braunschweig, Germany, July 6-9, 2010.
- [6] C. Thipyopas and J.M. Moschetta, "A Fixed-Wing Biplane Micro Air Vehicle for Low Speed Missions," International Journal of Micro Air Vehicles, Vol.1, p. 33.
- [7] W. Null, A. Noscek and S. Shkarayev, "Effects of Propulsive-Induced Flow on the Aerodynamics of Micro Air Vehicles," 23rd AIAA Applied Aerodynamics Conference, Toronto, June 6-9, 2005.
- [8] S. Shkarayev, J.-M. Moschetta, and B. Bataillé, "Aerodynamic Design of Micro Air Vehicles for Vertical Flight", Journal of Aircraft, 2008, Vol. 45, no.5, p. 1715-1724.
- [9] R. Albertani et al., "Validation of a Low Reynolds Number Aerodynamic Characterization Facilities", 47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Orlando, Florida, Jan 5-8, 2009.
- [10] Documentaion of Sarb Wind Tunnel, internal report, ISAE, Toulouse, France
- [11] S. Watkins, M. Abdulrahim, M. Marino, and S. Ravi, "Flight Testing of a Fixed Wing MAV in Turbulence with Open and Closed Loop Control," Proceedings of IMAV 2010, Braunschweig, Germany, July 6-9, 2010.
- [12] J.N. Ostler, "Flight Testing Small, Electric Powered Unmanned Aerial Vehicles," Master thesis, Brigham Young University, 2002.
- [13] J. Montinen, "On the Performance of Micro-Aerial-Vehicles," Mechanical and Aerospace Engineering Thesis, Arizona State University, 2003.
- [14] D. Gomez Ariza and J.M. Moschetta, "The Lateral Force Effect on Rotors at Incidence: Application to a Coaxial Rotor Mini-UAV Tail Sitter," 46th Symposium of Applied Aerodynamics, Orleans, France, March 2011.

- [15] J.B. Barlow, W.H. Rae, and A. Pope, "Low-Speed Wind Tunnel Testing," Wiley 1999, ISBN0471557749.
- [16] B.W. McCormick, Aerodynamics of V/STOL Flight," Dover Publications, INC, 1999, chapter 8.

APPENDIX A: WIND TUNNEL SCHEMATIC



APC 6x4: Result of *Qprop* is compared with the experimental test result. *Qprop* well predict static thrust but it is not good for calculate the torque of motor-propeller.

APPENDIX B: WIND TUNNEL SCHEMATIC

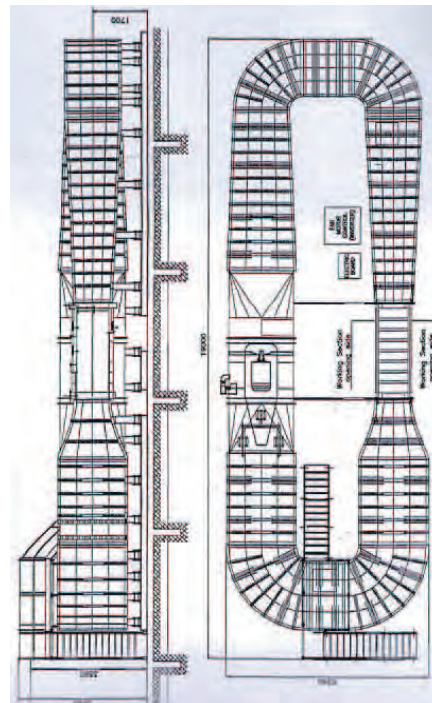


Figure A: Closed-loop low speed wind tunnel of Department of Aerospace Engineering, Kasetsart University at Sriracha Campus.

One useful propeller mathematical model for MAV

S.V. Serokhvostov* and T. E. Churkina**

*Moscow Institute of Physics and Technology, DAFE, Gagarina 16, Zhukovsky, Russia

**Moscow Aviation Institute, Volokolamskoe shosse 4, Moscow, Russia

ABSTRACT

A model for Micro Air Vehicles (MAV) propeller performance evaluation is proposed and investigated. This model is derived on the basis of simple analytical relationships and is examined on a set of experimental data. It allows to predict the main propeller characteristics (for example, thrust coefficient, power coefficient, efficiency as functions of advanced ratio) and gives the method for investigation of the powerplant as whole at the stage of preliminary design.

1 INTRODUCTION

Micro Air Vehicles (MAVs) have some peculiarities compared to other aircrafts. First of all, the characteristic Reynolds numbers Re are small enough and correspond to the region of laminar-turbulent transition for the wing and below this region for the propellers. This leads to higher values of the wing drag and lower propeller efficiency compared to larger airplanes. The next one is rather high dimensions of propeller compared to the wing span (up to 1/3-1/2 of wing span). Thirdly, the amount of energy onboard corresponds to flight times of order of 0.5-1 hour. So, to make a small airplane with high performance a thorough investigation and optimal matching of all the aircraft components is required, and one of critical components is propeller.

To save time and money during the design process the preliminary investigation and calculation based on some mathematical models of aircraft components should be carried out. As for the wing, useful simple formulas for the preliminary design already exist. The situation with the propellers is not good enough. So, the investigation proposed is an attempt to create rather simple but useful mathematical model of MAV propeller for preliminary MAV design and investigations.

2 PROPELLER CHARACTERISTICS AND PROPELLERS INVESTIGATED

Any model is useful if it describes a wide range of objects in/or a wide range of conditions. So, our propeller model must give and predict the characteristics for various propeller blade shapes, angles of blade installation, rotational frequencies, flight velocities and Reynolds numbers Re . On the other hand, it is not required to describe all the propellers that man can make but only "good" ones because nobody will use "bad" propellers. So, authors have decided to make and verify the model on the basis of well known propellers with rather good characteristics (efficiency, thrust etc.).

Nowadays a huge amount of experimental data for the

propeller characteristics at low Re can be found in the articles, books and at the Internet sites, but authors have chosen only the sources with enough amount of required information ([1]-[6]).

First of all, data for the model of AV-31 propeller was chosen [1],[2]. This propeller is 20-sm copy of the larger propeller (of 1.1 meter diameter). One of the peculiarities of this model propeller is the ability to change the angle of blade installation. This small propeller was used in the experiments of Reynolds number influence on the propeller characteristics [1],[2]. As one of the authors took part in these investigations he possesses the required geometrical characteristics and enough experimental data for this propeller. Reynolds numbers in these experiments for the blade chord at $\frac{3}{4}$ propeller radius were $Re=5 \cdot 10^4 \div 9 \cdot 10^4$.

Other propeller data used in the investigation are those of Black Widow propeller [3] as this propeller has very high value of the efficiency for its range of Re (more than 80%). Also the shape of its blades is "unusual" comparing, for example, with model aircraft propellers (large chords, high pitch). Reynolds number range was $Re \approx 10^4 \div 3 \cdot 10^4$.

The last part of data chosen is taken from [4]-[5]. It is a database of model airplane propellers experimental investigations. Reynolds number range in these experiments was $Re=2.5 \cdot 10^4 \div 10^5$. Dimensionless chord length distribution along the radius is the same within one series of propeller (for example, for all the APS Thin Electric). Also for vast majority of propellers examined (APC, JWS, Kyosho, Graupner) the value of $r \cdot (r)$ remains practically constant in the range of $0.2R < r < 0.9R$ (r is the distance from rotational axis, R is propeller radius, (r) is angle of profile installation distribution along the blade). This fact is not valid only for Master Aircscrew propellers. Such a parameters distribution is conventional for model airplane propellers as it gives simple way for design and fabrication of the propeller with acceptable efficiency but is not good enough. It gives zero induced drag (and torque) at zero thrust. But it is much more better to have minimal drag (and torque) and, as consequence, the highest efficiency at working regime.

To check data from [4,5] some data from [6] were used.

Very useful for the propeller performance description are dimensionless characteristics: thrust coefficient C_T , power coefficient C_P and efficiency as function of advanced ratio :

$$= \frac{V}{nd},$$

$$C_T = \frac{T}{n^2 d^4},$$

Email address: serokhvostov@mail.ru

$$C_p = \frac{P}{n^3 d^5},$$

$$= \frac{C_T}{C_p},$$

where V – air velocity at infinity, n – frequency of propeller rotation, ρ – air density, d – propeller diameter, T – propeller thrust, P – propeller power.

It is well known that for the fixed Reynolds number these coefficients are independent of dimensions and velocities. Also for $Re > 10^6$ these characteristics are independent also of Reynolds number, i.e. all the geometrically similar propellers in this range of Re have the same dimensionless characteristics. At lower Re the characteristics can change with Re (see, for example [1],[2]). Figures 1 and 2 show this change. Here $\alpha_{0.75}$ is blade installation angle (commonly measured at 0.75 of the blade radius R). To see the change in characteristics one can compare, for example, C_T and efficiency at $\alpha_{0.75}=20^\circ$ for 2500 and 5000 revolutions per minute (RPM)) (corresponding $Re=5 \cdot 10^4 \div 10^5$).

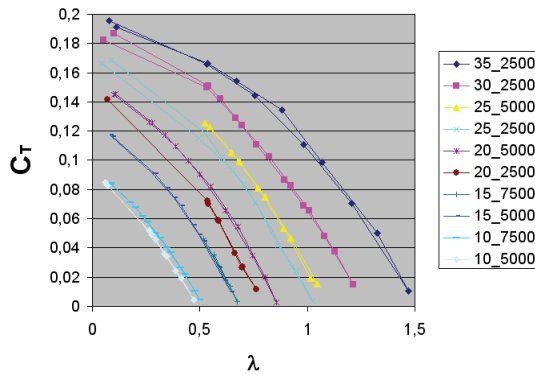


Figure 1: Thrust coefficient as function of advanced ratio and angle of blade inclination for AV-31 propeller (legend: $\alpha_{0.75}$ _RPM)

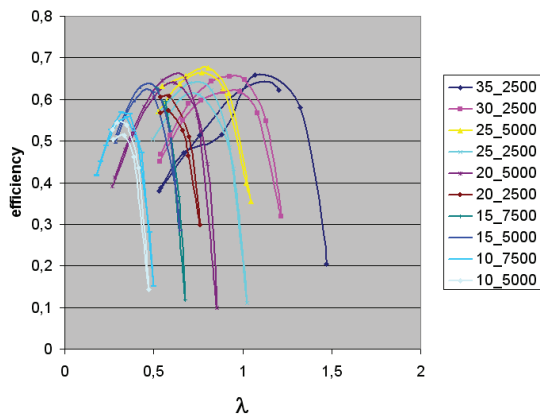


Figure 2: Efficiency as function of advanced ratio and angle of blade inclination for AV-31 propeller (legend: $\alpha_{0.75}$ _RPM)

But this change is noticeable only for the propellers with thick blades. Figures 3 and 4 show the dimensionless characteristics of Black Widow propeller (derived from data of [3]), Reynolds number changes in the range of 10000-30000. It can be seen that for the propeller with thin blades

this change of characteristics with Re is negligible.

So, we can postulate that it is preferable to use propellers with thin blades on MAVs and one can use dimensionless coefficients without the correction on Re changes in this case.

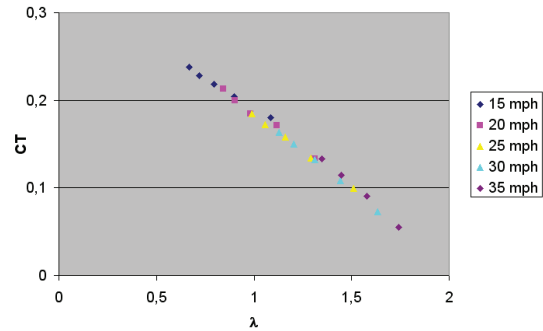


Figure 3: Thrust coefficient as function of advanced ratio for Black Widow propeller at various free stream velocities

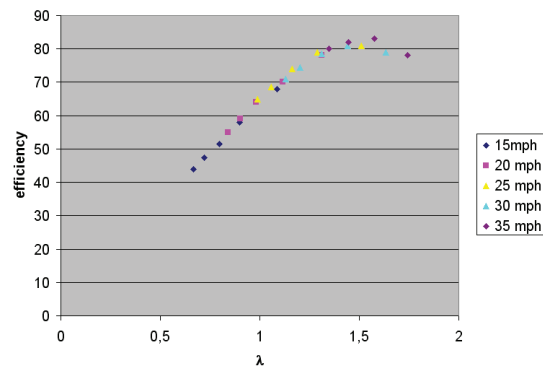


Figure 4: Efficiency as function of advanced ratio for Black Widow propeller at various free stream velocities

3 THRUST COEFFICIENT

One can see from the graphs of $C_T(\lambda)$ (in Figures 1 and 3, for example) that there exist two characteristic regions of $C_T(\lambda)$. Near the zero values of thrust coefficient the dependence $C_T(\lambda)$ is linear:

$$(1) \quad C_T = C_{T0} - C_1 \lambda,$$

where C_{T0} and C_1 are some coefficients.

For the propeller with rather high blade inclination angles near the zero advanced ratio values the thrust coefficient is nearly constant (it can be explained by flow separation on the blade at high angles of attack).

The first region is more interesting in the case of high efficiency requirements. The second region is of interest for the hovering mode.

The behavior of the graph in the first region can be explained theoretically. Let's consider a small part of propeller blade (profile) at the distance r from the axis of rotation. If the angular velocity is ω , then the rotational velocity of the profile considered is ωr .

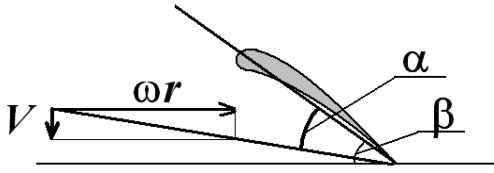


Figure 5: Angles of α and β definition for the propeller blade profile at distance r from the axis of rotation.

If the profile is at the angle β with respect to the plane of rotation (angle of inclination) (see Figure 5), then the angle of attack α for this profile is

$$\alpha = \beta(r) - \arcsin\left(\frac{V}{\sqrt{V^2 + (\omega r)^2}}\right).$$

If $V < \omega r$, the last formula can be simplified to

$$\alpha = \beta(r) - \frac{V}{\omega r}.$$

So, the thrust of this profile dT is

$$dT = C_L \left(\beta(r) - \frac{V}{\omega r} \right) \times \frac{V^2 + (\omega r)^2}{2} \frac{r}{\sqrt{V^2 + (\omega r)^2}} b(r) dr.$$

As $V < \omega r$, one can write this expression as

$$dT = C_L \frac{1}{2} (\omega^2 r^2 b(r) r^2 dr - V b(r) r dr).$$

From this formula one can obtain

$$(2) \quad C_{T0} = \frac{1}{8} C_L \int_{-1}^1 (\bar{r})^2 \bar{b}(\bar{r}) \bar{r}^2 d\bar{r},$$

$$(3) \quad C_l = \frac{1}{8} C_L \int_{-1}^1 \bar{b}(\bar{r}) \bar{r} d\bar{r},$$

where letters with overline are dimensionless with respect to the propeller radius R .

One can also take into account the drag force on the profile. If the drag force coefficient at zero lift for profile is C_{D0} , the projection D_T of viscous drag force on the axis of thrust can be expressed as

$$D_T = \frac{1}{2} \rho V^2 d^4 \frac{1}{8} C_{D0} \int_{-1}^1 b(\bar{r}) \bar{r} d\bar{r}.$$

Corresponding coefficient C_{1D} is

$$C_{1D} = \frac{1}{8} C_{D0} \int_{-1}^1 b(\bar{r}) \bar{r} d\bar{r}.$$

Assuming that C_{D0} is about 0.03 and C_L is about 2 (theoretical result for 2D flat plane), the value of C_{1D} is about 0.5% of C_l , and C_{1D} cannot be taken into account at the first approximation.

The last expression and formula (3) show that coefficient

C_l must be practically the same for the series of propeller with the same $b(r)$ but various $\beta(r)$. Moreover, from (3) C_l must be equal for all the propellers with the same value of

$$C_L \int_{-1}^1 \bar{b}(\bar{r}) \bar{r} d\bar{r}$$

Data from [1], [2], [4], [5] show that for small values of C_T this fact is valid within the experimental errors.

Expressions (2)-(3) allow not only to explain the behavior of $C_T(\beta)$, but also to obtain the values of C_{T0} and C_l for the given $\beta(r)$, $b(r)$ and profile shape. So, on the other hand, these formulas can be verified using the experimental data for propellers.

The verification was made by authors for a set of propellers from UIUC database [5]. Here, geometrical parameters $\beta(r)$, $b(r)$ and experimental data of APC, JWS, Kyosho, Graupner and Master Airscrew propellers can be found. The two problems were the absence of information on the profile shape of propeller "chords" and the value of C_L for profile. So, it was assumed that the curvature of the profile corresponds to the additional angle of attack of 0.05 radians ($\sim 3^\circ$). As C_L is in both formulas then the ratio of C_{T0}/C_l is independent of C_L and corresponds to β_0 for zero thrust coefficient (denote it as β_0). Analysis shows that for the propeller examined the theoretical value coincides with the experimental data good enough, the errors are less than 5%. Also, it was found that for $C_L = 5$ the theoretical and experimental data of C_T coincide with good accuracy.

It should be mentioned that the comparison of theoretical and experimental data with the results from program PropCalc [7] shows that the theoretical predictions are not worse than numerical calculation from PropCalc, but only for the region of low C_T values.

Also one can see from the graph in Figure 1 that the values of β_0 for zero thrust are nearly proportional to the angle of blade installation. This fact is very useful for the stage of the preliminary aircraft design as it can help to find the best propeller rather quickly. The abovementioned proportionality becomes evident if we look at the expressions for C_{T0} and C_l . As C_l is independent of the angles of profile installation $\beta_{0.75}$, the value of β_0 is determined by C_{T0} .

$$(4) \quad C_{T0} = C_l \beta_0.$$

In turn, the value of C_{T0} is proportional to this angle (see (2)).

More accurate investigation shows that for zero angle of installation the corresponding value of β_0 is not zero (it is due to the profile centerline curvature), but the shape of $\beta_0(\beta_{0.75})$ remains linear function for the fixed Re . Figure 6 shows this dependence for AV-31 and APC Thin Electric 11" propellers.

Unfortunately, because of thick blade chords profile at AV-31 propeller one cannot obtain the right result of the Reynolds number influence on the function of $\beta_0(\beta_{0.75})$, but the results for Black Widow propeller give the hope that this influence is small enough. (See Fig. 3, all the points are nearly on the one line.)

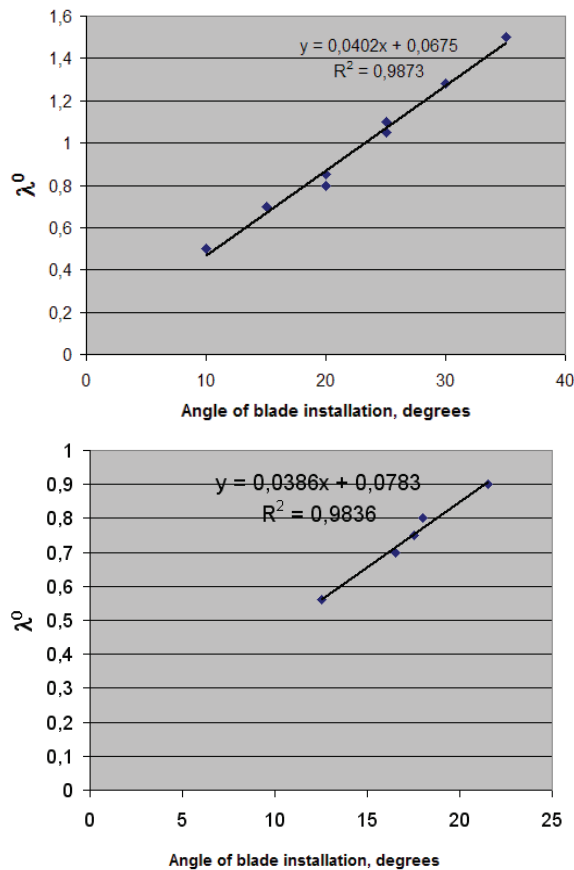


Figure 6: Zero-thrust advanced ratio as function of angle of blade installation for AV-31 (upper) and APC Thin Electric 11" (lower)

Results from [4], [5] show that λ_0 becomes ~5% lower when Re becomes 2 times lower for the same propeller. Also coefficient C_1 becomes slightly lower when Re becomes 2 times lower.

4 POWER COEFFICIENT AND EFFICIENCY

For the propellers at high Reynolds numbers the following semi-empirical formula exists [8]:

$$\frac{C_P}{2} = a \frac{C_T}{2} + b.$$

This expression can be rewritten in the form of

$$(5) \quad C_P = aC_T + b^2.$$

For the power and thrust this formula looks like

$$P = nd(aT + b^2V^2)$$

Authors examined these formulas on the experimental data [3], [5], [6] for low Re propellers. It was found that real dependency of $C_P(C_T)$, (λ) is not a straight line for some kinds of propellers, but its curvature is small enough so one can assume it as straight line at least for the region of high efficiency. As an example, for the Black Widow propeller it is "very good" straight line, for set of APC Electric, Graupner and Master Airscrew propellers the graph angle of incidence becomes lower at high $C_T/2$, but at least for APC 8x8.5 propeller this angle becomes little higher at high

$C_T/2$. Also one must keep in mind that the measurements near zero give higher value of the relative error, so the higher values of the error will be in the verification of formula (5) for low values of λ .

For the region near the zero values of formula (5) becomes

$$C_P \approx aC_T,$$

and (as C_P and C_T at this region do not change significantly) the expression for efficiency can be written as

$$(6) \quad \eta \approx \frac{1}{a}.$$

One can see that the graph of (η) is linear at low λ (see, for example, Figure 2), so expression (6) can be really used for this region of λ . Also one can use it for value of the coefficient a determination (if required).

But all the above mentioned in this chapter facts are valid for the fixed propeller and fixed Re . So, first of all, it is interesting to test this dependency for various Reynolds numbers.

Figure 7 shows the corresponding data for Black Widow propeller [3]. Shown in Figure 8 is the result of APC Thin Electric 11x10 propeller data comparison [5] for various frequencies of rotation. The lowest frequency for APC 11x10 (3000 RPM) corresponds to $Re \approx 36000$. One can see that the change of Re from 36000 to ~60000 gives practically no effect, the graphs are practically the same.

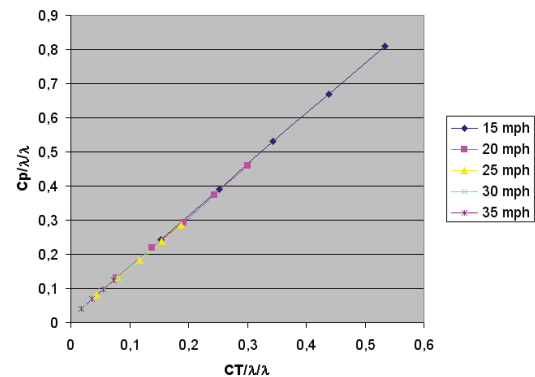


Figure 7: $C_P/2$ as function of $C_T/2$ for Black Widow propeller

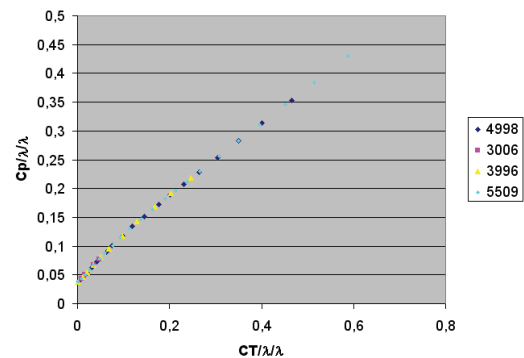


Figure 8: $C_P/2$ as function of $C_T/2$ for APC 11x10 propeller for various RPM

Also it is rather important to know the behavior of a and b coefficients with blade inclination angle change. In other words, the question is what is the difference of these coefficients, for example, for APC 11×5.5 and APC 11×10 propellers.

Figure 9 shows this dependency for APCE 11" propellers family.

One can see that values of a and b change with the propeller pitch change. To make more accurate investigation it is better to use data for AV-31 propeller. Figure 10 shows the values of a as functions of λ^0 . (Here "eff" corresponds to the points near maximal efficiency, "all" corresponds to all the points from experiment.) One can see that a is nearly proportional to λ^0 :

$$(7) \quad a = k \lambda^0,$$

where k is proportionality coefficient.

The same result can be obtained also from other data. For example, Figure 11 shows the efficiency as function of advanced ratio for various angles of blade installation [9]. It can be easily derived from these graphs the same dependency (7).

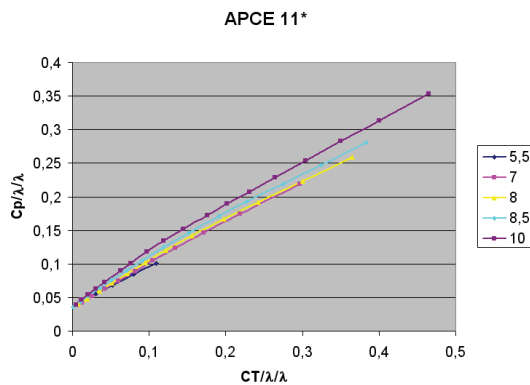


Figure 9: C_p/λ^2 as function of C_T/λ^2 for APCE 11" of various pitches

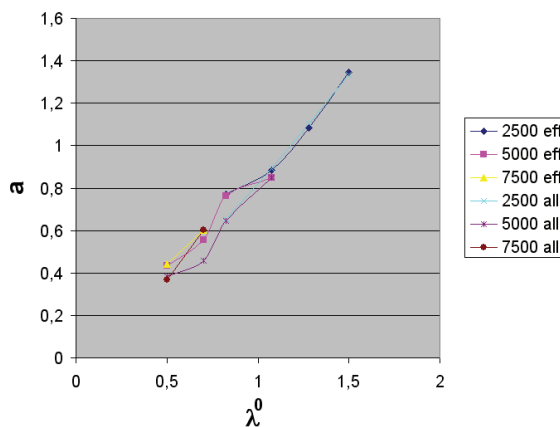


Figure 10: Coefficient a as function of λ^0 for AV-31 propeller for various RPM.

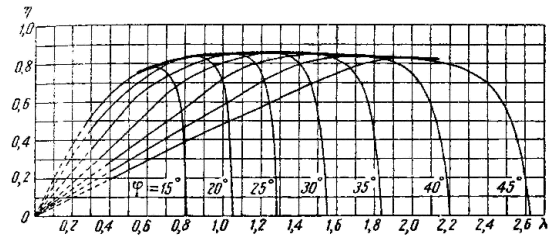


Figure 11: Efficiency as function of λ^0 and $\lambda_{0.75}$ for one of propellers from [9].

Characteristic values of coefficient k obtained from data analysis are 0.5-0.9, and characteristic values of coefficient b are about 0.02-0.08. It is evident that coefficient b can be found from experimental data with less accuracy as coefficient a . So, it is rather difficult to understand the b dependency on λ^0 (or $\lambda_{0.75}$). The coefficient b obtained on the basis of experimental data for AV-31 is shown in Figure 12. One can see the uncertainty in its definition. So, the alternative ways of b definition must be found. One of them will be proposed below. But on the basis of data analyzed one can assume that for rather high angles of installation (and, as consequence, high λ^0) the value of b does not change significantly.

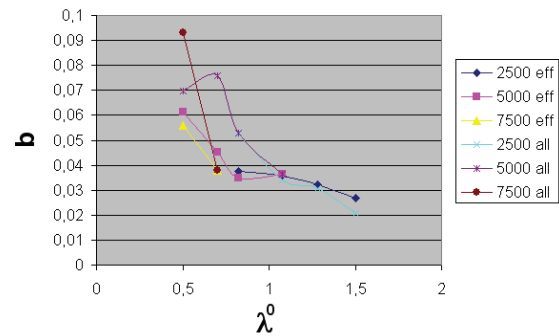


Figure 12: Coefficient b as function of λ^0 for AV-31 propeller for various RPM.

5 ANALYTICAL INVESTIGATION OF MAXIMAL EFFICIENCY

One can obtain a set of useful results on the basis of equations (1)-(7). First of all, it is easy to find the value of advanced ratio λ_{eff} corresponding to the maximum of propeller efficiency η_{max} . Substituting (1) into (5) and setting its derivation with respect to λ to zero one can find

$$(8) \quad \eta_{eff} = \frac{C_{T0}(C_1 a - \sqrt{C_{T0} a b})}{C_1^2 a - C_{T0} b} = \frac{C_{T0} \sqrt{a}}{C_1 \sqrt{a} + \sqrt{C_{T0} b}}.$$

Corresponding value of λ_{max} is

$$(9) \quad \lambda_{max} = \frac{C_{T0}}{C_1 a + 2\sqrt{C_{T0} a b}}.$$

Calculations based on data from [3]-[6] show the good matching of these formulas with experimental data.

Combination of (4), (7) and (8) gives

$$(10) \quad \eta_{eff} = \frac{\sqrt{C_1 k}}{\sqrt{C_1 k} + \sqrt{b}} \lambda^0.$$

So, within our model, the value of η_{eff}/λ^0 remains constant

for all the propellers of the same blade geometry and various angles of blade inclination. This fact is valid not only for the propellers examined. For example, the graph from [9] can lead to the same conclusion (see Figure 11).

Combination of (4), (7) and (9) leads to

$$(11) \quad \eta_{\max} = \frac{C_1}{C_1 k + 2\sqrt{C_1 k b}} = \frac{\sqrt{C_1}}{\sqrt{C_1 k + 2\sqrt{k b}}}.$$

One can see that this expression is explicitly independent of α , but it can depend on α through b and, may be, through C_1 . From Figures 2 and 11 it is seen that the value of $\eta_{\max}$ remains constant within rather wide range of α (0.8–1.5), corresponding to the working range of α for the propellers. On the other hand, this coincidence proves our assumption made above that the value of b does not change significantly with α for rather high α .

The expression for C_T at η_{eff} can be found from (1), (4), (10) as

$$(12) \quad C_{T_{\text{eff}}} = \frac{\sqrt{b}}{\sqrt{C_1 k + \sqrt{b}}} C_1 \eta_{\text{eff}}.$$

It is useful to compare the values of $\eta_{\max}$, η_{eff} and $C_{T_{\text{eff}}}$ through formulas (10), (11), (12) and experimental values.

For Black Widow propeller [3] $a=1.49$, $b=0.0134$, $C_1=0.168$, $k=0.71$, $\alpha=2.1$. For these values $\eta_{\max}=0.83$, experimental value is 0.82–0.83; $C_{T_{\text{eff}}}=0.088$, experimental value is ~ 0.9 ; $\eta_{\text{eff}}=0.75$, experimental value is ~ 0.74 .

For AV-31 propeller with $\alpha_{0.75}=35^\circ$ $a=1.34$, $b=0.025$, $C_1=0.205$, $k=0.89$, $\alpha=1.5$; $\eta_{\max}=0.65$ (0.656 experiment); $C_{T_{\text{eff}}}=0.085$ (0.0987 experiment); $\eta_{\text{eff}}=0.73$ (0.69 experiment).

For APC Thin Electric 11×8.5 ($\alpha_{0.75}=18^\circ$) $a=0.605$, $b=0.052$, $C_1=0.2124$, $k=0.756$, $\alpha=0.8$; $\eta_{\max}=0.619$ (0.622 experiment); $C_{T_{\text{eff}}}=0.061$ (0.0675 experiment); $\eta_{\text{eff}}=0.64$ (0.625 experiment).

So, one can see that the analytical results are rather close to the experiment. So, the conclusion can be made that the model proposed is adequate enough.

We need to know the values of C_1 , k and b . Using the formulas obtained the required coefficients can be defined. Coefficient C_1 can be found if we know the geometry of propeller, but assume that we have no such information. Assume that we have the experimental data of $C_T(\alpha)$ and $\eta(\alpha)$. From the first graph one can obtain C_1 using formula (1). From the second graph with the help of (6) and (7) one can find the values of a and k . One of formulas (10) or (11) gives then the value of b . The second of (10) or (11) can be used to check and correct the results if necessary.

It should be emphasized that only three coefficients C_1 , k and b must be known to describe the characteristics of all the "good" propellers with the same blade geometry for all the "good" operating regimes. So, formulas obtained can be used for the whole powerplant characteristics determination in a wide range of flight conditions.

6 PROPELLER MATCHING FOR MAV

Now we have enough knowledge to proceed to the main task of this investigation. Assume that we have the aircraft

that fly at the velocity V and it produces drag D , and we need to find the propeller with maximal efficiency and with predefined diameter d .

First of all, we must choose the propeller family (with the same blade geometries but different $\alpha_{0.75}$) that provide the maximal efficiency (above we have seen that this value of efficiency is practically constant in rather wide range of α). The next step is to match the angle of blade inclination $\alpha_{0.75}$ (or, in other words, α).

As the drag must be equal to the thrust ($D=T$), then with the help of (10) and (11) one can obtain

$$(13) \quad \eta = \frac{V^2 d^2}{k T} (\sqrt{C_1 k b} + b).$$

This expression gives us enough information for the best choice of propeller. The only thing we must do is to check if this value of η is within the region of maximal η_{eff} . If this value is outside this region one must change the propeller diameter.

Also from the last formula and (10) one can obtain

$$(14) \quad \eta_{\text{eff}} = \frac{V^2 d^2}{k T} \sqrt{C_1 k b}$$

for the propeller chosen.

From this, the required frequency for such propeller is

$$(15) \quad n_{\text{eff}} = \frac{V}{d_{\text{eff}}} = \frac{k T}{V d^3 \sqrt{C_1 k b}}$$

Formulas (13) - (15) lead to the conclusion that lower values of thrust for the fixed diameter and velocity require higher angles of blade installation and, as consequence, lower rotational frequencies.

7 EXPERIMENTAL ERRORS AND INACCURACY INFLUENCE ON MATHEMATICAL MODEL

All the data used are obtained in experiments. So, there can be experimental errors in this data. For example, the results from [5] and [6] slightly differ from each other (one can compare the maximal values of efficiency in both experiments). Unfortunately, authors have not found the values of errors in most of experimental data used in this investigation.

So, on the one hand, the analytical model and corresponding expressions can be not very accurate as they use some experimental data and verified on experimental data. On the other hand, authors used the experimental data from different researchers for different propellers. As the main results are nearly the same one can hope that the mathematical model proposed is rather adequate.

Also at the stage of preliminary design it's better to have less accurate but simpler and quicker methods rather than more accurate but more complex and slower methods. So, anybody who uses the above formulas must keep in mind that this method do not provide the best accuracy and must be used only for the first approximation. In the case of higher accuracy requirements one must use more complex methods but can use the "simple results" as the starting point.

8 FUTURE WORK

Now authors are working on the theory for b and k determination through the propeller geometrical parameters and flow conditions. So, they hope to obtain these expressions in their future work.

9 CONCLUSION

Obtained is the mathematical model of propeller's thrust and power coefficients as functions of geometrical and kinematical parameters for the preliminary MAV design. It needs minimal set of parameters required and can be used for rapid determination of propeller characteristics. This model was verified on a set of experimental data and has shown the adequate accuracy in the range of $Re=30000\div100000$. On the basis of this model the method for the best propeller choosing is proposed. Formulas obtained can be used for the whole powerplant characteristics determination in a wide range of flight conditions.

Investigation conducted also allows to make the following recommendations concerning the propeller design.

First of all, for low- Re propellers thin blades are preferable as they give higher value of efficiency.

Next, for the best efficiency the angle of blade installation is proportional to flight velocity squared, to propeller diameter squared and inversely proportional to thrust.

It was also found that model aircraft propellers are not good enough as they does not provide maximal efficiency because of nonoptimal shape of the blade (blade twist).

Finally, it should be noted that additional special experiments are required for more thorough investigation of Re and θ influence on C_1 , k , b .

REFERENCES

- [1] Ya. Flaxman, Al. Lipin, S. Ostroukhov, M. Ustinov, A. Shustov, A. Kornushenko, S. Serokhvostov. Experimental Investigations Of Propellers At Low Reynolds Numbers. In Proc. of 3rd US-European Competition and Workshop on Micro Air Vehicle Systems (MAV07), Toulouse, France, 2007
- [2] A.B. Липин, С.П. Остроухов, С.В. Серохвостов, М.В. Устинов, А.В. Шустов, Я.Ш. Флакман. Экспериментальное исследование зависимости характеристик воздушного винта от числа Рейнольдса. *Ученые записки ЦАГИ*, XXXVIII (3-4), 102-110, 2007.
- [3] J.M. Grasmeyer, M.T. Keennon. Development of the Black Widow Micro Air Vehicle. AIAA Paper No. 2001-0127, 2001
- [4] J.B.Brandt, M.S.Selig. Propeller performance data at low Reynolds numbers. In 49th AIAA Aerospace Sciences Meeting, Orlando, FL, 2011, AIAA 2011-1255
- [5] J.B.Brandt, M.S.Selig. Small-Scale Propeller Performance at Low Speeds – Online Database. <http://www.ae.illinois.edu/m-selig/props/propDB.html>, 2010
- [6] M.P. Merchant, L.S. Miller. Propeller Performance Measurement for Low Reynolds Number UAV Applications. In 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2006
- [7] PropCalc program. <http://www.drivecalc.de/PropCalc/index.html>
- [8] <http://www.allstar.fiu.edu/aero/BA-Background.htm>
- [9] Александров В.Л. Воздушные винты. М., Оборонгиз, 1951

TRANSIENT ANALYSIS of NYLON 6/6 for a THIN SHELL STRUCTURE by FEM

H.R. Montazer Hojjat
Ruhr Universität Bochum, Bochum, Germany

ABSTRACT

In the design of Micro Aerial Vehicles, decreasing the size is one of the most common challenging aims and to approach this aim, the weight of the whole aircraft shall decrease and structure is the second part which has the most weight [13].

In this paper, the use of a shell structure made from Nylon 6/6 is studied. Nylon was chosen because of its good mechanical properties and low mass density which is of great importance. Also due to the low aspect ratio, the shell structure is used without any kind of ribs or spars to let the user put the instruments easily inside the structure which facilitates assembly and maintenance.

Then the aerodynamic forces are calculated by CFD, Fluent, at Reynolds number 300000 at 5 AOA in a laminar flow. Afterwards the nodal results of Fluent after a nodal interpolation by MATLAB are imported in ANSYS for a full transient analysis. The model in ANSYS is a simple thin shell structure with low aspect ratio, AR=2, with 0.29 mm thickness and volume of 37.2 cubic centimeters, which is calculated in SOLIDWORKS.

Finally Nylon 6/6 is modeled as a nonlinear visco-elastic material and the model is assumed to have large deformations, and the constitutive model is based on the Prony series, which consists of 5 Maxwell elements parallel to a spring. The dynamic bulk and shear modulus are then defined in ANSYS.

Finally, the weight of structure is reduced to 42.40 grams, for the half of aircraft which is equal to about 16.74% of the whole aircraft's weight. Simulation results are presented to prove the efficiency of this material for this structure.

1 INTRODUCTION

Micro Air Vehicles (MAVs) are widely used in civilian and military fields for their light and agile characteristics [1]. Recently some researchers have focused on the development of MAVs and one of the biggest challenges is the reduction of size and weight of these aircrafts [8].

Some efforts have been done to decrease the weight and then the size, like using some light batteries or electrical engines; however, in this paper it is tried to decrease the weight by using Nylon 6/6, which is a light visco-elastic polymer and also has appropriate mechanical properties [2].

Also employing a thin shell structure is suggested. This structure is assumed to be like a sea shell and therefore user can easily open it, assembles the wing and its components which makes maintenance easy as well. Therefore, if one of the components like servos or engines does not work properly, there is no need to destroy the structure. So any change can happen in some seconds.

The procedure is done by the FEM software, ANSYS and Fluent, and the mesh generation of CFD part is done in Gambit and it is tried to have mapped meshing in all the

procedure. So this paper analyzes the effects of aerodynamic forces on a simple shell structure wing as a Fluid Structure Interaction problem. Also Nylon 6/6 is modeled with Prony series, 5 Maxwell elements parallel to a spring, to consider the visco-elastic behavior of Nylon 6/6 in the analysis during the time [3].

The root of the model is 30 cm and the span is supposed to be 50 cm, "Figure 1."

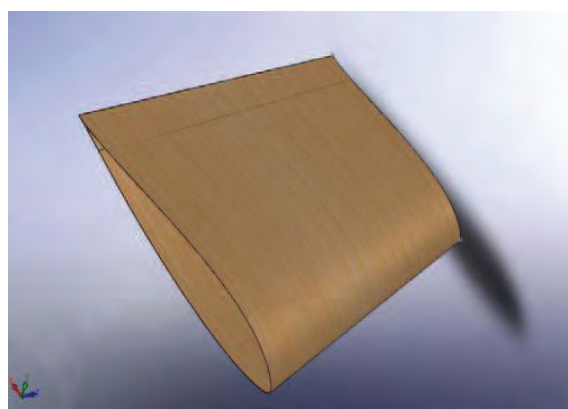


Figure 1: Half of the wing modeled in SOLIDWORKS

Also for calculating the aerodynamic forces the model is located in a laminar flow at 5 AOA with $Re=300000$ [4].

The airfoil which is used for this wing is named MH-91 and is a thick proper airfoil with high thickness for use in MAVs [9].

2 Nylon 6/6

Nylon is the common name of linear polyamides which all have in common the carbonamide group $-CO-NH-$ recurring in a chain of methylene groups. In Nylon 6/6, $-[HN-(CH_2)_6-NHCO-(CH_2)_4-CO]_n-$, the most common nylon, which is a polycondensation product of hexamethylenediamine $H_2N-(CH_2)_6-NH_2$ and adipic acid $HOOC-(CH_2)_4-COOH$ the first digit is derived from the number of carbon atoms in the diamine and the second digit from the number of carbon atoms in the dibasic acid [6].

For this analysis, the dynamic young modulus of Nylon 6/6 is obtained from literature [7] and then by using MATLAB, a curve is fitted on this diagram at $T=25^\circ C$ as shown below, "Figure 2", and then the exact data of E with respect to time are obtained for the final structural analysis. Finally by assuming that the Poisson ratio is constant in this analysis, $\nu=0.4$ [11], the shear and bulk modulus are obtained by the following equations with respect to time.

- (1) $\mu = E/2(1+\nu)$
- (2) $K = E/3(1-2\nu)$

Hamidreza.Montazerhojjat@RUB.de

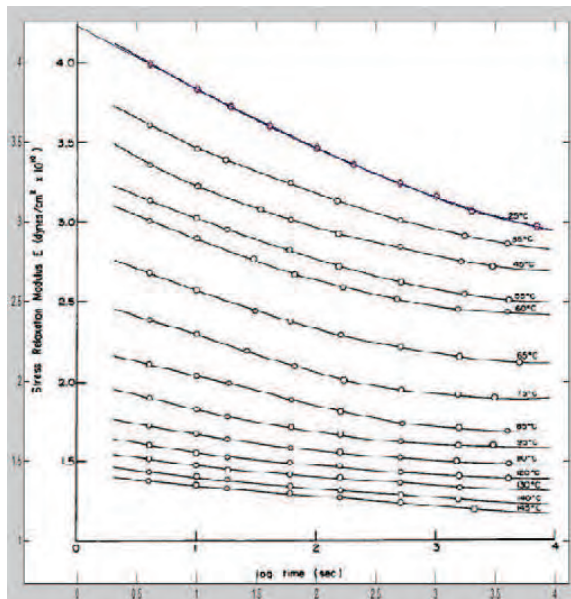


Figure 2: Dynamic young modulus with respect to time [7].

In the structure analysis by ANSYS, two curves are fitted on these data, "Figure 8".

3 CFD ANALYSIS

3.1 Modeling and Mesh Generation in Gambit

In this part, the wing is modeled in Gambit and then it is reduced from three other volumes, one half cylinder and two cubes. These volumes are presented as the flow of air around the wing. Also to have a mapped meshing and to prevent some elements with sharp edges, the last node of airfoil on the trailing edge is divided to two separated key points [5].

Also this wind tunnel is assumed to have 22.4c width and 9c height and the radius of half circle is 8c, where c is the chord of the root airfoil [5].

After defining successive ratio on the edges of geometry, the areas are meshed by Mapped Quad elements and afterwards the volumes are meshed by Mapped Hex elements, "Figures 3, 4".

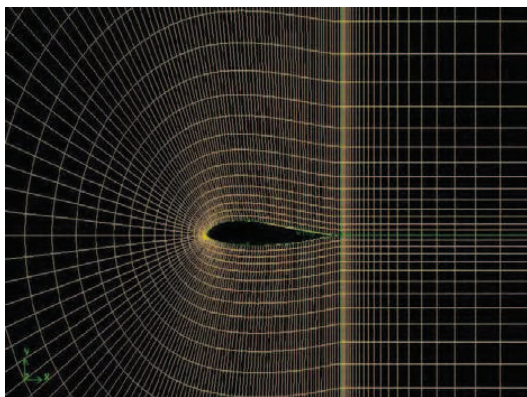


Figure 3: Elements around the wing.

Also as shown in "Figure 4", four boundary conditions, three types, are defined in this problem.

The outer surface of the half cylinder and the cubes, whose norms are parallel to the norm of wings' surface, are defined as far fields: 1(velocity inlet), 2 (velocity inlet) and 3(pressure outlet), respectively. Also all the side walls are defined as symmetry wall to prevent distortion along the edges [5].

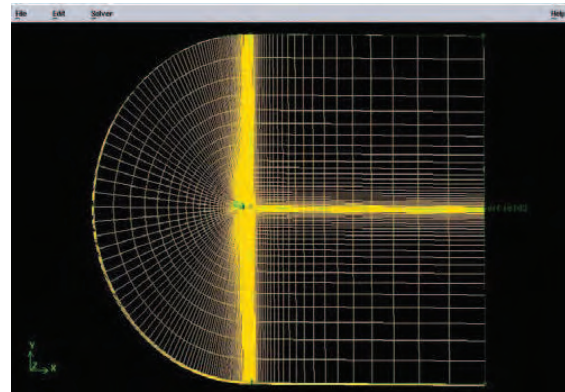


Figure 4: Meshed geometry and far field conditions.

3.2 Solution in Fluent

As already mentioned, the wing is located at 5 degree angle of attack in a laminar flow and the Reynolds number is assumed to be 300000 [4] and this value would lead us to calculate the V_∞ by the following equation. Also the Mach number is then equal to 0.0319.

$$(3) \quad V_\infty = Re \cdot \mu / \rho c = 10.95 \text{ m/s}$$

For the CFD solution in Fluent, due to the low Mach number, the energy equation is turned off and the model is solved in steady state with a pressure based and implicit solver.

The result is converged with lift coefficient of the wing equal to 0.54122. So the total lift produced by this wing at these conditions could be calculated as following.

$$(4) \quad L = 1/2 \rho V_\infty^2 SC_L = 4.9684 \text{ N}$$

In "Figure 6" the static pressure around the wing, as the output of Fluent, is plotted and the maximum static pressure is about 69.8 Pascal.

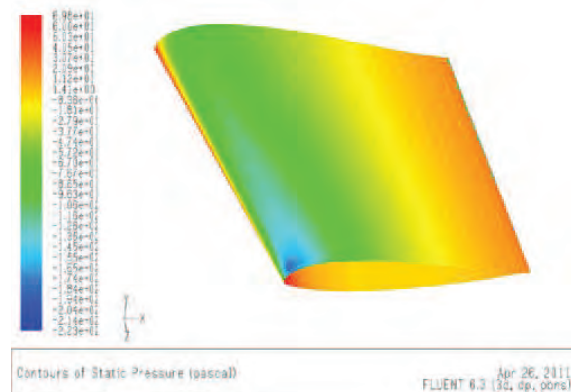


Figure 5: Contours of static pressure around the wing

Finally the nodal forces and pressure are written on a file to use in other software.

4 INTERPOLATION OF NODAL RESULTS

After calculating nodal values in CFD, Fluent, it is desired to import these values on the nodes or elements in structural software but the problem which exists is that the nodes in CFD analysis are different in ANSYS.

The first way to solve this dissimilarity is a mesh refinement. Although, mesh refinement increases the accuracy and the procedure time there is another problem in this analysis which makes the interpolation necessary.

The element which is chosen to capture the nonlinear behavior of shell structure is a 8-node element with 4 mid nodes, "Figure 7", and ANSYS could not use this element, when the results are exported from Fluent to ANSYS.

On the other hand, each node on the trailing edge is divided to two separated nodes, to have a better mesh generation in CFD, but in structural analysis there is no tendency to increase the thickness of trailing edge. Hence to use this powerful element, it is decided to do a simple linear interpolation on the nodal results of CFD to find the nodal forces which should be imported on structural nodes. To do the interpolation, a code is written in MATLAB and then the ASCII code is read and the interpolation results are exported as the ANSYS format.

5 STRUCTURAL ANALYSIS

5.1 Element and Mesh Generation

In this part, the wing is modeled in ANSYS and the first step is choosing an appropriate element to capture the nonlinear behavior of thin shell structure and the chosen element should support visco-elastic material behavior and large deformations as well. For this purpose the 8-node shell element, SHELL 281, is selected.

To have more accuracy it is suggested not to use triangular shapes [10]; therefore, in this paper, meshing is done by mapped quadrilateral elements, "Figure 6".

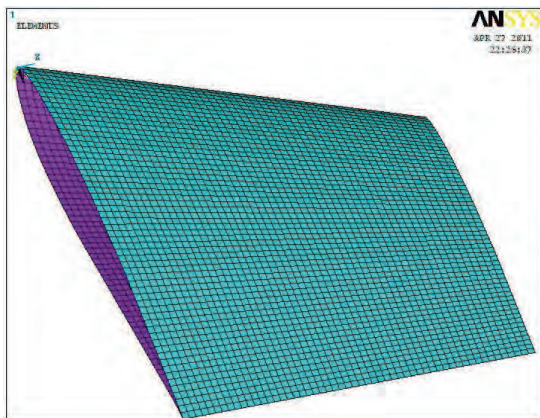


Figure 6: Modeled wing in ANSYS with quadrilateral elements

SHELL 281, "Figure 7", is suitable for analyzing thin to moderately-thick shell structures. It is an 8-node element with six degrees of freedom at each node; translations in the

x, y, and z axes, and rotations about the x, y, and z-axes. (When using the membrane option, the element has translational degrees of freedom only.) SHELL 281 is well-suited for linear, large rotation, and/or large strain nonlinear applications. Change in shell thickness is accounted for in nonlinear analyses [10].

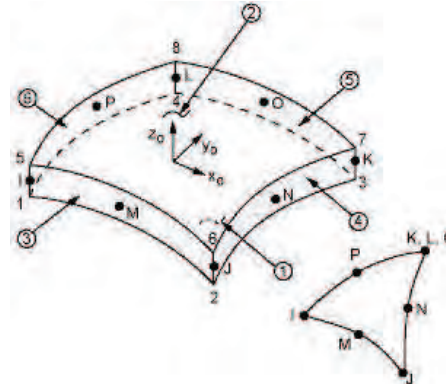


Figure 7: mid nodes in shell 281 [10]

5.2 Material Modeling in ANSYS

For the constitutive model of Nylon 6/6, due to its visco-elastic properties, Prony series is used. Here five Maxwell elements parallel to a spring are modeled by curve fitting in ANSYS. For this approach the Bulk modulus and shear modulus with respect to time are imported in ANSYS and two 5th order curves are fitted on these data, "Figure 8".

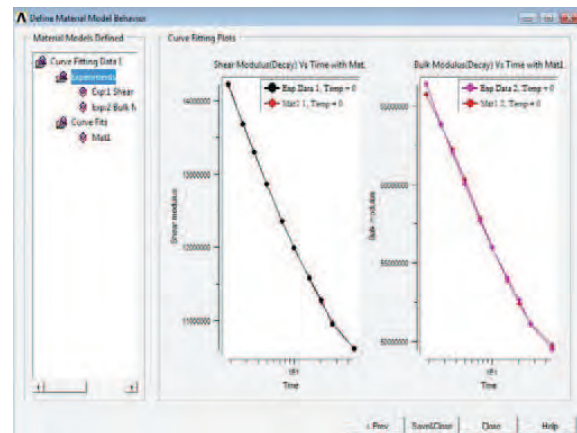


Figure 8: Bulk and Shear modulus with respect to time

In the last part, the solution is started with the "full dynamic" solution in 2 steps. It is assumed that at $t=0$ the forces are zero and they increase linearly until they reach nodal results, which are interpolated by MATLAB, and at time $t=4$ sec, the forces reach their maximum value and afterwards remain constant.

6 RESULTS

Finally solution is done and the desired results are plotted. As shown in "Figure 9", the stress is reduced with respect to time during the second step of loading. Also in this analysis all the values are converted to cm and therefore the final results shown in figures are in cm.

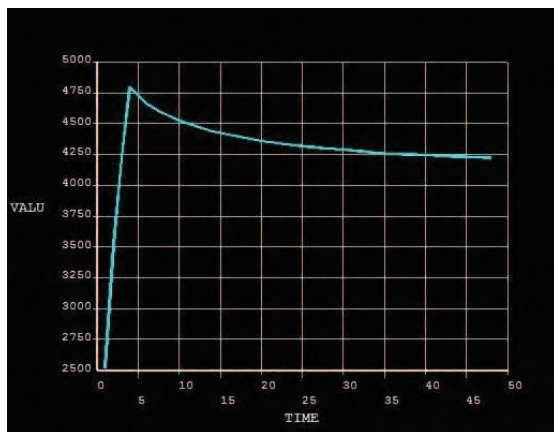


Figure 9: Stress relaxation for a point on the upper surface of wing

At time $t=48$ sec, the maximum value of von misses stress is 1.5266 MPa, and the maximum displacement in Y direction at this time is 0.1412, "Figure 10".

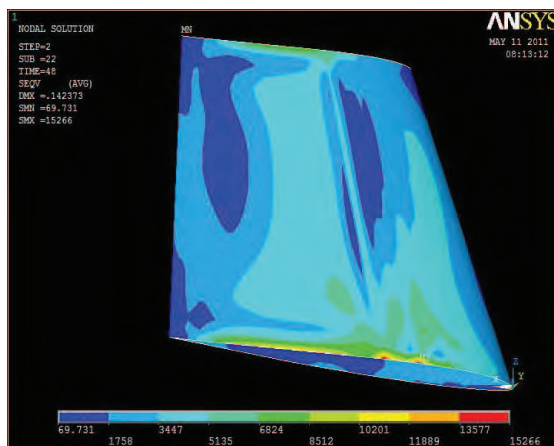


Figure 10: Von misses stress at $t=48$ sec.

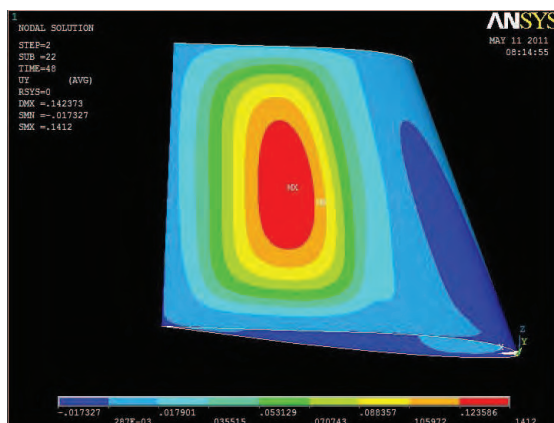


Figure 11: Maximum displacement in Y direction at $t=48$ sec.

Also the maximum value of the von misses total mechanical strain is 0.00057. This value is very small in comparison with yield strain of the material (5% at dry conditions) [11]. On the other hand, the maximum von misses stress is much smaller than yield/break stress of PA 66 (Nylon 6/6), $\sigma_y > 54\text{MPa}$ in dry/humid conditions [11].

7 Weight

After calculating forces and stresses, for this thin shell structure with 0.29mm thickness, the weight is calculated with the density of Nylon 6/6 ($\rho=1.14\text{ gr/cm}^3$). The volume is calculated with SOLIDWORKS, $V=37.2$ cubic centimeters and then it is clear that the weight of structure would be 42.4 grams for half of the wing.

If the total weight of aircraft is considered equal to the lift, then the weight of structure divided by the total weight would be 16.74%.

This value is estimated in other MAVs about 17% [12],[13]. Thus by this shell structure the weight of aircraft is decreased.

8 CONCLUSION

In this paper we tried to show the high mechanical properties of Nylon 6/6 on a simple thin shell structure for MAVs. Also this structure omits some difficulties during the assembly, maintenance and after crashes.

Finally, the weight of the aircraft is reduced to 42.4 grams. Considering that one wing accounts for about 16% of the whole aircraft's weight, this means a significant reduction in weight however, the wing has still very good strength and reliability for Micro Aerial Vehicles.

ACKNOWLEDGEMENT

I am heartily thankful to Prof. Dr. Ing. Holger Steeb, chair of Continuum Mechanics (Ruhr University Bochum), for his kind helps throughout this research and I would also like to thank Dr. Fritz Menzer for his help with the curve-fitting.

REFERENCES

- [1] J. Luo, Z. Jiang, W.M. Cheng, Z.B. Gong, Y.Z. Deng and Q.C. Liang: *Journal of Shanghai University (Natural Science)*, Vol.7-4 (2001), pp.293-296.
- [2] Takayuki Murayama, John H. Dumbleton, Malcolm L. Williams. The viscoelastic properties of oriented nylon 66 fibers. Part III: Stress relaxation and dynamic mechanical properties. *Chemstrand Research Center, Inc., Durham, North Carolina, Journal of Macromolecular Science, Part B*
- [3] Tzikang Chen, Determining a Prony Series for a Viscoelastic Material From time Varying Strain Data. NASA/TM-2000-210123 ARL-TR-2206
- [4] X.Q. Zhang, L. Tian. Three-dimensional Simulation of Micro Air Vehicles with Low-Aspect-Ratio Wings. *Key Engineering Materials* Vol. 339 (2007) pp 377-381 Trans Tech Publications, Switzerland
- [5] Nathan Logsdon, A procedure for numerically analyzing airfoils and wing sections. *University of Missouri, Columbia*.
- [6] Gerhard Hopf, Nylon 12-Huels in comparison to other nylons, Technical representative of chemische werke HUELS AG West-Germany.
- [7] Takayuki Murayama, John H. Dumbleton, and Malcolm L. Williams, The Viscoelastic Properties of Oriented Nylon 66 Fibers. Part 111: Stress Relaxation and Dynamic Mechanical Properties. *Chemstrand Research Center, Znc, Durham, North Carolina*
- [8] <http://www.compositesworld.com/articles/composites-enable-micro-air-vehicle> (May 01, 2011)
- [9] <http://mh-aerotoools.de/> (May 01, 2011)
- [10] Release 11.0 documentation for ANSYS, Element library, shell 281. ANSYS help
- [11] BASF Campus Data bank: <http://campusi.plasticsportal.net/matdb/matdb.php> (May 01, 2011)
- [12] Francis Barnhart, Michael Cuipa, Daniel Stefanik, Zachary Swick. Micro-Aerial Vehicle Design with Low Reynolds Number Airfoils. 7th March 2004.
- [13] Joel M. Grasmeyer and Matthew T. Keennon. Development of the Black Widow Micro Air Vehicle. *AeroVironment, Inc. AIAA-2001-0127*

Inverse Dynamics Approach to Adaptive Damage-Tolerant Control for Unmanned Aerial Vehicles

Alexey Kondratiev and Yury Tiumentsev¹

Moscow Aviation Institute (MAI),

Flight Dynamics and Control Department,

Volokolamskoye Shosse 4, 125993, GSP-3, Moscow, Russia

ABSTRACT

A control problem is discussed for unmanned aerial vehicle (UAV) as applies to its short-period longitudinal motion. The problem is formulated taking into account various uncertainty factors such as imperfect knowledge about UAV parameters and characteristics as well as environment exposure. One more class of the uncertainty factors includes failures of UAV systems and its structural damages. A preliminary step is needed to synthesize adaptive damage-tolerant control systems. This step consists in plant identification using artificial neural network (ANN) techniques. Next step is a synthesis of appropriate neural controller. An adaptive control scheme based on inverse dynamics problem (IDP) approach is used to achieve control goals for the conditions mentioned above. The scheme is implemented basing on ANN tools. Simulation is carried out to confirm efficiency of the adaptive damage-tolerant neural control. Appropriate computer experiment results are presented and discussed to demonstrate features of the proposed approach.

1 INTRODUCTION

We need to provide a motion control for modern and advanced UAVs under such conditions as considerable various uncertainties in values of their parameters and characteristics, flight regimes and environment exposure. Besides numerous failure conditions can emerge during UAV flight including equipment failures and structure damages. These conditions should be counteracted by means of reconfiguration for control system and control surfaces of the UAV.

Therefore the UAV faces each time circumstances that can vary considerably and unpredictably. The UAV control system must be able to conform efficiently to these variations by means of on-line changes in parameters and/or structure of control laws used to manage UAV behavior. A way based on theory of adaptive control allows us to satisfy these requirements [1]–[9].

An approach basing on neural network simulation and control is one of the most effective tools to implement adaptive systems [7], [8]. An important part of the implementation process for the proposed approach is generation of artificial neural network based model (ANN-model) for the UAV which is interpreted as a plant. An

example of ANN-model generation is presented in subsequent sections as applies to simulation of longitudinal short-period motion for a mini-UAV.

2 GENERAL ADAPTIVE CONTROL SCHEME BASED ON INVERSE DYNAMICS APPROACH

There are problems which consist in generation of control laws providing realization some prescribed motion for a plant [10], [11]. We designate these problems as inverse dynamics problems (IDP). As applies to the IDP problem we need to implement a motion of the dynamic system which fits to some desired or reference motion

$$(1) \quad x_m(t), \dot{x}_m(t), t \geq 0$$

with prescribed accuracy. The desired motion in this case can be defined by means of a reference model (RM). In other words we need to satisfy relationship

$$e = \|x(t) - x_m(t)\| + \lambda \|\dot{x}(t) - \dot{x}_m(t)\| \leq \varepsilon,$$

where ε is the prescribed accuracy for tracking of the reference model output. In the perfect case we have $e = 0$ therefore

$$(2) \quad \dot{x}(t) = \dot{x}_m(t).$$

This relationship is satisfied only if following conditions are fulfilled:

- mathematical model of the plant fits precisely to the plant itself;
- initial conditions for the plant model and reference model coincide precisely;
- no disturbances affect on the plant.

However these conditions are not satisfied usually for real world application problems because of uncertainties in the DS behavior caused by external factors as well as approximate nature of the plant model.

To prevent a rise of the tracking error in time we need to add in Eq. (1) an auxiliary member providing elimination of the tracking error:

$$(3) \quad \dot{x} = \dot{x}_m + K(x_m - x).$$

We have in such case, that

$$(4) \quad \dot{e} = -Ke.$$

If source equations of motion for the plant had the form

$$(5) \quad \dot{x} = f(x, u)$$

then we can rewrite Equation 2 in the form

$$(6) \quad \dot{x} = \dot{x}_m + K(x_m - x) = f(x, u(x, x_m, \dot{x}_m)).$$

We cannot derive analytically the required control function (control law) $u(x, x_m, \dot{x}_m)$ from Equation 6. Therefore we

¹ E-mail: tium@mai.ru

have to build an approximate solution which we can obtain using a feedforward multilayer neural network named here as neurocontroller (NC) and learned with the error backpropagation algorithm as a standard tool for such kind of the networks.

3 SYNTHESIS OF A NEURAL NETWORK BASED MOTION MODEL FOR UAV

A plant model is needed in many of adaptive control schemes. Deriving of the plant model basing on some experimental data corresponds to the classical identification problem for dynamic systems [12]. We know from experience that using of ANN-based techniques and tools is very efficient way to solve this identification problem with regard to nonlinear systems [13]–[15]. Neural network simulation allows us to build rather accurate and computationally effective models of dynamic systems.

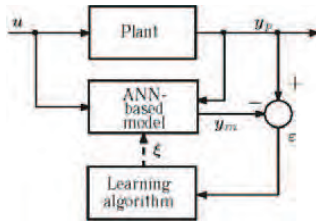


Figure 1: Neural network based scheme for plant identification.

Here u is control, y_p is plant output, y_m is reference model output; ε is divergence between outputs of plant and ANN based model; ξ is corrective action.

Computational efficiency roots of ANN-based models is based on the following fact: an artificial neural network is algorithmically universal mathematical model [16], [17] which allows us to represent with arbitrary accuracy any nonlinear mapping $\varphi: R^n \rightarrow R^m$. In other words we can represent with arbitrary accuracy any nonlinear relationship between n -dimensional input vector and m -dimensional output vector.

A synthesis of ANN-based model for controlled nonlinear plant motion is interpreted below as generation of a neural network approximation for some source mathematical model of UAV motion. This source model is formulated frequently as a system of ordinary differential equations.

General scheme of neural network plant identification is presented on Figure 1.

Squared difference between plant output y_p and ANN-based model output y_m both under control signal u is used as an error signal ε guiding a learning process for the ANN-based model. A trained ANN-based model realizes recurrent-type computational scheme using output signal $\hat{y}$ and control signal u values for instant time t_i to compute output signal $\hat{y}$ value for instant time t_{i+1} .

The NARX (Nonlinear AutoRegressive network with eXogeneous inputs) model was chosen to represent the dynamic plant because it corresponds well with UAV control problem. This model is a recurrent dynamic layered

neural network with feedbacks between layers and with TDL (Time Delay Line) units before its inputs.

Validation of the ANN-based model is carried out with regard to angular longitudinal motion of UAV described with a mathematical model which is rather common for aircraft flight dynamics [18]:

$$\dot{\alpha} = q - \frac{\bar{q}S}{mV} C_L(\alpha, \varphi) + \frac{g}{V},$$

$$\dot{q} = \frac{\bar{q}Sc}{I_{yy}} C_m(\alpha, \varphi, q),$$

$$T^2 \ddot{\varphi} = -2T\zeta\dot{\varphi} - \varphi + \varphi_{act},$$

where α is angle of attack, deg; q is pitch angular velocity, deg/sec; φ is deflection angle of elevator or elevons, deg; C_L is lift coefficient; C_m is pitching moment coefficient; m is mass of UAV, kg; V is airspeed, m/sec; $\bar{q} = \rho V^2 / 2$ is airplane dynamic pressure; ρ is mass air density, kg/m³; g is acceleration of gravity, m/sec²; S is wing area of UAV, m²; c is mean aerodynamic chord, m; I_{yy} is pitching-moment inertia, kg·m²; dimensionless coefficients C_L and C_m are nonlinear functions with respect to their arguments; T , ζ are time constant and relative damping factor for actuator, φ_{act} is command signal value for elevator actuator limited in the $\pm 20^\circ$ range. Variables α , q , φ and $\dot{\varphi}$ in the model are plant states and variable φ_{act} is plant control.

This ANN-based model were built and described in [18] as applies to the considered UAV control problem. Validation of the model is carried out for X-04 mini-UAV [18] with airborne weight 4.2 kg.

It was suggested some special way to generate training samplings intended to learn considered ANN-based UAV model. This way relates to using of very aggressive actions (often and strong random variations) which are carried out with elevator as longitudinal motion control surface to obtain command signal φ_{act} for the relevant actuator. The purpose of such approach to command signal generation is to ensure diversity of simulated system states as large as possible and to cover the system state space as uniformly and tightly as possible. Besides it is necessary to provide variety of differences between states in adjacent instant times as large as possible to represent dynamics of the simulated system in the ANN-based model with maximum adequacy.

The similar approach under similar reasons is used below to generate training samplings for neurocontroller in the IDP-based scheme. It also was used for two another adaptive control schemes (MRAC and MPC) considered in [18].

Validation results presented on Figure 2 were obtained for closed-loop ANN-based UAV model using simulation. These data demonstrate the model efficiency as applies to the UAV angle of attack tracking problem for dynamically specified reference values of this angle. The results show us

rather high simulation accuracy of the suggested approach. Simulation error which equals a difference between UAV state and ANN-model output does not exceed 0.3 deg for all experimentally studied conditions.

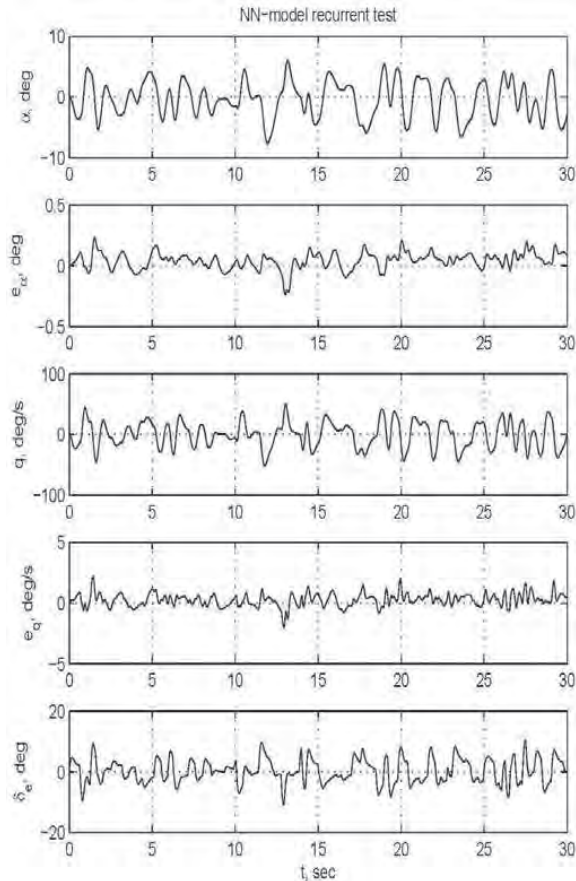


Figure 2: Synthesis of ANN-based plant model for X-04 UAV in respect to flight regime with indicated airspeed $V_i = 70$ km/h and altitude $H = 10$ m. Here α is angle of attack, deg; e_α is difference between angle of attack values for the plant and ANN-based model, deg; q is pitch angular velocity, deg/sec; e_q is angle of elevator deflection, deg; e_q is difference between pitch angular velocities for the plant and ANN-based model, deg/sec; t is time, sec.

4 SYNTHESIS OF INVERSE DYNAMICS BASED DAMAGE TOLERANT ADAPTIVE CONTROL FOR UAV

An application of traditional control theory requires us to know plant mathematical model as well as values of plant and environment parameters and characteristics. These requirements can be satisfied not always in practice. Besides values of plant parameters and characteristics can change in the course of its operation. Traditional control theory methods lead often to unacceptable results in that case.

Because of such situation a demand arises to build control systems which do not require full a priori knowledge about the plant and its environment. These systems must afford to adjust themselves to changing conditions including plant

and environment properties. Adaptive systems satisfy such demand. They use current available information not only to generate control actions just as it occurs in traditional control systems but to correct a control law.

A general structure of adaptive system can be represented as it is shown on Figure 3. As we can see from Fig. 6, corrective action $\xi(t)$ for the controller is generated by means of some adaptation mechanism which uses control signal $u(t)$, plant output signal $y(t)$ and some additional "external" information $\psi(\lambda)$, $\lambda \in \Lambda$ to provide the corrective action. The additional information can be necessary to take into account some data enter into the UAV motion model as parameters, e.g. airspeed and altitude in model of UAV angular motion.

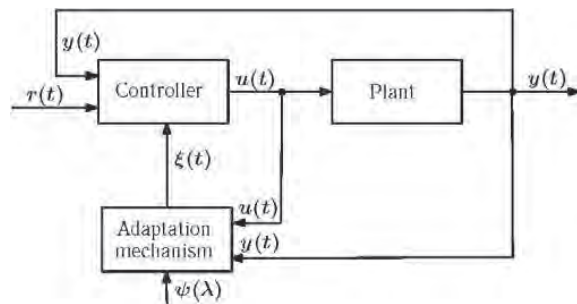


Figure 3: A controlled system scheme with adjustable control law: Here $r(t)$ is reference signal; $u(t)$ is control; $y(t)$ is plant output; $\xi(t)$ is corrective action for controller; $\psi(\lambda)$, $\lambda \in \Lambda$ is some additional information we need to take into account while generating control signal value, for example, velocity and altitude values for UAV as applies to angular motion control problem.

There are numerous adaptive control schemes including ANN-based ones [1]–[8]. The MRAC (Model Reference Adaptive Control) and MPC (Model Predictive Control) schemes belong to the most frequently used ones (see Figures 4 and 5 respectively).

A controller in the MRAC scheme can be implemented basing on an artificial neural network. A learning process for the ANN-based controller named below as neurocontroller is accomplished to satisfy proximity condition for motions realized with the reference model and the plant under synthesized control law. The reference model shows an idea of control system designer about "good" or appropriate behavior of the plant which need to be tracked with the neurocontroller.

The MPC scheme exploits a plant model used to predict future behavior of the plant together with some optimization algorithm to choose appropriate control actions providing best values of predicted characteristics for the considered system.

We have considered MRAC and MPC schemes as applies to control UAV longitudinal short-period motion in our previous paper [18]. One more scheme is discussed in this article. This scheme is based on the inverse dynamics problem (IDP) approach. It is used to stabilize a prescribed value for UAV angle of attack which transmits from the pitch control channel.

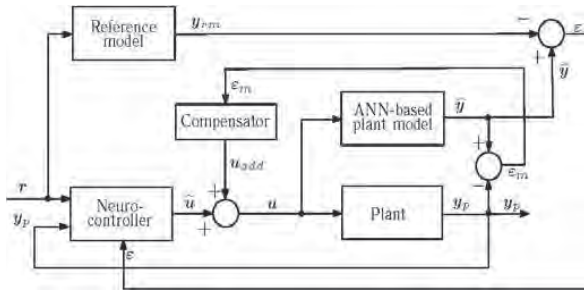


Figure 4. General scheme for a model reference adaptive control. Here $r(t)$ is a reference signal; $y_p(t)$ is a plant output; $\hat{y}(t)$ is an output of the ANN-model; $y_m(t)$ is a reference model output; $u(t)$ control signal generating with neurocontroller; $u_{add}(t)$ is additional control signal generated with a compensator; $u(t)$ is combined control input acting on the plant; $\varepsilon(t) = y_p(t) - y_m(t)$ is a difference between outputs of plant and reference model.

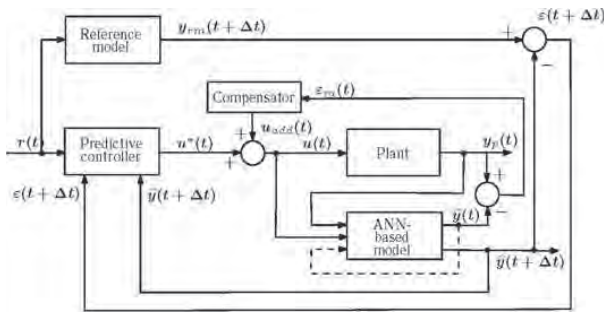


Figure 5. General scheme for a model predictive control. Here $r(t)$ is a reference signal; $y_p(t)$ is a plant output; $\hat{y}(t)$ is an output of the ANN-model; $y_m(t)$ is a reference model output; $u(t)$ control signal generating with predictive controller based on optimization algorithm; $u_{add}(t)$ is additional control signal generated with a compensator; $u(t)$ is combined control input acting on the plant; $\varepsilon(t) = y_p(t) - y_m(t)$ is a difference between outputs of plant and reference model.

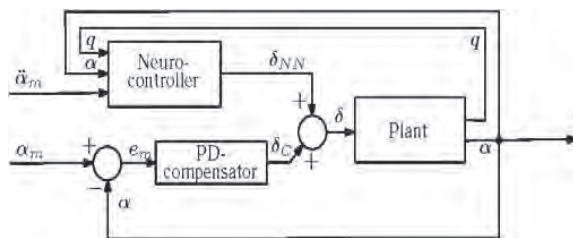


Figure 6: Structure of the IDP-controlled system. Here α and α_m is angle of attack from UAV and reference model, q is pitch angular velocity, e_m is divergence between outputs of plant and reference model; δ , δ_c and δ_{nn} are control signals.

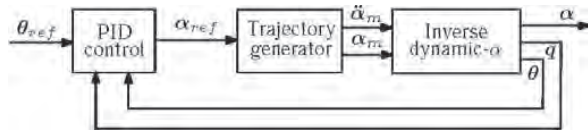


Figure 7: Neural network based scheme for plant identification.

Here θ_{ref} and α_{ref} are required values for pitch angle and angle

of attack; α_m is angle of attack from reference model; q is pitch angular velocity. 'Trajectory generator' here is the reference model and 'Inverse dynamic- α ' is the IDP-based neurocontroller.

A flowchart for the IDP-controlled system is shown on Figure 6. The neurocontroller generates here a control signal to track precisely the reference trajectory generated by the reference model and the dynamic PD-compensator $K(x_m - x)$ adjust the control signal to decrease a value of difference between actual trajectory $x, \dot{x}$ and reference one $x_m, \dot{x}_m$, i.e. a value of the tracking error.

The neurocontroller shown on Figure 6 is a part of the UAV pitch control channel presented on Figure 7. Input of the pitch channel is prescribed value for UAV pitch angle transmitted from the trajectory control channel.

The controller in the IDP scheme is implemented basing on an artificial neural network. A learning process for the controller named here as neurocontroller is accomplished to satisfy proximity condition for motions realized with the reference model and the plant under synthesized control law. The reference model shows an idea of control system designer about "good" or appropriate behavior of the plant which need to be tracked with the neurocontroller.

The reference model can be defined in a variety of ways. Within this article the reference model is built basing on an oscillatory link with rather high damping ratio in aggregate with an aperiodic link interpreted as a prefilter. It is accepted that the reference model defined as

$$(7) \quad W_\alpha = \frac{\omega_{RM}^2}{((1/\omega_{PF})p + 1)(p^2 + 2\omega_{RM}\zeta_{RM}p + \omega_{RM}^2)}$$

if the UAV motion is described by means of equations mentioned above. In model (7) parameter values are specified as $\omega_{RM} = 51/\text{sec}$, $\omega_{PF} = 80$, $\zeta_{RM} = 0.8$ for mini-UAV X-04. Here ω_{RM} and ω_{PF} are natural frequencies of the oscillatory and aperiodic links; ζ_{RM} is damping ratio for the oscillatory link.

The ANN-based plant model obtained above is used to implement learning process for the neurocontroller. The adjustment purpose specified for the neurocontroller consists in minimization of the error $y_{rm} - \hat{y}$. In other words it is needed to bring the plant under neurocontroller behavior nearer as possible to the reference model behavior. If the ANN-based model has appropriate accuracy then the neurocontroller will minimize "genuine" error $y_{rm} - y$ too, i.e. it will try to reduce a difference between behavior of the ANN-based plant model and the real plant under the same neurocontroller actions.

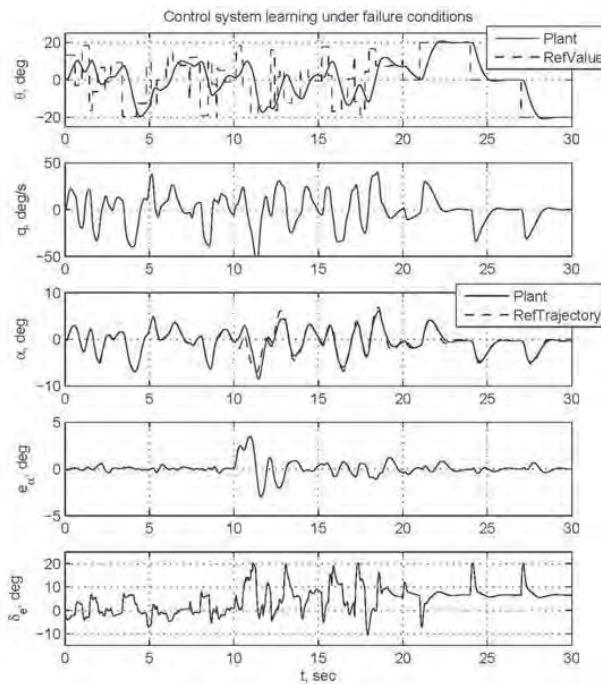


Figure 8: Simulation results for the IDP-based system with adaptation in the angle of attack stabilization loop.

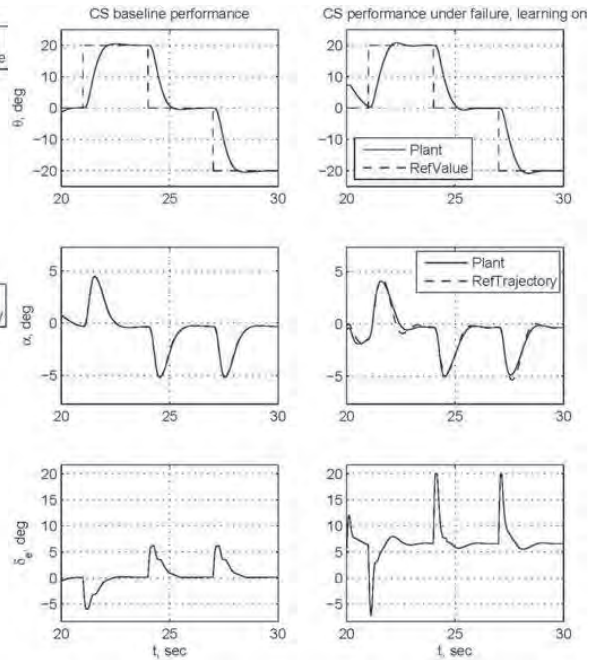


Figure 10: Comparison of control quality for pitch angle before and after damage with adaptation in the angle of attack stabilization loop.

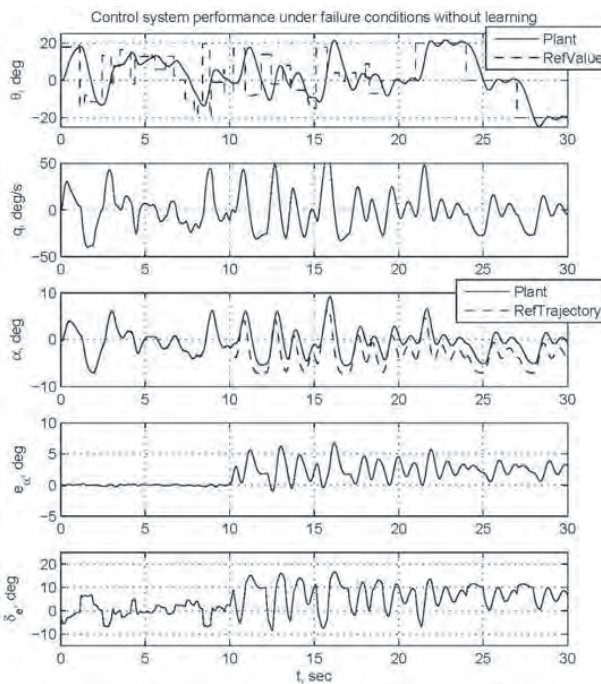


Figure 9: Simulation results for the IDP-based system without adaptation in the angle of attack stabilization loop.

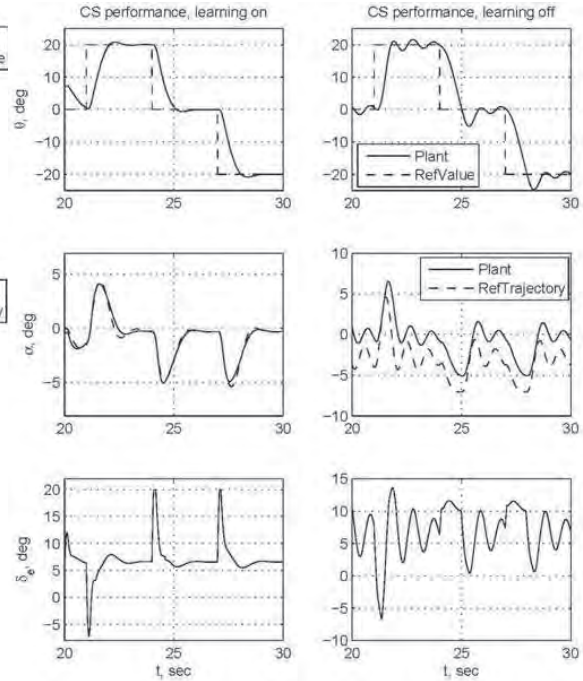


Figure 11: Comparison of control quality for pitch angle before and after damage without adaptation in the angle of attack stabilization loop.

Simulation results for the IDP-based adaptive control scheme are presented on Figures 8 and 9. All of computational experiments were carried out for X-04 mini-UAV [18]. An indicated airspeed V_i is used here as an external parameter to specify aircraft flight regime. All data presented on Figures 8 and 9 were obtained for $V_i = 70$ km/h. Here θ is pitch angle, deg; α is angle of attack, deg; e_α is tracking error for angle of attack, deg; q is pitch angular velocity, deg/sec; δ_e is elevator deflection angle, deg; t is time, sec; *RefTrajectory* is output of the reference model.

Capabilities are demonstrated here for adaptation to abrupt changes of plant dynamics: 1) center of gravity aft shift on 15% in instant time $t = 10$ sec; 2) decreasing of elevator efficiency on 50% at the same instant time.

Figures 10 and 11 shows us comparison results for control quality as applies to pitch angle channel with and without adaptation in the angle of attack stabilization loop (Figures 8 and 9 respectively). Besides we can see an influence of damage on control quality for conventional controller to compare it with IDP-based one.

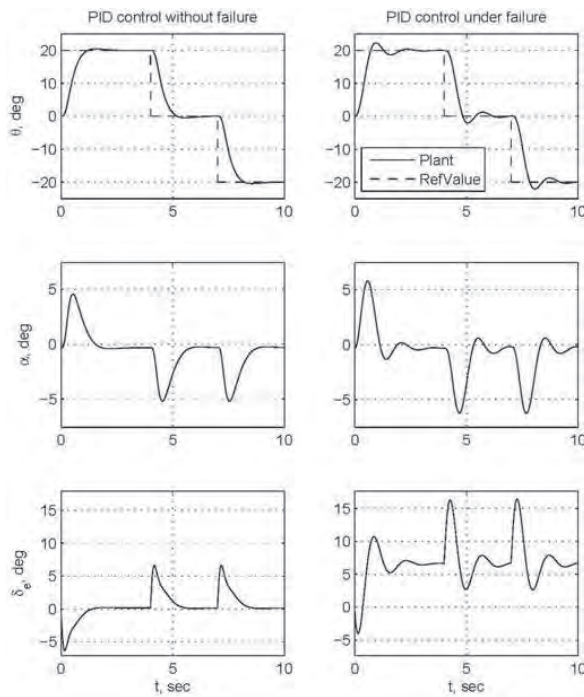


Figure 12: Influence of damage on control quality for conventional controller.

Simulation results (see Figures from 8 through 12) demonstrate how the IDP-based system equipped with the PD-compensator manage effects of two simultaneous damages influencing significantly on the plant dynamics. First of the damages leads to UAV center of gravity aft shift on 15%. It occurs in instant time $t = 10$ sec. The second damage causes decreasing of longitudinal control efficiency on 50% at the same instant time. We can see that the IDP-based

scheme provides operation with a slight error (as a rule $e_\alpha \approx \pm 0.05$ deg) until the first failure occurs. Adaptation to the plant dynamic change in this case executes quite rapidly taking 1.2–1.5 sec approximately. The tracking error is now larger than before the failure but its value still lies in the range $e_\alpha \approx \pm 0.2$ deg and the system stability is unbroken. After the second failure the system stability is still unbroken although the tracking error values become rather large, their values belong mostly to the range $e_\alpha \approx \pm 0.5$.

Thus the suggested reconfiguration scheme for the UAV motion control law proves its efficiency as a tool which allows us to suppress on-the-fly effects of equipment failures and structural damages. Therefore we can ensure some specified level of fault tolerance and damage tolerance for the UAV control system.

We can compare these results with simulation results obtained in [18] for the MRAC and MPC control systems equipped with the PD-compensator as applies to the same X-04 UAV under the same failure cases. Both MRAC and MPC schemes provide operation with the same error value ($e_\alpha \approx \pm 0.05$ deg) until the first failure as it occurs for the IDP scheme. Adaptation to the plant dynamic change for MRAC and MPC schemes executes taking 1.2–1.6 sec approximately. The tracking error values are $e_\alpha \approx \pm (0.18 - 0.22)$ deg after the first failure and the system stability is unbroken. After the second failure the system stability is still unbroken for both MRAC and MPC cases [17]. The relevant tracking error values are $e_\alpha \approx \pm (0.48 - 0.52)$ deg after the second failure. Thus the MRAC-based and MPC-based reconfiguration schemes for the UAV motion control law have in whole very similar properties in comparison with IDP-based scheme.

The most important conclusion following from the simulation results for the IDP-based system as well as for MRAC-based and MPC-based systems (see [18]) consists in the fact that all of these systems can operate successfully including cases with UAV equipment faults and structural damages.

5 CONCLUSION

Investigations considered above show us that the ANN-based approach to build models of complex nonlinear dynamic systems is very effective from the standpoint of simulation accuracy as well as processing speed while using these models. Such ANN-based model features are especially important for on-board implementation of UAV control laws.

The obtained results demonstrate clearly that the ANN-based approach to control complex nonlinear dynamic systems under uncertainty conditions using adaptation mechanisms allows us to adjust control systems effectively in respect to a current situation including emergence of various failures and damages in UAV equipment and structure. Neural network based techniques and tools show us very high efficiency concerning adaptive fault-tolerant and damage-tolerant control for nonlinear systems under various kinds of uncertainty.

Comparison of the MRAC, MPC, and IDP systems do not allow us to prefer explicitly one of these adaptive control

schemes. Each of these systems has both positive and negative properties. Some final choice between MRAC, MPC, and IDP control systems can be carried out only with regard to specific application problem performing sufficiently large sequence of computational experiments.

REFERENCES

- [1] V.N. Fomin, A.L. Fradkov, and V.A. Yakubovich. *Adaptive control for dynamic plants*. Moscow: Nauka, 1981, 441 pp. (In Russian)
- [2] A. Astolfi, D. Karagiannis, and R. Ortega. *Nonlinear and adaptive control with applications*. Berlin a.o.: Springer, 2008, 290 pp.
- [3] P.A. Ioannou and J. Sun *Robust adaptive control*. Prentice Hall, 1995, 848 pp.
- [4] P.A. Ioannou and B. Fidan. *Adaptive control tutorial*. SIAM, 2006.
- [5] E. Mosca. *Optimal, predictive and adaptive control*. Englewood Cliffs, New Jersey: Prentice Hall, 1994, 480 pp.
- [6] G. Tao *Adaptive control design and analysis*. John Wiley & Sons, Inc., 2003, 618 pp.
- [7] V.A. Terekhov, D.V. Efimov, and I.Yu. Tiukin. *Neural network control systems*. Moscow: Radiotekhnika, 2002, 480 pp. (In Russian)
- [8] V.A. Terekhov and I.Yu. Tiukin. *Adaptation in nonlinear dynamic systems*. Moscow: LKI Publishers, 2008, 384 pp. (In Russian)
- [9] V. Gavrillets. Damage tolerant flight control systems for unmanned aircraft. *Proc. of the 26th International Congress of Aeronautical Sciences (ICAS 2008)*, Anchorage, Alaska, USA, 14–19 Sept. 2008.
- [10] H.K. Khalil. *Nonlinear systems*. 2nd Ed, Prentice Hall, 1996, 747 pp.
- [11] P.D. Krutko. *Inverse problems of dynamics in automatic control theory*. Moscow: Mashinostroyeniye, 2004. – 576 pp. (In Russian)
- [12] L. Ljung. *System identification: Theory for the user*. University of Linköping, Sweden, 1987, 412 pp.
- [13] K.S. Narendra and K. Parthasarathy. Identification and control of dynamic systems using neural networks. *IEEE Trans. on Neural Networks*, 1 (1): 4–27, 1990.
- [14] S. Chen and S.A. Billings. Neural networks for nonlinear dynamic systems modelling and identification. *International Journal of Control*, –56 (2): 319–346, 1992.
- [15] S. Omatu, M. Khalid, and R. Yusof. *Neurocontrol and its applications*. Springer, 1996, 242 pp.
- [16] A.N. Gorban. Generalized approximation theorem and computational capabilities of neural networks. *Siberian Journal of Computational Mathematics*, 1 (1): 11–24, 1998 (In Russian).
- [17] S. Haykin. *Neural networks: A comprehensive foundation*. 2nd Ed. Prentice Hall, 1999, 823 pp.
- [18] A.I. Kondratiev and Yu.V. Tiumentsev. Adaptive nonlinear fault-tolerant neural control for unmanned aerial vehicles, *Proc. of the International Micro Air Vehicle Conference (IMAV 2010)*, 6–9 July 2010, Braunschweig, Germany, 20 pp.

Design Approach for Selection of Wing Airfoil with Regard to Micro-UAVs

Vladimir Brusov¹ and Vladimir Petruchik
Moscow Aviation Institute (MAI),
Flight Dynamics and Control Department,
Volokolamskoye Shosse 4, 125993, GSP-3, Moscow, Russia

ABSTRACT

A reasonable selection of wing airfoil is very important part of aerodynamic design process for micro-UAVs. The selected airfoil predetermines substantially performances of the designed UAV. This reason causes us to pay attention to the problem of UAV wing airfoil selection taking into account properties specific for micro-UAVs. A concept of multitask design is suggested to solve this kind of problems. This concept is explained in regard to selection process for micro-UAV airfoil. Presented simulation results demonstrate that using of multitask approach to aerodynamic design of micro-UAV enables us to enhance UAV efficiency due to improvement of its aerodynamic perfection.

1 INTRODUCTION

A class of very-small unmanned aerial vehicles (micro-UAVs) includes UAVs with a mass in the range from several dozen grams up to 1 kg or up to 5 kg according to other appraisals. Micro-UAVs are equipped mostly with electric propulsion system consists of rechargeable battery and electric engine to drive propeller. There are many papers related to reasonable selection of design parameters for a micro-UAV including selection of its airfoil [1]–[8], [10], and [14]. A reasonable selection of wing airfoil is very important part of aerodynamic design process for any micro-UAV. The selected airfoil predetermines substantially lift to drag ratio, altitude-airspeed performance, stalling performance as well as takeoff and landing performance for the designed UAV. These reasons stimulate us to investigate the problem of UAV wing airfoil selection taking into account properties specific for micro-UAVs.

2 AN INFLUENCE OF FLIGHT REGIMES ON SELECTION OF WING AIRFOIL FOR MICRO-UAVS

The wing airfoil selection problem for micro-UAVs has some peculiarities caused by reasons discussed below.

1. *Low airspeeds and low Reynolds number values.* Airspeed values for typical micro-UAVs are usually in the range from 8–10 m/sec to 25–30 m/sec. This range is specified by requirements which are formulated usually to the UAV. According to these requirements the UAV should have capability to carry out flight tasks both during calm and at strong enough wind. In combination with small UAV dimensions it leads to a situation when UAV flies in about critical Reynolds number values if it is near to the bottom of the UAV airspeed range. It is important because of almost

all aerodynamic characteristics change considerably for critical Reynolds number values [1], [2], and [14].

Critical Reynolds number values are from 80000 to 140000 for various wing airfoils. A transition from subcritical Reynolds number values to supercritical ones causes essential enhancement of UAV aerodynamic characteristics. For example UAV lift to drag ratio rises approximately on 50% in this case.

For subcritical Reynolds number values, i.e. for low airspeed, airfoils with small relative thickness (5–7%) and with large relative concavity (even downstream airfoil face is concave in this case) are preferable if we need to maximize airfoil lift to drag ratio. These airfoils become inefficient if airspeed increases because of enhancement for airfoil profile drag.

For middle airspeed values ($U=15\text{--}20$ m/sec) airfoils with relative thickness approximately 14–16% and with almost flat or even convex downstream face are most preferable.

Finally, for large airspeeds (for micro-UAV airspeed about $U=25\text{--}30$ m/c is large enough) sufficiently thin airfoils (8–12%) become preferable again but this time they need to be close to symmetrical one with small relative concavity. The fact is that we do not need high values of UAV lift coefficient for these relatively large airspeeds but it is very desirable to decrease airfoil profile drag.

Thus, the requirements claimed to airfoils in regard to various flight regimes are obviously contradictory.

2. *Nonstationarity of aerodynamic characteristics for micro-UAV with respect to Reynolds number and angle of attack values.* If micro-UAVs flew all the time on small Reynolds numbers including subcritical ones then it would leads only to some decreasing of their lift to drag ratio in comparison with the flight within supercritical Reynolds number region. However, micro-UAVs have very small flight weight and small wing load values. Then micro-UAV under a gust influence can transits very quickly (for few seconds) from subcritical Reynolds numbers to supercritical ones and back to subcritical Reynolds numbers. These transitions cause significant changes in UAV aerodynamic characteristics.

Micro-UAVs have small values for moments of inertia about X, Y, and Z body axes. For this reason micro-UAVs have large angular accelerations and large angular velocities p , q , and r . Moreover, instantaneous center of rotation for micro-UAV do not coincide as a rule with center of gravity for the UAV. Quick changes in angle of attack and sideslip values caused by the rotation lead to emergence of additional aerodynamic moments which depend not only from these angles but also from their rate of change as well as from roll angular velocity. We can see in this case that

¹ E-mail: vsb@mai.ru

the transitions from subcritical Reynolds numbers to supercritical ones and back to the subcritical Reynolds numbers run in different ways. This phenomenon is named as the aerodynamic hysteresis [14]. Almost all aerodynamic characteristics of micro-UAV become highly nonstationary and they depend also from dynamics of changes for Reynolds number values. Moreover large UAV angular velocities about Z axis can cause critical angle of attack value and stalling of the UAV. Various airfoils have different kinds of $C_L(\alpha)$ function with more steep or more smooth function value changes for $\alpha > \alpha_{cr}$ i.e. for angle of attack values exceeded critical ones. As a rule airfoil with relatively large radius of its forebody have more smooth stall characteristics while airfoil with small forebody radius have more steep characteristics.

We can see now that airfoil has an influence on non-stationary aerodynamic characteristics of UAV because of dependence between these characteristics and Reynolds number.

Thus, the reasonable selection of airfoil for micro-UAV must be some compromise between desired airfoil properties for each UAV flight regime and actual characteristics for the selected airfoil.

The design philosophy suggested in our paper is based on usage of some collection of alternative airfoils with known aerodynamic characteristics to choose a reasonable alternative contained in the collection. A synthesis of new airfoils additional to the alternatives of the collection is a separate problem which is not considered in this paper.

3 DESIGN PHILOSOPHY BASED ON MULTITASK APPROACH FOR REASONABLE AIRFOIL SELECTION IN REGARD TO MICRO-UAVS

Thus, as it was stated above, we need to solve the problem of compromise airfoil selection for the wing of a micro-UAV. It is necessary to make this selection for some range of flight regimes which differ one from another with airspeed values. Let us consider this selection problem in wider statement. We will suppose that airspeed is only one element from a set of flight tasks and application conditions for designed micro-UAV.

A problem of adequate representation for the source set of flight tasks and application conditions are very important for the designed micro-UAV as well as for its components especially for airfoil. This problem has great significance for micro-UAVs both for their theoretical issues and applications. A solution of this problem determines immediately requirements specification for the designed micro-UAV. In addition it predetermines optimization approaches and techniques used for micro-UAV design.

Two different approaches are used in contemporary design activity. First of them is based on replacement (by means of aggregation technique) of source set X of flight tasks and application conditions by another set X' . The X' set has fewer elements than the X set. For marginal case of the X' set it consists of only one element x^* which is some typical task named usually as 'design task' or 'nominal task' (see Figure 1).

However this approach causes the problem of appropriate selection for the design task x^* to represent sufficiently the source set X .

An alternative approach suggested and developed in the USSR in the middle of 1960s is presented in [11], [12], and [13].

This multitask approach takes into account the set of flight tasks and application conditions through introducing of the external set X . We choose values of design parameters for UAV according to the multitask approach by means of appropriate optimization problem solving.

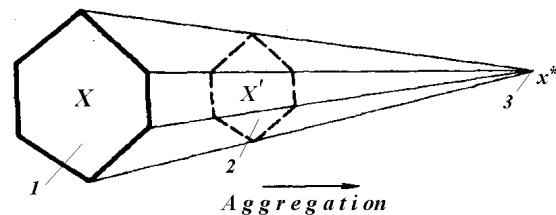


Figure 1: Replacement of the external set X by the design task x^* :
1 – external set X ; 2 – aggregated external set X' ; 3 – design task x^*

This problem involves some unified operation criterion [12], [13]

$$(1) F[X, y, u(t)],$$

required to be minimized or maximized subject to UAV design parameters; here y is vector of design parameters and $u(t)$ is vector of control law parameters for UAV.

Following the aggregated approach [13] we take into consideration tasks and application conditions as some design task $x^* \in X$ in a mathematical model of optimal design. This design task x^* is usually the main element of requirements to the developed UAV and it is derived by means of design task analysis as well as analysis of the source set of flight tasks and conditions of their accomplishment.

As an example of design task x^* for micro-UAV we can specify a flight operation to search some small surface object for prescribed search range D with predetermined UAV payload. A design task in airfoil case can be stated as a cruising flight with some prescribed airspeed U . The influence of airspeed value on reasonable airfoil selection for wing of micro-UAV was discussed above.

According to this approach the optimal design process consists in selection of some alternative micro-UAV version, which is most effective for the prescribed design task. However in real flight conditions our micro-UAV has to be capable to run not only this design task but a set of other flight tasks. Therefore parameter values of the UAV must ensure some design compromise to run efficiently every flight task of the set although these values are not possibly the best for any task.

On Figure 2 we can see how concepts introduced above relate to such important micro-UAV design element as wing airfoil. This example uses a one-dimensional continuous set of tasks $U = [U_0, U_1]$ which is the range of UAV airspeeds. Figure 2 shows us that the first airfoil version (Profile 1) is

most efficient for the design task U_r^* , but this airfoil has large drag losses (they marked with hatching an Figure 2) for off-nominal values of airspeed, therefore the Profile 2 airfoil is preferable in regard to discussed case.

There are no doubts that it is necessary to consider quantitatively the whole set of flight tasks and application conditions for the designed micro-UAV and its elements. If we build the optimal design model taking into account the diversity of prescribed tasks then we can avoid considerable losses of effectiveness for developed UAV. This problem can be solved using the concept of external set X and appropriate optimization algorithms.

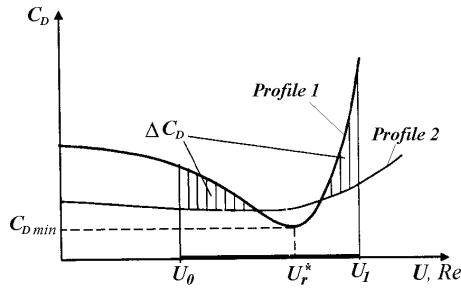


Figure 2: A drag losses for the Profile 1airfoil due to off-nominal values of airspeed

However such approach complicates obviously the optimization model for micro-UAV design parameters as well as it enhances efforts needed to solve optimal design problem. The question is natural about reasonability of this complication as compared with the conventional optimal design problem. It was shown in [11], [12], and [13] as an answer to this question that deriving of the design task x^* basing only on the information about external set X leads to a considerable error in UAV effectiveness estimation.

4 PROBLEM OF MULTITASK OPTIMIZATION AND A WAY TO SOLVE IT

Thus, we can see that a choice of optimal parameters can be represented as an optimization problem for some simple scalar multitask system with external set X , set of strategies Y , and strategies $A=\{y_i \in Y\}, i=1, \dots, m$.

An efficiency index for this multitask system can be specified in two ways according to [12] as some efficiency function in regard to the optimal design problem for a system of UAV airfoils or wings.

First of all, the efficiency function in case of *integrated multitask system* can be stated as

$$(2) F(X, A, E(x)) = \int_X p(x) G(x, y_i) dx$$

or

$$(3) F(X, A, E(x)) = \int_X p(x)(G(x, y_i) - \bar{G}(x))dx$$

Variable $F(X, A, E(x))$ defined by Equation 2 corresponds to the mean value of the functional $G(x, y_i)$. This value is related to a single task from the external set X which is a

region of flight tasks in the considered problem. The optimization problem using this efficiency index is equivalent to the well-known unification problem [11], [12], and [13]. Solving this problem it is possible to derive optimal values for design parameters of the system of UAV airfoils if we know the $p(x)$ function.

Variable $F(X, A, E(x))$ defined by Equation 3 corresponds to the value of absolute deviation of the functional $G(x, y_i)$ from the value $\bar{G}(x)$ which this functional could possess for the UAV wing optimized in regard to the flight regime $x \in X$.

In the second case which is *guaranteed multitask system* one, the efficiency function for UAV system of airfoils/wings $A = \{y_i\}, i = 1, \dots, m$ can be stated as

$$(4) F(X, A, E(x)) = \max_{x \in X} \rho(x, y_i),$$

where $E(x)$ is a *distribution function*, which connects airfoil alternatives with their reasonable usage regions for the predetermined external set X .

Here we introduce the functions

$$(5a) \rho(x, y_i) = \frac{G(x, y_i)}{\bar{G}(x)}$$

or

$$(5b) \rho(x, y_i) = \frac{G(x, y_i) - \bar{G}(x)}{\bar{G}(x)}$$

which define the nonoptimality degree of arbitrary UAV airfoil y_i for a flight regime (flight task) $x \in X$ in comparison with the airfoil optimized for the same regime $x \in X$.

Then efficiency function defined by Equation 4 is maximal nonoptimality degree for the UAV airfoil y_i in regard to the whole region of tasks X .

We can formulate now a general optimization problem named also as strategy optimization problem for described system of airfoils treated as multitask system.

The system of airfoils $A = \{y_i\}, i = 1, \dots, m$ is optimal for the set X of flight regimes $x \in X$ if:

- 1) the collection of airfoils $A = \{y_i\}, i = 1, \dots, m$ provides maintenance of the UAV for all flight regimes of X ;
- 2) the efficiency function value is maximal (see Eq. 6) or minimal (see Eq. 7) for this collection, i.e.

$$(6) F(X, \bar{A}, \bar{E}(x)) = \min_{A \subset Y, E(x)} \int_X p(x) G(x, y_i) dx$$

$$(7) F(X, \bar{A}, \bar{E}(x)) = \min_{A \subset Y, E(x)} \int_X p(x) (G(x, y_i) - \bar{G}(x)) dx$$

or the value of maximal nonoptimality degree is minimal for this system of airfoils

$$(8) F(X, \bar{A}, \bar{E}(x)) = \min_{A \subset Y, E(x)} \max_{x \in X} \rho(x, y_i)$$

Let us notice that we can write expressions according to [12]

$$F(X, \bar{A}, \bar{E}(x)) = \min_{A \subset Y} \int_X p(x) \min_{i=1, \dots, m} G(x, y_i) dx$$

$$F(X, \bar{A}, \bar{E}(x)) = \min_{A \in Y} \max_{x \in X} \min_{i=1, \dots, m} \rho(x, y_i)$$

due to specific nature of relationships described by Equations 2, 3, and 4.

We need to specify realization rate function $p(x)$ for all flight regimes $x \in X$ to determine indices (6) and (7).

However, the efficiency index specified by Equation 8 allows us to derive optimal design parameters of the airfoil system $A = \{y_i\}, i = 1, \dots, m$ without information about distribution of flight tasks in the region X . It is very convenient on early stages of UAV design process when it is especially important to take into account expected region of flight tasks. Optimization according to this efficiency index provides UAV maintenance for each flight regime of the region X . This approach ensures that nonoptimality degree described by Equation 5 do not exceed some limiting value obtained from solution of the problem defined by Equation 8.

The suggested optimality criteria is coincide with conventional minimality conditions if the external set X is reduced to one flight task, i.e. the UAV will operate only in one flight regime.

5 SOLUTION EXAMPLE OF MULTITASK OPTIMIZATION PROBLEM FOR MICRO-UAV

We will discuss in this section a solution example for multitask micro-UAV optimization problem. Our goal is to select the most reasonable airfoil from the predetermined collection of airfoils with known aerodynamic characteristics. Lift to drag ratio $K = C_L / C_D$ is used here as an efficiency function $G(x)$ together with ΔK as a value of relative loss for a wing with arbitrary airfoil $y_i \in \{y_i\}, i = 1, \dots, m$

as applies to a flight regime $x \in X$ in comparison with the airfoil optimized for the same regime $x \in X$.

Airfoil 1	 Eppler 61
Airfoil 2	 Eppler 193
Airfoil 3	 FS 60100
Airfoil 4	 Göttingen 495

Table 1: Dominance of low speed flights ($U = 8\text{--}12$ m/sec)

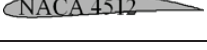
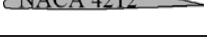
Airfoil 1	 G 532
Airfoil 2	 N-81
Airfoil 3	 NACA 4512
Airfoil 4	 NACA 4212

Table 2: Dominance of middle speed flights ($U = 15\text{--}20$ m/sec)

Airfoil 1	 NACA 19
Airfoil 2	 NACA 20
Airfoil 3	 NACA 15
Airfoil 4	 NACA 17

Table 3: Dominance of large speed flights ($U = 25\text{--}30$ m/sec)

Example. A rational choice of the most suitable airfoil from the predetermined collection.

We have some predetermined collection of airfoils with known aerodynamic characteristics (see Tables 1, 2, and 3) as well as three versions of the weight function $p(x)$ which is interpreted here as a quantity N of flights carried out with different airspeeds (see Figures 3, 4, and 5).

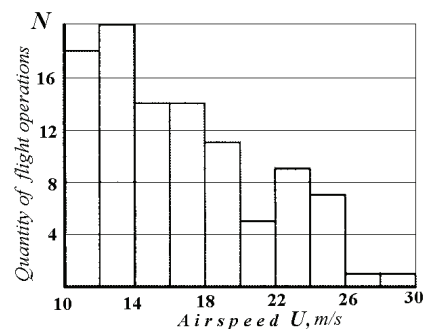


Figure 3

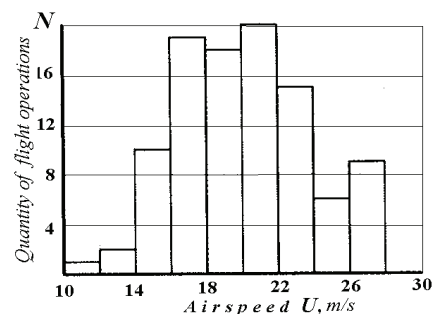


Figure 4

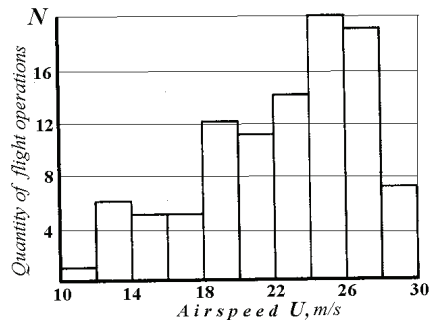


Figure 5

Aerodynamic characteristics for airfoils presented in Table 4 allow us to make up some preliminary conclusions about preferable regions of airspeed values for each subset of the airfoil collection.

Airfoil	Integrated relative L/D losses for micro-UAV wing, ΔK %		
	Weight function, Figure 3	Weight function, Figure 4	Weight function, Figure 5
Eppler 61	42.15	54.51	68.19
Eppler 193	42.85	13.17	16.55
FS 60100	9.41	6.61	18.81
Göttingen 495	37.6	42.74	49.62
G 532	63.89	74.96	78.50
N-81	68.73	82.54	80.93
NACA 4512	14.27	5.99	2.58
NACA 4212	13.33	11.96	11.28
NACA 19	12.97	12.74	18.14
NACA 20	42.81	52.65	56.24
NACA 15	8.09	10.04	28.39
NACA 17	74.84	76.34	74.82

Table 4: Computational experiment results for collection of alternative micro-UAV airfoils with regard to the integrated approach

We can use the integrated approach (see Eq. 7) with weight functions $p(x)$ defined with Figures 3, 4, and 5 to choose more precisely the most reasonable wing airfoil for micro-UAV. Appropriate simulation results are presented in Table 4.

As we can see the NACA 15 airfoil is preferable for the region with dominance of low speed flights (see Figure 3). This airfoil has minimal integrated relative L/D loss which equals to $\Delta K = 8.09\%$. As regards to the dominance of middle speed and large speed flights the NACA 4512 airfoil is preferable. It has $\Delta K = 5.99\%$ and $\Delta K = 2.58\%$ integrated relative L/D losses, respectively (see Figures 4 and 5).

Airfoil	Maximal relative losses ΔK_1 , %	Reasonable airfoil
Eppler 61	96.6	
Eppler 193	36.8	
FS 60100	54.0	
Göttingen 495	82.8	
G 532	96.6	
N-81	30.0	
NACA 4512	94.3	
NACA 4212	77.0	
NACA 19	19.5	NACA 19
NACA 20	82.8	
NACA 15	36.8	
NACA 17	97.7	

Table 5: Collection of alternative airfoils for micro-UAV

We can apply also the guaranteed approach (see Eq. 8), which does not require weight functions $p(x)$. An estimation of relative L/D losses in this case is carried out for the whole region of airspeed values from 8 m/sec to 28 m/sec. Only maximal values of the losses ΔK_1 are essential for each airfoil in the guaranteed case. Simulation results to obtain ΔK_1 maximal values are presented in Table 5.

As we can see from Table 5 the smallest maximal relative L/D loss equals to $\Delta K_1 = 19.5\%$. This value belongs to the NACA 19 airfoil which is the most reasonable choice according to the guaranteed approach. This airfoil is not the best design alternative according to the integrated approach, however its relative losses are rather small. The losses values make up 12.97 %, 12.74 %, and 18.14 % for the weight functions presented on Figures 3, 4, and 5 respectively.

6 CONCLUSION

The problem of most reasonable selection for airfoil was stated and solved in this paper to ensure UAV efficiency for some range of its flight regimes, for example for some range of the micro-UAV airspeed values. The selection is carried out from predetermined collection of airfoils with known aerodynamic characteristics. Appropriate methodology was suggested to solve this kind of design problems using multitask approach.

The multitask approach is based on a set-theoretic statement of design optimization problem which allows to take into account diversity and uncertainty of UAV flight regimes as well as a set of effectiveness criteria.

We can apply the integrated multitask approach to select the most reasonable alternative airfoil if we have information about realization rate function $p(x)$ for all flight regimes. Otherwise we can use the guaranteed approach, which does not require information about $p(x)$ function.

Simulation results show us that suggested multitask approach to select reasonable airfoil enables us to enhance UAV efficiency due to improvement of its aerodynamic perfection.

REFERENCES

- [1] T.J. Mueller. Aerodynamic measurements at low Reynolds numbers for fixed wing micro aerial vehicles. RTO AVT/VKI Special Course on Development and Operation of UAVs for Military and Civil Applications, VKI, Belgium, September 13-17, 1999.
- [2] C. O'Neill. Low Reynolds number airfoils. MAE 5233 November 30, 2001.
- [3] T.F. Burns. Experimental studies of Eppler 61 and Pfenninger 048 airfoils at low Reynolds numbers. Master's Thesis, The University of Notre Dame, January 1981.
- [4] J. de Vries, G. H. Hegen and L. M. M. Boermans. Preliminary results of wind tunnel measurements at low Reynolds numbers on airfoil section E61. Interim Report LSW 80-5, 1980.
- [5] Mueller T. J. and T. F. Burns. Experimental studies of the Eppler 61 airfoil at low Reynolds numbers. *AIAA Paper* No 82-0345, January 1982.
- [6] M.W. Prazak and T.J. Mueller. Experimental studies of an Eppler 61 wing at chord Reynolds numbers from 12,000 to 63,000. Technical Note UNDAS-TN-256-1, July 1997.
- [7] M.S. Selig et al. *Summary of low-speed airfoil data*. Volume 1, Virginia: Soar Tech Publications, 1995.
- [8] M. Selig. High-lift low Reynolds number airfoil design. *J. of Aircraft*, vol. 34, pp. 72-79, Jan-Feb 1997.
- [9] H.P. Buckley, B.Y. Zhou, and D.W. Zing. Airfoil optimization using practical aerodynamic design requirements. *Journal of Aircraft*, Vol. 47, No.5, Sept-Oct 2010, pp. 1707-1719.
- [10] V.S. Brusov and V.P. Petrushik. Theoretical and experimental investigations of aerodynamic characteristics for micro-UAV. 3rd US-European Competition and Workshop on Micro Air Vehicle Systems (MAV07) & European Micro Air Vehicle Conference and Competition (EMAV2007), Toulouse, France, 17-21 September 2007.
- [11] V.S. Brusov and S.A. Piyavsky. Computational algorithm of optimal coverage for regions of the plane. *Journal of Computational Mathematics and Mathematical Physics*, 11 (2): 304-313, 1971 (In Russian).
- [12] S.A. Piyavsky, V.S. Brusov, and E.A. Khvilon. *Optimization of parameters for multitask flying vehicles*. Moscow: Mashinostroyeniye, 1974 (In Russian).
- [13] V.S. Brusov and S.K. Baranov. *Optimal design of flying vehicles. Multiple-goal approach*. Moscow: Mashinostroyeniye, 1989 (In Russian).
- [14] V.S. Brusov, V.P. Petrushik, and N.I. Morozov. *Aerodynamics and flight dynamics for small unmanned aerial vehicles*. Moscow: MAI-PRINT, 2010 (In Russian).

Flight Data Acquisition System for Small Unmanned Aerial Vehicles

Vladimir Brusov¹, Józef Grzybowski², and Vladimir Petruchik¹

¹Moscow Aviation Institute, Russia

²Rzeszow Technical University, Poland

ABSTRACT

Flight investigations of aerodynamics and flight dynamics for micro-UAVs and mini-UAVs stimulate us to use automatic data acquisition systems to obtain valid estimations for UAV performances and characteristics. There exist many kinds of microprocessor-based and microcontroller-based data acquisition systems but all of them do not satisfy specific requirements of UAV flight tests. A Flight Data Acquisition System (FDAS) is suggested to provide support for flight data gathering and registration processes. This FDAS consists of microcontroller-based flight data recorder equipped with SD/MMC memory card to store experimental data, set of sensors to measure UAV flight parameters and software utility providing experiment planning, processing and visualizations of recorded data. Some examples related to UAV flight tests are presented and discussed to demonstrate features of the proposed approach.

1 INTRODUCTION

Flight investigations of aerodynamics and flight dynamics for small UAVs demand usage of automatic data acquisition systems to support valid estimation of UAV aerodynamic and flight performances. There exist many kinds of microprocessor-based data acquisition systems but all of them do not satisfy specific requirements of UAV flight tests.

Investigation of aerodynamic and flight performances for small UAVs is rather complicated problem because of severe dimensions, mass and power restrictions for a Flight Data Acquisition System (FDAS) needed to support flight data gathering and registration [1], [7]–[9]. Another difficult problem is a selection of composition and placement for FDAS sensors.

A programmable micro-controller unit (MCU) based flight data recorder (FDR) is the main component of the FDAS. The FDR is intended to measure and record analog voltage signals incoming from sensors and converts dealing with various physical quantities. The PRP-J5 recorder described in the paper is based on the PRP-J1 type of FDR developed and tested earlier. The tests of PRP-J1 device had revealed necessity of real-time verification for measured and recorded data. In addition we use plug-in

memory card to provide quick and convenient data reading from the card outside of FDR. In addition plug-in FDR data memory allows preprogramming of flight experiment schedules to enhance efficiency of flight tests.

2 DESCRIPTION AND PERFORMANCES OF THE FLIGHT DATA ACQUISITION SYSTEM

The FDAS is composed of several units including MCU-based flight data recorder (FDR) equipped with flash card external memory, card reader to transfer recorded data from the memory card into memory of personal computer, software to manage measurements and to process obtained experimental data, storage battery, gyroscopic motion sensor card, linear accelerometer card, pressure sensor card, voltage stabilizer to supply external devices, temperature sensor card and three converter cards to transform remote control radio commands into voltage signal for recording with FDR.

The PRP-J5 allows us to register up to 24 UAV flight parameters. An SD/MMC flash memory card is used as plug-in recording media in the FDR.



Figure 1: PRP-J5 flight data recorder placed into container housing.

An acquisition of values for needed physical quantities is carried out using such devices as:

- integrated Motorola MPX4115A and Freescale Semiconductor MPX7007 absolute and differential pressure sensors to measure velocity and barometric altitude values [2, 3];

¹ E-mail: vsb@mai.ru

¹ E-mail: jozef.grzybowski@pilc.pl

- high performance STMicroelectronics LIS344ALH 3-axis linear accelerometer to measure accelerations along UAV body axes [4];
- two STMicroelectronics LPY530AL dual axis analog gyroscopes to measure angular velocities around UAV body axes [5];
- deflection sensors for UAV control surfaces implemented through conversion of PWM (Pulse Width Modulator) command signal obtained from remote radio control unit.

Block diagram of the Flight Data Acquisition System is presented on Figure 4.

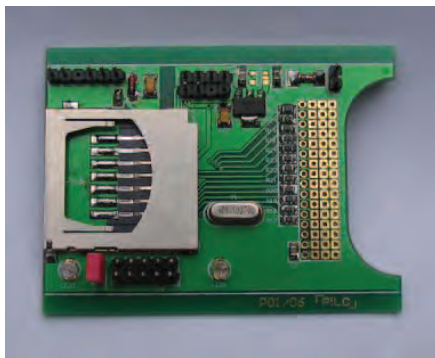


Figure 2: Flash card side of the FDR board.

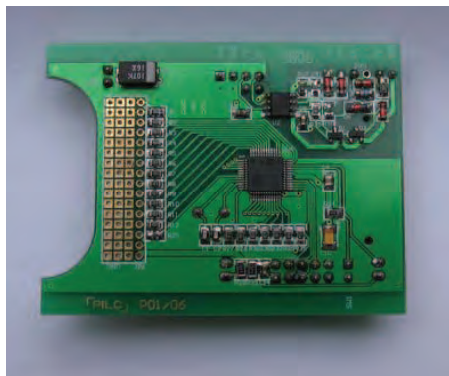


Figure 3: MCU side of the FDR board.

The described FDR is characterized by following features:

- input signals – 24 programmable external analog inputs, ADC with 12-bit resolution, programmable amplifier gains of 16, 8, 4, 2, 1, and 0.5 for each channel;
- memory – plug-in SD/MMC memory card with capacity up to 512 MB;
- measurement/recording frequency – programmable time intervals (1 ms, 5 ms, 10 ms, 50 ms, 100 ms, 200 ms, 500 ms, 1 s, 5 s, 60 s) for each channel;
- reading of recorded data – with SD/MMC PC-connected card reader;
- power supply – Li-Po battery or DC source with 4.5-12 V voltage and 20 mA maximum operating current;
- dimensions – 57 x 37 x 9 mm;
- weight – 17 g without container housing and battery.

3 MICRONTROLLER BASED CORE OF THE FDAS

Flight data acquisition system presented in the paper is based on the C8051F206 micro-controller unit of the Silicon Laboratories C805F2xx family [6], which is a family of fully integrated, mixed-signal System on a Chip MCUs. The C8051F206 is available with a true 12-bit multi-channel ADC. It features an 8051-compatible microcontroller core with 8 kbytes of flash memory. There are also UART and SPI serial interfaces implemented in hardware. The C8051Fxxx family matches well to build systems with high throughput and low power consumption providing high-precision measurement and recording of experimental data. The C8051F206 microcontroller of this family was chosen for PRP-J5 FDR because it allows us to use SD/MMC card as an external memory for experimental data recording.

On-board JTAG debug support allows non-intrusive (uses no on-chip resources), full-speed, in-circuit debug using the production MCU installed in the final application. This debug system supports inspection and modification of memory and registers, setting breakpoints, watchpoints, single steppings, run and halt commands. All FDAS peripherals are fully functional when emulating using JTAG.

The C8051F206 microcontroller used as the FDAS core has following features:

- high speed 8051 microcontroller core – pipelined instruction architecture; executes 70% of instructions in 1 or 2 system clocks; up to 25 MIPS throughput with 25 MHz clock; expanded interrupt handler; up to 22 interrupt sources;
- memory – 256 bytes internal data RAM; 1024 Bytes extended data RAM; 8k bytes FLASH, in-system programmable in 512 bytes sectors;
- analog peripherals – 12/8-bit resolution; up to 100 ksp/s; up to 32 channel input multiplexer, each port I/O pin can be an ADC input; programmable amplifier gains of 16, 8, 4, 2, 1, and 0.5 for each channel; two comparators (16 programmable hysteresis states; configurable to generate interrupts or reset);
- digital peripherals – 32 port I/O, all are 5 V tolerant; hardware SPI and UART serial ports available concurrently; three 16-bit counter/timers; dedicated watch-dog timer; bi-directional reset;
- clock resources – internal programmable oscillator, 2-to16 MHz; external oscillator (crystal, RC, C, or clock); can switch between clock sources on-the-fly;
- on-chip JTAG debug – on-chip debug circuitry facilitates full speed, non-intrusive in-system debug; provides breakpoints, single stepping, watchpoints, stack monitor; inspect/modify memory and registers;
- supply voltage is 2.7V to 3.6V, typical operating current is 9 mA at 25MHz, and 0.1 μ A at sleep mode;
- temperature range is from -40°C to $+80^{\circ}\text{C}$.

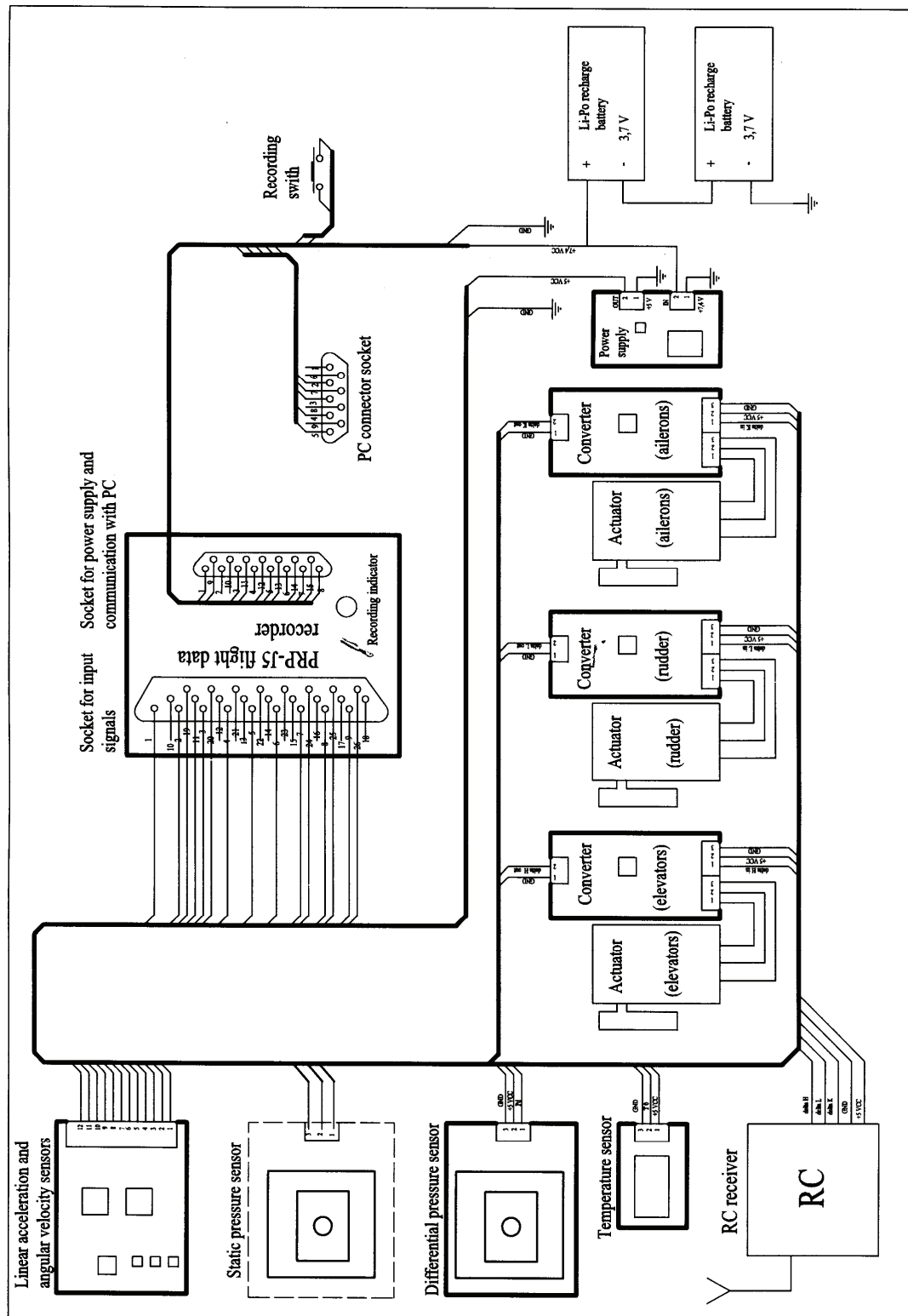


Figure 4: General block diagram of the Flight Data Acquisition System.

4 SOFTWARE USED TO MANAGE FLIGHT DATA ACQUISITION WITH FDAS

Management of measurement and recording processes for UAV flight data is implemented using a configuration file. This file is generated by means of PC-running utility program and it is stored in flash memory card pulled into

FDR socket. The utility program allows us to perform such operations as setting of parameter values to handle data acquisition, to read experimental data recorded on the SD/MMC memory card, and to convert source (raw) data into appropriate text and graphical format. Screenshots presented on Figures from 5 through 7 demonstrate usage of the FDAS software.

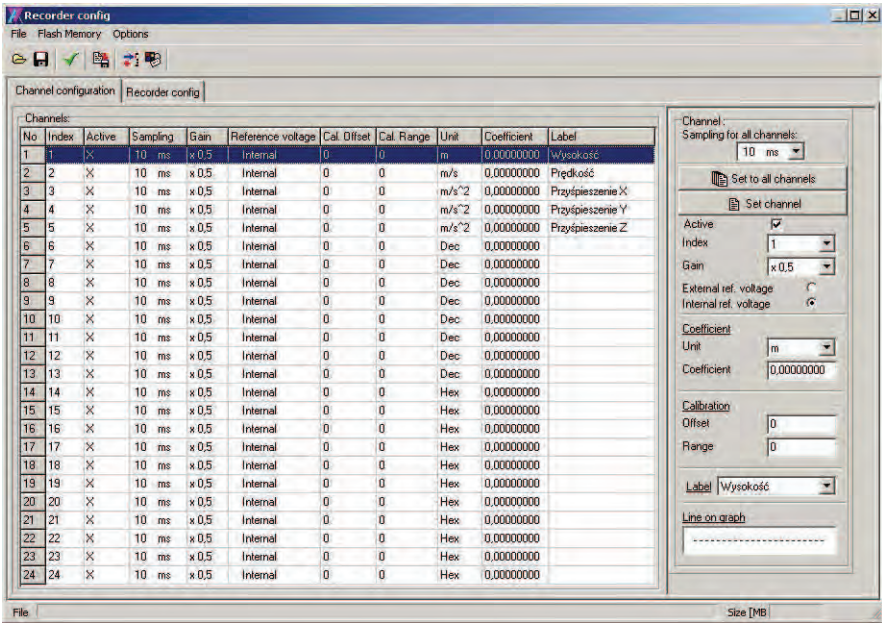


Figure 5: Adjustment window for parameters of recording channels.

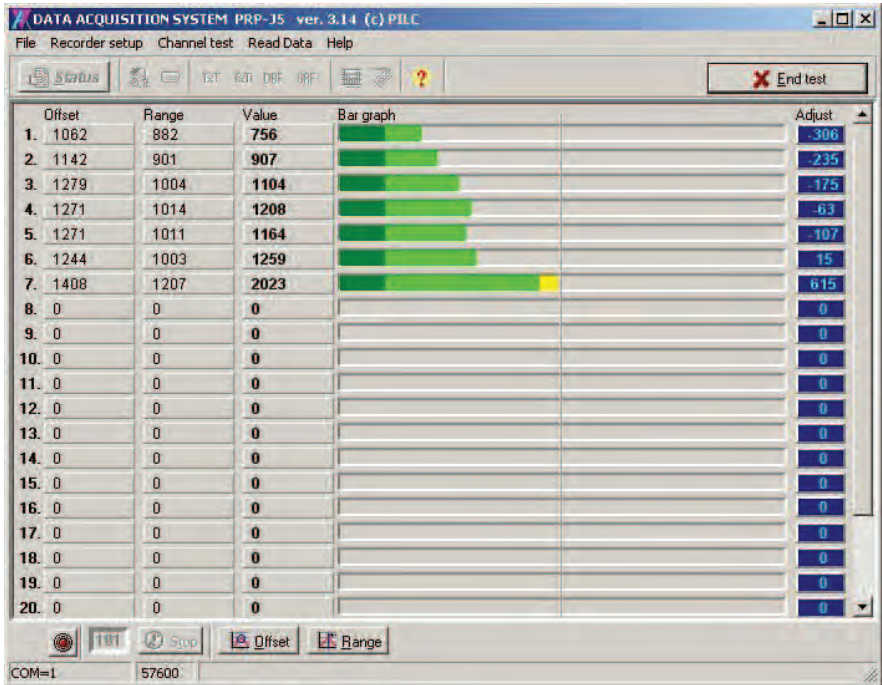


Figure 6: Range adjustment for measured UAV flight parameters.

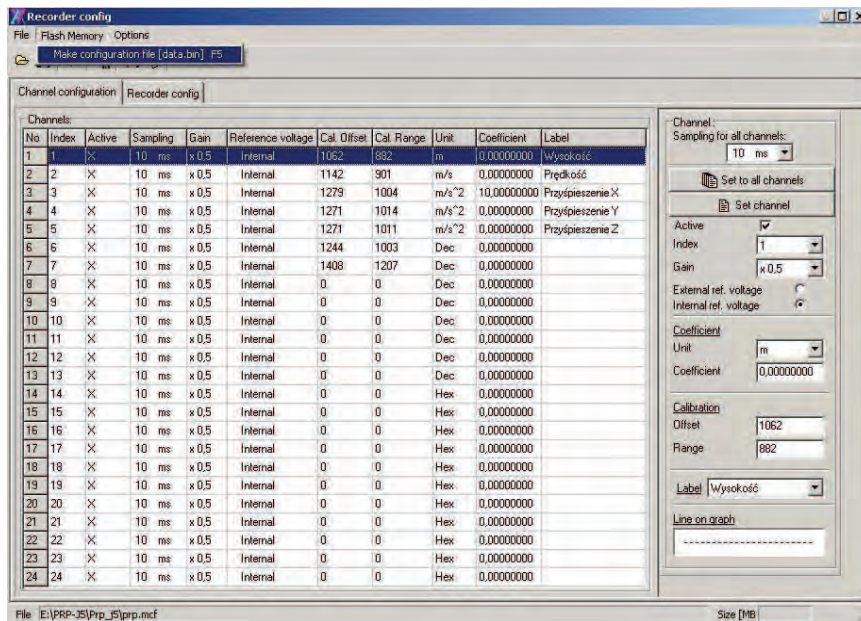


Figure 7: Generation of file with experimental data.

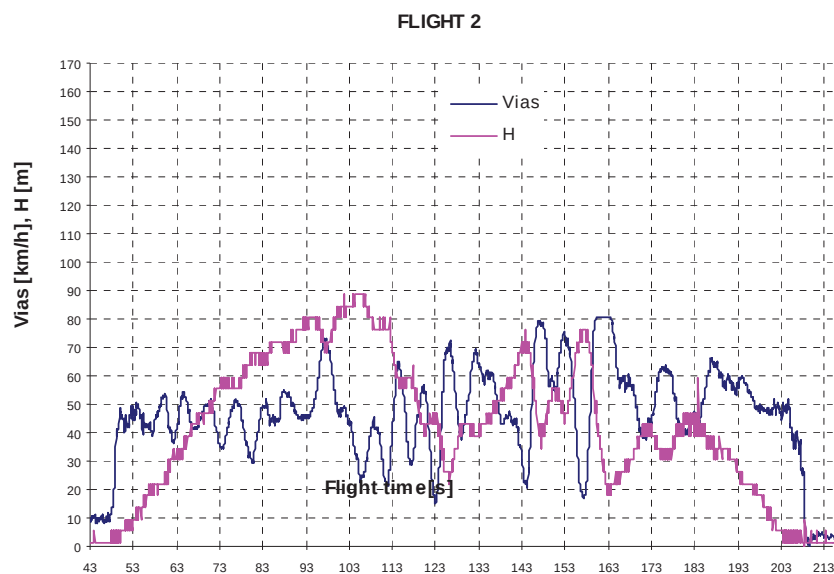


Figure 8: Micro-UAV altitude and airspeed registered in a flight test.

Flight data recording is started by means of FDR power switch on. The recording process is terminated with power switch off then SD/MMC card is pulled out of FDR and is processed off-line with PC to process and visualize obtained experimental data.

Flight test results are presented on Figures 8 and 9 as an example of acquisition and visualization of micro-UAV flight data.

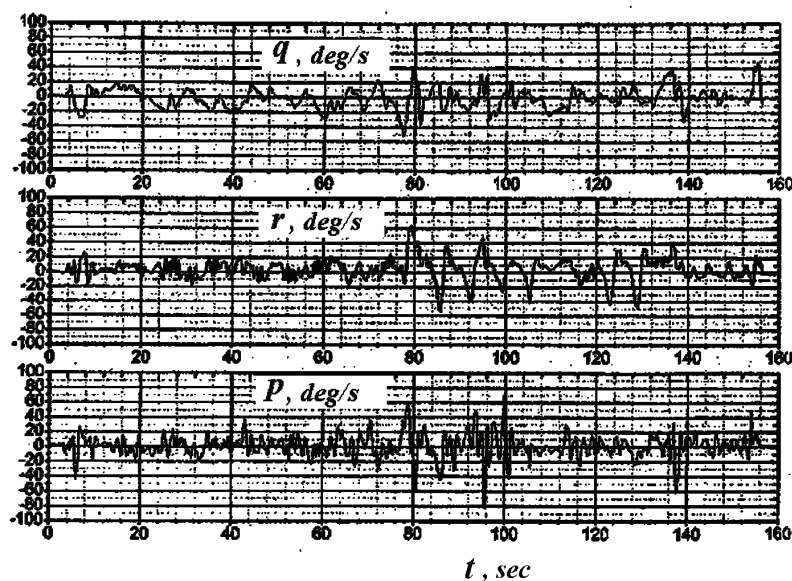


Figure 9: Micro-UAV angular velocities registered in a flight test.

5 CONCLUSION

1. The MCU-based automatic data acquisition system (FDAS) is developed and tested to record flight parameter values for small UAVs.

2. Flight Data Recorder as the main part of FDAS has small dimensions (57 x 37 x 9 mm) and weight (17 g without FDR case and battery).

3. If appropriate sensors are available then the FDR provides recording up to 24 flight parameters of a small UAV using a memory card as well as reading of the recorded data on a personal computer and visualization of the measured data.

4. The FDAS discussed in the paper is equipped with such kind of sensors as:

- absolute and differential pressure sensors to measure air speed and barometric altitude values;
- three-axis linear accelerometer to measure accelerations along UAV body axes;
- two dual axis analog gyroscopes to measure angular velocities around UAV body axes.

5. The FDR can also record angle of attack and sideslip values if appropriate sensors are in the FDAS.

6. Recording of deflection angles for UAV control surfaces (elevator, rudder, ailerons) is carried out through conversion of autopilot control signal or remote radio control pulse signal into analog signals. Three conversion units is used in the FDAS for elevator, rudder and ailerons channel.

7. Innovation of the described FDAS system is to record configuration parameters in the same file as the recorded data. While making experiments it allows us to configure the system very quickly, by inserting an appropriate programmed SD memory card. Working ON-LINE allows to load initial values from the sensors and calibrate their offset and gain, which are then credited to the configuration file on some SD card data. This increases the system modularity and allows us to install other sensors to perform different measurement tasks.

8. The FDAS can be used not only for flight tests but to support wind tunnel tests of real micro-UAVs [10].

REFERENCES

- [1] V.A. Deriabin, R.B. Zolotukhin, and V.N. Chetvergova. Experimental investigation of airplane flight dynamics under large angle of attack values by means of free-flight models. In Proc. of *All-Russian Conf. "Modern Problems of Flight Dynamics, Aerodynamics and Flight Tests"* dedicated to the 100th Anniversary of I.V. Ostoslavsky. – Moscow Aviation Institute, 2004 (In Russian).
- [2] MPX4115A – Integrated Silicon Pressure Sensor for Manifold Absolute Pressure, Altimeter or Barometer Applications On-Chip Signal Conditioned, Temperature Compensated and Calibrated. Data Sheet by Motorola, 2009.
- [3] MPXV7007 – Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated. Data Sheet by Freescale Semiconductor Inc., 2009.
- [4] LIS344ALH – MEMS inertial sensor high performance 3-axis $\pm 2/\pm 6g$ ultracompact linear accelerometer. Data Sheet by STMicroelectronics, 2008.
- [5] LPS330AL – MEMS motion sensor: dual axis pitch and yaw $\pm 300^\circ/s$ analog gyroscope. Data Sheet by STMicroelectronics, 2009.
- [6] C8051F206 – Mixed-Signal 8KB ISP FLASH MCU Family. Data Sheet by Silicon Laboratories, 2003.
- [7] J. Gruszecki, J. Grzybowski, and P. Grzybowski. Mikrokomputerowy rejestrator parametrów lotu do zastosowania w badaniach mikro-samolotów. VI Seminarium poświęcone problematyce badawczej i dydaktycznej katedr i zakładów szkół wyższych oraz instytutów naukowo-badawczych o profilu lotniczym. Bezmiechowa, 25-28 maja 2011.
- [8] J. Grzybowski, L. Baranowski. Wykorzystanie systemu akwizycji danych do badań dynamiki pocisków balistycznych. Zeszyty Naukowe Politechniki Rzeszowskiej, MECHANIKA z.71, AWIONIKA t1, V Konferencja Awioniki, Rzeszów-2007, s.277-284.
- [9] J. Grzybowski, T. Rogalski, and P. Rucido. Pokładowy system rejestracji PSR-04E, Polskie Towarzystwo Diagnostyki Technicznej, Diagnostyka 1(41)/ 2007, str. 75-80.
- [10] V. Brusov, V. Petrukhin, Yu. Tiumentsev. Theoretical and experimental investigations of aerodynamics and flight dynamics for micro-UAVs. Proc. ICAS-2010 Congress, Nice, France, Sept. 2010.

An Architecture with Integrated Image Processing for Autonomous Micro Aerial Vehicles

Christian Dernehl*, Dominik Franke, Hilal Diab, Stefan Kowalewski
RWTH Aachen University, Germany

ABSTRACT

This paper presents an overall MAV design with an integrated camera system. It shows the integration of the camera into the hardware and software architecture and how camera information can be used within the logical design for improving flight control. The presented architecture will be tested and evaluated during the International Micro Aerial Vehicle Conference and Competition 2011 (IMAV 2011)¹.

1 INTRODUCTION

During the last decade research and development of micro aerial vehicles (MAV) have increased and new application areas have been discovered. With the improvement of smaller and better cameras, camera systems found their way as additional and important components in MAVs. By having a camera system available on a MAV the efficiency of this air vehicle increases and new fields of applications become available. For example, this is needed in military operations, where targets have to be identified. Such an identification is often done by a human on ground, to reduce the probability of mistakes. But a camera system is also helpful if a MAV shall autonomously fly through an arch. In such a scenario camera image evaluation can be integrated into the flight control system to help navigation. In addition tiny cameras today are not only getting cheaper, but also capable of high resolution pictures. Together with improved computing power of embedded systems on-board video processing becomes possible. By having video processing on-board of a MAV the reliability of the video processing system increase, since it is not necessary to send video data via a network connection to some other device (e.g. base station) for evaluation. Further, such an approach improves the performance of the overall system, because the time and effort of sending video data to another device and receiving results can be omitted.

In this work we present an autonomous MAV system design with an integrated camera system. On the one hand we show the integration of the camera system in the hardware architecture. On the other hand we also show how to integrate the camera system into the MAVs software architecture. The camera is integrated into the logical system by introducing

sensor weighting in the MAVs decision making process. We also indicate, how one can use the additional, and often powerful, camera system for load balancing of the overall system.

This architecture is implemented in a tilt wing MAV, which takes part in the International Micro Aerial Vehicle Conference and Competition (IMAV) 2011. During this competition MAVs have to perform specific tasks, described in the section below.

The remainder of the paper is organized as follows. Chapter 2 presents functional and non-functional demands on the presented MAV system. In Chapter 3 we list some of the related work on this area. Chapter 4 introduces the MAVs hardware architecture, which is the basis for the software architecture, presented in Chapter 5. Chapter 5 further explains the integration of the camera system into hardware and software. Chapter 6 concludes this work.

2 REQUIREMENTS

There are several missions which the MAV has to accomplish in the IMAV 2011 competition. These include take-off, landing, flying through an arch, hitting a balloon, drop a listening device, record audio from the listening device, identify a vehicle and observe a group of humans. In addition to these tasks, there is an endurance mission, focusing on energy consumption and speed of the MAV. In all tasks, extra points are given for autonomy. In fact, autonomy can win the competition, since the factor 12 is multiplied to the total score if full autonomy is given, i.e. the MAV needs no instructions from humans. From these missions, requirements on the MAV can be derived.

There are various functional requirements. One goal is to have a MAV, which is capable to control itself with respect to environmental interferences, such as wind. In addition autonomy needs to be implemented to fulfill the missions. For instance autonomy means, that the MAV is capable to observe humans and depending on their actions, the MAV shall react appropriately without human interaction. A camera system is crucial with respect to the observation missions. But the camera can also be used to improve the flight control system during other tasks, such as landing. Finally safety regulations, including a safe landing in case of GPS loss, have to be fulfilled. Furthermore, an emergency system is required, allowing a human operator to operate the MAV in case the autonomous flight control does not work properly.

In addition to functional requirements, some non-functional requirements can also be derived. Only small ve-

*Email address(es): {dernehl, franke, diab, kowalewski}@embedded.rwth-aachen.de

¹See www.imav2011.org.

hicles of a limited size are allowed. Furthermore, vehicles, which are longer than $1m$, have a disadvantage in the rating process. This implies a limited size of the MAV. Moreover, there is a maximum weight of $25kg$ and a momentum, defined as mass times speed, of $20kg\frac{m}{s}$ is allowed. As our goal is to achieve high points at the endurance mission with a high air speed, we have strict limitations to the mass of the MAV. The computation hardware components of the MAV shall only have a total weight of less than $150g$.

3 RELATED WORK

The related work are separated into two parts. In the first part we present work related to the architecture of MAV. The second part presents related work done in the area of object recognition.

Architecture Autonomous aerial vehicles are being developed since multiple decades. In 1996, Johnson et al. [1] built an aerial vehicle capable of taking off, landing and discovering certain items with an attached camera system. In their work, the UAV transmits video data to the base station, which processes the images and sends the results back to the UAV. There is no automatic image processing component on-board of the MAV. Instead images are processed at the ground. In our approach video processing is performed on-board of the MAV, which we expect to be faster and more reliable (no transmission delays and faults).

Pastor et al. [2, 3] developed an architecture for UAVs, which includes a base station, an on-board camera system, a communication subsystem and a mission controller. The authors use the IEEE 802.3 standard² (Ethernet) for communication between their components. On top of the IEEE 802.3, application layer protocols are used, such as web services [4]. This architecture allows a high degree of flexibility, since components are easily exchangeable by other components implementing the suitable application protocol. In contrast to our work, an implementation of such a complex protocol stack is not necessary, since we face strong hardware constraints and do not make use of high-level protocols like Ethernet. This is due to the fact that in our project MAVs are considered, whereas the work of Pastor was applied to UAVs in general. Compared to their work our approach is on one side more low-level, but on the other side also more lightweight.

Maranhão et al. [5] designed in their work a hardware and software architecture for UAVs. The main idea is to connect components with the *Universal Serial Bus (USB)*, in which a normal PC, performing fast image processing, is the master while other microcontrollers and the camera act as slaves. Compared to our work, Maranhão focuses on the USB connection between the PC and the microcontroller, controlling the UAV. In our work we use the USB only to connect the camera module to the camera system and use the I²C bus for

other purposes. Our focus is on the integration of the camera system into a flight control system.

Handling multiple UAVs at the same time is the focus of the work of Tisdale et al. [6]. The authors developed an architecture containing a communication system via which the UAVs transmit data to each other. With respect to the communication, a hybrid approach was chosen, featuring communication between UAVs and data transmission to a base station. The base station assigns tasks to the UAVs. These tasks are then processed by one of the UAVs, being capable of accomplishing this mission. Our work, however, focuses on camera evaluation within one single MAV, so no other sensor data from other MAVs are available and, for video evaluation, no connection to a base station is necessary.

Object Recognition In 2011 Chiu and Lo [7] developed a system, in which a camera module is attached to an UAV, transmitting data to a base station. The base station evaluates the video and is connected to a adapted RC receiver in a way the base station can control the UAV. With the detection of the skyline, flight control only by video evaluation is possible. Similar research has been done by Bao et al. [8], focusing primary on horizon extraction. In our work, we incorporate other sensors as well and perform video evaluation on board.

Tisdale et al. [9] used in 2008 multiple UAVs to identify and locate objects in a certain area. In their work, the authors focus on data fusion, which arises from the different available data sources and implement a particle filter, performing in their scenario better than a default Kalman filter.

Chen and Dawson [10] analyzed the tracking problem with UAVs. In their scenario, one UAVs has an attached camera and follows another UAV. The main focus of their work is, however, based on the coordinate system transformations between real world coordinates and the image planes of the UAVs.

4 HARDWARE ARCHITECTURE

A flight control system is responsible for the stability of an aerial vehicle by reading and interpreting sensor data and controlling the engines and other actuators. Since the behavior of the environment, i.e. wind and other interferences, cannot be modeled perfectly, a closed loop control model is chosen. In this model the impact of the environment on the aerial vehicle is measured and fed back into the controller. Basically, there are three types of components: controllers, sensors and actuators. The main component is the controller itself, described in the next subsection. In the second subsection sensors are introduced, measuring the environment and transmitting the measurement data to the controller. Thereafter actuators, executing the actions of the MAV are presented. Finally in the last paragraph a summary of the architecture is provided.

²See www.ieee802.org/3.

4.1 Microcontroller

In our work an Arduino³ based platform was chosen consisting of a software library, providing hardware abstraction, and hardware, the *ArduPilot Mega* board. The board can be programmed via USB, e.g. by a desktop computer. A (*FTDI*)-chip, converting USB data streams to other data streams like RS232, connects the hardware to the USB. Besides, the *ArduPilot Mega* board features two *Microcontroller Units (MCUs)*, an ATmega1280 and an ATmega328 from Atmel. The ATmega1280 offers 16MHz frequency and 128K RAM, while the ATmega328 provides up to 20MHz frequency and 32K RAM. The less powerful ATmega328 increases reliability by working independent of the ATmega1280 as an emergency system. If the ATmega1280 is not available due to heavy overload, the ATmega328 is still capable of controlling the MAV via forwarding incoming RC signals in a well-specified way to the corresponding actuators. The ATmega1280 is used for autonomous flight control, which includes reading data from the sensors and telemetry devices and the evaluation of this data for controlling purposes. All devices within the MAV are connected by wires and various bus systems are used for communication. The ATmega1280 features four universal asynchronous receiver and transmitter (UART) devices and offers registers to attach the I²C bus to the MCU. The ATmega328 provides one single UART.

For the camera system we decided to chose a dedicated embedded system. Our choice is the *BeagleBoard xM*⁴, including a 1GHz ARM A8 MCU and 512 megabyte of RAM. Further, the Beagle Board xM features an I²C port and an USB host chip. Having its own floating point unit the ARM A8 is wide spread for image processing.

Concluding, the airborne computer consists of three microcontrollers, an ATmega1280, ATmega328 and an ARM A8 processor. Each microcontroller is responsible for specific tasks. The ATmega1280 is the main airborne computer and designed for controlling and the decision making process. The decision making process evaluates the current sensor data, e.g. camera data, with respect to the current mission goals and chooses an action. The ATmega328 is used for backup in case the ATmega1280 fails. While both ATmega microcontrollers are responsible for controlling the MAV, the ARM A8 microprocessor has to perform different tasks, i.e. evaluating data from the camera.

4.2 Sensors

Choosing the appropriate sensors is important especially for flight control, in order to get as many reliable information about the environment as needed to follow the current mission. Sensor data is critical for the closed control loop. They include air speed, acceleration, altitude, attitude and position of the aerial vehicle. Each of these *environment variables* can

be measured by a suitable sensor. A very important sensor unit is the *Inertial Measurement Unit (IMU)*, which consists of *gyroscopes*, measuring the attitude, and an *accelerometer*, measuring the acceleration. Only with the IMU already all six degrees of freedom can be determined. Additional sensors are an *air speed sensor* with an attached *pitot tube* and a *pressure sensor*, which can be used for height sensing. With respect to low altitudes, an *ultrasonic sensor* might also be used in combination with the pressure sensor to improve the height estimation. Finally a *GPS receiver* locates the aerial vehicle in space.

For this scenario the *ArduPilot Oil Pan IMU Shield*, which has also been developed within the *ArduPilot* project, has been chosen. This board contains gyroscopes, accelerometer, pressure sensor and a 12-bit *Analog Digital Converter (ADC)*. For instance, ADC is used to convert the sensors on the IMU Shield. The IMU board is located on top of the *ArduPilot Mega* board.

4.3 Actuators

After the controller has evaluated data from the sensors, the appropriate actions are performed. Data is transmitted to actuators in order to change the MAV's fly route accordingly. With respect to the type of the MAV, available actuators differ. For example, a rotary based aircraft has two actuators, motors for the main rotor and the tail rotor. On the contrary a fixed wing aircraft has two motors, a rudder, an elevator and the aileron. One result of the composition of the rotary and fixed wing concept is the *tilt wing aircraft*, which can flip its wings up to 90 degree during flight. Compared to the fixed-wing concept, there exist additionally one tail rotor and another motor, regulating the degree of the wings. Furthermore, there might be actuators for specific mission tasks, for example a chute to drop specific items.

In our scenario, only electric motors are considered, even for the main propulsion. This fact allows the usage of *Pulse Width Modulation (PWM)* to control all actuators. PWM signals can be generated by the ATmega1280 MCU. In this way the controller can directly interact with the actuators without an external, additional engine controller.

4.4 RC System and Telemetry

There are two ways to control the actuators of the MAV, the first is via the airborne computer and the second is via the RC system. The RC system operates in Europe on either 35MHz or 2.4GHz and consists of a remote and a receiver. In most cases data transmission on these frequencies is analogous. When using an airborne computer in autonomous flight mode, the RC system is used as an emergency system, allowing the operator to control the MAV in case the airborne computer is not available (e.g. due to a failure).

Since the RC system serves for manual control of the MAV and as an emergency system, other data (e.g. video data) have to be transmitted to the base station in a different way. Video streaming requires a high transmission rate and

³See www.arduino.cc.

⁴See beagleboard.org/hardware-xM.

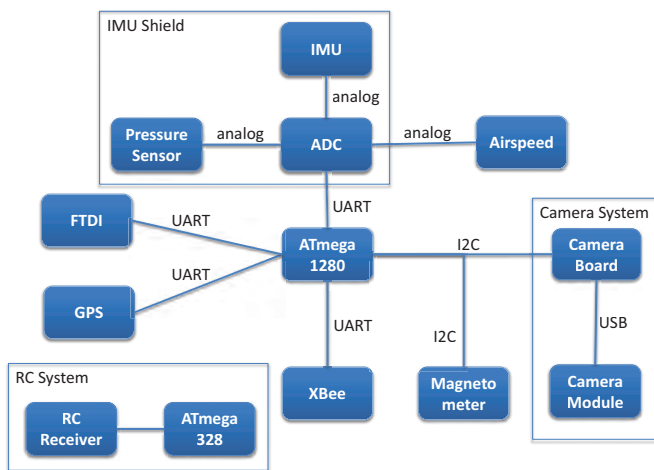


Figure 1: Hardware Architecture

is usually done via digital protocols. The *ZigBee specification* includes a protocol for digital data transfer and is comparable to other protocols like Bluetooth. In fact, Bluetooth and ZigBee reside both in the IEEE 802.15 working group. Unlike analogous RC transmission, the ZigBee specification is packet based and includes its own MAC layer, so multiple devices can be addressed. This is useful, when operating multiple MAVs with a single base station. ZigBee operates on the 868MHz, 902-928MHz, 2.4GHz frequencies.

In our scenario two *XBee modules*, operating at 2.4GHz, are provided. XBee modules implement the ZigBee specification. One module serves as the base station controlling the MAV, and one communicates with the airborne computer. The base station in this scenario is a laptop.

4.5 Overall Hardware Setup

The hardware architecture presented in Figure 1 describes the interactions and connections between the different hardware components. The essential part of the architecture is the airborne computer (ATmega1280) and the IMU board (IMU Shield). The ATmega1280, providing four UART modules, which are used for communication between two hardware components, is responsible for the manual as well as autonomous flight control. In the ArduPilot project the first UART module is assigned to communicate with the FTDI chip. This allows the developer to upload Arduino programs, i.e. programs written in the C language and linking the Arduino libraries, via USB to the ATmega1280. The second UART module is used to communicate with the GPS receiver. The third UART module is utilized to read data from sensors. Analogous sensor data is first transmitted to a 12-bit ADC, before processed by the ATmega1280. In comparison to the UART modules, supporting data transmission between two devices, the I²C bus can handle multiple devices in a master-slave fashion. The ATmega1280 was chosen to be the master device, whereas the magnetometer and the camera system

work as slave components. Note that the camera system is computationally more powerful than the airborne computer. However, the airborne computer with its access to all sensors and actuators is responsible for the decision making process. Therefore, the airborne computer was chosen as the master component.

5 SOFTWARE ARCHITECTURE

After providing an overview of the hardware architecture, this section describes the software architecture. The ArduPilot Mega project provides a general purpose flight control system, capable of navigating to given GPS locations. However, the software architecture needs to cover the additional camera system, as well.

5.1 Camera System

The camera system consists of two components, first a camera board for video processing and second a camera module, connected via USB to the camera board. In contrast to the ATmega MCUs, the powerful camera board is capable to execute a full Linux operating system. It has no hard disc drive, so Linux is installed on a flash card. Since a RS232 port is available, the console can be accessed via the serial interface to communicate with the operating system during the development process. The USB host chip on the camera can be accessed through the Linux USB drivers. So the developer can run software on the camera board, which reads RGB frames from the camera module. For image processing a software module is implemented, which takes advantage of the OpenCV library. Image processing in this context includes the recognition of objects and deriving from this a flight control suggestion, which is sent to the ATmega1280.

For this scenario a camera module with 30 frames per second with a resolution of 640x480 pixels is used. The computer vision algorithms of OpenCV can process 7 frames per second on our hardware, if color frames are evaluated. Instead of calculating color images, about 15 greyscale images can be calculated in a second. Evaluation means here that items, e.g. an arch, within a taken picture are recognized and their position in the picture determined (x- and y-coordinates in pixels relative to the image border). Assuming a cruise speed of about $10 \frac{m}{s}$ this allows 0.7 frames per meter in color mode or 1.5 frames per meter in greyscale mode.

5.2 Airborne Computer

The software of the airborne computer is based on the Arduino Mega platform, which again is based on the open source Arduino platform. Next to the introduced hardware the ArduPilot Mega provides also various libraries, e.g. for autonomous stabilization and GPS navigation.

The airborne computer in our setup consists of a telemetry module, the flight controller, mission controller and is influenced by the camera system. Further, it has access to all sensors and actuators (see Figure 2). The camera system transmits flight control suggestions to the mission controller. This

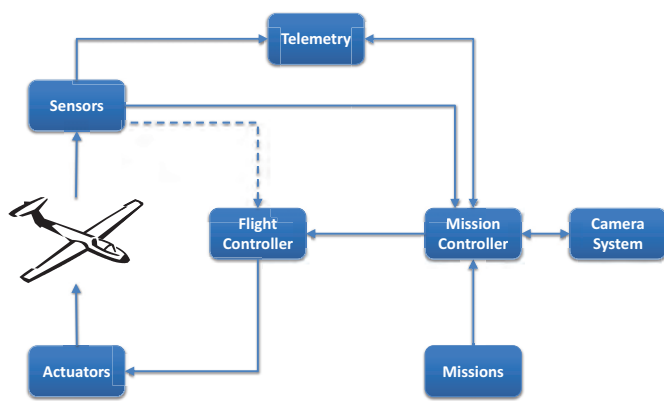


Figure 2: Software Architecture

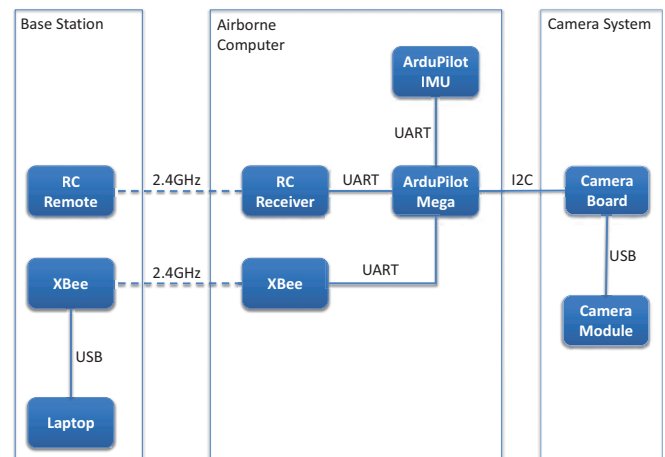


Figure 3: Overall Architecture

is explained in detail in the next subsection. Consider the MAV approaching a balloon. The coordinates of the balloon are fixed and known, stored in the mission database. Then together with the IMU and GPS the flight route can be set accordingly. However, when the MAV is very close towards the balloon the frequency of data measured from the IMU and GPS might not be sufficient to hit a balloon with a diameter of 60cm flying in about 10 meter height. From then on, the MAV can fly assisted by the camera system. Additionally, plausibility tests have to be performed on the mission controller to check the reliability of the camera suggestions or to detect a failure in the camera system. For instance, the camera system suggests to fly lower to hit the balloon. Then the mission controller checks, if this can be confirmed by the current IMU and GPS data (e.g. flight direction and last GPS coordinates indicate that the MAV is flying towards the expected GPS position of the balloon). Furthermore, the mission controller is capable of turning the camera system on or off, e.g. for energy efficiency reasons if the camera is not needed in a mission.

Missions are stored in a database, attached to the mission controller. In this way delays, resulting from loading mission data, are minimized and do not interfere or depend on the current communication load. Mission data is also used by the mission controller for further improvements of the flight control system. For example, in some missions certain sensors are more important to flight control than others. The sensor importance can be expressed by a weighting function, assigning a weight to each sensor before the decision making process (e.g. which actuator is activated) starts. Therefore, the airborne computer can behave differently in each mission. Consider a mission to land the MAV and another one to observe an area. During landing the altitude is crucial, whereas during observation, the exact altitude is of lower importance, but camera data becomes significant. Therefore, in a landing mission, the altimeter and ultrasonic sensors have a higher weight than the camera.

In the autonomous mode the mission controller, knowing what the next task is, works together with the flight controller, which has access to all actuators and sensors (details in next subsection). Since the flight controller implements a closed control loop, current state values are read from the sensors and output data is transmitted to the actuators. The telemetry module allows the user to set the current state of the mission controller, e.g. skip mission x and proceed with mission y.

The decoupling of the mission controller from the flight controller increases, by encapsulation, the reusability of the software modules. Consider the type of the MAV changes from fixed wing aircraft to rotary wing aircraft. In this case the flight controller module needs to be replaced with an according module. The mission controller, however, does not need any additional changes.

In Figure 3 the overall structure is depicted. On the left side, the system used for the base station is illustrated. Signals are transmitted either with the RC remote or the XBee module to the airborne computer. The airborne computer is attached via the I²C bus to the camera system. For interconnections within the airborne computer UART is used, except for the magnetometer, which is not included in the illustration.

5.3 Integration of the Camera System into the Airborne Software Architecture

In this chapter we explain the interaction of the camera system with the airborne computer in detail. If the current mission, chosen by the mission controller, needs the camera system, it will be switched on by the mission controller. Further, the mission controller tells the camera system, which object shall be recognized by sending an object ID. For this purpose a set of corresponding objects with IDs are already specified in the flash storage of the camera system. The camera system is then responsible for recognition, processing and evaluation of the object (e.g. with OpenCV, as ex-

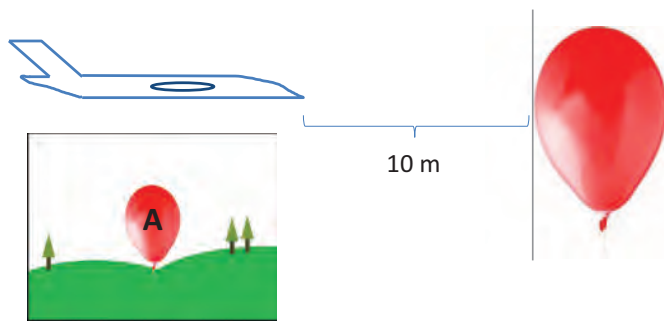


Figure 4: Calibration of Object Size

plained above). Finally evaluation results has to be send to the mission controller, which is capable of interpreting them as flight suggestions. The interpretation in the mission controller is dependent on the current flight mode (manual or autonomous), the current mission (does camera play a role?) and reliability and relevance of the camera system information (do GPS and IMU also confirm, that the MAV is approaching the balloon position?).

To get a better idea of this concept, we take a close look on the communication protocol between the camera system and the airborne computer. Our communication protocol between these two systems contains the following elements:

Status Bit This bit is set to 0, if the camera system does not recognize the wanted object. It has the value 1, if the object is recognized by the camera system. Only in this case, the other three parameters are available to the mission controller.

Size of Object This parameter contains the size of the recognized object, which is needed in the mission controller to estimate the distance to the object and derive from this the intensity of future actions, like adjusting the route.

The size is given in a relative manner. In a calibration process on the ground each object is placed exactly 10 meters ahead of the MAV. The object is then recognized by the MAV and the area that the object takes on the camera images during this recognition is defined as 100%. Figure 4 shows the setup during calibration and recognition of a balloon. In our project the balloon recognition is based on color and shape features. The balloon area is labeled A. We chose area as measurement unit, since it can be fast computed with our camera system. A distance of 10 meters is chosen with respect to the flight speed and size of the available objects during the IMAV competition. If an object is closer than 10 meters to our flying MAV, then reacting on corresponding images becomes difficult as less than one second is left to pass the object in fixed wing mode. But values greater than 100% are possible as well. This is the case, if the object is closer than 10 meters.

We chose a relative measure to keep the encapsulation of

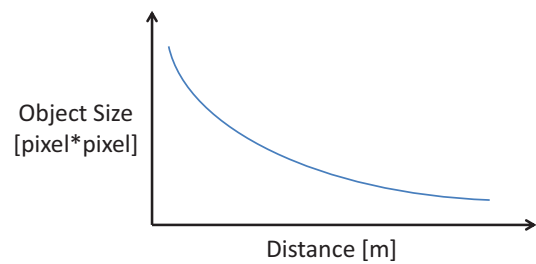


Figure 5: Relation between Object Size and Distance

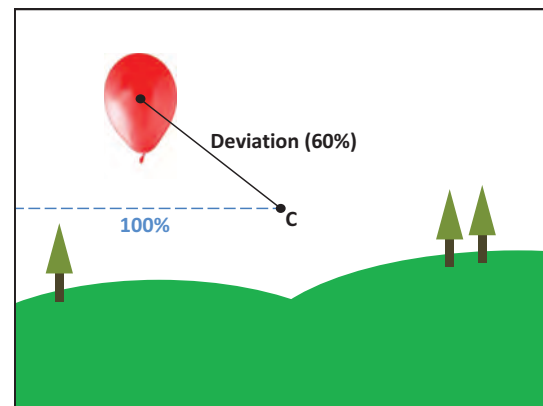


Figure 6: Computation of Deviation

our software and hardware modules high. For instance, if a camera module is replaced, then no adjustments have to be made to the mission controller, since the relative value will not be changed. In case of a camera replacement only the calibration on ground has to be repeated to define the size of 100% area of each object and store this value in the camera processing hardware.

Concluding, the size of the recognized object, as area A on the camera picture, is passed as a relative value to the mission controller. The mission controller has a model containing the size of objects relative to distance. The model is sketched in Figure 5. This model is based on the fact that size of objects recognized on camera images, relates exponential to the distance of the objects. The current size of the object together with the size of the object during calibration, are used by the mission control system to derive the current distance to the object. Based on this knowledge the strength of the necessary action (e.g. steer hard or soft left) is computed by the mission control algorithm.

Deviation Another parameter passed by the camera system to the mission controller is the deviation of the recognized object in the camera image from the center of the image.

Figure 6 sketches the computation of the deviation parameter. It is computed as the distance between the center of the camera image and the center of the recognized object. We

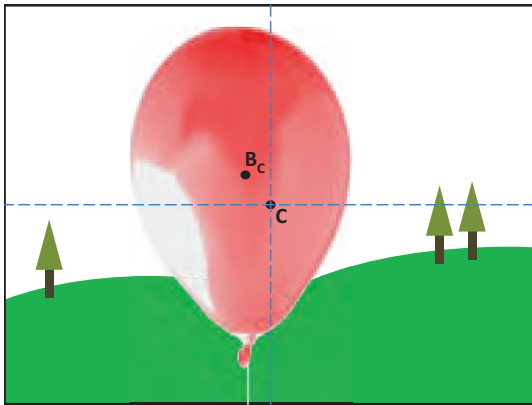


Figure 7: Close to Object

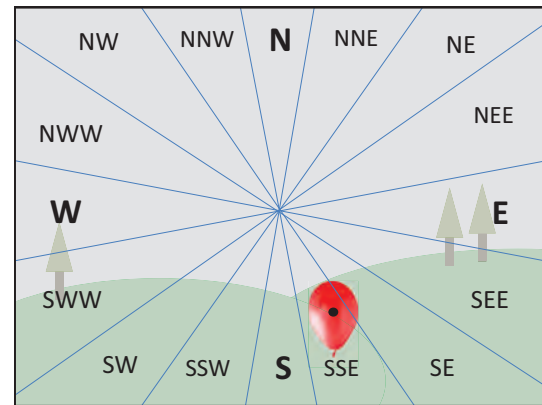


Figure 8: Identification of Direction Parameter

again pass a relative value to the mission controller, for the same reason as with the size of the recognized object. For the deviation 100% are defined as the distance from the center of the image to the left and right border of the image (see Figure 6). The deviation parameter is larger than 100%, when the center of a recognized object is located one of the corners of the image, since the diagonal of a rectangle is larger than its width. Together with the parameter *size of object*, the deviation parameter is used within the mission control algorithm to compute how hard a corresponding reaction of the flight controller has to be executed. For instance, if the mission controller detects with the parameter *size of object*, that it has a large distance to the recognized object (as explained above), and the deviation parameter has a value of 80%, then, although deviation is large, the mission control algorithm instructs the flight controller for a soft reaction, since the distance to the object is large.

If the center of the camera image is within the recognized object (see Figure 7), deviation is likely to be still greater than zero (figure 7, distance between center of image C and center of recognized balloon object B_C). This is due to the fact, that it is within fixed wing flight mode and some certain speed improbable that the center of the recognized object remains exactly on the center of the camera image. Therefore, the mission controller additionally has for each stored object, depending on its size, a certain deviation tolerance. This means, if it for example approaches the balloon, for which it computes that the distance is about 12 meters, and the deviation is 5%, no reaction is necessary. As another example, if it approaches orthogonally an arc, with a width of 10 meters and a height of 5 meters, and computes out of the previous parameters a left distance of 12 meters, then even a deviation of 30% does not imply any reaction, since the arc is large enough to pass properly through with the current route.

Direction The fourth parameter passed by the camera system to the mission controller is a suggestion, to which direc-

tion the plane should adjust its route. Therefore, we divide the camera image to different areas, as presented in Figure 8. The center of this pattern complies with the center of the camera image. The different areas are named by cardinal points (N = north, NNE = north north east, NE = north east, NEE = north east east, ...). To keep the protocol between the camera system and airborne computer short and high-performance these abbreviations are passed from the camera system to the mission controller.

For instance, if a balloon is recognized in the bottom right corner of the camera image (see Figure 8), then depending on the area, in which the center of the balloon is placed, the corresponding abbreviation is passed to the mission controller. The mission controller interprets this direction parameter as a flight suggestion. In case of the balloon, the corresponding flight suggestion would be SSE, meaning that the flight route has to be adjusted down and a bit right.

With the first three parameters the plane knows if an object is recognized (status bit), what the distance to the object is (derived from the size of the object) and the deviation of the object from our current flight route. So the mission controller can compute if a reaction is needed and the strength of corresponding actions. However, it has no information about the direction, in which the action shall be applied. With the direction parameter the mission controller gets a suggestion, in which direction the flight route has to be adjusted. Depending on the current speed of the MAV and the distance to the target, the mission controller decides, if an action is performed, or not.

With these parameters from the camera system together with sensor data (position, orientation, speed), the model of size-distance relation (see Figure 5) and the current mission objectives the mission control system is able to autonomously control the MAV via the flight control system.

6 CONCLUSION

This work presents an overall MAV architecture with a completely integrated camera system. We present the integration on hardware as well as on software layer. While the integration on hardware layer is often given by the available connection types, the software architecture integration can become pretty challenging.

For the communication between the camera system and the airborne computer, we introduce a modular and lightweight protocol. The structure of this protocol consists of four elements: status bit, size of object, deviation and direction. All four values are computed within the camera system. We show how this parameters can be used in the mission control system together with the flight control system, to integrate image processing into autonomous flight control.

By introducing a modular and lightweight protocol for the communication between the camera system and airborne computer, we also clearly separate the tasks of these two modules. We separate the following tasks (on software layer):

1. image recognition
2. image processing
3. image evaluation
4. deriving flight suggestions based on the evaluated images
5. sensor weighting
6. and decision making

Such an encapsulated approach has different benefits on hardware as well as on software layer. On hardware layer, if the camera itself has to be replaced, then only the modules 1 and possibly 2 have to be adjusted. Furthermore, image processing hardware, like the BeagleBoard xM, is often more powerful than the flight control hardware (here Atmega1280). This is why we execute tasks 1-4 on the BeagleBoard xM and tasks 5 and 6 on the Atmega1280. Such a load balance can even be enhanced by outsourcing even more tasks from the flight control hardware to the video hardware (e.g. sensor fusion). This depends on the overall architecture of each single MAV system.

Next to load balancing our approach also increases the safety and reliability of the MAV. If the BeagleBoard xM crashes due to complex tasks or heavy load, this has just a small effect on the separated flight control system. In our case the Atmega1280 will simply notice, that one sensor, namely the camera system, is not available any more.

As future work of this project the evaluation of this presented approach in different flight competitions like the IMAV 2011 is planned.

ACKNOWLEDGEMENTS

This work has been supported by the UMIC Research Centre, RWTH Aachen University, Germany.

REFERENCES

- [1] E.N. Johnson, P.A. DeBitetto, C.A. Trott, and M.C. Bosse. The 1996 MIT/Boston University/Draper Laboratory Autonomous Helicopter System. In *Digital Avionics Systems Conference, 1996., 15th AIAA/IEEE*, pages 381–386, oct 1996.
- [2] E. Pastor, J. Lopez, and P. Royo. An Embedded Architecture for Mission Control of Unmanned Aerial Vehicles. In *Digital System Design: Architectures, Methods and Tools, 2006. DSD 2006. 9th EUROMICRO Conference on*, pages 554–560, 0-0 2006.
- [3] E. Pastor, J. Lopez, and P. Royo. UAV Payload and Mission Control Hardware/Software Architecture. *Aerospace and Electronic Systems Magazine, IEEE*, 22(6):3–8, june 2007.
- [4] Francis McCabe, David Booth, Christopher Ferris, David Orchard, Mike Champion, Eric Newcomer, and Hugo Haas. Web Services Architecture. W3C Note, W3C, February 2004. <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/>.
- [5] D. Maranhao and P. Alsina. Project of a Hardware and Software Architecture for an Unmanned Aerial Vehicle. In *Robotics Symposium (LARS), 2009 6th Latin American*, pages 1–6, oct. 2009.
- [6] J. Tisdale, A. Ryan, M. Zennaro, Xiao Xiao, D. Caveney, S. Rathinam, J.K. Hedrick, and R. Sengupta. The Software Architecture of the Berkeley UAV Platform. In *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, pages 1420–1425, oct. 2006.
- [7] C.-C. Chiu and C.-T. Lo. Vision-only automatic flight control for small uavs. *Vehicular Technology, IEEE Transactions on*, 60(6):2425–2437, july 2011.
- [8] Gui-Qiu Bao, Shen-Shu Xiong, and Zhao-Ying Zhou. Vision-based horizon extraction for micro air vehicle flight control. *Instrumentation and Measurement, IEEE Transactions on*, 54(3):1067–1072, june 2005.
- [9] J. Tisdale, A. Ryan, Zu Kim, D. Tornqvist, and J.K. Hedrick. A multiple UAV System for Vision-based Search and Localization. In *American Control Conference, 2008*, pages 1985–1990, june 2008.
- [10] Jian Chen and D.M. Dawson. UAV Tracking with a Monocular Camera. In *Decision and Control, 2006 45th IEEE Conference on*, pages 3873–3878, dec. 2006.