COVER SHEET

Gonzalez, Luis F. and Damp, Lloyd and Srinivas, Kavita (2006) Aero-structural optimisation of unmanned aerial vehicles using a multi-objective evolutionary algorithm. In Proceedings 2nd Australasian Unmanned Air Vehicles Conference, Melbourne.

AERO-STRUCTURAL OPTIMISATION OF UNMANNED AERIAL VEHICLES USING A MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM

L. F.González[1], L. Damp[2],K. Srinivas[3]

1. Lecturer AMME, University of Sydney, NSW 2006, Australia,
gonzalez@aeromech.uysd.edu.au
2. Postgraduate Student, AMME, University of Sydney, NSW 2006, Australia,
lloyd.damp @aeromech.uysd.edu.au
3. Senior Lecturer AMME, University of Sydney, NSW 2006, Australia gonzalez@aeromech.uysd.edu.au
ragh@aeromech.uysd.edu.au

Abstract

This paper describes the practical application of Hierarchical Asynchronous Parallel Evolutionary Algorithms for Multi-objective and Multidisciplinary Design Optimisation (MDO) of UAV Systems using high fidelity analysis tools. The project looked at the aerodynamics and structure of two production UAV wings and attempted to optimise these wings in isolation to the rest of the vehicle. The two vehicles wings which were optimised were a High Altitude Long Endurance (HALE) UAV similar to the Global Hawk and a Medium Altitude Long Endurance (MALE) UAV similar to the Altair. The optimisations for both vehicles were performed at cruise altitude with MTOW minus 5% fuel and a 2.5g load case. The work was carried out by integrating the current University of Sydney designed Evolutionary Optimiser (HAPMOEA) with Computational Fluid Dynamics (CFD) and Finite Element Analysis (FEA) tools. The variable values computed by APMOEA were subjected to structural and aerodynamic analysis. The aerodynamic analysis computed the pressure loads using a Morino class panel method code named PANAIR. These aerodynamic results were coupled to a FEA code, MSC.Nastran® and the strain and displacement of the wings computed. The fitness were the overall mass of the simulated wing box structure and the inverse of the lift to drag ratio. Furthermore, six penalty functions were added to further penalise genetically inferior wings and force the optimiser to not pass on their genetic material. The results indicate that given the initial assumptions made on all the aerodynamic and structural properties of the HALE and MALE wings, a reduction in mass and drag is possible through the use of the HAPMOEA code. Even though a reduced number of evaluations were performed, weight and drag reductions of between 10 and 20 percent were easy to achieve and indicate that the wings of both vehicles can be optimised.

Biography

L. F. Gonzalez is a Lecturer at the School of Aerospace Mechanical and Mechatronic Engineering, The University of Sydney. His research interest focuses in aerodynamic and UAV/UAS optimisation..

L. Damp is a Masters student at the School of Aerospace Mechanical and Mechatronic Engineering at The University of Sydney.

K. Srinivas is a Senior Lecturer at the School of Aerospace Mechanical and Mechatronic Engineering, The University of Sydney. His research interest focuses in aerodynamic and biomedical optimisation.

Introduction

Optimisation is a sensitive integrated part of global aeronautical design as small changes in geometry gain in structural weight and reduction of aerodynamic drag. In aerospace engineering design and optimisation the engineer is usually presented with a problem which involves not only one single objective but also numerous objectives and multi-physics environments. Hence a systematic approach, which accounts for the interaction and trade-offs between multiple objectives, variables, constraints and disciplines, is required. This approach is called Multi-objective (MO) and Multidisciplinary Design Optimisation (MDO).

Capturing the solution of a MO and MDO problem in aeronautics requires the use of CFD and FEA computations which are time consuming, and involve the evaluation of candidate solutions of non-linear equations with several millions of mesh points and the computations of prohibitive gradients. There are different approaches for solving a MDO problem using traditional deterministic optimisation techniques [1-3].

New algorithms such as Evolution Algorithms (EAs) are good for complex cases problems where the search space can be multi-modal, non-convex or discontinuous, with multiple local minima and with noise. There are also problems where we look for a set of Pareto solutions, a Nash equilibrium point or other solutions like ones issued from Stackelberg games. Optimisation techniques can be combined with approximation techniques for expensive computations, for multi-fidelity analysis, for complex MDO problems incorporating additional compatibility constraints and variables into the system and in applications with complicated search spaces where the design space dimension varies.

Evolutionary Algorithms (EAs) have been pioneered in the late 60' by J. Holland [4] and I. Rechenberg [5] anf Goldberg [6]. EAs are based on Darwinian evolution, whereby populations of individuals evolve over a search space and adapt to the environment by the use of different mechanisms such as mutation, crossover and selection. An attractive feature of EAs is that they evaluate multiple candidate members of a population and are capable of finding non dominated solutions distributed on the so called Pareto front. EAs have been successfully applied to different aeronautical design and CFD problems and due to their robustness properties they have been recently used to explore the capabilities of EAs for aircraft, wing, aerofoil and rotor blade design [7-9]. However one drawback of EAs is that they are slow to converge as they require a large number of function evaluations and have poor performance with increasing number of variables. Hence the continuing challenge in the scientific community has been to develop robust and fast numerical techniques to overcome these difficulties and make the complex task of design and optimisation with EAs operational in aeronautics. In this paper we describe the implementation of a numerical method for the optimisation of aeronautical systems that uses a robust evolutionary technique, which is scalable to preliminary design studies with higher fidelity models for the solution.

The rest of this paper is organised as follows: section 2 provides the reader with an overview of needs for a robust framework to solve multi-objective and multi-disciplinary design optimisation problems, section 3 describes the main components of a framework, section 4 provides details on how his requirements are satisfied within the current framework , section 5 presents the application of the methodology to problems related to the aero-structural wing design. Conclusions are then presented in section 6.

Requirements For A Multi-Criteria Multidisciplinary Design Optimisation Framework

Complex optimisation problems in engineering may involve non-linearities, multi-criteria and multidisciplinary considerations. In order to handle these complexities it is desirable to develop of a system, which facilitates integration of a series of design and analysis tools, graphical user interfaces (GUI) and post-processing capabilities. This section focuses on the requirements, development and implementation of such a framework using Evolutionary Algorithms in which different multidisciplinary and multi-criteria problems can be analysed.

The fundamental idea with the framework is to simplify the task of integration to the design team so that it can focus on the problem itself. The idea on the development of this framework is a generic system that can be easily developed, maintained and extended.

The basic requirements for a MDO framework can be subdivided in architectural design and information access, optimisation methods, problem formulation and execution [10-11]. In Architectural Design some of the considerations are that the framework should be developed using object-oriented principles, provide an easy to use and intuitive GUI, be easily extensible to integrate new processes and numerical methods into the system, not impose unreasonable overhead on the optimisation process handle large problem sizes and be based on standards.

Information Access refers to the framework on providing facilities for database management, capabilities to visualize intermediate and final result from the analysis or optimisation, allow capabilities for monitoring the status of an execution and mechanisms for fault tolerance. On Optimisation Methods, the framework should allow ease of integration of robust optimisation tools, allow an easy coupling of different disciplinary analysis with optimiser, provide schemes for sub-optimisations within each design module and allow the user to incorporate legacy codes which can be written in different programming languages and proprietary software where no source code is available. A final consideration is on Problem Formulation and Execution where the framework should allow the user to configure and reconfigure different multi-criteria and MDO formulations easily without low level programming, allow the execution and movement of data in an automated fashion, should be able to execute multiple processes in parallel and through heterogeneous computers and execute different optimisation runs in a batch mode. In the following sections we will describe how the framework satisfies these requirements.

Design

With these requirements in mind the general scope for the framework was identified. The framework was designed to address all of these requirements to some extent. Figure 1 shows a representation of different components. The framework has a GUI, a robust optimisation tool, several analysis modules and capabilities for parallel computing, mesh generation, Design of Experiments (DOE) and post-processing.

Implementation

Integrating these components is a complex task. This work considers the development of the architecture, the GUI, implementation and extension of a robust optimisation tool, a general formulation for MDO and multi-criteria problems, and capabilities for pre and post-processing. The DOE capability has been accounted for, but has been evaluated only for simple mathematical test cases. The following sub-sections detail how the requirements are satisfied.

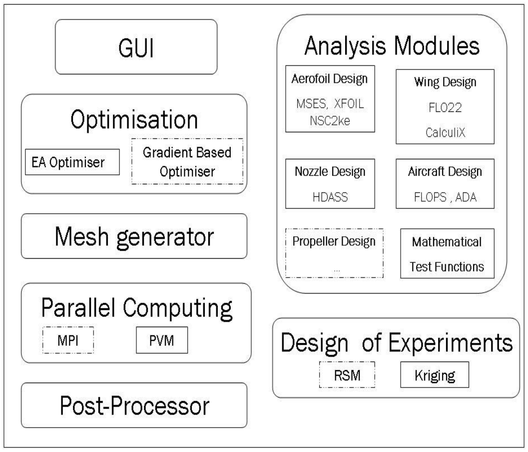
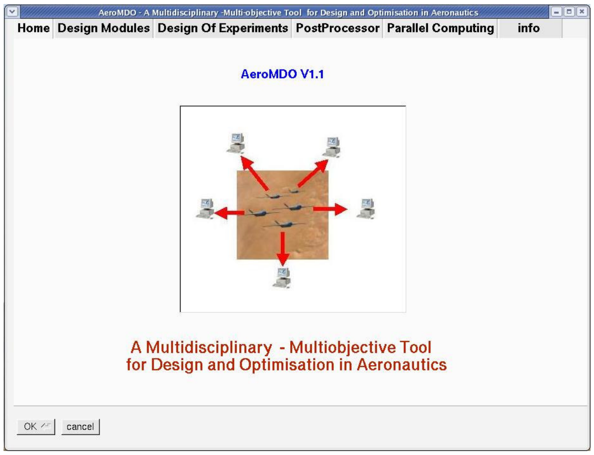


Figure 1: MDO Framework



AeroMDO - A Multidisciplinary -Multi-objective Tool for Design and Optimisation in Aeronautics

AeroMDO V1.1

A Multidisciplinary - Multiobjective Tool
for Design and Optimisation in Aeronautics

Figure 2: GUI Sample.

Architectural Design And Information Access

To satisfy the architectural design requirements the platform uses an object-oriented approach in C++. The benefits of using object-oriented software are the ease of implementation and extension of software in a modular fashion by the use of classes and methods. In an industrial and academic environment the need for a user-friendly application is required hence a simple GUI was designed. There were many considerations and options for the GUI development, but knowledge in C++ and the use of object-oriented principles were the main considerations. The Fast Toolkit (FLTk) library [12] was selected for this task. This toolkit provides a friendly and easy to use environment for different implementations. The GUI is simple and modular on its implementation and consists of five main modules as illustrated in Figure 2. The main modules are: Design and Analysis, Design of Experiments, Post-processing and Parallel Processing. The GUI facilitates development, extension and modifications of modules in a rather simple manner. The user has to create only a few subroutines within the corresponding module.

Design and Analysis Module

The aerofoil design module allows the user to conduct a single design and optimisation for different aeronautical applications and mathematical test cases. So far this module contains five sub-modules for aerofoil, multi-element aerofoil, nozzle, wing, aircraft and mathematical functions design or optimisation. As designed the framework is flexible and provides for ease of implementation of other design modules. Modules currently under development are such as those for propeller, cascade aerofoil and rotor blade design.

Development of Aeronautical Design Modules

Before implementing a sub-module it is necessary to develop a design module interface, this comprises a series of files written in C++ that allow communication between the GUI, analysis codes, the optimiser and the parallel processing capability. When designing the interface a choice has to be made depending if the source code for the analysis tool was available or not. In the current implementations minimal modification to the source code was required, ideally it is desirable to operate only through the input/output files of the analysis tool. In the implementations considered, a design template was used in conjunction with one or two additional files which contain the necessary linking subroutines allowing a rather fast implementation of the design modules. So far, there are subroutines for aircraft, nozzle, wing and full aircraft configuration design. Each of these options allows the user to perform a single design analysis or a full optimisation.

Aerofoil Design and Optimisation Module: This module allows the user to perform a single analysis or a full aerofoil optimisation routine. Three different CFD codes at a combination of them can be used: A panel method (XFOIL) [13], an Euler + boundary layer (MSES) [14] or Navier-Stokes analysis (NSC2ke [15]).

Wing Design and Optimisation Module: This module allows the user to conduct a single analysis on a wing or an optimisation study. These could be studies in one or several objectives or with multiple disciplines. Figure 3 illustrates this module. Details on the analysis tools used in this module and its application to multi-criteria and multidisciplinary wing design are presented in section 5.

Aircraft Design and Optimisation Module: This module allows the user to analyse and optimise different problems related to aircraft external configuration design. It can be used to design and optimise different subsonic, Unmanned Aerial Vehicles, transport or supersonic aircraft. Single or multi-criteria optimisation studies can be performed. Comparison of different multi-criteria analysis such as Pareto optimality and Nash equilibrium approach are possible. The user can select from two different analysis codes: An object-oriented Aircraft Design and Analysis Software (ADA) developed by the first author or using the Flight Optimisation System (FLOPS) software developed by A. McCullers at NASA Langley. ADA is conceptual design and analysis software written using object-oriented principles and is based on the formulation described in Raymer [16]. FLOPS [17], a more robust solver, is a workstation-based code which has capabilities for conceptual and preliminary design and evaluation of advanced concepts. The sizing and synthesis analysis in FLOPS are multidisciplinary in nature. It has a numerous modules and analysis capabilities for takeoff, performance, structural, control, aerodynamic and noise.

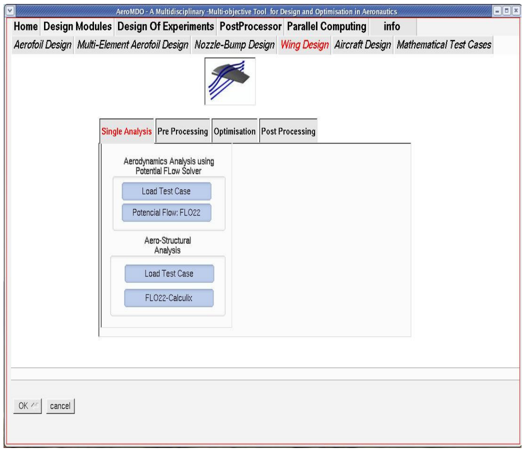


Figure 3: Wing Design and Optimisation Module

Multi-element Aerofoil Design and Optimisation Module: Similar to the aerofoil design module, it allows the user to perform a single analysis or a full optimisation, the user can choose from an Euler or Navier-Stokes analysis.

Nozzle-Bump Design and Optimisation Module: The Nozzle -Bump design module allows a single two-dimensional analysis or optimisation using the CUSP solver developed by Srinivas [18].

Mathematical Test Functions Module: This module allows the user to design, and evaluate single, or multi-criteria mathematical test functions which give confidence in the robustness and performance of the optimisation method before deciding on its application to real world problems. The current implementation includes mathematical test function for single or multiple criteria, constrained optimisation, DOE and non-linear goal programming problems.

Design of Experiments Module: In the implementation considered in this research, the designer uses an EA for the optimisation, but as discussed in section 1, one of the drawbacks of EAs is that they suffer from slow convergence. By providing a DOE capability into the framework we wish to hybridise the desirable characteristics of EAs and surrogate models such as RSM to obtain an efficient optimisation system. Within this context, the DOE samples a number of design candidates at which the analysis code (CFD will run), the surrogate model is then constructed for the computationally expensive problem. Different sampling and DOE strategies can be used; Latin hypercube, Response Surface Methods or DACE/Kriging. There is sufficient literature and software developed specifically for DOE, after a careful selection of software packages it was decided to implement the DACE tool box [19] which is robust and allows different options for sampling strategies and DOE. This software was ported to Octave (a mathematical package common in most UNIX installations) and then integrated with the framework. If desired, the user can design and implement different DOE methods.

Parallel Computing Module: One of the drawbacks of EAs is slow convergence but this module allows the users to dynamically create, add or delete nodes on the parallel implementation. Recent work on multi-criteria parallel evolutionary algorithms has allowed significant performance and robustness gains in global and parallel optimisation [20]. The framework considers the implementation of a cluster of PCs, wherein the master carries on the optimisation process while remote nodes compute the solver code. The message-passing model used is the Parallel Virtual Machine (PVM) [21].

Post Processing: The approach considered for post-processing was to use a combination of visualisation capabilities within each analysis software, and the use of GNU plot (a graphics software common in most UNIX installations). Common to all design modules is visualisation of the evolution progress of the fitness function and Pareto fronts for multi-criteria problems. Post-processing tools on each analysis module include a top view of the wing plan forms and a general 3D view of the resulting aircraft configurations. Visualisation tools within each analysis software module include the pressure coefficient distribution on the aerofoil using an Euler + Boundary Layer solver or pressure or Mach contours using a Navier-Stokes solver. Examples of some visualisation capabilities are presented in section 7.

Implementation of Different Legacy codes: The framework also implements legacy codes in different programming languages C, C++, Fortran 90, and Fortran 77. The optimiser has been successfully coupled with the following aerodynamic and analysis software: FLO22 [22], FLOPS, ADA, XFOIL, MSES, PANAIR, MSC. NASTRAN and NSC2Ke. One of the benefits of using an Evolutionary optimiser is that EAs require no derivatives of the objective function. The coupling of the algorithm with different analysis codes is by simple function calls and input and output data files.

Optimisation Methods

The second requirement is the incorporation of obust optimisation tools. In this research we used and extended the Hierarchical Asynchronous Parallel Evolutionary Algorithm (HAPEA) approach developed by Whitney [23,24] and Gonzalez [25,26] The foundation of this algorithm lies on traditional evolution strategies and incorporate the concepts of multi-criteria optimisation, hierarchical topology, parallel computing and asynchronous evaluation.

This algorithm uses a hierarchical approach with three levels, on the bottom level a coarse type analysis to direct the exploration, at the top level more precise model that better describes the physics involved and at an intermediate level, a compromised balance between top and bottom layers is used. Initially the system will specify the design variables $\boldsymbol{x}$, constraints g_i, g_{ij} ,and parameters p , then it will generate random sub population of individuals μ_o at each layer, then it defines the number of subpopulations (nodes) *i* and number of hierarchical levels which for simplicity is equal to the number of analysis *k*. Once these initial populations are generated the algorithm will go through the steps described in the previous algorithms. the scheduler first determines whether given stopping conditions have been met, and if so the evolutionary loop is exited and the entire process is stopped. If no stopping conditions are met, the scheduler updates the asynchronous solver so that further progress may be made.

Then the scheduler determines whether or not candidate solutions which have been solved are ready for *incorporation* into the population. If such solutions exist, the *incorporation* routine is called and available candidates which now have had a fitness assigned are processed; it receives the individual, ages the population and buffer, performs Pareto tournament selection, deletes the oldest from the buffer and if the acceptance is true it is inserted in the population which I subsequently sorted, it then updates the CMA parameters. Finally, the scheduler determines whether it is possible to generate more candidates, by polling the asynchronous solver.

If this is possible, then the generation routine is called and individuals are generated via the evolutionary operators, by doing recombination, mutation via CMA, checking upper and lower bounds. During evaluation, the optimiser will take output analysis a_i and parameters p to guarantee satisfaction of constraints and compute the overall fitness function. If the problem is multi objective the algorithm will find the non-dominated individuals and will calculate the Pareto fronts.

On a hierarchical topology with three levels, when the epoch is finished or the migration criteria is satisfied, the migration phase occurs: Layer 1 gets best solutions from Layer 2 and re-evaluates them using type of analysis one, Layer 2 gets random solutions from Layer 1 gets best solutions from Layer 3 and re-evaluates them using type of analysis two, Layer 3 gets random solutions from Layer 2 and re-evaluates them using type of analysis three. This process continues until a stopping condition is reached. These can be equal to a limited number of function evaluations, hours or a prescribed value on the fitness function.

Problem Formulation and Execution

A third requirement is on how to incorporate different multi-criteria and MDO formulations. There are many strategies proposed for multi-criteria and MDO and the development of these optimisation methods, architectures and decomposition methodologies has been an active field of research. The framework developed in this research is applicable to an integrated analysis or distributed MDO analysis [1-3]. Examples on the application of the method for these formulations are presented by the optimiser in section 5. The framework also satisfies the requirement of multiple executing processes in parallel; different candidate members of the population can be sent to remote parallel heterogeneous computers. Once a solution is computed it is returned to the optimiser and framework for database storage, manipulation.

Applications

The framework has been used to evaluate several real world problems including inverse and direct problems for aerofoil, high- lift aircraft system, multidisciplinary and multi-criteria wing and aircraft design and optimisation problems [23-26]. In the following we illustrate the application of the method for three real world examples; two test cases related to UAV aerofoil design and one test case related to UAV wing design.

Multidisciplinary Wing Design

Problem Formulation: This case considers a multi-criteria wing design optimisation for a UAV. The cruise Mach number and altitude are *0.69* and *10000 ft*. The wing area is set to *2.94 m2* and the corresponding *CL* is fixed at *0.19*. For the solution we initially compute the pressure distribution over the wing using a potential flow solver to obtain the wing aerodynamics characteristics that include the span wise pressure distribution, *CL* and total drag coefficients *CDw*. Concentrated loads replace the lift distribution and the spar cap area is calculated to resist the bending moment. The weight is then approximated as the sum of the span-wise cap weight. The strong interaction between the aerodynamic pressure distribution and the structural deflections is ignored.

Definition of the EA Strategy: A Simple EA Single/Multi-objective EA, Parallel EA: The complexity, non-linearity and multi-objective characteristics of this problem make it suitable to be solved by an EA optimiser. The computational cost of a Navier-Stokes or Euler solution around one of these geometries involves high computational expense therefore it is also desirable to use parallel computations and a multi-fidelity approach. In this case we use a multi-objective parallel EA (MOPEA) and select the HAPEA approach which has all these capabilities.

Design Variables and Constraints: The wing geometry is represented by three aerofoil sections and five variables for the wing planform. In total fifty-three design variables are used for the optimisation. Figure 5 illustrates the main design variables and table 1 indicates their upper and lower bounds for the wing plan form. The aerofoil geometry is represented by two Bezier curves, one for the mean line and one for the thickness distribution. The mean line--thickness distribution is a standard method for representing aerofoils [27], as it closely couples the representation with the results; the mean line has a powerful effect on cruise lift coefficient and pitching moment, while the thickness distribution has a powerful effect on the cruise drag. Put simply, the aerofoil is obtained by perpendicular offset of the thickness distribution about the mean line.

For a given mean line point (x_m , y_m) and matching thickness distribution height y_t, an upper and lower surface point can be obtained:

$$x_{u,l} = x_m \pm y_t \sin(\theta) \tag{1}$$

$$y_{u,l} = y_m \pm y_t \cos(\theta) \tag{2}$$

where θ is the angle of the mean line at (x_m,y_m). We select the *x*-positions of the Bezier control points in advance; the y- positions remain as the unknowns. The only restrictions are that the first and last points are fixed to (0, 0) and (1, 0) to provide leading and trailing edges respectively, and that the first control point on the thickness distribution must be directly above the leading edge (i.e. (0,$y_{c,1}$)) to provide a rounded geometry (Bezier curves are by definition always tangent to the extreme edges of their defining envelopes).

We bound the vertical heights to range $y_c \in [0.01, 0.10]$ giving a very wide range of possible geometries (theoretically spanning aerofoils from 2% to 20% thick).The advantage of using single high--order Bezier curves for the representation rather than piecewise splines or others is their geometric stability. A Bezier curve must by definition always be contained within the bounding envelope of control points. Furthermore, if the bounding envelope is not re--entrant, then the curve will also have this property. Also, Bezier curves do not 'kink' like a piecewise spline, and the defining equations are not stiff (ill-conditioned). Therefore, a small change in control point location will always result in a small change in surface representation. This provides a favourable interface between the optimiser and the flow solver. For this case, four evenly spaced interior (free) control points were taken for the mean line, and five for the thickness distribution, giving a problem in nine unknowns

Constraints are imposed on minimum thickness (*t/c ≥ 0.14* root aerofoil, *0.12* intermediate aerofoil, and *0.11* tip aerofoil) and position of maximum thickness. (*20% ≤ t/c ≤ 55%).* If any of these constraints is violated both fitness are linearly penalized to ensure an unbiased Pareto set.

Fitness Functions: The two fitness functions to be optimised are defined as minimisation of wave drag (CDw) and minimisation of the sum of the span wise cap weight (Wsc) to resist the bending moment.

$$\min(f_1): \quad f_1 = c_{Dwave} \tag{3}$$

$$\min(f_2): \quad f_1 = \sum W_{SC} \tag{4}$$

Aerodynamics and Weight Analysis: The aerodynamic characteristics of the wing configurations are evaluated using FLO22, a 3-D full potential wing analysis software. This program uses sheared parabolic coordinates and accounts for wave drag [22]. FLO22 was developed by Jameson and Caughey for analysing inviscid, isentropic, transonic shocked flow past 3-D swept wing configurations. The algorithm is based on free stream Mach numbers limited by the isentropic assumption and weak shock waves are automatically captured wherever they occur in the flow. Also the finite difference form of the full equation for the velocity potential is solved by a relaxation method, after the flow exterior to the aerofoil is mapped to the upper half plane. The mapping procedure allows exact satisfaction of the boundary conditions and use of transonic free stream velocities. Details on the formulation and implementation can be found in Jameson et al [22].

The fixed lift requirement can be satisfied by performing an extra two function evaluations by varying the angle of attack at the wing root and assuming a linear variation of the lift coefficient. The lift distribution is summed into concentrated loads. The wing weight is estimated from the wing spar cap area designed to resist the bending moment. The local stress has to be less than the ultimate tensile stress in this case for Aluminium Alloy 2024 -T6 ≤ σadm

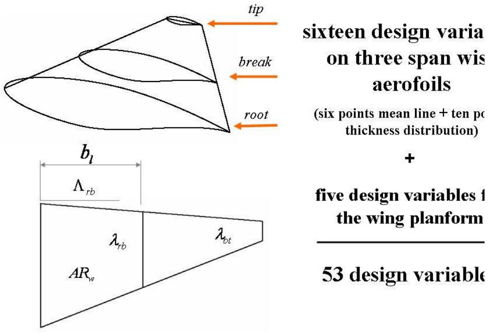


Figure 5: Design variables for multidisciplinary wing design.

Implementation –Design and Optimisation Rationale: The problem was implemented using the procedure described in algorithm 12. Details on the multi-fidelity –hierarchical tree are: (EA with CMA/Pareto tournament selection, Asynchronous Evaluation) on each node of the hierarchical tree. We use the wing design and optimisation module to solve this problem and considered two approaches for the solution; in the first approach the optimiser is configured as a traditional EA with a single population model and computational mesh of 96 x 12 x 16 for the FLO22 code. The second approach uses a hierarchical topology of resolutions with the following settings:

Top Layer: A population size of 30 and a computational mesh of 96 x 12 x 16.
Middle Layer: A population size of 30 and a computational mesh of 72 x 9 x 12.

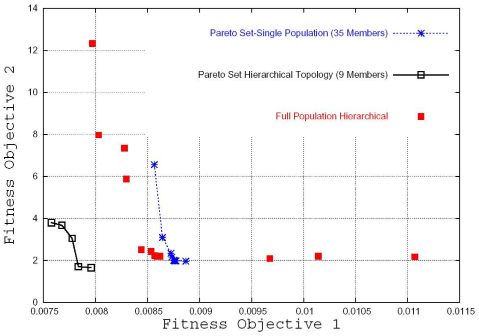


Figure 6: Pareto fronts after 2000 function evaluations.

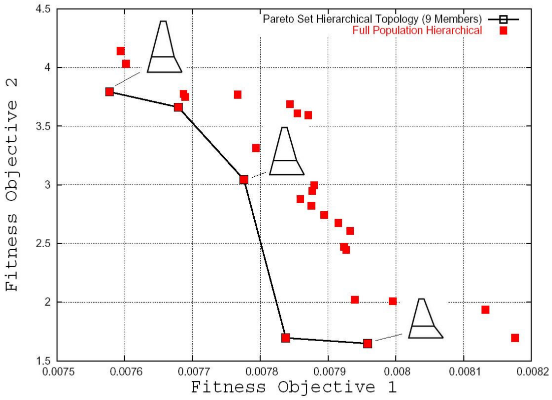


Figure 7: Pareto fronts and wing plan forms.

Optimisation Results and Post-processing of Optimal Solutions: The algorithm was run five times for 2000 function evaluations and took in average six hours to compute. Figure 6 shows the Pareto fronts obtained by using the two approaches. It can be seen how the optimisation technique gives a uniformly distributed front in both cases. By inspection we can see that the use of a hierarchical approach gives an overall lower front as compared to a single model approach. Figure 7 illustrates the Pareto front for the hierarchical approach and a representative top view of the wing geometries. Figure 8 shows the corresponding aero foils at root, break and tip for some of the Pareto configurations and table 2 indicates the final values design variables.

This problem demonstrates the use the framework for UAV wing design and optimisation. Results indicate a computational gain on using a hierarchical topology of fidelity models as compared to a single model during the optimisation. Results also show how the algorithm was capable of identifying the trade-off between the multi-physics involved and provide classical aerodynamic shapes as well as alternative configurations from which the design team can choose and proceed into more detailed phases of the design process.

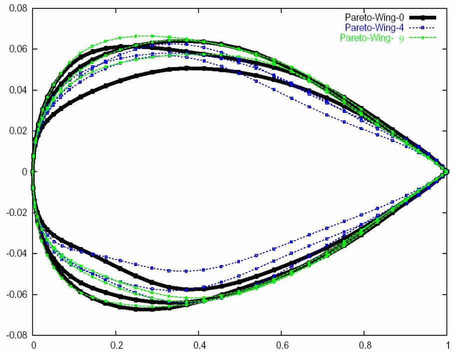


Figure 8: Aerofoil geometries root, break tip for three members of the Pareto front.

Aero-Structural Wing Optimisation for a High Altitude Long Endurance UAV

Problem Formulation: In this case, we consider the detailed analysis and optimisation of a wing for a high altitude long endurance (HALE) UAV application. This is a multi-objective problem where we want to maximise the lift-to-dag ratio and minimise wing weight. The operating conditions and data for the baseline wing are based on reference [28]. The aircraft has a wingspan of approximately 17 m, a mean chord of approximately 2.432 m, a wing area of 50.194 m2 and a plan form shape with 5.9 deg sweep.

We assume the aircraft operating at M =0.5983 and a cruise altitude of 15840 m approximately. It is also assumed that this UAV uses a single aerofoil, the LRN1015 aerofoil, as the only wing section aerofoil along the wing span. Table 3 indicates the operating conditions while table 4 summarises some of the design variables for the external geometry. Table 5 indicates some of the parameters for the internal structure. Some details on the structural model are based on Reference [29] and the baseline is indicated in Table 6.

Definition of the EA Strategy: A Simple EA Single/Multi-objective EA, Parallel EA.: The complexity, non-linearity and multi-objective characteristics of this problem make it suitable to be solved by an EA optimiser. The computational cost of a Navier-Stokes or Euler solution around one of these geometries involves high computational expense therefore it is also desirable to use parallel computations and a multi-fidelity approach. In this case we use a multi-objective parallel EA (MOPEA) and select the HAPEA approach which has all these capabilities.

Aerodynamics and Weight Analysis: An aero-structural solver was developed. This solver integrates two analysis tools for FEA and CFD namely MSC. Nastran, and the high order panel method PANAIR [30]. The entire aero-structural program is controlled through a Matlab® script file. This allows for an easy coupling of the different required programs as one continuous Matlab® data structure could be utilized to define all the information passed between programs. A flow diagram if this process is indicated in Figure 9.

Design Variable: Each candidate wing is represented by some design variables that define the internal and external geometry. We consider three aerofoil sections and five variables for the wing plan form. The aerofoil geometry is represented by 8 free control points on the mean line and 10 free control points on the thickness distribution.

External shape: Figure 10 and table 7 Illustrates the upper and lower bounds for the design variables that define the external geometry.

Internal structure: For the structural analysis and internal geometry, a simplified finite element model consisting of varying number of ribs and up to 7 spars are used. The model consists of shell elements with simplicity the spars and ribs caps. As expected the number of nodes and elements varies depending on the wing geometry. The number of design variables in the structural analysis is related to the nodes, elements and depends also on the number of internal spars and ribs. In the examples considered in this work, the ribs and spars are modelled as single panels with varying thickness. The structural model is represented, indicated in figure 11 can be represented with 11 design variables that define the skin panels, spar panels, spar and rib caps. The upper and lower bounds are indicated in table 8.

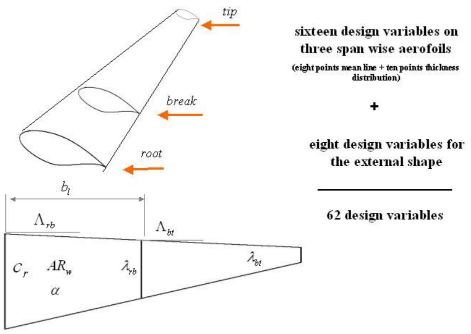


Figure 10: Design variables for multidisciplinary wing design.

In total 72 design variables are used for the optimisation.

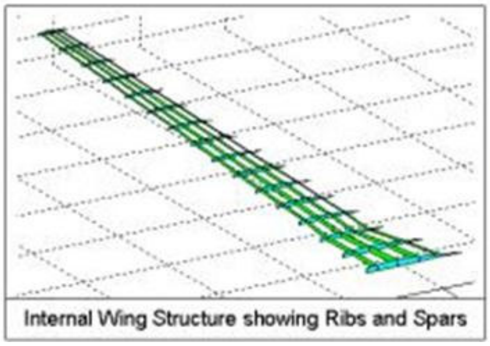


Figure11: Design variables for the structural model (# ribs, number of spars, thickness internal geometry design.

Constraints: There are several, geometric, aerodynamic and structural constraints for this problem

Geometrical constraints: Constraints are imposed on minimum thickness ($t/c \geq 0.14$ root aerofoil, 0.12 intermediate aerofoil, and 0.11 tip aerofoil) and position of maximum thickness. ($20\% \leq t/c \leq 55\%$). If any of these constraints is violated both fitness are linearly penalized to ensure an unbiased Pareto set.

Aerodynamics constraints: Moment coefficient: the wing was linearly penalized by increasing the fitness value defined by the inverse of the lift to drag ratio for every percent over the aerodynamic moment benchmark value calculated.

Lift coefficient: a penalty was added to the fitness functions if the lift coefficient vale of a candidate wing was below that for the baseline geometry.

*Structural constraints:*Buckling of Wing Skin Panels: a simple analytical expression was used to test whether or not the stresses in the wing skin sections was large enough to cause buckling. These local buckling expressions are described by Eqn. 5

$$\frac{\sigma_1}{\sigma_{cr}}-1\leq 0,\quad \frac{\tau_1}{\tau_{cr}}-1\leq 0,\quad \frac{\sigma_1}{\sigma_{cr}}+\left(\frac{\tau_{12}}{\tau_{cr}}\right)^2-1 \qquad (5)$$

where:

$$\sigma_1=\frac{N_1}{2t},\quad \sigma_{cr}=\frac{3.6kE}{\left(\frac{b}{t}\right)^2},\quad \tau_{12}=\frac{N_{12}}{2t},\quad \tau_{cr}=\frac{4.85kE}{\left(\frac{b}{t}\right)^2}$$

and b is the width of each wing skin section and k was an additional safety factor to guard against delamination of the plies making up the carbon graphite composite material. The safety factor was set at four.

Tip Twist: as the wing bends, so the pressures generated by the wing would change and hence the forces within the wing would also change. If the wing tip twisted, there is a possibility that the wing would greatly increase the drag created by the wing, but also increase the moment about the Z axis. As a full aero-elastic analysis was not part of the simulation runs, any further increase in the angle at which the wing would make with the freestream flow would be disadvantageous to the computation as the final wing position and internal forces could not be computed. It was therefore necessary to penalize wings in which after the structural simulation using MSC.NASTRAN, the wing tip angle of attack was greater than one degree from the initial wing setting angle. This penalty took the form of additional mass added to the structure at the rate of ten percent for each additional degree.

Deflection: the wing was allowed to deflect a maximum of twenty percent of the span for the 2.5g load case before a penalty value was added to the fitness value. As with the tip twist penalty an additional mass was added linearly at a rate of ten percent per percent deflection over twenty percent.

Failure of Internal Component Panels: if any components making up the internal structure within the wing, spars, ribs, skin, etc failed due to excessive strains, the wing was heavily penalized by the addition of extra wing mass. This mass was increased at a rate of ten percent for each panel making up a component which failed. Hence, if ten CQUAD4 components making up a spar failed, the mass of the wing structure was increased one hundred percent.
Table 9 summarises the values allowed for the constraints.

Fitness Functions: The two fitness functions to be optimised are defined as minimisation of he inverse of lift-to-drag ratio and minimisation of wing weight (W_{wing}):

$$\min\left(f_1\right)=\frac{1}{\left(L/D\right)} \qquad (6)$$

$$\min\left(f_2\right)=W_{wing} \qquad (7)$$

Imple*mentation - Design and Optimisation Rationale:* We use the wing design and optimisation module to solve this problem. etails on the multi-fidelity –hierarchical tree are: (EA with CMA/Pareto tournament selection, Asynchronous Evaluation) on each node of the hierarchical tree with the following parameter settings for the EA and CFD solver:

Top Layer: A population size of 30 and a computational mesh of 96 x 12 x 16.
Middle Layer: A population size of 30 and a computational mesh of 72 x 9 x 12.

Optimisation Results and Post-processing of Optimal Solutions: The algorithm was run for 400 function evaluations and took in average three days to compute on a cluster of three, 3.2 GHz machines. Figure 12 compares the Pareto front and the baseline geometry. It can be seen how the optimisation technique gives a uniformly distributed front, the baseline geometry is at the boundary of the Pareto front. Figure 13 shows the evaluation progress for objective one, each step in the figure roughly corresponds to a migration step-good information from the bottom levels have been seeded to the upper levels. Figure 14 compares the aerofoils at root break and tip for some of the members of the Pareto front and the aerofoil for the baseline geometry. Table 10 indicates the final values of design variables and objective functions for the baseline geometry and some members of the Pareto front. Figures 15, 16 and 17 show the pressure distribution, displacement and von Mises stress for one of the members of the Pareto front.

This problem demonstrates the use the framework for UAV wing design and optimisation using high fidelity analysis tools. Results indicate the workings of the method and the coupling of aero-structural solver with the framework. Results also show how the algorithm was capable of identifying the trade-off between the multi-physics involved and provide better results as compared to the baseline design. The simulation was stopped at 1000 function evaluations due to limitations on the computational resources, further test is underway.

Conclusions

This paper described the basic concepts of a hierarchical, asynchronous parallel multi-objective evolutionary algorithm used to solve aero structural design problems. The method can be used as an alternative option to satisfy some of the needs for robust multi-objective and multidisciplinary design optimisation problems. As described the method is easily coupled, particularly adaptable, easily parallelised, require no gradient of the objective function(s), have been used for multi-objective optimisation and successfully applied to different aeronautical design problems. The methodology is integrated in a single framework that allows:

- Solving single and multi-objective problems that can be deceptive, discontinuous, and multi-modal.
- Incorporation of different game strategies- Pareto, Nash, Stackelberg
- Implementation of multi-fidelity approaches
- Parallel Computations
- Asynchronous evaluations

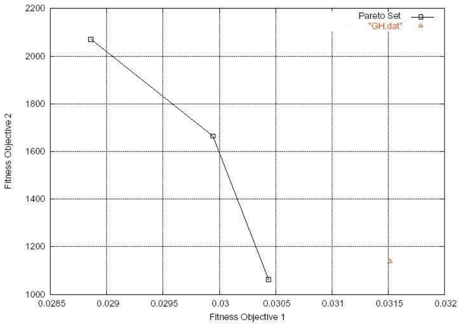


Figure 12: Comparison of Pareto front and baseline geometry.

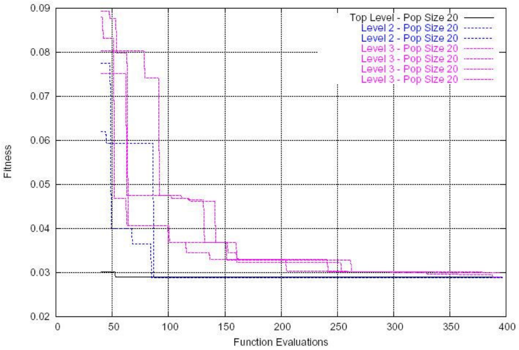


Figure 13: Evaluation progress for objective one.

Further extensions of the structural model and its coupling with the flow solver are presently under investigation. The hierarchy of structural models and its parallel properties taken into account with game strategies within the optimisation procedure are under implementation and both efficiency and accuracy will be compared with single structural model approach.

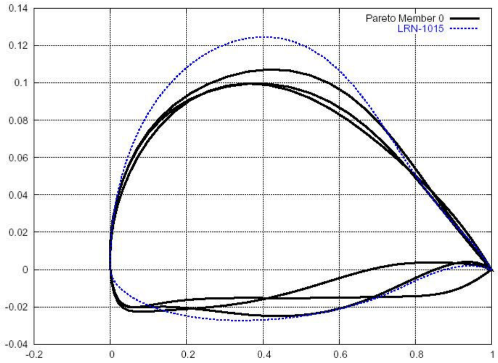


Figure 14: Comparison of aerofoil geometries at root, break and tip for Pareto Member 0 and baseline aerofoil.

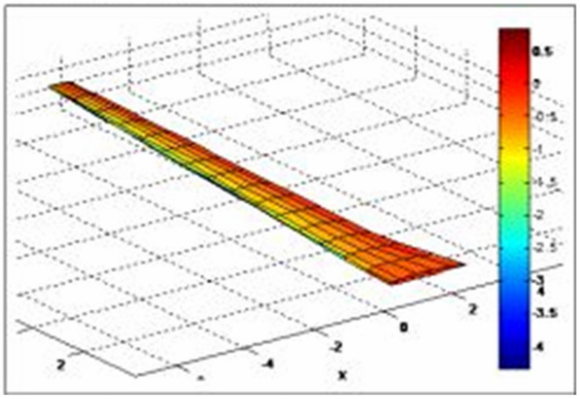


Figure15: pressure distribution for one of the members of the Pareto front.

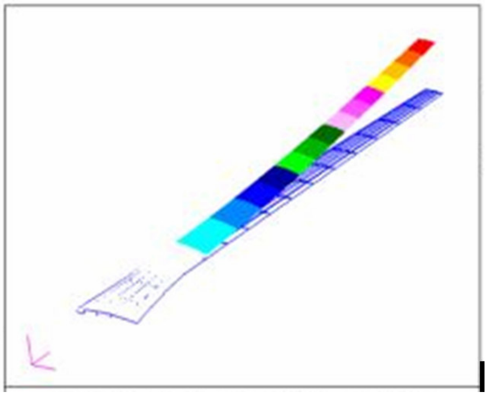


Figure 16: displacement for one of the members of the Pareto front.

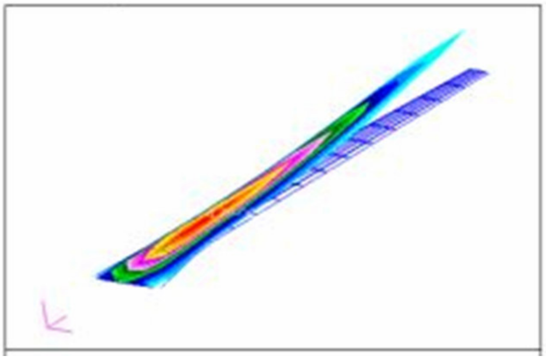


Figure 17: von Mises stress for one of the members of the Pareto front.

Acknowledgements

The authors gratefully acknowledge Mourad Sefrioui, Dassault Aviation for fruitful discussions on Hierarchical EAs. Steve Armfield the University of Sydney for allowing access to the cluster of computers. A. Jameson, Stanford University and S. Obayashi, Tohoku University for the use of the *FLO22* software to provide the full potential flow analysis around a wing. The project was supported by the Asian Office of Aerospace Research & Development (AOARD) Detachment of the US Airforce through the contract AOARD-044078 administered by LtCol Bill Nace,

References

[1] P. Bartholomew "The Role of MDO within Aerospace Design and Progress towards an MDO Capability, AIAA-98-4705, pp 2157-2165, 7th AIAA/USAF/NASA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, St. Louis, Mo,1998.

[2] J. Sobieski and RT Haftka, Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments, AIAA Paper No. 96-0711, 1996.

[3] Z. Thomas and A. Green, "Multidisciplinary Design Optimization Techniques: Implications and Opportunities for Fluid Dynamics Research" AIAA Paper-1999-3798, Jun, 1999

[4] J.H. Holland, Adaption in Natural and Artificial Systems. University of Michigan Press. 1975.

[5] I. Rechenberg, Evolution Strategie: Optimierung technisher Systeme nach Principen der biologischen Evolution, Frommann-Holsboog Verlag, Stuttgart, 1973.

[6] D. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley,1989.

[7] S. Obayashi. Multidisciplinary Design Optimization of Aircraft Wing Planform Based on Evolutionary Algorithms . In Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics, La Jolla, California, IEEE, October 1998

[8] A. Oyama, M.-S. Liou, and S. Obayashi. Transonic Axial-Flow Blade Shape Optimization Using Evolutionary Algorithm and Three-Dimensional Navier-Stokes Solver, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, Georgia, September 2002.

[9] L. F. González, E. J. Whitney, K. Srinivas, K. C. Wong and J. Périaux "Multidisciplinary Aircraft Conceptual Design Optimisation Using a Hierarchical Asynchronous Parallel Evolutionary Algorithm (HAPEA)" In I.C. Parmee, editor, Proceedings of the Sixth International Conference on Adaptive Computing in Design and Manufacture (ACDM'2004), volume 6, Bristol, UK, April 2004. Springer-Verlag.

[10] A. O. Salas and J. C. Townsend, Framework Requirements for MDO Application Development, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, Missouri, AIAA 98-4740, September 2-4, 1998, pp. 11.

[11] M.R. Sankar, I. Amitay, PM. Mujumdar and K. Sudhakar, MDO Framework Development - A Case Study With An Elementary Model Of Airborne Early Warning System Optimization, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, Georgia, September 4-6, 2002

[12] M. Sweet, C.P Earls and B. Spitzack. FLTK 1.1.6 Programming Manual, Revision 6. www.fltk.org.

[13] M. Drela. XFOIL 6.94 User Guide. MIT Aero Astro, 2001 .

[14] M. Drela. A User's Guide to MSES V2.3," Feb. 1993.

[15] B. Mohammadi. Fluid Dynamics Computation with NSC2KE, User-Guide, Release 1.0. Technical Report RT-0164, INRIA, May 1994.

[16] D. Raymer "Aircraft Design: A Conceptual Approach", American Institute of Aeronautics and Astronautics American Institute of Aeronautics and Astronautics, Third Edition, 1999.

[17] A. McCullers. FLOPS User's Guide, Release 6.02, NASA Langley Research Center, March 2003.

[18] K. Srinivas. Calculation of Cascade Flows by a Modified CUSP Scheme, Computational Fluid Dynamics Journal, 1999,2,pp 285-295.

[19] S.N. Lophaven, H. B. Nielsen and J. Sondergaard, Aspects of the Matlab Toolbox DACE, 2002, 44, IMM-TR-2002-13.

[20] D. A. Van Veldhuizen, J.B. Zydallis and G. B. Lamont. Considerations in Engineering Parallel Multi objective Evolutionary Algorithms, IEEE Transactions on Evolutionary Computation, Vol. 7, No. 2, pp. 144--173, April 2003.

[21] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek and V. Sunderam. PVM: Parallel Virtual Machine. A User's Guide and Tutorial for Networked Parallel Computing. Massachusetts Institute of Technology, 1994.

[22] A. Jameson, D. Caughey, P. Newman and R. Davis . A Brief Description of the Jameson Caughey NYU Transonic Swept-Wing Computer Program FLO22. NASA Technical Memorandum, NASA TM X-73996, Dec. 1976.

[23] E. J.Whitney. A Modern Evolutionary Technique for Design and Optimisation in Aeronautics, PhD Thesis, The University of Sydney, 2003.

[24] E. J. Whitney, M. Sefrioui, K. Srinivas, J. Périaux: "Advances in Hierarchical, Parallel Evolutionary Algorithms for Aerodynamic Shape Optimisation", JSME (Japan Society of Mechanical Engineers) International Journal, Vol. 45, No. 1, 2002.

[25] L .F. Gonzalez, Robust Evolutionary Methods for Multi-objective and Multidisciplinary Design in Aeronautics., PhD Thesis, University of Sydney, 2005.

[26] L. Gonzalez, E W Whitney, K Srinivas, S Armfield and J Periaux, A Robust evolutionary technique for coupled and multidisciplinary design optimisation problems in aeronautics Computational Fluid Dynamics Journal, July 2005

[27] I. H. Abott and A. E. Von Doenhoff. Theory of Wing Sections. Dover,1980.

[28] K. Munson, Jane's Unmanned Aerial Vehicles and Targets, Jane's Information Group Limited, 1998.

[29] www.heseweb.nrl.navy.mil/glast/CALPeerDesignRev/RFAs/RFA%20CAL006-030503.pdf

[30] G. R. Saaris A502I User's Manual - PAN AIR Technology Program for Solving Problems of Potential Flow about Arbitrary Configurations, Boeing, 1992.

Table 1: Upper and lower bounds for multidisciplinary wing design variables.

Description	Lower Bound	Upper Bound
Wing aspect ratio, [AR]	3.50	15.00
Taper ratio root to break, [λbr]	0.65	0.80
Taper ratio break to tip, [λbt]	0.20	0.45
Wing 1/4 chord sweep, *deg* [Λi]	10.00	25.00
Break location, [b_l]	0.30	0.45

Table 2. Optimum design variables for some members of the Pareto front .

Description	Pareto Member 0	Pareto Member 4	Pareto Member 15
Wing aspect ratio [AR]	6.92	6.07	2.56
Wing 1/4 chord sweep, deg [Λi]	10.83	10.02	20.30
Wing semi span, m	2.14	2.00	1.30
Taper ratio root to break, [λbr]	0.74	0.68	0.69
Taper ratio break to tip, [λbt]	0.31	0.24	0.35

Table 3: Operating flow conditions for the baseline wing.

Altitude (m*)*	15849.8
Freestream Mach number	0.5983
Chord reference point	0.6080
Wake data	[1 1 10]
Angle of Attack (*deg*)	3
Maximum maneuver (g's)	2.5

Table 4: External geometry parameters for the baseline wing.

Semispan (*m*)	16.9461
Root chord (*m*)	2.432
Dihedral angle (*rad*)	0*pi/180
Number of cranks	1
Position of Crank as function of the wing span	0.0881
Chord ratio. [root,crank,tip]	[1 0.78 0.287]
Sweep angle (*rad*), [root, crank]	[0.1030 0.1030]
Aerofoil Type	NASA LRN 1015 at both root and tip

Table 5: Internal geometry parameters for the baseline wing.

Material	Graphite-Epoxy (T300 - 1K carbon/M76 epoxy)
Youngs Modululs (Pa)	1.53e11
Poissons Ratio	0.3
Density (kg.m-3)	1.31e3
Ultimate Tensile Strength (Pa)	1532e6
Longitudinal Compressive Strength (Pa)	947e6
Strain allowed	0.00333

Table 6: Material specifications for the baseline wing.

Material	Graphite-Epoxy (T300 - 1K carbon/M76 epoxy)
Youngs Modululs (Pa)	1.53e11
Poissons Ratio	0.3
Density (kg.m-3)	1.31e3
Ultimate Tensile Strength (Pa)	1532e6
Longitudinal Compressive Strength (Pa)	947e6
Strain allowed	0.00333

Table 7: Upper and lower bounds for external shape.

	Upper Bound	**Lower bound**	**Baseline**
Rootchord,[*cr*]	2.375	2.5	2.432
sweeptostation1, [*Λrb*]	0.0	0.1745	5.9 deg
sweeptostation2, [*Λbr*]	0.0	0.1745	5.9 deg
chordratiostation2, [*λbr*]	0.55	0.9	0.78
chordratiostation3, [*λbt*]	0.2	0.8	0.287
cranklocation1, [b_l]	0.06	0.6	0.0881
alfa,[α]	0.0	6.0	3

Table 8: Upper and lower bounds for structural model.

Internal Geometry	**Lower Bound**	**Upper Bound**	**Baseline**
Numberofspars	3	7	5
Numberofribs	12	18	16
Ribrootthickness	0.001	0.005	0.0015
Ribthicknesstaperratio	0.01	0.15	0.05
sparrootthickness	0.045	0.095	0.09
sparthicknesstaperratio	0.01	0.15	0.05
skinthickness	0.00001	0.05	0.001
skinratioroottip	0.0095	0.15	0.01
skinratioleadingtrailing	0.0095	0.15	0.01
sparcaprootarea	0.004	0.0125	0.012
ribcaprootarea	0.0005	0.002	0.0005

Table 9: Baseline Wing Specification.

Description	**Values Allowed**
Allowed wing tip twist in degrees	1
Allowed wing deflection as a percentage span	20
Allowed wing moment (benchmark)	-0.3041
Minimum lift to be generated by the wing	0.89
Wing Mass per degree twist Penalty Values	0.1
Wing Mass per degree over 20 span	0.1
Wing Mass per failed panel	0.1
Additional Cd per over allowable	0.001
Additional Cd per less than the required minimum	0.005

Table 10: Optimum design variables for some members of the Pareto front and comparison with baseline design.

Variable	**PM0**	**PM5**	**PM9**	**PM9**
Semispan (m)	18.4003	16.4019	15.8084	17
Rootchord (m)	2.41617	2.43837	2.40254	2.432
sweeptostation1 (*rad*)	0.113513	0.0567227	0.0443384	0.102972
sweeptostation2 (*rad*)	0.0706541	0.121511	0.0655513	0.102972
chordratiostation2	0.60491	0.610763	0.698868	0.78
chordratiostation3	0.276425	0.267427	0.279098	0.287
cranklocation1	0.3673	0.213602	0.205515	0.0881
Alfa (def)	5.99981	5.91514	5.88858	3
numberofspars	4.79014	5.8396	3.00028	5
numberofribs	17.1381	15.8344	15.0798	16
ribrootthickness	0.00394079	0.00374714	0.00167289	0.0015
ribthicknesstaperratio	0.108462	0.125728	0.053825	0.05
sparrootthickness	0.0861813	0.088058	0.0695108	0.09
sparthicknesstaperratio	0.122835	0.108673	0.131612	0.05
skinthickness	0.0429143	0.0462582	0.0242976	0.001
skinratioroottip	0.134344	0.0300034	0.146426	0.01
skinratioleadingtrailing	0.0596622	0.0976194	0.143986	0.01
Sparcaprootarea (m^2)	0.00875312	0.00569291	0.00549552	0.012
Ribcaprootarea (m^2)	0.000693718	0.00100964	0.00123374	0.0005
1/ (L/D)	*0.028859*	*0.0299429*	*0.0304375*	*0.031516909*
Weight (Kgs)	*2070.48*	*1663.73*	*1061.72*	*1138.15*

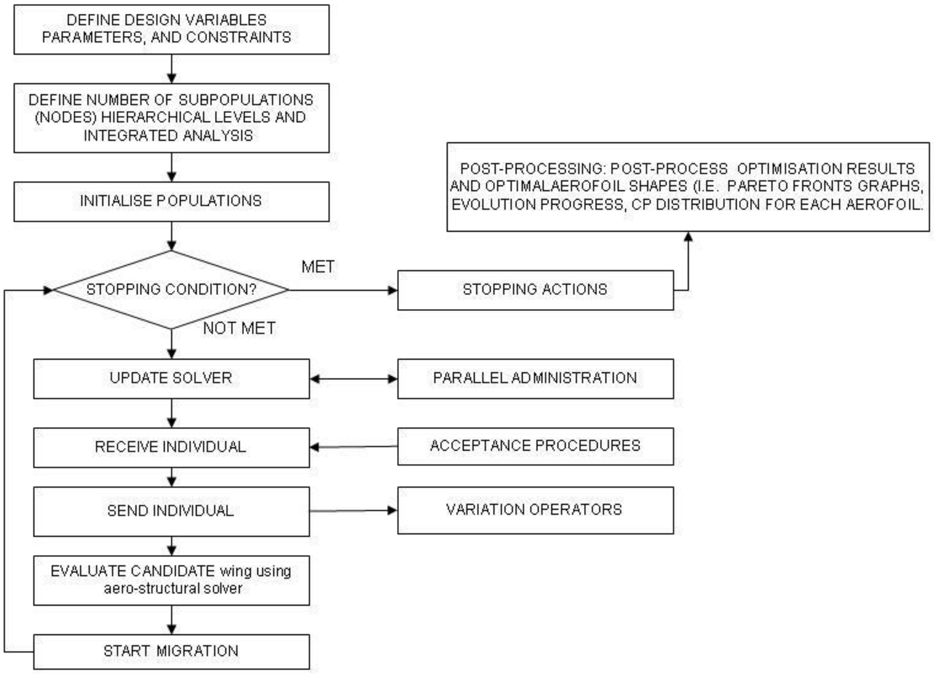


Figure 4: Overall optimisation process.

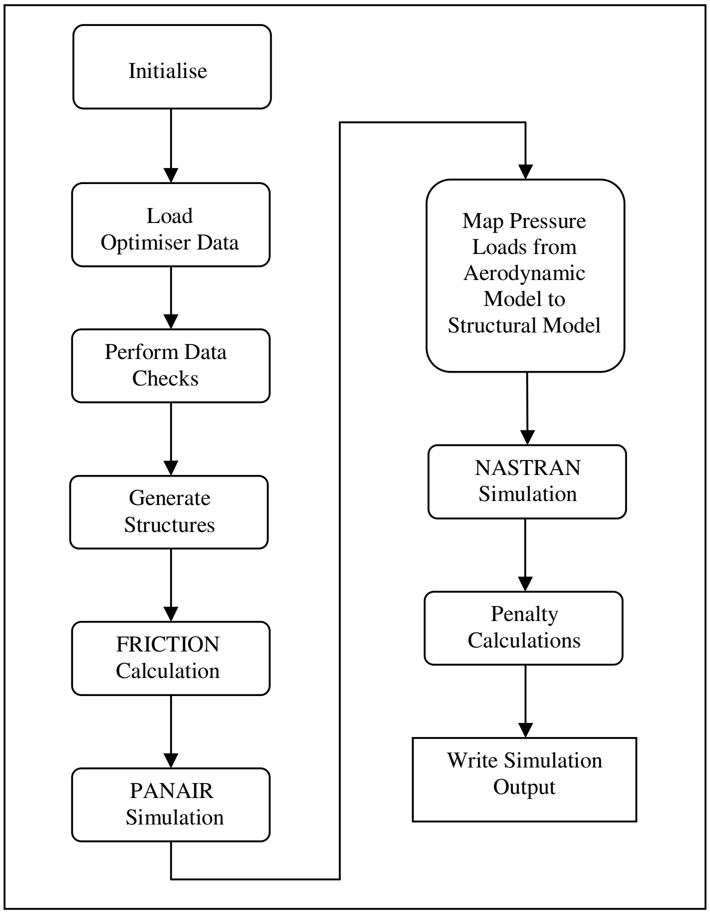


Figure 9: Aero-Structural Analysis Program Layout.