

Automation of Design and Prototyping of Micro Aerial Vehicle

D. Lundström^{*}, K. Amadori[†] and P. Krus[‡]
Linköping University, Linköping, 581 83, Sweden

The presented work is centered on the automation of the design process of Micro or Mini Aerial Vehicles (MAV). A design optimization framework that links together a CAD system for airframe design and a panel code for aerodynamic evaluations has been developed. This paper is based on research and results previously published by the research team. It describes the experiences made so far, and demonstrates with a case study, how fully automated design is indeed possible. The user is required to enter the initial requirements into the system that will then optimize the MAV design. Both the geometry and the propulsion system are taken into account. Finally, a 3D printer is used for manufacturing of the aircraft. The optimization comprises both discrete and continuous variables. Validation of common propulsion system models is also presented.

Nomenclature

α	=	Angle of attack
<i>CAD</i>	=	Computer Aided Design
<i>COTS</i>	=	Commercial Off The Shelf
<i>DDM</i>	=	Direct Digital Manufacturing
c_L	=	Lift coefficient at given angle of attack α
c_{di}	=	Induced drag coefficient at given angle of attack α
c_m	=	Pitching moment coefficient at given angle of attack α
c	=	Chord length
E	=	Endurance
<i>FDM</i>	=	Fused Deposition Modeling
<i>FEM</i>	=	Finite Element Method
I_o	=	Electric motor zero load current
K_v	=	Electric motor Rpm constant
<i>PWM</i>	=	Pulse Width Modeling
R_m	=	Electric motor internal resistance
<i>Rpm</i>	=	Revolutions per minute
W	=	Aircraft weight

I. Introduction

Design automation is of general interest in aeronautics, and automated methods for coupling aerodynamic calculations, CAD modeling, FEM analysis etc are getting an increased usage in the design of manned aircraft, but primarily during the conceptual and preliminary design phases. Completely automating the design, from concept to production is, however, far from possible. Micro Air Vehicles (MAVs) on the other hand are small, simple to build, and requires relatively few components. This is an application where fully automated design can be implemented. This can be regarded as a stepping stone from which design automation of more advanced vehicular systems can be developed. The ideal MAV design automation procedure is described in Figure 1.

^{*} PhD Student, Dept of Management and Engineering, David.Lundstrom@liu.se, AIAA Student Member.

[†] PhD Student, Dept of Management and Engineering, Kristian.Amadori@liu.se, AIAA Student Member.

[‡] Professor, Dept of Management and Engineering, Petter.Krus@liu.se.

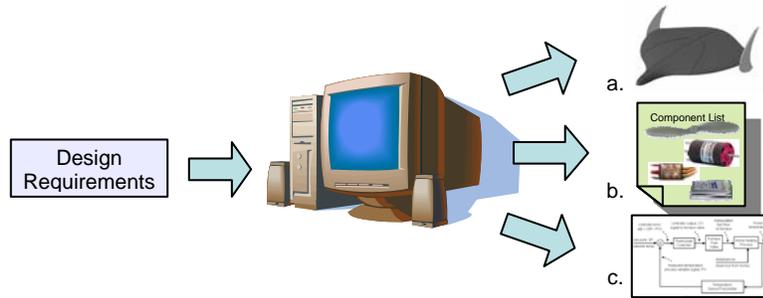


Figure 1. MAV Design automation

From a mission requirement a design specification is created. This is information such as performance, payload requirements etc. A design tool, or design framework, then uses several coupled computer software's to generate the geometric design in CAD, a list of off the shelf components for propulsion, and computer code for autonomous flight ready to upload in the intended autopilot. Such a framework is being developed at Linköping University. In a previous paper by the authors¹, MAV design optimization was demonstrated using a Genetic Algorithm to configure an optimal propulsion system from a database of components, while simultaneously establishing the optimum geometrical plan form. In a later study² this optimization method was scaled up to a design framework including CAD software and panel code for aerodynamic evaluation. The result was inspiring and showed that based on a few specific mission requirements (endurance, payload etc) it was possible to automatically generate a complete MAV design with CAD models for production. There was however two important issues that at the time was not dealt with and that was to insure that during the optimization, the proper balance and size restrictions were met. The MAV must have a large enough volume to accommodate it's intended components, as well as balance with a proper stability margin.

In the present paper both the balancing and volume criteria has been taken care of. The optimization procedure has been expanded from a single objective function to a multi objective pareto front optimization. Finally the loop is closed by actually manufacturing and test flying an automatically generated MAV. The manufacturing is done using a 3D printer, a novel approach to MAV manufacturing that truly allows for "button click" design automation.

II. Distributed Design Framework

Conceptually the design framework has been kept unchanged from the one presented in Ref.2. The structure of the framework could still be illustrated as in Figure 2, where the core is a user-friendly Excel spreadsheet. Though, in the present paper some significant changes have been implemented. Previously the spreadsheet served as an input interface for the user and stored the different design variables. It was also used to link together the calculations between CAD software and the Panel Code. The calculations of the different propulsion system components were made in Excel. After parameters were input in the spreadsheet, Excel called the CAD software to update a parametric CAD model. From this Excel retrieved computed weight, center of gravity, and a mesh that was exported to the panel code for aerodynamic calculations. Lastly Excel summarized the MAVs performance.

In this work the design framework has been implemented with modeFRONTIER v.4.1.0. Thus a more detailed picture of the process could be drawn, as seen in Figure 3. Moreover modeFRONTIER was used to control and govern all connections between all tools, rather then employing VBA-scripts coded into Excel macros.

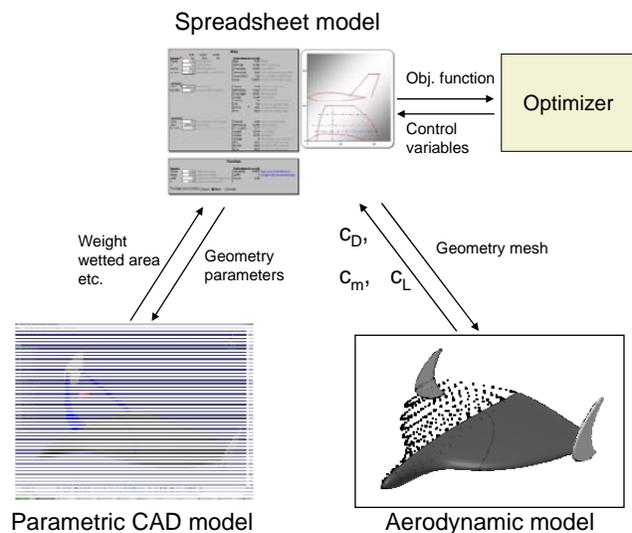


Figure 2. The design framework

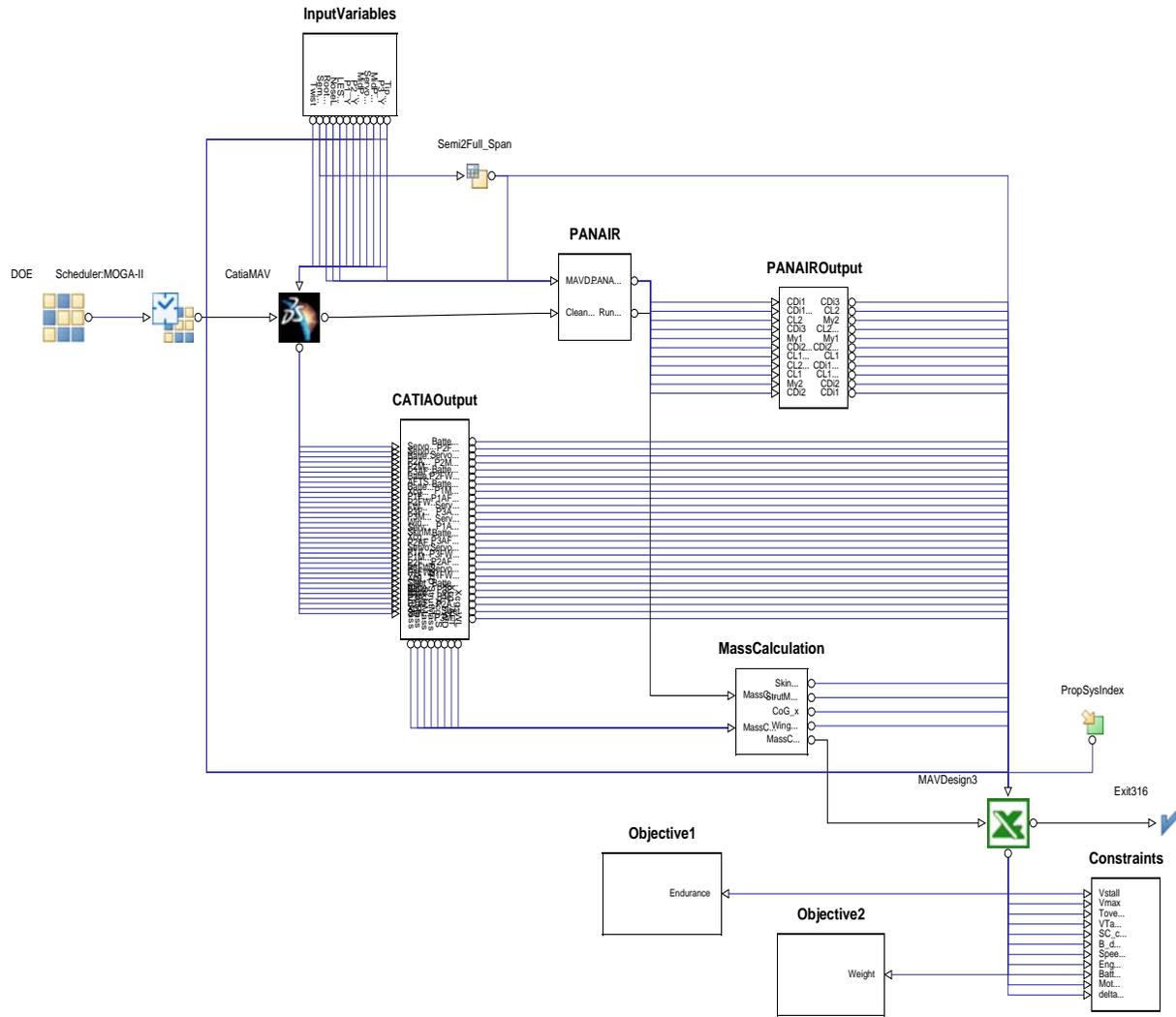


Figure 3. The design framework process as pictured in modeFRONTIER

Comparing the two framework pictures, the latter one appears more complex. This is mostly due to the very large number of variables and connections being pictured. The reason for this can be traced to a shortcoming of the software that did not allow sending or extracting vector variables from the CATIA-node, therefore requiring that every single variable, going into the model or being retrieved from it, to be explicitly pictured in the process.

On the other hand, modeFRONTIER offered drag-and-drop capabilities when creating the work-flow and, more importantly, it includes a multi-objective genetic algorithm (MOGA II¹¹) that provides increased analytical possibilities (see Chapter II.C). The CAD model and the Excel workbook pictured in Figure 3 are basically the same as those in Ref. 2, with only the addition of new balancing and payload fitting tools that are described in the following chapter.

Another useful feature of the software is that it keeps track of all optimization steps, ensuring that basically no data is loss in case of a system failure. This was very important, since the system suffered from frequent and inexplicable interruptions during the optimizations. The most stable conditions were found when running the system on a Dell Precision M2400 laptop, rather than on a much faster Dell Precision PW390 64bits workstation. Together with the support from ESTECO¹², the authors are still investigating the matter and hopefully an explanation will be found. The downside of logging all steps is that the evaluation speed is decreased. Compared to the cycle-time

registered with framework from Ref.2, using modeFRONTIER each function evaluation took 40-50% more, being the cycle-time between 70 and 90 seconds.

A. Parametric CAD Model

The MAV is limited to be a tailless aircraft. The wing is defined by total area, aspect ratio, dihedral and twist angle. Two parameters are also controlling the curvature and shape of the leading and trailing edges, allowing the wing to be shaped with a “non trapezoidal” contour. The wing profiles at the wing root and tip can be chosen from a catalog and are controlled through two dedicated parameters.

The fuselage is completely blended with the wing and its size depends on the wing root length and thickness. It is also possible to specify the cross section size and length of the portion of the fuselage ahead of the wing.

The fins are placed at the wing tips and their dimension is defined by a tail volume coefficient. Other parameters that can be set are wing sweep and taper ratio, plus a coefficient that controls how the fins surface is distributed above and below the chord line. Figure 4 and Figure 5 show a principal sketch of a typical geometry created in the geometry module.

The geometry includes also three different payload boxes and all control system equipment. Each component is represented as a rectangular box of given length, height, width and weight, all taken from a database included in the spreadsheet. It is then possible to review their placement in order to balance the aircraft and to verify that everything fits inside the outer surface.

The generic CAD model of the MAV has been developed using CATIA V5 r18. The geometric parameters entered in the excel spreadsheet, are used to determine the outer surfaces in this model. Then the internal systems and structural elements are placed within them. The catalogues in the spreadsheet includes the following components (between parentheses is the number of choices available) :

- Motor (151)
- Motor PWM Controller (9)
- Battery (26)
- Propeller (29)

Since the aircraft is of very simple nature and of a very small size, the structural requirements are quite simple.

From the CAD model it is possible to retrieve a precise measure of the aircraft weight⁹ and mass distribution that can be used for both performance prediction as well as flight simulations.

The CAD model plays a central role for the balancing and packing tool that has been developed. For each one of the components that have to be fitted inside the aircraft, the tool checks if the component can be fitted and - if so – between which extreme positions in the longitudinal direction it can be moved for balancing purposes. To do so in a simple, fast and reliable way, only the most outboard side of each component is analyzed. It is therefore assumed that the surface of the aircraft is continuous and monotone in the y-direction. For the considered model this is a fair assumption. The

MAV’s outer surface is intersected with a plane lying on the outboard face of the component and then the available height distribution is approximated with a polynomial curve. This equals to trace how the total thickness of the aircraft changes with the x-coordinate at a given y-location. On this curve it is then possible to check whether the component fits or not and how far it can be moved in the x-direction (Figure 6). During optimization an internal

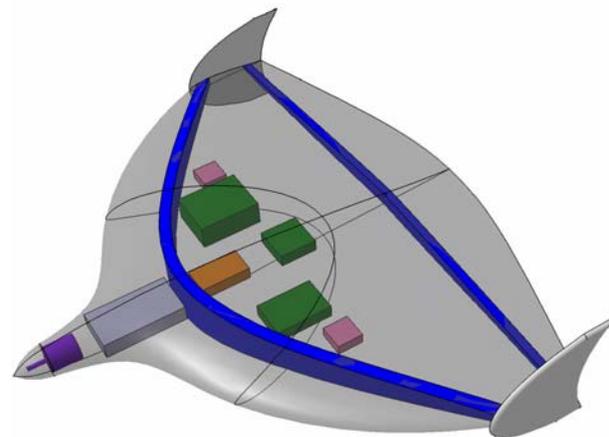
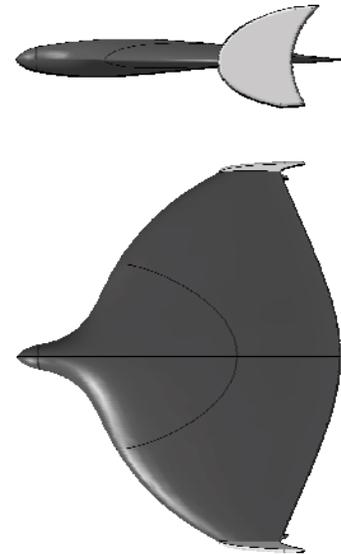


Figure 5. The parametric MAV CAD model

balance loop is executed, for each function evaluation, which automatically seeks to balance the aircraft within a predefined stability constraint.

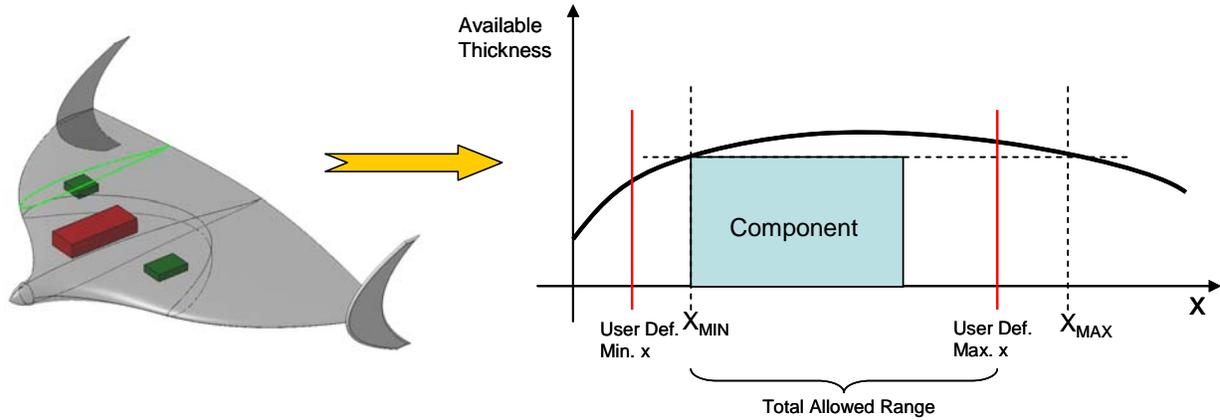


Figure 6. The balancing and fitting tool.

B. Aerodynamic Analysis

At the moment, the aerodynamic analysis tool adopted is a panel code, PANAIR. Panel codes are numerical schemes for solving (the Prandtl-Glauert equation) for linear, inviscid, irrotational flow about aircraft flying at subsonic or supersonic speeds¹⁰. As pointed out by Amadori et. al.^{3,4}, panel codes are not as precise as modern CFDs can be, but they have other advantages. During the optimization, the aircraft geometry and its outer shape is not precisely defined and the allowed range of variation is very large. It is clear that it can be unpractical and not justified to use tools that have a much higher accuracy. Moreover CFDs requires the space around the studied body to be accurately meshed, while for a panel code it is sufficient to approximate the aircraft's outer surfaces with proper rectangular panels. Therefore the meshing time required by a panel code is lower by several orders of magnitude, compared to a CFD code. When much powerful and faster computers will be available or if higher accuracy was required, PANAIR could be substituted with other solvers, thanks to the modular nature of the framework.

The CAD model described previously is also responsible for generating a mesh of the surfaces of the aircraft. This is performed by an in-house tool developed at Linköping University. This grid is then used by the panel code algorithm to calculate basic aerodynamic coefficients for a given mission section. The parameters that are required for an analysis to be carried out are angle of attack, yaw angle, air speed and altitude. Outputs of this module are lift coefficients c_L and $c_{L,\alpha}$, induced drag coefficient c_{di} and pitching moment coefficients c_m and $c_{m,\alpha}$. PANAIR returns also the pressure values and speed vectors in each node of the mesh that is input.

PANAIR is run at three different and arbitrary angles of attack, i.e. 3,6 and 9 degrees (Figure 7). The only requirement here is that the angles must be within the linear range of the lift coefficient. The results from this analysis permit to retrieve the slope of the c_L -curve as function of the angle of attack ($c_{L,\alpha}$) and the shape of the c_{Di} -curve as function of α . The latter is used for performance calculations other than at the cruise condition. Given now the weight of the aircraft, its cruising speed and altitude, the cruising angle of attack α_{cruise} is calculated. Then it is also possible to predict the induced drag coefficient ($c_{di-cruise}$) in cruise condition.

The panel code described gives the induced drag but the parasite drag Parasite drag has been calculated using Prandtl and VonKarman low Reynolds number flat plate skin friction equations⁵ corrected with form factor for the

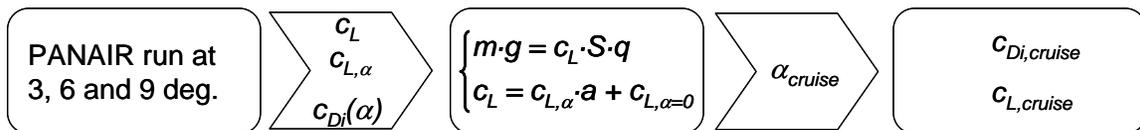


Figure 7. Each configuration is analyzed in two steps.

pressure drag.

It should be pointed out that using panel code as described above has been a way to demonstrate an automated MAV design, but the panel code itself may not be the optimal choice for modeling MAV aerodynamics. Its main drawback is that it does not accurately capture effects of low Reynolds number flows, and vortex lift/drag. Both effects can have a fairly large influence on the overall lift and drag of a MAV. Work is being done to better take these effects into consideration. Inspiration comes from Cosyn⁶ and Lupo et. al.⁷. Both are using panel codes combined with two dimensional airfoil data. Cosyn uses panel code to compute lift distribution and then uses experimental two dimensional airfoil data to compute the drag. Lupo uses a similar method where panel code gives the lift distribution but where two dimensional data from X-foil⁸ is used for drag estimation. Lupo also iterates the results taking into account effects on boundary layer thickness found in x-foil calculations. Both have achieved good results without using heavy computational power. Similar techniques may be a suitable extension to the PANAIR code used in this work.

C. Optimization Algorithms

The design optimization task comprises two different aspects: the optimization of the aircraft shape and the optimization of the components of control and propulsion system. These can be run separately, in sequence or simultaneously.

The design optimization has been divided into two successive phases. First the framework is run without invoking CATIA and PANAIR. In this initial mode, a larger number of parameters are involved in the optimization which comprises both the geometry layout of the aircraft as well as the selection of the propulsion system components. During these calculations traditional lifting line equations are used for aerodynamics and simplified weight estimation formulas are used for weight and centre of gravity. Then the system is restarted, this time involving CATIA and PANAIR, using a slightly different set of input variables. This approach was selected for two main reasons. Firstly, because each function evaluation involving CATIA and PANAIR takes between 70 and 90 seconds, depending mostly on how many parameters are changed from the previous configuration. Running the system using the simpler models embedded in the Excel spreadsheet takes instead only 2,5 seconds, allowing to take into account a much larger number of variables. Secondly, it was noted that when components were chosen from the full database, a very large number of design resulted unfeasible, due to violations of constraints related to the propulsion system. Many function evaluations are necessary for the algorithm to orient itself among the large number of combinations possible. Therefore, in the second step, the propulsion system is frozen into a short list of predefined configurations that are deduced from the results from the first phase. Among all the pareto designs from the first run, the unique propulsion systems were saved in a list that was later used during the second phase, instead of permitting the algorithm to choose among all different components. In this way, a complete description of the whole propulsion system could be obtained, with one single parameter, thus reducing the number of optimization variables. Moreover, in the second phase, design parameters that could not be evaluated using the lifting-line theory can be included, such as the tip chord twist or the wing profile. The process is illustrated in Figure 8.

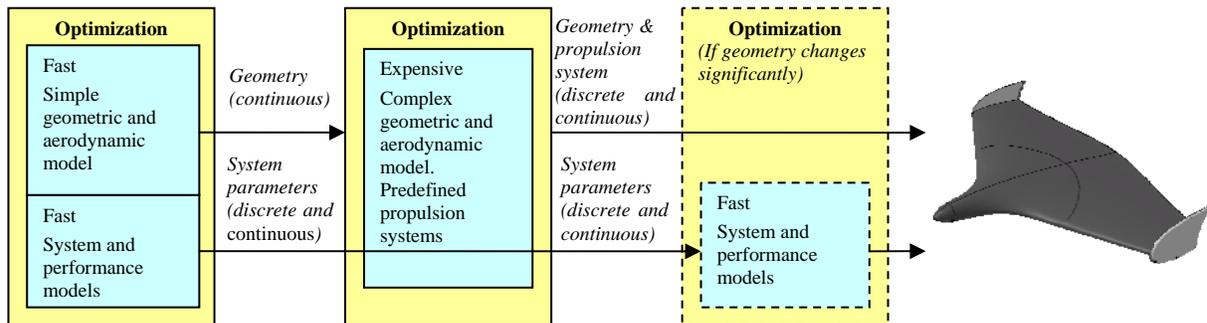


Figure 8. Optimization procedure.

As said above, in previous work, during the optimization the aircrafts total centre of gravity was never checked to stay within reasonable static margin values². Also there was no check that the internal components actually fitted within the outer surface of the MAV. Both these issues have been solved in a fairly simple way. The chosen approach is to implement an internal balance loop that for each function evaluation tries to balance the MAV. This lies better in line with the “button click“ automated design, but it could be a bit difficult to achieve since it may

require many function call to the CATIA model. This would slow down each iteration step even further. To avoid incurring in such problems, only the external side of the component boxes are considered, as explained in chapter II.

Thanks to the algorithms available in modeFRONTIER, in this work it was possible to use a multi-objective genetic algorithm (MOGA II¹¹). This allowed studying the problem without being required to synthesize all objectives in one single objective function or goal. The result of a multi-objective optimization is not a single optimal design, but it rather is a curve where all dominant designs are located. A design is dominant if there aren't any improvements that can be made to any of its characteristics without worsening at least another one. In this work it was chosen to study the relationship between weight (W) and endurance (E) of the vehicle. The objective function of the optimization was set as following:

$$\max \left\{ \left(\frac{W_{REF}}{W} \right)^a \right\} \quad \text{and} \quad \max \left\{ \left(\frac{E}{E_{REF}} \right)^b \right\} \quad (1)$$

$$s.t. : g_i \leq g_{i,REF}$$

Initially the coefficients a and b were set equal to unity, but that led to the optimization algorithm not to spread the solutions evenly on the pareto frontier. This is a fairly common problem when working with multi-objective algorithms. Most designs appeared to converge to the left lower corner of the design space, where low-weight and low-endurance designs are found (Figure 9).

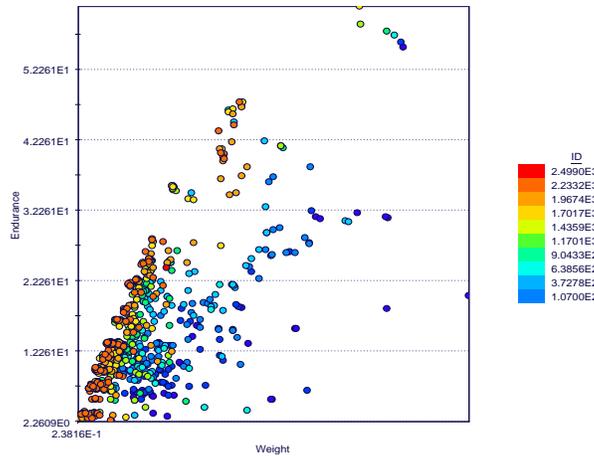


Figure 9. Initial results showed the pareto designs concentrated in the lower left corner

This was believed to be related to the fact that it is easier for the algorithm to reduce weight rather than trying to increase the endurance, since most variable selections have a direct impact on weight, but only a combination of settings will produce an enhancement of endurance performances. Therefore it was later selected to set $a = 1$ and $b = 2$, in order to augment the effects on the endurance. This solution gave positive results.

As explained above, the optimization was run in two phases. The first with a slightly limited set of geometrical variables (due to the simplified methods implied) and with a complete set of variables for the propulsion system. The result from one of these runs is shown in Figure 10a. The number of individual in each generation is 40 and the algorithm was allowed to run 16000 iterations. The different colors of the dots indicates to how early in the optimization the corresponding design was found: old designs tend to blue colors, while the newer they are, the more they tend to red. Figure 10b shows the results from the second phase, where the generation size was unchanged, but the system was allowed 2000 iterations. For this step the number of possible propulsion systems had been limited to 13.

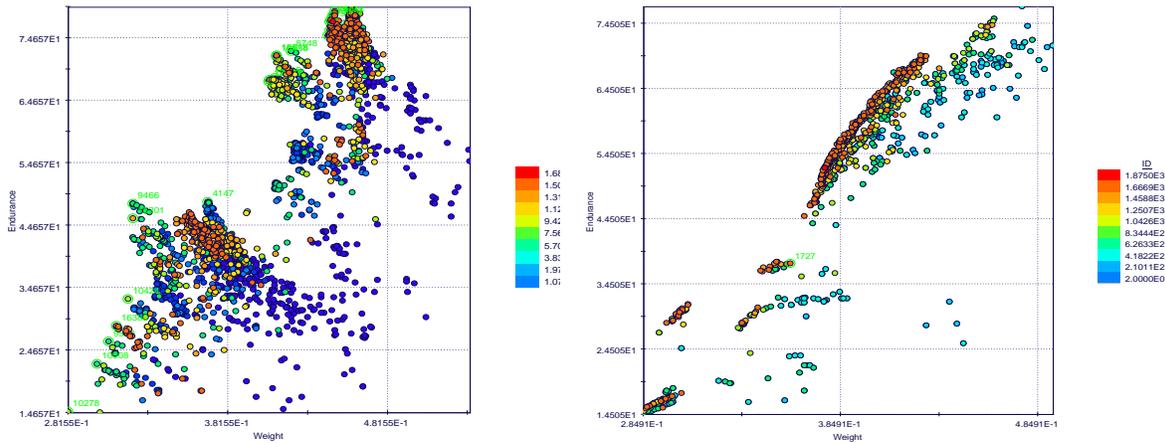


Figure 10. Left: the pareto frontier after the initial optimization (10a) and (right) the final result after the second phase involving both CATIA and PANAIR (10b)

It is interesting to note that in Figure 10b the pareto frontier seems to be composed by several curves. If plotting the same diagram with the color-coding being relative to the chosen propulsion system, it appears that each part of the pareto frontier is actually a front where the propulsion system is kept constant (Figure 11).

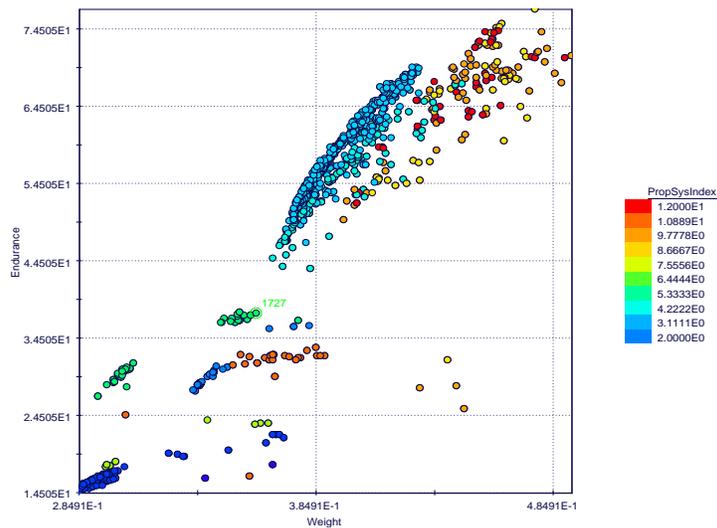


Figure 11. The pareto frontier plotted with colors corresponding to the different propulsion systems

III. Validation

A. Manufacturing methods

To realize design automation in reality, improved methods for airframe manufacturing are needed. An attractive method to small scale manufacturing would be if modern 3D printers could be used to create MAVs. To test this idea and to test the design framework with a real design example, a MAV of 270 mm span was picked from a preprocessed pareto front optimization and printed using a Stratasys “Dimension Elith” FDM¹³ technology 3D printer . The size of 270 mm was picked from limitations in build size of the 3D printer. FDM machines build

components by melting a thin thread of ABS-plastic to thin layers of thickness 0,18-0,35 mm. Two different structure layouts of the 270 mm MAV was produced in this machine (Figure 12). One with fully covered structure, and one with an open rib and spar structure. For both versions a wall thickness of 0.7 mm was utilized for most structural elements.

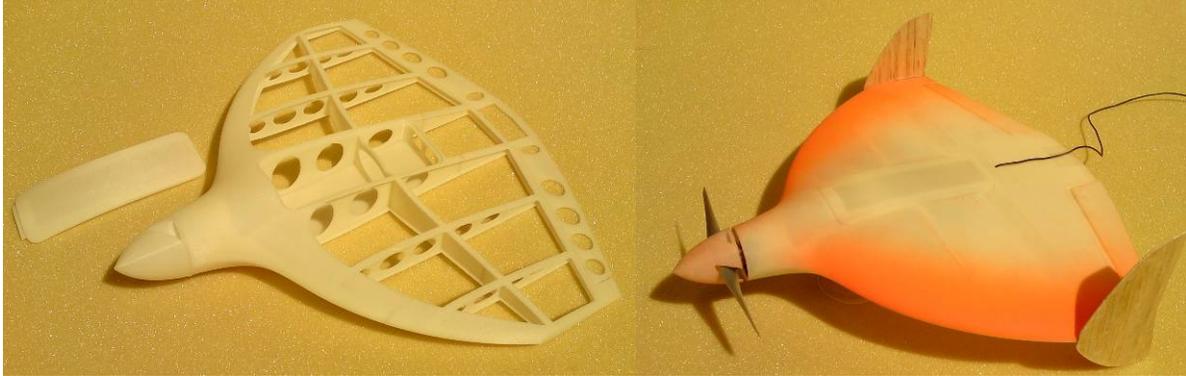


Figure 12. FDM printed MAVs.

The fully covered structure obtained a weight of 90 g while the open structure got 60 g. These weights are heavier than what could be achieved with more traditional manufacturing techniques, but with some tweaking such as reducing wall thickness, optimizing structural layout etc weight could be reduced further. The great benefit is that absolutely no hand craft skills are needed and details such as hatches, cooling channels, control surfaces, attachment points for servos etc could easily be integrated in the design at no cost.

The covered structure version has been installed with equipment for remote controlled flight and has undergone several test flights. Total take-off weight resulted in 175 g. During flight testing the MAV was observed to have an endurance and cruise speed similar to the predictions. The main focus was however to evaluate 3D printing as a way of producing real flying MAVs. Other than verifying that it could be produced and flown, it was of interest to see how robust it would be to typical MAV handling. The MAV has never been crashed, but it has survived a few misshaped take-off launches without damage. The authors' impression is that its strength can not be compared with other building methods, such as composite fabrication, but is sufficient for prototype testing. A disadvantage of the FDM technology is that the material is inhomogeneous in strength. While it is strong in x- and y-axis, it is brittle in z-axis (coordinate system of the printer). If its printing method would result in homogeneous strength it would be easier to save weight and getting a better robustness. For this reason other 3D printing systems than FDM may be better suited. Although 3D printing MAVs is an exciting possibility, it can not today compete with building methods such as composites in terms of strength and low weight, but it is well suited for producing demonstrators or smaller production batches where it is hard to motivate the expensive tooling needed for industrial production. With the rapid evolving technology, and reduction in cost, of 3D printing systems (or direct digital manufacturing, DDM), it will likely be a possible method for producing MAVs in the future. In addition to be a convenient way of producing prototypes, the introduction of DDM technology as such, also has some interesting implications for future designs. With DDM, the cost of geometric complexity goes away, since the cost is only proportional to the weight of the prototype. This means that, if found useful, more complex geometries can be exploited. In general, it can be expected that designs produced for DDM will move closer to biological systems in appearance, and in function.

B. Propulsion system validation

When working with MAV optimization it has become evident that the propulsion system has the highest influence on the performance. There seem to be more to gain on optimizing the propulsion system than fine tuning the optimal aerodynamics layout. The mathematical models that have been used for propulsion system modeling are fairly straight forward and commonly used. There is however little information on how accurate these models are. Also it is interesting to study approximately how trustworthy specifications from different manufacturers are. The database of propulsion system components, used in the design framework, is entirely based on information from manufacturers. For instance electric motor manufacturers usually supply information about their motors performance using the common motor parameters K_v , I_o and R_m . This data is used in the optimization, but the question is if all motor manufacturers define their motor parameters in the same way.

In order to validate the models a test bench has been built (Figure 13). The test bench is built around a set of carefully arranged strain gauges. A 10 mm diameter tube of 0,1 mm wall thickness is used for torque strain measurement. This tube is connected in series with a beam type load cell for thrust measurement. A Medusa Products “Power Analyzer”¹⁴ is used for measuring the electric parameters to the motor. In total the following parameters are monitored: supply voltage, motor current, motor *rpm*, thrust, motor output torque, PWM controller signal, and motor surface temperature. Much effort has been put into achieving high resolution and accuracy in the measurements. The rig can be used to measure battery performance, motor characteristics, motor controller characteristics, as well as propeller characteristics. Validating the models for these components is an ongoing task. In this paper the motor model will be studied more closely.



Figure 13. Propulsion system test rig.

1. Motor

The ideal motor is said to have its *rpm* proportional to voltage and its torque proportional to current. In reality, due to internal losses in the motor, torque and *rpm* are coupled and an increasing torque, naturally, decreases *rpm*. These characteristics are described with the classical motor parameters K_v (*rpm* proportionality constant), I_o (zero load current) and R_m (motor internal resistance). See Ref. 1 for additional explanation of the model. These motor constants have traditionally been used to characterize conventional brushed motors, but are today also used with brushless 3-phase motors. For brushless motors it makes less sense to use the designation K_v , I_o and R_m as the parameters are not completely representing the electrical properties originally intended. Nevertheless the behavior of the motor, in terms of the shape of its performance curves are said to be the same, hence using the same parameters and model.

A motor that often score well in the optimization trials is the AstroFlight 010, but it is also a motor that the authors have experience with and have noticed to have a performance lower than the theoretical predictions. As an example this motor has been studied in the test bench.

a: Constant supply voltage, varying load

With a constant supply voltage the motor model says that the *rpm* should drop linearly with load and that the power output is increasing following a second order polynomial curve. The general shape of these functions is displayed in Figure 14. By testing a series of propellers of varying size (3x3 to 7x5) different loads were put to the AstroFlight 010 motor, while input power was kept constant at 10V. Figure 15 shows the results of the tests for both *rpm* and power. Each dot represents a tested propeller. As can be seen, there is a large difference between theoretical results and motor model. The *rpm* curve is not linear but drops with increasing current. The power curve also deviates drastically with increasing current. Although the agreement between real motor and the modeled motor using known motor constants is quite inaccurate the general behavior of the curves are not too far from the general behavior of the motor model. By doing a curve fit on the test values it was possible to establish new motor constants that very well reflected the motor's behavior even if the accuracy decreases at the higher current levels. It should be mentioned that the two last propellers in Figure 15 (two last dots to the right) are clearly too much load for the AstroFlight 010 as the motor is operated beyond its point of maximum output power. On this side of the power curve the accuracy of the model is of lesser interest since it is a region not interesting for optimization due to poor efficiency. Nevertheless it is important that the model captures the peak of the maximum power.

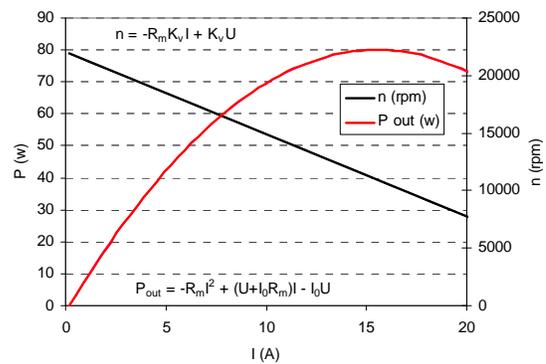


Figure 14. General characteristics of motor model

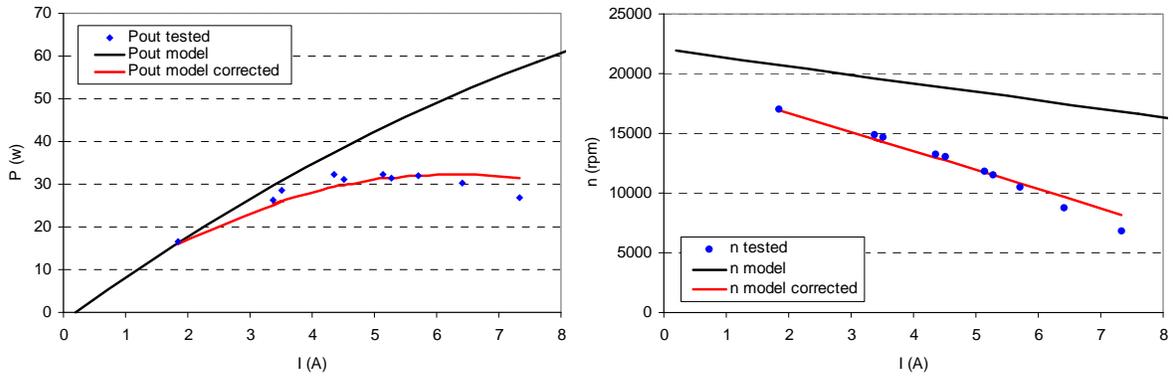


Figure 15. Test results at 10V input voltage and varying load.

b: Varying input voltage, known load

The curves in Figure 15 are useful for determining motor constants, but a more representative way to illustrate the motor characteristics would be to look at a test run with a known load. Figure 16 shows how motor input power relates to *rpm* of an APC 5.25x4.75 propeller as the input power is varied. First a test run is made to record real data. This data is then used to simulate the same test with the theoretical model. First with original motor constants, and then with motor constants established in (a). Again the original motor constants clearly under predict the power needed to turn the propeller. The corrected motor constants come closer and are reasonably accurate, although they are a compromise to agree with motor data over the entire span

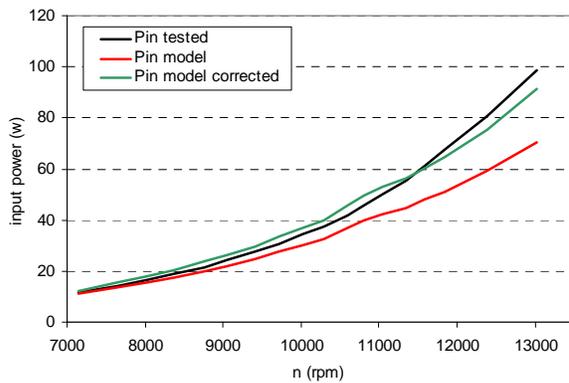


Figure 16. Motor Characteristics with APC 5.25x4.75 propeller

c: Temperature effects

In (a) and (b) it is evident that at higher power outputs the theoretical model deviates from the real test values. Not even the corrected motor constants managed to mimic the motor behavior in this range. This is likely due to increasing temperature as power increases. To demonstrate the effect of temperature the motor was ran at a fixed power input while the efficiency was monitored over time. The tests were made at two different power levels, 30W and 60W. 60W was chosen as it corresponds closely to the point of maximum output power in Figure 15. Results are given in Figure 17. At the 30W level the temperature rise is moderate and does not affect efficiency much but at 60W the effect is significant. The AstroFlight 010 motor is labeled for up to 70W input power, but for reasonable efficiency it appears it should not be used for over 40W. In the Figure 16 the temperature curves are the motor's surface temperature and should only be seen as an illustration of the temperature trends. The internal temperature is much higher as can be seen from the temperature rise when the motor is turned off and the effect of prop wash cooling is cancelled.

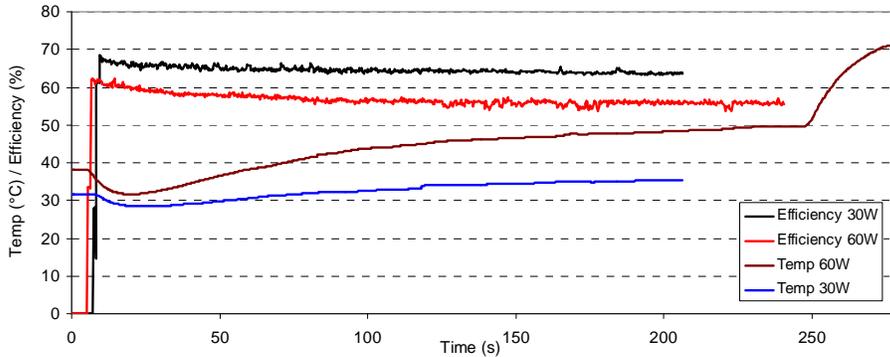


Figure 17. Effect of temperature on motor efficiency

2. *Motor controller efficiency*

On all the tests made in (a), (b) and (c) the motor controller was set to 100% power output hence eliminating any losses depending on the PWM controllers characteristics related to duty cycle length. Instead input Voltage was regulated using an adjustable DC power supply. In flight the power is regulated by adjusting the duty cycle length on the motor controller, thus it is important to map its effect on controller efficiency. To exactly measure the efficiency of the motor controller is very complex, but one way to illustrate the effect of duty cycle length is to compare the efficiency curves over output power for a case when duty cycle is fixed at 100% with varying input voltage and the case when input voltage is fixed but with varied duty cycle length. The top curve in Figure 18a displays the efficiency of the motor and motor controller combination when input voltage is varied from 5 to 10 V at 100% duty cycle. The 5 volt limit comes from that it is the lowest input voltage at which the motor controller can operate. The lower curve represents the same efficiency curve but when the input voltage is fixed at 10V and the controller's duty cycle length is varied between 0 and 100%. If the upper line in 18a is extended for results below 5v, using the motor model, an approximate shape of the controller efficiency curve can be extracted for the entire duty cycle range. This is shown in Figure 18b. Evidently the duty cycle length plays a large effect on total efficiency. Initial comparison with the motor controller model used in Ref. 1 shows large deviations at lower duty cycle values. More experiments are needed to validate the model.

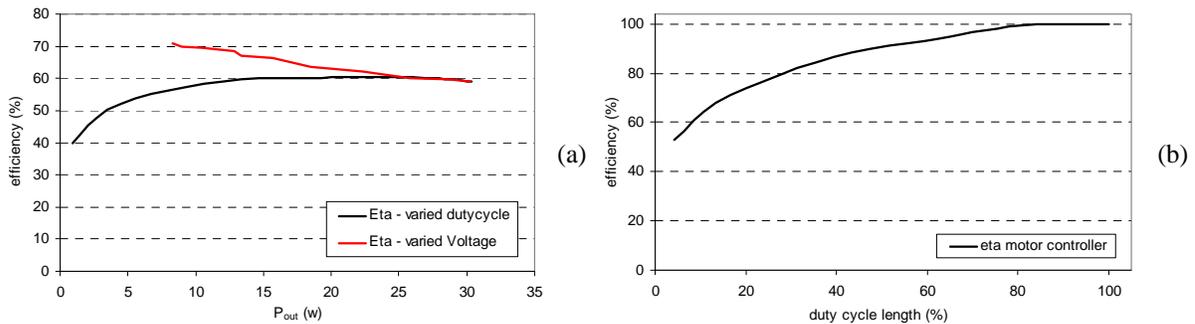


Figure 18. Effect of motor controller duty cycle length

3. *Propulsion system testing summary*

The above explained testing of the AstroFlight 010 motor does not provide enough statistical data to draw any definitive conclusions, but to sum it up it appears that the classical motor model is valid for lower power outputs where temperature effects are low. This is acceptable since the range where motors run most efficiently does not generate much heat and this is also the range interesting for optimizations. The accuracy for a specific motor is completely depending on with what accuracy the motor constants are defined, and there is no standardized method of doing this, consequently the manufacturer's data can not be fully trusted. The best way to determine motor constants for brushless motors appear to be from practical testing where curve fitting is done for the region of interest. The motor constants will then be representing coefficients in a polynomial curve rather than the electrical properties they originally were defined to represent (resistance, zero load current), hence K_v , R_m and I_0 could as well be called A, B and C.

In the tests above the timing of the motor controller is never changed. A fact that could explain the large difference from the manufacturer is that a different controller timing is used, as motor constants are depending on timing used or even what brand of motor controller is used. More testing needs to be made to study the effect of the motor controller on different types of motors.

IV. Discussions and Future Work

It has been proven that full design automation is indeed possible for Micro Aerial Vehicles. The use of multi objective pareto front optimization has proven powerful as it allows the user to preprocess optimal designs for a wide range of scenarios. To completely close the loop, the output of the optimization is sent to a 3D printing machine that manufactures the aircraft. The manufactured prototype can be used, not only as a product in its own right, but also to calibrate the models.

Initial flight tests have also been carried out with a 3D printed MAV. From the experience gained so far, it seems that the weight does not become as low as it would if the aircraft would be built using composites, but the finished aircraft performed as expected in terms of endurance and cruise speed. Robustness was also sufficient to withstand normal handling, missed take-offs and some harder landings as well.

Finally initial validation of the models of the propulsion system components has been presented. Since properly selected components have the largest impact on performances, it was necessary to control how well the commonly adopted models represent the real performances. Initial results indicates that there can be discrepancies between the data published by manufacturers and the real values. The motor model itself, however, has proven to give good validity for the typical power range used during optimizations. Additional validation of models used will be a large part of the future work. More MAVs will be 3D printed and systematically analyzed to benchmark the aerodynamic analysis. The propulsions system validation will focus at testing a larger variety of motors to see how representative the results in this paper are. Validating the propeller model has been completely left out in this paper, but plans are to use the testbench (Figure 13) to also perform propeller wind tunnel testing. In future work a more systematic approach will be used to further investigate the impact of uncertainty in parameters and models, using sensitivity analysis as previously demonstrated by Krus¹⁸.

V. Conclusions

This paper presented a design framework for Micro or Mini Aerial Vehicles where a automated design process is set up, that includes prototyping through 3D-printing, resulting in a flyable prototype. The presented framework only requires the user to introduce initial mission requirements, the system will optimize both the propulsion system and the geometrical layout to best meets the requirements. For analysis of a multidisciplinary system that is used in an optimization loop, it is necessary to have models of all the relevant subsystems, and it is very important that the models have the right balance of complexity and accuracy. In this paper, this has been discussed, and a balanced system model has been produced, that include models with a fidelity that can be justified with respect to the level of uncertainty inherent in e.g. low Reynolds number aerodynamics and, the propulsion system. Testing and validation of real propulsion components was used in this step. To connect all different software modeFRONTIER was tested as an alternative to in-house VBA scripts. The introduction of the design environment software made it easy to use e.g. a multi-objective genetic algorithm. This made possible to carry out more advanced optimizations, compared to what had previously been done.

References

¹Lundström, D., Krus, P., "Micro Aerial Vehicle Design Optimization Using Mixed Discrete and Continuous Variables", 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Sept. 2006, Portsmouth, VA, USA

²Lundström, D., Amadori, K., Krus, P., "Distributed Framework for MAV Design Automation", 46th AIAA Aerospace Sciences Meeting and Exhibit, Jan. 2008, Reno, NV, USA

³Amadori, K., Jouannet, C., Krus, P., "Use of Panel Code Modelling in a Framework for Aircraft Concept Optimization", 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Sept. 2006, Portsmouth, VA, USA

⁴Amadori, K., Johansson, B., Krus, P., "Using CAD Tools and Aerodynamic Codes in a Distributed Conceptual Design Framework", 45th AIAA Aerospace Sciences Meeting and Exhibit, Jan. 2007, Reno, NV, USA

⁵Hoerner, F., "Fluid Dynamic Drag: Practical Information on Aerodynamic Drag and Hydrodynamic Resistance", Published by the Author, 1965.

⁶Cosyn P., Vierendeels J., “*Numerical Investigation of Low-Aspect-Ratio Wings at Low Reynolds Numbers*”, Journal of Aircraft, Vol. 43, No. 3, May-June 2006

⁷Lupo S., Nyberg H., Karlsson A., Mohseni K., “*Xwing – A 3D Viscous Design tool for Wings*”, 46th AIAA Aerospace sciences Meeting and Exhibit, January 7-10 2008, Reno, Nevada, USA

⁸X-foil: web.mit.edu/drela/Public/web/xfoil/

⁹Jouannet, C., Silva, S.E.R., Krus, P., “*Use Of CAD For Weight Estimation In Aircraft Conceptual Design*”, 24th International Congress Of The Aeronautical Sciences, Linköping University, Linköping, Sweden, 2004.

¹⁰Erikson, L.L., “*Panel Methods – An Introduction*”, NASA Technical Paper 2995, 1990

¹¹Poles, S., “*MOGA II: An Improved Multi-Objective Genetic Algorithm*”, ESTECO Technical Report, Dec. 2004

¹²ESTECO: www.esteco.com

¹³Stratasys: www.stratasys.com

¹⁴Medusa Products: www.medusaproducts.com

¹⁵Krus, P., “*Aircraft System Optimization and Analysis for Traceability in Design*”, 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Sept. 2006, Portsmouth, VA, USA.