Evaluation of Automatically Designed Micro Air Vehicles and Flight Testing

D. Lundström*, K. Amadori† and P. Krus‡
Linköping University, Linköping, 581 83, Sweden

The presented work is centered on the evaluation of Micro or Mini Air Vehicles (MAV) that have been automatically designed and manufactured. An in-house developed design framework uses several coupled computer software's to generate the geometric design in CAD, a well as list of off the shelf components for the propulsion system, and computer code for autonomous flight ready to upload in the intended autopilot. The paper describes the experiences made so far regarding automation of the design process and of manufacturing. Furthermore, it presents results from evaluation and analysis of the optimization algorithm and flight testing, and from continuing work with the framework to achieve deeper understanding of the process and to fine-tune the design automation performance. The flight data is correlated to the predicted performances to validate the models and design process.

Nomenclature

α	=	Angle of attack
CAD	=	Computer Aided Design
COTS	=	Commercial Off The Shelf
DDM	=	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
FDM	=	Fused Deposition Modeling
FEM	=	Finite Element Method
I_o	=	Electric motor zero load current
K_v	=	Electric motor Rpm constant
PWM	=	Pulse Width Modeling
R_m	=	Electric motor internal resistance
Rpm	=	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 are increasingly used in the design of manned aircraft, although 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.

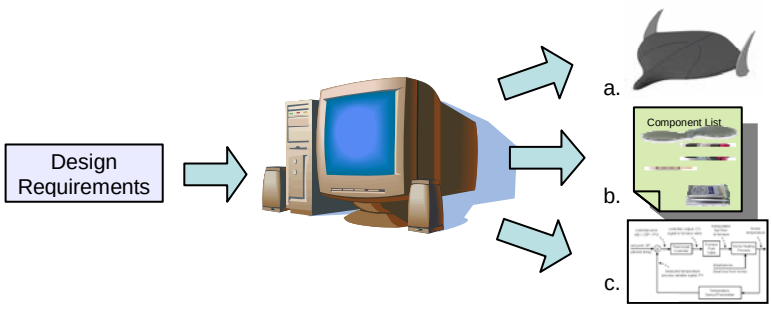

Figure 1. MAV Design automation .

From a mission requirement a requirement 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, as well as 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 authors[1,13,14], 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 study[13] this optimization method was scaled up to a design framework including CAD software and panel code for aerodynamic evaluation.

The MAV must have a large enough volume to accommodate it is intended components, as well as balance with a proper stability margin. In this paper both the balancing and volume criteria has been taken care of. The optimization procedure has been expanded 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.13 and Ref.14. 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.

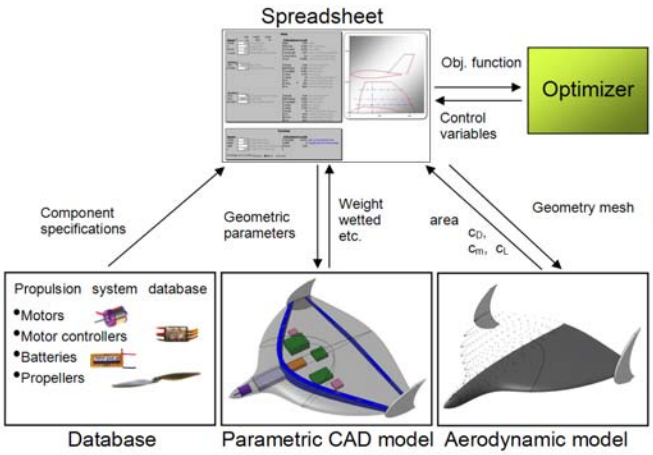

Figure 2. The design framework.

In this work the design framework has been implemented with modeFRONTIER v.4.1.0[8]. 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.

For the presented work, modeFRONTIER could be run on a Dell Precision PW390 64bits workstation instead of the Dell Precision M2400 laptop previously used. The cycle-time registered during the optimization trials was comparable with that from Ref. 13, between 70 and 90 seconds. A very 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.

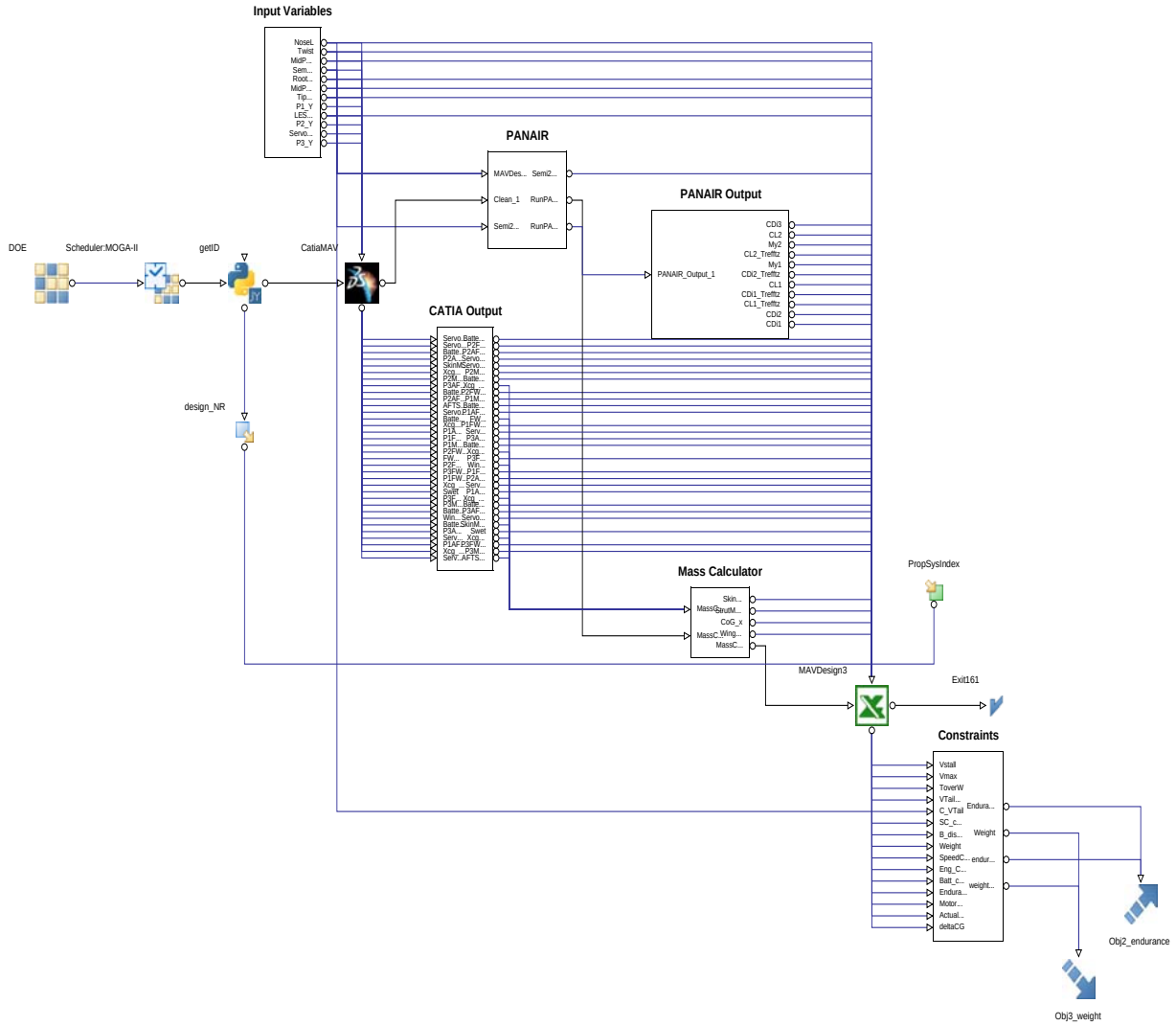

Figure 3. The design framework process as pictured in modeFRONTIER.

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 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 (121)
- 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 weight[9] and mass distribution that can be used for both performance prediction as well as flight simulations.

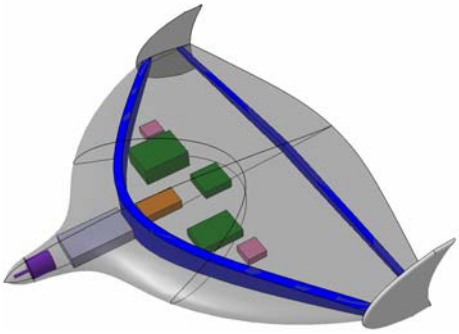

Figure 4. The parametric MAV CAD model.

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 5). During optimization an internal balance loop is executed, for each function evaluation, which automatically seeks to balance the aircraft within a predefined stability constraint.

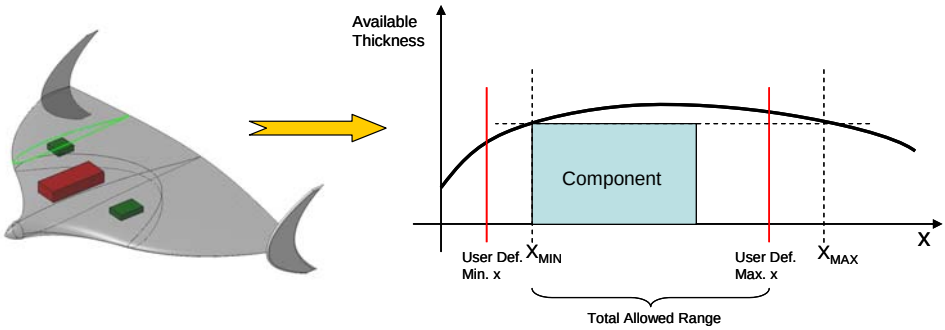

Figure 5. The balancing and fitting tool.

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 speeds[7]. As pointed out by Amadori et. al.[1,2], 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 6). 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 ($c_{di\text{-}cruise}$) in cruise condition through interpolation of the c_{Di}-curve.

The panel code described gives the induced drag but the parasitic drag has been calculated using Prandtl and VonKarman low Reynolds number flat plate skin friction equations[9] corrected with form factor for the pressure drag.

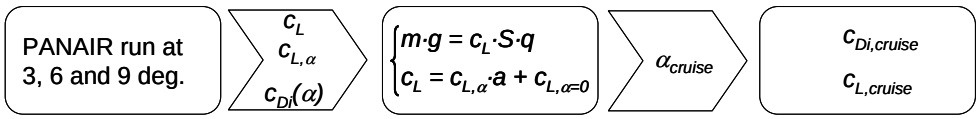

Figure 6. Each configuration is analyzed in two steps.

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 in to consideration. Inspiration comes from Cosyn[5] and Lupo et. al.[15]. 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-foil[20] 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.

III. Design Optimization

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. In this section results will be presented from the optimization performance analysis and it will be explained how the optimization strategy has been modified to refine the framework.

A. General Considerations

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 a much larger number of variables to be taken into account. Secondly, it was noted that when components were chosen from the full database, a very large number of design resulted in unfeasible designs, 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-optimal 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 7.

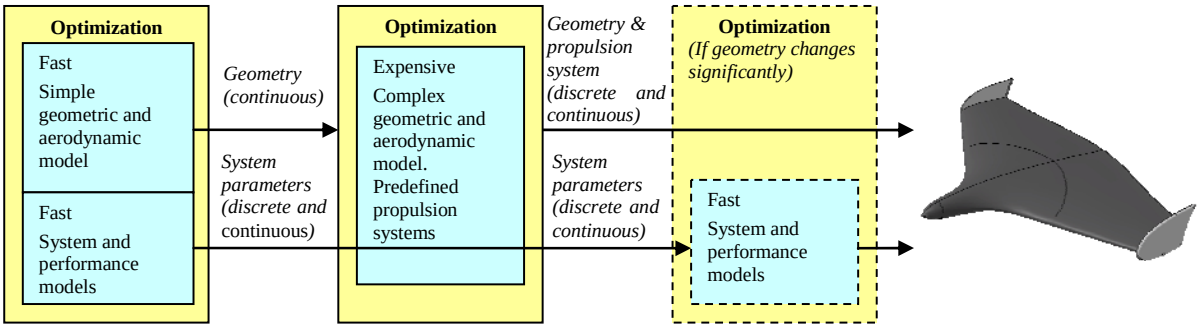

Figure 7. Optimization procedure.

B. Optimization Performance Analysis

Thanks to the algorithms available in modeFRONTIER, it was possible to use a multi-objective genetic algorithm (MOGA-II[17]). This allowed for studying the problem without being required to synthesize all objectives into one single objective function. 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 are not any improvements that can be made to any of its characteristics without degrading at least one other characteristics. In this work it was chosen to study the relationship between weight (W) and endurance (E) of the vehicle. Initially[13] the objective function of the optimization was set as following:

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

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

The coefficients a and b were set equal to *one* and *two* respectively, in order to try to counter the algorithm's tendency to lock itself into the lower-left region of the design space, where low-weight and low-endurance designs are found (Figure 8). Clearly the optimizer struggled to find feasible high-endurance and higher-weight solutions, leaving the upper-right region of the design space relatively uninvestigated. Several combinations of coefficients a and b were tried, but no one presented sufficient repeatability nor stable performances. Therefore a completely different approach was selected (see next paragraph for details).

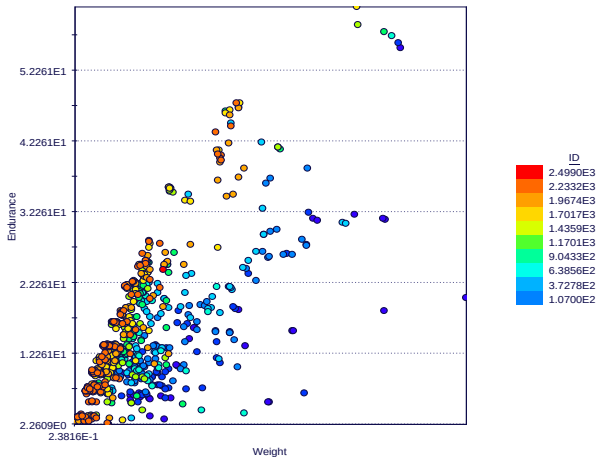

Figure 8. Early results showed the pareto designs concentrated in the lower left corner.

To study how MOGA-II operates to satisfy the different objectives a test was carried out. The two goals (minimizing weight and maximizing endurance) were weighted together using the following expression:

$$obj = \max\left(K_1 \cdot \frac{E}{E_{REF}} - K_2 \cdot \frac{W_{REF}}{W} \right) \tag{2}$$

The weights K_1 and K_2 were then varied from 1 to 0 and from 0 to 1 respectively; for each combination the optimization was then tested, evaluating several settings. Table 1 below shows a summary of the findings.

The ratio explains the values used for the weights K_1 and K_2: for example a ratio 08:03 means that $K_1 = 0.7$ and $K_2 = 0.3$. As it can be seen MOGA-II was tested running 5000 or 10000 iterations per single objective with different combinations of generation size and number of generations. It can be seen that in these conditions the algorithm was quite successful in finding good feasible solutions even in the upper-right region of the design space (see also Figure 9 to Figure 11). Each optimal solution of each one of the different ratio-combinations in the table above is in fact points of the pareto front. The issue is that it is extremely computationally expensive to need to run thousands of iterations per each single point front. So it is necessary to find a strategy that guarantees confident results still allowing to run the multi-objective analysis all at once with a reasonable number of iterations required. One important parameter that was found having a sensible effect on general performances is the so called "*Directional Cross-Over*". While being a genetic algorithm, MOGA-II incorporates methods that aim at speeding up search and convergence. Among them there is the Directional Cross-Over which assumes that a direction of improvement can be found comparing the fitness values of different designs. The problem is that the first optimization step (see Figure 7) includes both continues and discrete variables. The discrete ones represent the database indexes of several components that are needed in the aircraft and for which it is hardly possible to define an ordering. Hence it is not correct to assume a direction of improvement.

		MOGA-II			MOGA-II			MOGA-II		
		10000 iterations Directional X-over = 0,5 Pareto Point			5000 iterations Directional X-over = 0,0 Pareto Point			10000 iterations Directional X-over = 0,0 Pareto Point		
Nr	Ratio	ID	Endurance	Weight	ID	Endurance	Weight	ID	Endurance	Weight
0	10:00	9109	104,51	430	3864	78,64	400	8865	99,10	470
1	09:01	443	76,40	490	3385	105,35	420	9010	105,25	420
2	08:02	344	89,21	490	4213	82,70	420	9625	57,83	360
3	07:03	5124	90,84	500	4355	103,33	440	7877	101,16	470
4	06:04	8075	105,63	420	3971	92,50	500	8854	81,97	420
5	05:05	9305	79,36	410	1304	95,60	460	6255	81,94	360
6	04:06	5246	79,49	360	4477	99,20	430	7695	102,83	420
7	03:07	9784	46,50	300	4170	62,53	340	6856	86,16	380
8	02:08	7790	85,57	380	4078	55,33	320	9726	18,03	270
9	01:09	7265	21,00	280	4824	16,80	270	9950	42,26	320
10	00:10	5485	18,10	280	4596	14,75	300	7169	15,29	270
		200 generations *50 individuals x generation* *New starting gen. each time* *SOBOL*			*100 generations* *50 individuals x generation* *Same starting gen. each time* *SOBOL*			*100 generations* *100 individuals x generation* *Same starting gen. each time* *SOBOL*		

Table 1. Summary of evaluation runs for MOGA-II

Another optimization algorithm called "*Evolution Strategy*" was also tested. As for GA, Evolution Strategies (ES) are optimization technique based on the concepts of adaptation and evolution, but their main search procedure is a smart mutation operator. For single-objective optimization problems ES require only one individual per generation. Hence they could potentially be more efficient than GA, since they should converge faster. Table 2 below shows the results from the trials carried out.

		Evolution Strategy			Evolution Strategy		
		2000 iterations Nr. of Offsprings = 4 Pareto Point			2000 iterations Nr. of Offsprings = 8 Pareto Point		
Nr	Ratio	ID	Endurance	Weight	ID	Endurance	Weight
0	10:00	551	43,54	380	1040	44,62	430
1	09:01	551	43,54	380	1505	51,26	410
2	08:02	551	43,54	380	1505	51,26	410
3	07:03	551	43,54	380	1012	44,23	430
4	06:04	551	43,54	380	899	44,12	420
5	05:05	551	43,54	380	899	44,12	420
6	04:06	1421	41,69	360	899	44,12	420
7	03:07	1421	41,69	360	616	90,33	470
8	02:08	1453	14,66	320	616	90,33	470
9	01:09	1453	14,66	320	955	27,35	370
10	00:10	1453	14,66	320	955	27,35	370
		500 generations *1 individuals x generation* *Same starting gen. each time* *SOBOL* *Initial Stepsize = 0,1*			*250 generations* *1 individuals x generation* *Same starting gen. each time* *SOBOL* *Initial Stepsize = 0,05*		

Table 2. Summary of test results with Evolution Strategies algorithm.

As the table shows the Evolution Strategy was completely unsuccessful at finding the pareto front points. It can be seen that the algorithm keeps on finding exactly the same solution for several different combinations of the weight and endurance weight factors. Therefore the idea of using ES was discarded.

Figure 9 and Figure 10 hereby show a summary of the weight and endurance values for the three tests conducted with the MOGA-II algorithm. These are the values corresponding to the pareto front points found during the tests reported in Table 1. The two diagrams also include an average curve that approximates all points with a third-order polynomial. Apart from a slight dip in the weight curve for the aircraft that are supposed to weight the most, the two curves present the right shape. It should be pointed out that the mentioned dip is actually accentuated by the approximation with the polynomial curve.

Finally the test results were used to create three pareto fronts in an Endurance Vs. Weight diagram (Figure 11) and compared with a multi-objective test run with the following settings (Table 3):

MOGA-II
50000 iterations Directional X-over = 0,0 Pareto Front
500 generations 100 individuals x generation SOBOL

Table 3. Multi-objective test run settings with MOGA-II.

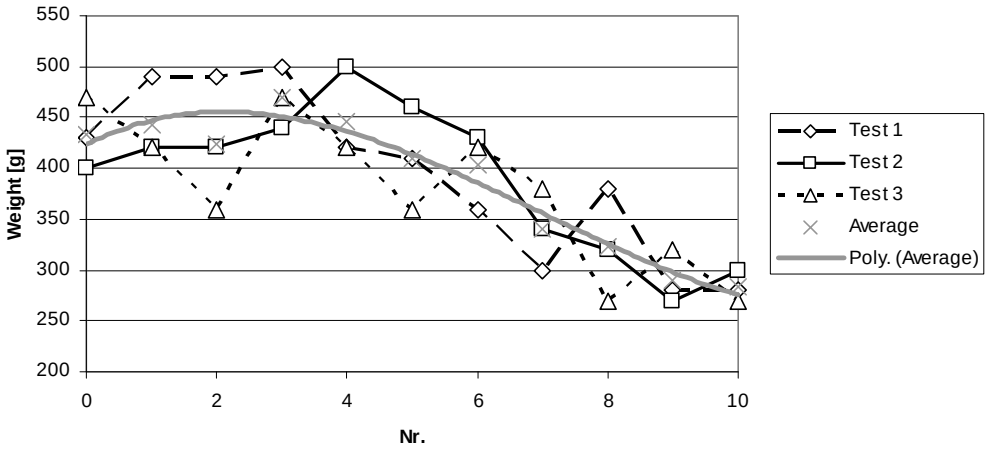

Figure 9. Results from the test runs with the MOGA II algorithm: weight variation is showed as function of different weight factors combinations.

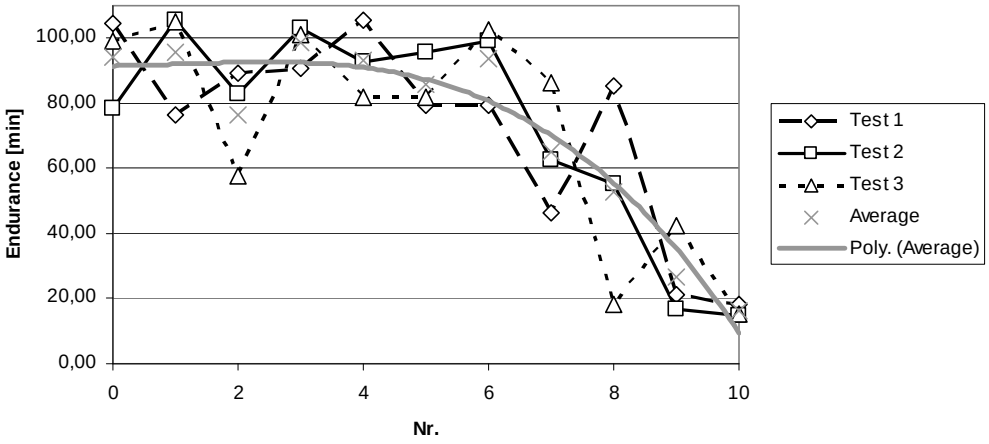

Figure 10. Results from the test runs with the MOGA II algorithm: endurance variation is showed as function of different weight factors combinations.

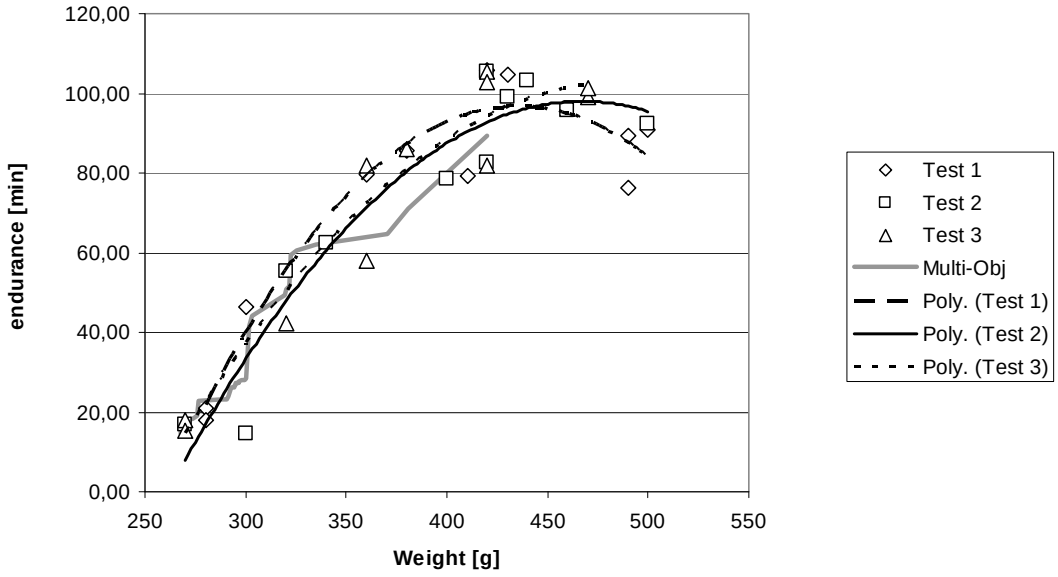

Figure 11. The pareto fronts resulting from the three test runs compared with a multi-objective optimization front.

The three fronts are obtained approximating the single-objective pareto points with a second-order polynomial. From the comparison between the three test runs it can be seen that the differences are not very large. Nevertheless it seems that in "Test 1" the algorithm struggles in the upper-right region, where aircraft are heavier and fly longer. On the other hand it draws a more advanced front in the other regions. "Test 2" and "Test 3" are very similar, as expected. The algorithm settings are the same but "Test 3" is run with population size of 100 individuals rather than 50 in "Test 2". The larger population size, given a fixed number of generations, implies double the number of iterations. Thus it is not surprising that "Test 3" shows slightly better results in the most difficult region, namely the high-weight high-endurance end of the front. It can be also noted that the multi-objective curve fits reasonably well in the lower part of the front, but is totally lacking any points in the upper region.

In the following paragraph a strategy to try to overcome this issue is presented.

C. Optimization Strategy Refinements

To force a thorough search of the higher region of the allowed space, the constraint on the total flight endurance was changed dynamically during the optimization itself, according to the following relationship:

$$E_{MIN}^{Actual} = E_{MAX}^{Absolute} - \frac{E_{MAX}^{Absolute} - E_{MIN}^{Absolute}}{ID_{MAX} - ID_{REF}} \cdot \left(ID - ID_{REF} \right) \tag{3}$$

where:

- E_{MIN}^{Actual} is the minimum allowed endurance value at a given time during the optimization
- $E_{MAX}^{Absolute}$ is the maximum value considered for the minimum allowed endurance
- $E_{MIN}^{Absolute}$ is the minimum value considered for the minimum allowed endurance
- ID_{MAX} is the final iteration number planned for the optimization
- ID_{REF} is the reference iteration number from which to start relaxing the endurance constraint
- ID is the iteration number at a given time during the optimization

The introduction of such a formulation is that the endurance constraint is kept at a maximum value ($E_{MAX}^{Absolute}$) in the beginning, forcing the optimization algorithm to concentrate on solutions lying in the upper-right region. Then, from a predefined point in the optimization (ID_{REF}) the constraint start to relax and the algorithm starts to move towards the lower-left region. The real constraint is that the endurance value should not be less than $E_{MIN}^{Absolute}$, so that the solutions that are discarded at the beginning are in fact feasible. Hence a large tolerance was added to the constraint that was formulated as:

$$\frac{E}{E_{MIN}^{Actual}} \geq 1 \tag{4}$$

with a tolerance of 0,5, meaning that the ratio of equation (4) could be as low as 0,5 before the solution becomes unfeasible.

This modification resulted in a much better search performance in the tougher upper-right region of the plot, presenting a higher solution density in that area that was previously very sparsely populated.

IV. Results

As a test case, MAVs were designed to carry out a flight test mission. The payload would then consist of a small USB flight data recorder from Eagle Tree Systems[6]. Figure 12 shows a picture of both components. Their sizes are reported in Table 4.

	Length	Width	Height	Weight
Flight Data Recorder	49 mm	35 mm	22 mm	24 g
GPS Antenna	35 mm	35 mm	11 mm	24 g

Table 4. Payload sizes and weights.

Figure 12. The USB flight data recorder (left) and GPS antenna (right) used as payload for the aircraft design.

The aircraft were optimized around the above listed components, in order to fulfill the requirements listed in Table 5. The manufacturing system for the MAVs is a Stratasys “Dimension Elite” FDM technology 3D printer, as already described and tested[14].

V_{Stall}	30	km/h
V_{Cruise}	60	km/h
Min Endurance	20	min
Max Weight	0,5	kg
T/W	0,6	
Max Wingspan	600	mm
Payload	48	g

Table 5. Design requirements.

With the knowledge gathered from the optimization algorithm test results, it was chosen to use the following optimization settings for the first optimization phase:

- Population size: 100

- Number of Generations: 500
- Probability of Directional Cross-Over: 0,25
- Probability of Mutation: 0,3
- Algorithm type: MOGA-II Adaptive Evolution

For the second phase that involves both CATIA V5 for the geometrical modeling, and PANAIR for the aerodynamic evaluation, the probability of cross-over was set to 0,5 since only one parameter was discrete (i.e. the propulsion system index). Moreover, the population size was diminished from 100 individuals to 24. This is because the pareto front individuals from the first step (that happened to consist of 24 points) were imported as start generation for the second phase. The total amount of iterations for the second optimization phase was set to 2000.

Figure 13 and Figure 14 show the resulting plots in the Endurance Vs. Weight diagrams for the first and second optimization phase respectively. The colors correspond to how recent the designs are. Bluer points are individuals that the algorithm found earlier in the optimization; the more the point move to the red scale, the later they appeared in the plot. In Figure 13 the pareto points have been highlighted. Only the feasible designs are plotted in the first phase result diagram.

It is interesting to observe that the final pareto front is not a monotone curve, but presents successive "bumps" or portions. It was previously pointed out[14] that each of these sections is characterized by having the same propulsion system. Within one of the "bumps" the different performances are determined by changes in the geometry and size of the MAV. This confirmed property of the pareto front emphasizes even more the importance of being able to carry out a detailed and automated optimization of the aircraft. Due to the complex relationship between all variables it would be very hard to find "by hand" the best combination of propulsion system components and geometrical variables for a given mission profile.

In comparison with Figure 8, the two figures below also show how differently the algorithm operates thanks to the modified endurance constraint. In Figure 8 the vast majority of points is located in the lower-left region, while only sparse individuals made it to the more interesting and challenging top-right corner of the plot. In the two latter figures it can be clearly seen how the algorithm is striving to fill the upper-right portion of the diagram and how the distribution of points along the pareto front is more even.

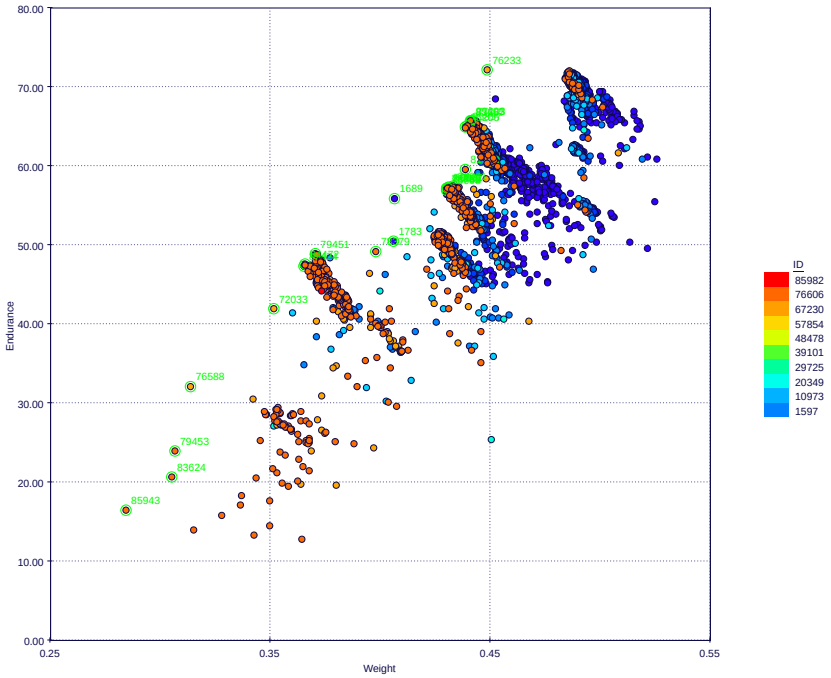

Figure 13. Results of the first optimization step (population size: 100; number of generations: 500).

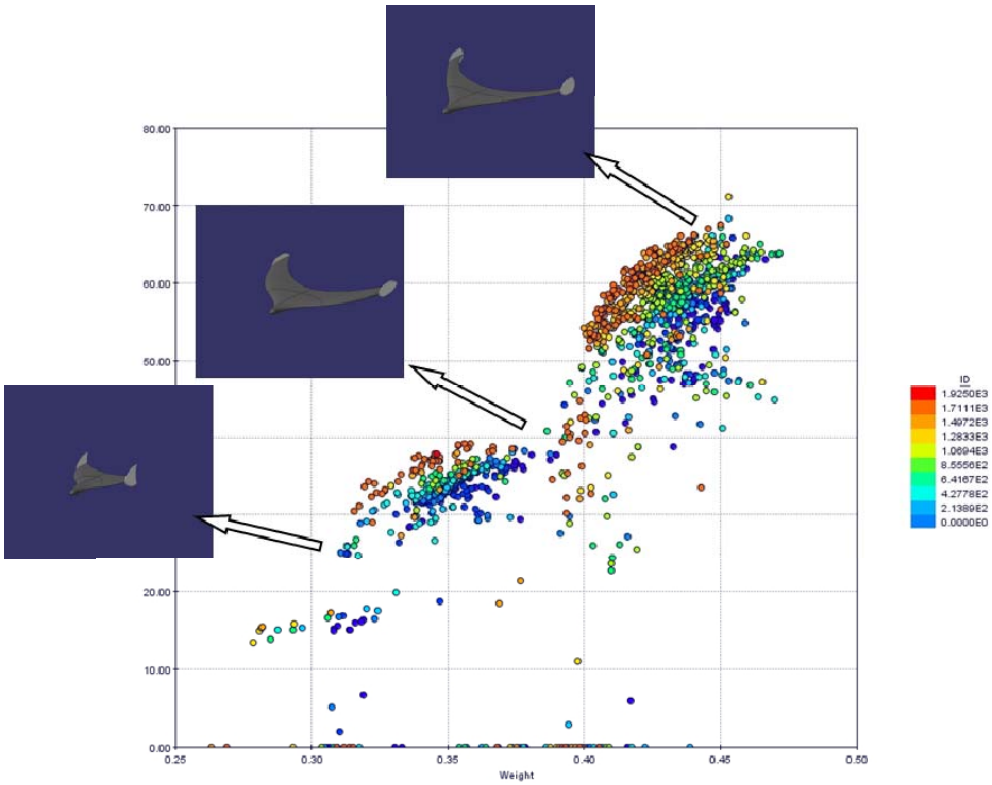

Figure 14. Results of the second optimization step (population size: 24; number of generations: 84).

V. Validation and Flight Testing

The main objective of the flight tests was to measure the overall accuracy of the design framework, which means to be able to measure the aircrafts' weight, endurance, maximum speed and stall speed. The purpose was not analyzing the single components and their mathematical modeling, since a detailed report on that approach and results can be found in Ref. 3. The focus was instead on getting a feeling for how the design tool could capture all the necessary phenomena and predict the overall performances of the MAV.

Some reflections are appropriate here. As in the majority of cases, the aerodynamic characteristics are calculated as if flying in a steady clean air. Considering the size of MAVs and how close to the ground they usually fly, this may be far from being true in reality. Moreover, the aerodynamic effects of the propeller are not considered or modeled at all. Observing a MAV, one can notice that it not unusual for the propeller diameter to reach sizes close to half the aircraft's wingspan. Therefore this may be another issue where model and reality do not match. Finally there are all the discrepancies that have been pointed out in Ref. 14 and Ref. 3: the models used for the propulsion system's components are quite correct but extremely dependent on the quality of the data provided by manufacturers. Unfortunately it appears that this data can not always be trusted.

Figure 15 and Table 6 show the aircraft used for the flight tests and its characteristics. To measure as accurately as possible the performance figures, all flights were carried out only in days with no wind.

Geometrical Specs			Propulsion System Specs	
Root Chord	208 mm		Motor	Turnigy C1822
Tip Chord	56 mm		Battery	FlightPower EVO Light 3s350Mah
Wing Span	270 mm		Propeller	APC 4.5x4.1
LE Sweep	38 Deg		ESC	Turnigy Plush 6A
Twist	1 Deg			
Nose Length	31 mm			
Weight	185 g			

Table 6. Test aircraft geometrical characteristics and propulsion system components list.

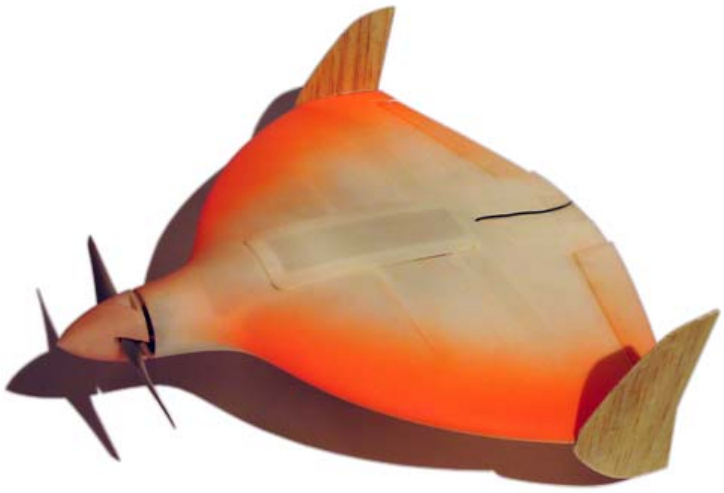

Figure 15. The MAV used for flight testing.

The measurements from the flights provided feedback on the aircraft's maximum speed, the endurance at maximum speed and its weight. Table 7 presents both the predicted values and the measured ones. As can be seen, the agreements between predictions and measurements are very good. It is not surprising that weight predictions are accurate, since the design tool was tuned by means of a scaling factor in order to have a good match between calculations and reality. It is very encouraging that speed and endurance also present such good accuracy. The larger error in the maximum speed prediction could be related to several factors, such as difficulties in precisely measuring the speed in flight and discrepancies between aerodynamic calculations and flight conditions. The authors believe anyway that the error margins are at such a level that other uncertainties that are very difficult to predict or control dominates, and that the design framework proves to be a reliable tool to continue to develop.

		Predicted	Measured	Error
Maximum Speed	**[m/s]**	26,4	25,0	5,3%
Endurance (V_{Max})	**[min]**	6,1	6,0	1,6%
Weight	**[g]**	185	187	1,1%

Table 7. Flight testing results showing agreement between predicted and measured data.

VI. Conclusions and Work In-Progress

At Linköping University a design automation framework for Micro Air Vehicles has been developed, and is being refined. Detailed analysis of the optimization algorithm and its performances have been carried out to find the best settings and conditions in order to speed up design time and ensure high reliability of the result.

The framework has been used to design MAVs for carrying out a series of flight test. These aircraft are optimized around a test equipment package that is used for measuring flight performance. The pareto front diagrams obtained from the optimization trials, confirm the results obtained previously. The final pareto front exhibit a discontinuous shape, being continuous only over limited sections. On each section the propulsion system is constant and only the size and shape of the aircraft is varied.

Different aircraft are being manufactured and equipped for remote controlled flight. They are being test flown and their performances compared to predictions. Initial flight test data reveal very good agreement with the forecasts. The aim of flight testing the automatically designed MAVs is to verify the overall precision of the framework and design process. Total endurance, maximum and stall speed and aircraft weight is measured. Together with in-depth knowledge of the propulsion system modeling quality, the planned flight testing campaign will also tell how well the aerodynamic models are performing.

To get further knowledge about the models used in the framework, the authors have been working on including Tornado[19] in the framework. Then comparisons will be made between Tornado and the already included vortex lattice method and PANAIR. Moreover advanced CFD analysis are also being carried out. Large Eddy Simulation (LES) are applied to compute the flow field around the aircraft.

Finally, Linköpings University has recently acquired a wind tunnel. Modification of the laboratory and installation of the tunnel are currently about to start. Once in place, the tunnel will permit to compare the analytical results to experimental ones. During flight testing, conditions are hard to oversee, while in a wind tunnel it is possible to have relatively good control on them. Thus it will be possible to have even more data with which to compare the analytical predictions.

VII. References

[1]Amadori, 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

[2]Amadori, 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

[3]Amadori, K., Lundström, D., *"Validation of Small Scale Electric Propulsion System Models*" - 48th AIAA Aerospace Sciences Meeting, Jan. 2010, Orlando, FL, USA

[4]ANSYS homepage: www.ansys.com

[5]Cosyn P., Vierendeels J., *"Numerical Investigation of Low-Aspect-Ratio Wings at Low Reynolds Numbers"*, Journal of Aircraft, Vol. 43, No. 3, May-June 2006

[6]Eagle Tree homepage: www.eagletreesystems.com

[7]Erikson, L.L., *"Panel Methods – An Introduction"*, NASA Technical Paper 2995, 1990

[8]ESTECO homepage: www.esteco.com

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

[10]Jouannet, 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

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

[12]Lundström, 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

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

[14]Lundström, D., Amadori, K., *"Automation of Design and Prototyping of Micro Aerial Vehicle*" - 47th AIAA Aerospace Sciences Meeting, Jan. 2009, Orlando, FL, USA

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

[16]Medusa Producs homepage: www.medusaproducs.com

[17]Poles, S., *"MOGA II: An Improved Multi-Objective Genetic Algorithm*", ESTECO Technical Report, Dec. 2004

[18]Stratasys homepage: www.stratasys.com

[19]Tornado homepage: http://www.redhammer.se/tornado/

[20]X-foil: web.mit.edu/drela/Public/web/xfoil/