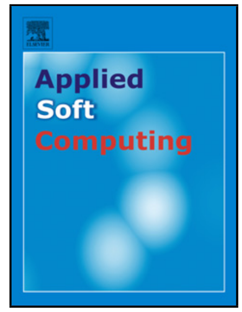
Accepted Manuscript



Title: State-of-the-Art in Aerodynamic Shape Optimisation Methods

Author: S.N. Skinner H. Zare-Behtash

PII: S1568-4946(17)30569-0
DOI: https://doi.org/doi:10.1016/j.asoc.2017.09.030
Reference: ASOC 4477

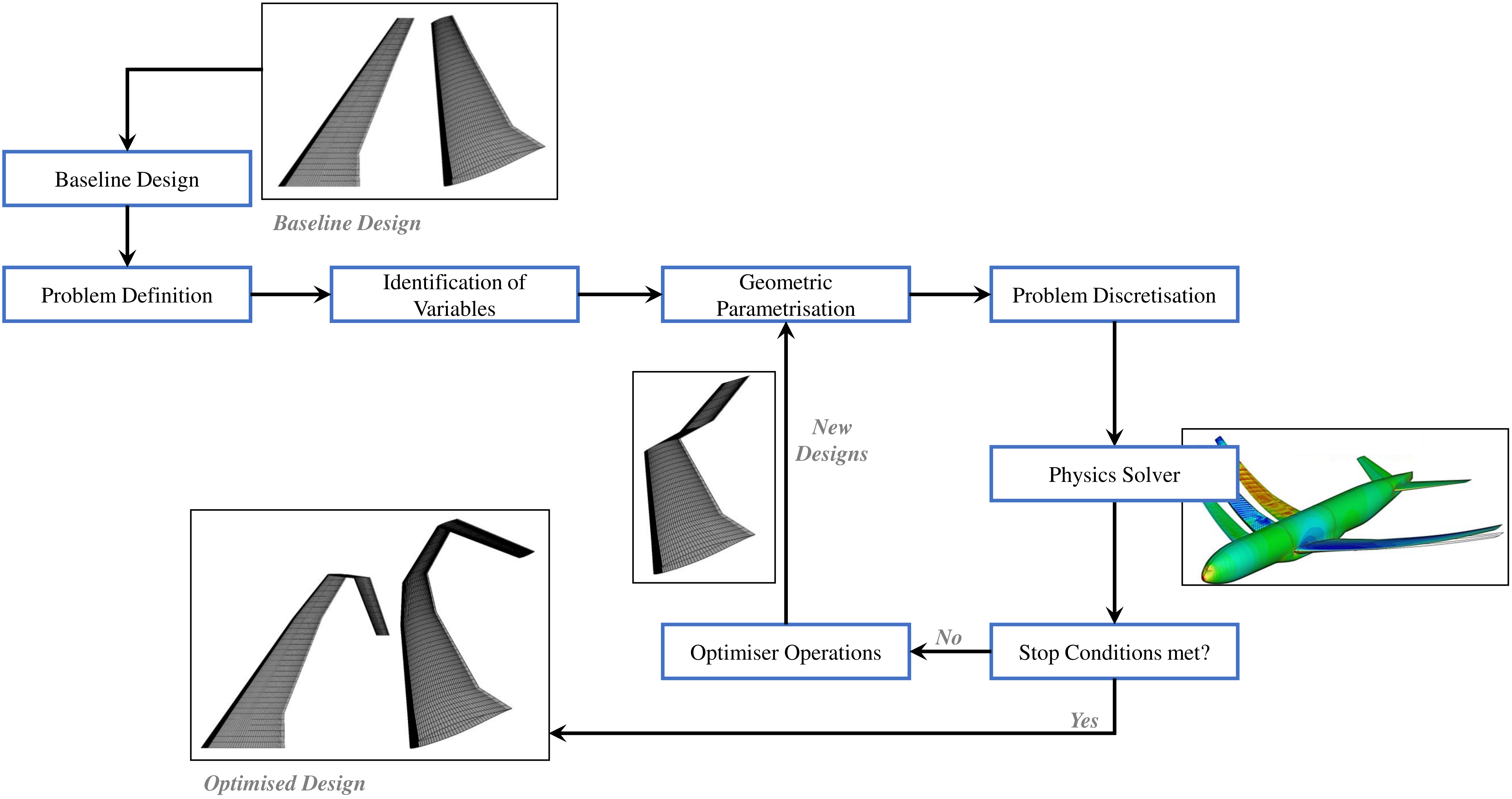
To appear in: *Applied Soft Computing*

Received date: 18-7-2016
Revised date: 13-9-2017
Accepted date: 16-9-2017

Please cite this article as: S.N. Skinner, H. Zare-Behtash, State-of-the-Art in Aerodynamic Shape Optimisation Methods, *<![CDATA[Applied Soft Computing Journal]]>* (2017), https://doi.org/10.1016/j.asoc.2017.09.030

- This paper examines various aerodynamic optimisation methods.
- Benefits and drawback of architectures are discussed relating to aerodynamic optimisation.
- Difficulties and applicability of algorithms to different design criteria are highlighted.

*Graphical abstract (for review)



State-of-the-Art in Aerodynamic Shape Optimisation Methods

S. N. Skinner* and H. Zare-Behtash
[1]*University of Glasgow, School of Engineering, Glasgow, G12 8QQ, UK*
**s.skinner.1@research.gla.ac.uk*

Abstract

Aerodynamic optimisation has become an indispensable component for any aerodynamic design over the past 60 years, with applications to aircraft, cars, trains, bridges, wind turbines, internal pipe flows, and cavities, among others, and is thus relevant in many facets of technology. With advancements in computational power, automated design optimisation procedures have become more competent, however, there is an ambiguity and bias throughout the literature with regards to relative performance of optimisation architectures and employed algorithms. This paper provides a well-balanced critical review of the dominant optimisation approaches that have been integrated with aerodynamic theory for the purpose of shape optimisation. A total of 229 papers, published in more than 120 journals and conference proceedings, have been classified into 6 different optimisation algorithm approaches. The material cited includes some of the most well-established authors and publications in the field of aerodynamic optimisation. This paper aims to eliminate bias toward certain algorithms by analysing the limitations, drawbacks, and the benefits of the most utilised optimisation approaches. This review provides comprehensive but straightforward insight for non-specialists and reference detailing the current state for specialist practitioners.

Abbreviations

ACARE	Advisory Council for Aeronautics Research in Europe
ADODG	Aerodynamic Design Optimisation Discussion Group
ALPSO	Augmented Lagrange Particle Swarm Optimisation
ARMOGA	Adaptive Range Multi-Objective Genetic Algorithm
CFD	Computational Fluid Dynamics
CRM	Common Research Model
EI	Expected Improvement
ES	Evolutionary Strategy
FFD	Free-Form Deformation
GBM	Gradient-Based Method
GDEA	Genetic Diversity Evolutionary Algorithm
GA	Genetic Algorithm
HM	Hybrid Method
LHS	Latin Hypercube Sampling
MDO	Multi-disciplinary Design Optimisation
MIGA	Multi-Island Genetic Algorithm
MO	Multi-Objective
MS	Multi-Start
NN	Neural Network
NSGA	Non-donated Sorting-based Genetic Algorithm
NURBS	Non-Uniform Rational Basis Spline
PDE	Partial Differential Equation
PSO	Particle Swarm Optimisation
RANS	Reynolds-Averaged Navier-Stokes
RBF	Radial Basis Function
SA	Simulated Annealing
SIMPSA	Simplex Simulated Annealing
SLSQP	Sequential Least Squares Programming
SNOPT	Sparse Nonlinear Optimiser
SQP	Sequential Programming
sGA	population Structured Genetic Algorithm
μGA	Micro-Genetic Algorithm

1. INTRODUCTION

Aerodynamic shape optimisation has become an indispensable component for any effective and robust aerodynamic design. Between now and 2030, there will be an estimated global demand for approximately 27,000 new passenger aircraft potentially worth up to £2.3 trillion. These aircraft must comply with strategic research agenda developed by the Advisory Council for Aeronautics Research in Europe (ACARE) which aim to enforce strict emission targets - CO_2 emissions per passenger kilometre to be reduced by 75%, NO_x emissions by 90% and perceived noise by 65%, all relative to the year 2000.[1] This can only be achieved by marrying novel light weight and flexible materials, e.g. composites, with a highly optimised aerodynamic aircraft configurations; the airframe contribution should be in the order of 20 to 25% for fuel consumption reduction. The impact of aviation on the environment is now a main driving factor affecting the aerodynamics of future aircraft.[2]

Optimisation is the process of obtaining the most suitable solution to a given problem, while for a specific problem only a single solution may exist, and for other problems there may exist multiple potential solutions. Thus, optimisation is the process of finding the 'best' solution, where 'best' implies that the solution is not the exact solution but is sufficiently superior.[3] Most current design optimisation approaches are heavily dependant on user training and experience requiring an array of specialised optimisation tools and compact shape parametrisation. This constitutes a major obstacle to robustness and reliability.[4] Another persistent difficulty in aerodynamic optimisation is the ability to define an analysis method that is capable of operating as many time as required (often thousands of times) and integrating it appropriately with an optimisation strategy. The methods employed must execute with realistic run times, dependant on computational resource, but must also be sophisticated enough to capture enough information to analyse local geometry that feeds into a globally optimal system.[5]

Many optimisation problems, especially those involved with large design spaces with coupled variables, inherently fall into the category of multi-disciplinary design optimisation (MDO), which in turn require a multi-objective (MO) compromise to be an effective design. The main motivation for applying MDO is that the performance of a real system is driven not only by the performance of individual disciplines but also by their coupled interactions. It is no longer acceptable to consider the aerodynamic analysis alone, its far reaching coupled effects to other disciplines must also be taken into account if a truly optimal design is to be reached. Studies, such as that by Werter and Breuker,[6] have shown that aeroelastic tailoring through aerostructural optimisation offer clear performance advantages. A survey of developments in multi-disciplinary optimisation for aerospace applications is presented by Sobieszczanski-Sobieski and Haftka.[7] Furthermore, Wunderlich[8] provide a summary of multi-disciplinary considerations.

Forums for comparing aerodynamic optimisation, such as the AIAA Drag Prediction Workshop[9,10], are well established and enable different research groups (such as the MDO laboratory led by Professor Martins at the University of Michigan) to validate their codes utilising common research models (CRM). Additionally, researchers in the aerodynamic design optimisation community have developed the AIAA Aerodynamic Design Optimisation Discussion Group (ADODG). The ADODG define benchmark optimisation problems enabling research groups to test and compare optimisation codes for various problems; a broad scope of flow conditions and geometries are considered. Several studies utilising common research benchmarks are suggested.[11–15] Furthermore, new comers into the field of optimisation are directed to Rios and Sahinidis[16], and Amaran et al.[17] offering comprehensive reviews of accessible generalised optimisation algorithms and their applications examining and contrasting relative performance.

This study reviewed a total of 304 papers considering more than 120 peer reviewed journals and several well-established conference proceedings; from which 229 papers have been selected to provide a comprehensive and well-balanced review of optimisation algorithms that are dominant in the field of aerodynamic optimisation. This review begins with a brief examination of general considerations required for aerodynamic optimisations. The fundamental operation, to-date research application, and challenges encountered relating to the optimisation strategies employed are analysed. The review discusses studies from several areas for aerodynamic design optimisation also covering associated multi-disciplinary analysis.

2. GENERAL CONSIDERATIONS OF AERODYNAMICALLY ORIENTATED OPTIMISATION

A. Basic Problem Formulation

The field of optimisation is expansive, and the choice of a suitable algorithm is highly problem dependant.[18] A general optimisation problem can be presented mathematically as:[19]

$$
\begin{array}{rlll}
\mathit{Minimise}\left\{\right. & F(\mathbf{X}) & & \mathit{Objective\ function} \\
\mathit{With\ respect\ to}\left\{\right. & \mathbf{X} & & \mathit{Design\ variables} \\
\mathit{Subject\ to}\left\{\begin{array}{l} \\ \\ \\ \end{array}\right. & \begin{array}{l} g_i(\mathbf{X}) \leq 0 \\ h_j(\mathbf{X}) = 0 \\ X_k^l \leq X_k \leq X_k^u \end{array} & \begin{array}{l} i = 1, l \\ j = 1, l \\ k = 1, N_{DV} \end{array} & \begin{array}{l} \mathit{Inequality\ constraints} \\ \mathit{Equality\ constraints} \\ \mathit{Parameterised\ constraints} \end{array}
\end{array}
$$

$$where\ \mathbf{X} = \left\{ \begin{array}{c} X_1 \\ X_2 \\ \vdots \\ X_{N_{DV}} \end{array} \right\}$$

Most optimisation methods use an iterative procedure. The initial set $\mathbf{X}$ design variables, which in the context of aerodynamic optimisation this is referred to as the baseline configuration, and is updated until a minimum of $F(\mathbf{X})$ is identified or the optimisation process runs out of allocated time/iterations. In the initial set-up of the optimisation problem consideration must be given to: 1) the level of information fidelity required from the flow solver, dependant on the type of problem; 2) scope of parametrised design space; 3) types of design variables, e.g. discrete and/or continuous; 4) single or multi-objective optimisation; 5) constraints handling; 6) properties of the design space, e.g. number of local optima, discontinuities. It is important to note that no optimisation procedure guarantees the global optima of the objective function $F(\mathbf{X})$ will be found: the process may only converge towards a locally optimal solution. Typically in this situation there are three possibilities: 1) restart the optimisation process to investigate if the same solution is found; 2) approach the design problem with a different optimisation methodology to compare solution quality at a high computational expense; or 3) accept the optimum found knowing that while it is superior to the baseline configuration it may not be the optimal solution.

B. Design Variables and Geometric Representation

In general, it is necessary to implement geometric parametrisations in such a way that reduce the complexity and cost of the optimisation process but do not restrict communication of variables or the degree to which aerodynamic performance is optimised. Parametrisation aims to balance the fundamental compromise between computational speed of the optimisation run-time, favouring a tight parametrisation. Zhang et al.[20] show that the defined dimensionality of a problem for shape optimisations can restrict the optimal design. Using too few variables may prove certain potential improvements impossible. Conversely, if too many design variables are used, particularly if variables are strongly coupled, the search landscape can become intractably complex to navigate. Increasing the dimensionality of a given problem excessively leads to a paradox, first addressed by Sobieszczanski-Sobieski,[21,22] in which increasing the number of design variables leads to a decrease in the number of variables that can be manipulated as a direct result of increased coupling. It is often desirable to limit the allowable design variables to avoid geometries that cannot be evaluated with sufficient accuracy by the flow solver: due to meshing limitations for example. Furthermore, this can help to avoid geometries that are unacceptable in terms of some criteria, or similarly, restrict the optimisation to geometries that are necessary for other criteria. Regardless of the user defined parametrisation, the final design is most definitely suboptimal - often limited by parametrisation.[4]

Chernukin and Zingg[23] conducted one of the few studies on how the number of variables used, and the related modality, can affect aerodynamic designs and highlight that distinguishing between multi-modality and poor optimiser convergence can prove problematic. By increasing the dimensionality of a design space it can be expected, but not guaranteed, to increase the modality of the search space. Initialising 224 random starting geometrics with 368 design variables they demonstrated the presence of at least 8 local optima for a blended wing optimisation, shown in Figure 1. All 8 solutions satisfied both optimality and feasibility tolerances with the objective value varying by approximately 5% between the local optima. The planform shapes are distinct and so demonstrate that geometric variation is significant between local optima which share similar performance characteristics.

Furthermore, the method of geometric parametrisation used to communicate a set of variables plays an important role in identifying optimal aerodynamics. It determines what shapes and topologies can be represented, and how many design variables are necessary for sufficient representation of the geometry. Thus, parametrisation dictates particular geometric requirements and has a strong influence on the design landscape. Therefore it cannot be precluded that different geometric parametrisations will increase or decrease the degree of modality, linearity, or discontinuity observed. Additionally, a complex geometry parametrisation may impose distinct computational costs. Representations of a geometry can be broken down into a number of categories but in a more broad sense they can be considered to be constructive, deformative, or volume based.

Constructive models include functions which define basic body shapes, spline methods (such as Bezier splines, basis splines (B-splines), non-uniform rational basis spline (NURBS)) and partial differential equations. Jansen et al.[24] used a medium-fidelity aerostructural panel code to perform optimisation of a conceptual wing configurations, shown in Figure 2. The basic wing topology was defined through a series of globally enforced geometric variables to manipulate a series of wing sections. Parametrising the entire geometry in this way typically allows for global shape control with few basic variables. This method is well suited to low-fidelity aerodynamic models if a wide allowable design scope is necessary - no need for mesh deformations.

Spline-based geometric parametrisations are used to represent two- or three- dimensional surfaces and are typically used in conjunction with higher-fidelity flow solvers, such as Euler and Navier Stokes solvers, with the control points being the design variables. Bezier splines are most efficient to evaluate requiring few variables and have been used for efficient aerofoil definition by Peigin and Epstein.[25] Modification of any single control point defining a Bezier spline will modify the entire curve and thus is inherently effective for global shape definition, but has very limited local control. B-splines address this issue of local control allowing single control point modifications to modify small portions of the overall curve. This allows for more complex aerofoil definitions, as demonstrated by Koziel at al.,[26] and can enable the use of hinged control surfaces to an otherwise rigid body. NURBS increase the local deformation control over surface definitions further in order to have more complex geometric shapes such as fairings or wing-fuselage junctions. Vecchia and Nicolosi[27] and Hashimoto et al.[28] adopt NURBS to parametrise the entire aircraft configuration in order to reduce drag of the vehicle through steam-lining fillets and fairings. Figure 3 shows an example of NURBS control points re-defining the surface over the upper section of the fuselage/wing juncture.

Geometry definition through the use of partial differential equations (PDEs) are not as commonly used as well-established spline-based methods but are just as versatile for geometry surface definition. Athanasopoulos et al.[29] show that for equivalently complex surface construction PDEs require fewer design variables, resulting in a more compact design space. Due to the small set of design parameters required by the PDE method the computational cost associated with the optimisation of a given aerodynamic surfaces can be reduced.[30] In a PDE-based method the parameters are boundary values to the PDE, hence the relationship between the value of the design parameter and the geometry can be unclear making methodical surface deformations tedious. This is likely why the aerodynamic definition of a body in an optimisation scheme does not used PDE representation even though it may initially seem a more appropriate method. Comparatively, spline-based methods are conceptually simpler and will provide a more direct relationship between design parameters and the resulting geometry and thus allow better control over the range of geometries that can be generated.

If optimisation establishes performance metrics from computational fluid dynamics (CFD), the simplest methods for body surface definitions are deformative ones. In deformative methods the mesh points on the surface of the body are directly treated as design variables,[31] and their position can be perturbed by the optimiser in order to generate new shapes. These approaches have the significant advantage that any geometry the mesh generation algorithm is capable of can be evaluated, however it is likely to require many hundreds of design variables; deformations are therefore usually limited to single-degree-of-freedom deformations. A common method used for aerodynamic optimisation is the free-form deformation (FFD) approach which is useful if the the geometry manipulations are particularly complex; FFD is covered in depth by Kenway and Martins.[32] This approach embeds the solid geometry within a FFD hull volume (volumes are typically trivariate analogues of Bezier splines, B-splines of NURBS), which are parametrised by a series of control points as shown in Figure 4. These control points deform the volume which translate to geometric changes of the solid geometry rather than redefining the whole geometry itself which can give a relatively more efficient set of design variables. A key assertion of the FFD approach, when applied within a CFD environment, is that a geometry has constant topology throughout the optimisation process;[12] this is typical of high-fidelity optimisations where the initial geometry considered is sufficiently close to the optimal solution. Figure 4 shows the FFD hull volume enclosing a wing with 720 geometric control points used by Lyu et al.[12] which control shape deformation in the vertical (z) axis. The initial random wing deformation and associated optimised wing cross-sections at select locations are also shown. A similar method is based on radial basis function (RBF) interpolation which defines data sets of design variables and their global relationships. Fincham and Friswell[33] use radial basis functions to optimise morphing aerofoils and report that they provide a means to deform both aerodynamic and structural meshes and interpolate performance metrics between two non-coincident meshes.

Volumetric-based body representation have been used for optimisation but rarely in the field of aerodynamics, a recent review of the applicability of volumetric parametrisation for aerodynamic optimisation is given by Hall et al.[34] The authors point out the limitations of volumetric representations stating that black-box optimisers cannot be used and even gradient-based methods can often be impractical.

C. Constraint Handling

Constraint handling in aerodynamic, and indeed any industrial optimisation problem, plays a consequential role in the quality and robustness of an optimised solution within the defined design space. Geometric parametrisation itself poses a constrained optimisation problem since, in addition to minimising the objective $F(\mathbf{X})$, the design variables must satisfy some geometric constraints. Constraint management techniques found in literature which have been classified by Koziel and Michalewicz[35] and Sienz and Innocente[36] as: 1) strategies that preserve only feasible solutions with no constraint violations: infeasible solutions are deleted; 2) strategies that allow feasible and infeasible solutions to co-exist in a population, however penalty functions penalise the infeasible solutions (constraint based reasoning); 3) strategies that create feasible solutions only; 4) strategies that artificially modify solutions to boundary constraints if boundaries are exceeded; and 5) strategies that repair/modify infeasible solutions.

Most commonly optimisations apply weighted penalties to the objective function if the constraint(s) are violated. The reason for this is that penalty functions are often deemed to ease the optimisation process, and bring the advantage of transforming constrained problems into unconstrained one by directly enforcing the penalties directly to the objective function. With this method Pareto-optimal solutions with good diversity and reliable convergence for many algorithms can be obtained easily when the number of constraints are small; fewer than 20 constraints. It becomes more difficult to reach Pareto-optimal solutions efficiently as the number of constraints increase, and the number of analyses of objectives and constraints quickly becomes prohibitively expensive for many applications. This is because the selection pressure decreases due to the reduced region in which feasible solutions exist.[37]

Kato et al.[38] suggest that in certain circumstances Pareto-optimal solutions may exist in-between regions of solution feasibility and infeasibility. This is illustrated in Figure 5, where it is seen that feasible and infeasible solutions could be evaluated in parallel to guide the optimisation search direction towards feasible design spaces. This is intuitively true for single discipline aerodynamic optimisation problems where often small modifications to design variables can largely impact the performance rendering designs infeasible. Algorithm understanding of infeasible solutions can help in the betterment of feasible solutions though algorithm learning/training and constraint based reasoning. Robinson et al.[39], comparing the performance of alternative trust-region constraint handling methods, showed that reapplying knowledge of constraint information to a variable complexity wing design optimisation problem reduced high-fidelity function calls by 58% and additionally compare the performance to alternative constraint managed techniques

Elsewhere, Gemma and Mastroddi[40] demonstrated that for the multi-disciplinary, multi-objective aircraft optimisations the objective space of feasible and infeasible design candidates are likely to share no such definitive boundary as shown in Figure 6. With the adoption of flutter constraints, structural constraints, and mission constraints solutions defined as infeasible under certain conditions would otherwise be accepted, hence forming complex Pareto fronts. Interdisciplinary considerations such as this help to develop and balance conflicting constraints. For example, structural properties which may be considered feasible, but are perhaps heavier than necessary will inflict aeroelastic instabilities at lower frequencies.

In the aerospace industry alone there are several devoted open-source aerodynamic optimisation algorithms with built-in constraint handling capability. For example COBYLA[41–43], DIRECT[44–47], NOMAD[48], and HAVOC[49,50] each offer derivative-free optimisation algorithms capable of handling constraints explicitly; each adaptable to the users aerodynamic solver whether commercial or in-house. Some studies[51–53] have also adopted MATLAB's optimisation tool-box for successful optimisation constraint management.

D. Problem Discretisation

Problem discretisation is achieved though surface and volume meshing/panelling of the fluid domain in order to formulate a discretised representation of the computational domain for which applied physics solvers provide a numerical solution too.

A persistent challenge in the application of numerical simulations (whether high- or low-fidelity) is the accurate evaluation of the optimisation objective function which is strongly dependant upon the body/domain discretisation. Given that multiple iterations are required within an individual optimisation process, it is necessary to maintain a balance between efficiency and accuracy; particularly if one employs multi-start or multi-point optimisation. For this reason, many aerodynamic shape CFD-based optimisations use a fixed mesh constructed around some near optimal baseline geometry. Zhang et al.[54] perform aeroelastic optimisation with flutter constraints, which they avoid convergence issues associated with dynamic mesh deformation by employing a fixed over-set mesh strategy. However, the task of creating a mesh *a priori* which remains appropriate for the whole design search space is often infeasible. Certain parameters, such as induced drag, are particularly sensitive to the grid size, hence aerodynamic properties of a body must be shown to be grid independent for the entirety of the design space for the mesh employed. If the aerodynamic properties of a body are not grid independent it is very easy for the mesh to become inappropriate as geometry and/or flow conditions change during the optimisation process. This can lead to ill-defined optimisation information and render the optimisation process useless as mesh discretisation errors are exploited leading to false designs.[55]

Developments in robust CFD mesh adaptation/refinement and deformation have helped make optimisation procedures more reliable and have made it possible to expand the scope of design parameters.[56,57] This enabling a more flexible design space. Adaptive meshes help in the progressive optimisation of a design, particularly when compressible aerodynamics is considered. Li and Hartmann[58] compare aerofoil optimisations for a fixed mesh and an adaptively refined mesh. They demonstrate that the adaptive meshing, reducing discretisation error, was not only computationally cheaper but capable of finding better solutions. The fixed mesh was unable to sufficiently evaluate shock discontinuities at the surface of the body. Nemec and Aftosmis,[55] in the optimisation of a supersonic inverse-design problem, also show that progressive mesh adaptation reduces the required computational effort during early design iterations and improves objective function convergence. Adaptive meshing can however introduce issues with optimiser convergence if poorly implemented. These issues may include unnecessary design evaluations for the course mesh, but also fine mesh iterations to undo/reverse false progress.[55]

An alternative to adaptive meshing is used by Lyu et al.[12] called multi-level optimisation. This involves the systematic increase in mesh fidelity as certain stages of optimality is achieved; the mesh stages used by Lyu et al.[12] are shown in Figure 7. These mesh stages would be developed before the optimisation and would allow the course grid to do much of the optimisation reducing the overall computational demand. When moving to the next mesh level for higher-fidelity optimisation, care must be taken as the updated mesh will compute different performance metrics from the previous mesh.

For aeroelastic problems in which the geometry is expected to structurally deform, or the definition of a perturbed volume mesh is needed, mesh deformations must be considered. Mesh perturbing algorithms are common practice however special attention must be given to how node translations and rotations effect the orthogonality of boundary layer mesh; this will drastically change results obtained and may give rise to a situations of false optimisation. Chiba et al.[59] highlight that independent modification to the surface and volume meshes, which are dictated by changes in the surface geometry, can result in surface mesh distortion. Figure 8 shows an example of surface mesh distortion around the leading edge of the wing. The authors claim this to be a primary reason for restrictive optimisation and therefore concluded that the mesh distortions highly limit the reachable design space consequently resulting in either sub-optimal or pseudo-optimal results. Elsewhere mesh deformations have been employed successfully for high-fidelity aerodynamic[60–62] and aerostructural design,[13,63] morphing wing structures,[64] and ducted propeller optimisation[65].

For reduced, lower-fidelity models, such as vortex lattice methods the discretisation of the surface is much simpler, however if the body geometry is allowed to vary dynamically throughout the optimisation the spatial resolution of the panelling should be adjusted accordingly to maintain consistent numerical accuracy. Ning and Kroo[66] achieve this by discretising the wing span length by the panel size. This forced the dimensions of the wing to vary discontinuously. Jansen et al.[24] allowed the span to vary continuously by dividing the wing into independent segments and then assessing their contribution to the overall continuous span. Similar methods were used by Skinner and Zare-Behtash.[67] Methods used by Vecchia and Nicolosi[27] fix the panels over the body and so do not allow significant geometric span changes; shed wake panels through the aerodynamic analysis also have a fixed size.

3. APPLICATION OF GRADIENT-BASED METHODS TO AERODYNAMIC OPTIMISATIONS

Gradient-based optimisation is a calculus-based point-by-point technique that relies on the gradient (derivative) information of the objective function with respect to a number of independent variables. The nature in which gradient-based methods (GBM) operate make them well suited to finding locally optimal solutions but may struggle to find the global optimal.[68] With gradient-based algorithms an understanding of the design space is assumed, as an appropriately pre-conceived starting design point must be given. Kenway and Martins[13] point out that with increasingly higher fidelity aerodynamic optimisations, a more refined initial design should be used so that the optimisation does not diverge too far from the baseline. If large changes in topology are expected lower fidelity panel codes, such as that described by Vecchia and Nicolosi,[27] can facilitate useful optimisation procedures. Typically, the higher the fidelity analysis used the more compact the design variables will need to be to allow effective optimisation with a gradient based optimiser.

Gradient-based optimisation is, in its most basic form, a two step iterative process which can be summarised mathematically as:

$$\mathbf{X}^{new} = \mathbf{X}^{old} + h\nabla f \tag{1}$$

where ∇f is the gradient of function $F(\mathbf{X})$, and $\mathbf{X}$ is a vector of the design variables. The first step is to identify a search direction (gradient), ∇f, in which to move. The second step is to perform a one-dimensional line search to determine a distance/step size h along ∇f that achieves an adequate reduction of some cost function, i.e. define how far to move in the search direction until no more progress can be made.[69] A schematic diagram illustrating the operation of a gradient-based optimisation is shown in Figure 9. In-depth benchmarking of gradient based algorithms for aerodynamic problems has been conducted by Secanell and Suleman[70] and Lyu et al.[71]

Gradient based algorithms are extensively used in aerospace optimisation as they exhibit low computational demands when handing many hundreds of design variables - this makes them well suited for optimising shapes based on deformative geometric parametrisations.[12,64] Significant difficulties arise if they are not applied within a restricted set of functions with well defined slope values due to a dependency upon the existence of derivative information via some sensitivity analysis. There are several different methods for sensitivity analysis for which four general classes can be distinguished: 1) finite-difference methods; 2) complex-step derivative approximation; 3) automatic/algorithmic differentiation; and 4) analytic methods. It is important to understand their relative merits since none are a clear choice for all classes of problem. Comparative studies on the numerical sensitivity analysis for aerodynamic optimisation has been conducted by Martins et al.[72] and Peter and Dwight,[73] while Martins and Hwang[74] offer a detailed discussion for computing derivatives within multi-disciplinary computational models.

The computational expense of evaluating gradients using finite-difference or the complex-step method provide a simple and flexible means of estimating gradient information, but are considered excessive with respect to hundreds of variables.[75] These approaches preserve discipline feasibility, but they are costly and can be unreliable. Finite-differencing, while not used to provide gradients for the optimisation itself have been used by Kenway and Martins[13] and Skinner and Zare-Behtash[67] to provide gradients for stability derivative constraints. For a restricted number of design variables the complex-step method is suitable for sensitivity analysis as demonstrated by Kenway and Martins.[76] They employ the complex step method to provided gradients for the Sparse Nonlinear Optimiser (SNOPT) algorithm, originally developed by Gill et al.[77], in the constrained optimisation of wind turbine blades. It is commented on that to increase the dimensionality of the problem an analytic sensitivity analysis would have to be adopted.

Finite-differencing or complex-step methods employed for providing sensitivity analysis for low-fidelity codes can be considered appropriate due to low computational demand. Ning and Kroo[66] optimise a series of wing topologies investigating fundamental wing design trade-offs for which sensitivity analysis of the objective and constraints were approximated by finite-differencing. Results provided by the sequential quadratic programming method show robust and quick convergence able to determine relative gradients between approximated area-dependant weight, effects of critical structural loading, and stall speed constraints.

In the presence of several hundred design variables and constraints the analysis code will require a particularly long time to evaluate sensitivities. Automatic/algorithmic differentiation or analytic derivative calculations (direct or adjoint) can be used to avoid multi-discipline analysis evaluations. Pironneau,[78] pioneered the adjoint method in fluid dynamics, showing that the cost of computing sensitivity information was almost completely independent of the number of design variables, and hence the overall cost of optimisation is roughly linearly proportional to the number of design variables. Lyu et al.[71] more recently demonstrated that both SNOPT and sequential least squares programming[79,80] (SLSQP) gradient-based algorithms with analytically derived adjoint gradients require far fewer total functions calls when compared to using finite-differencing for high-fidelity large scale aerodynamic optimisations. The adjoint form of the sensitivity information is particularly efficient for aerodynamic optimisation applications as the number of cost functions (outputs) is small, while the number of design variables (inputs) is relatively larger.

The discrete adjoint method (as opposed to continuous adjoint method) is generally favoured in aerospace-based optimisation as it ensures that sensitivities are exact with respect to the discretised objective function.[81,82] The implementation of the adjoint method for the governing equations of the flow analysis can often be difficult to derive and require direct manipulation; adjoint methods require much more involved detailed knowledge of the computational domain. One way to approach this difficulty is to use automatic/algorithmic differentiation, which is a method based on the systematic application of the differentiation chain rule to the source code to compute the partial derivatives required by the adjoint method. Mader et al.[83] developed a discrete adjoint method for Euler equations using automatic differentiation, later followed by Lyu et al.[84] who extended and developed this adjoint implementation to Reynolds-averaged Navier-Stokes (RANS) equations and introduced simplifications to the automatic differentiation approach. Methods developed have shown robust an efficient application to high-fidelity optimisation.[12,63]

Hicken and Zingg[85] adopted similar methods for the high-fidelity aerodynamic optimisation of non-planar wings addressing the non-linearity of wake shape and how it can impact the induced drag. Several non-planar geometries, inherently creating non-planar wake-wing interactions, are optimised using discrete adjoint sensitivities and the SNOPT algorithm. This work illustrates the drawbacks in static-wake assumptions, demonstrating that higher-order effects must be included for accurate induced drag prediction and hence for meaningful optimisations. This work was followed by Gagnon and Hicken[86] for the aerodynamic optimisation of un-conventional aircraft configurations; adapted optimisation results are shown in Figure 10. Here, for the aerodynamic metrics, the gradients are evaluated analytically by using the discrete-adjoint variables while other gradients are provided by the complex-step method. This work is notable as it enables used axial deformation combined with free-form deformation to achieve both local and global geometric manipulations, the effects of this can be seen in Figure 10, allowing span, sweep, dihedral, taper, twist and aerofoil sectional shape changes. These global geometric variables are generally not considered in high-fidelity simulation. The cost of allowing such geometric variation away from the baseline under high-fidelity optimisation limited how many variables could be considered in any one optimisation process. The authors observed limited optimisation in some wing configurations because of this.

Aeroelastic optimisation requires the coupling of aerodynamic and structural models for most effective sensitivity analysis in optimisation routines. Even small changes in aerodynamic shape can have a large influence on aerodynamic performance with various flow conditions resulting in multiple shapes. Wing flexibility impacts not only the static flying shape but also it's dynamics, resulting in aeroelastic phenomenon such as flutter and aileron reversal. Based on this principle, to enable high-fidelity aerostructural optimisation while encompassing hundreds of design variables, Martins et al.[87] proposed the use of a coupled adjoint method to compute sensitivities with respect to both the aerodynamic shape and the structural sizing. Kenway et al.[13,88] subsequently made several developments and demonstrated that the computation of coupled aeroelastic gradient calculations were scalable to thousands of design variables and millions of degrees of freedom, and since applied it to the aerostructural optimisation of high aspect ratio wings with different structural properties.[89] More recently, Burdette et al.[64,90] applied the coupled discrete adjoint method with the sparse non-linear optimiser SNOPT for wing morphology optimisation. This approach was capable of handling over a thousand design variables and constraints. The coupled adjoint method is also applicable to lower-fidelity models in studies like that by Elahm and Tooren,[91] where they used a vortex lattice method and finite element analysis tool capable of accurately mimicking high-fidelity accuracy at a greatly reduced computational cost.

The coupling of design constraints makes the optimiser additionally capable of considering more sophisticated criteria. Mader and Martins[92] included flight dynamics into the coupled adjoint sensitivity and explored the use of static and dynamic stability constraints. Their result showed that coupling stability constraint sensitivities into the adjoint formulation had a significant impact on optimal wing shape. Elsewhere, structural dynamics were considered by Zhang et al.[54] who used a coupled-adjoint formulation to include flutter constraints. The flutter constraints used the coupled aerodynamic/structural solver to suppress flutter onset by identifying dominant modes and adjusting variables such as the wing stiffness.

Grossman et al.[93] investigated using modular sensitivity analysis for aerostructural sequential optimisation of a sailplane. They showed that coupled aerostructural optimisation gave higher performance designs than those identified by sequential optimisation of aerodynamics followed by structural optimisation. Subsequently, Grossman et al.[94] optimised the performance of a subsonic wing configuration showing that while modular sensitivity analysis for sequential optimisation reduced the total number of function calls and sensitivity calculations, the wing performance gain was limited. When performing sequential optimisation the optimiser does not have sufficient information necessary for aeroelastic tailoring. This limitation of sequential optimisation is further explained by Chittick and Martins.[95]

A significant drawback of all gradient-based algorithms is the requirement for continuity and low-modality throughout the design space otherwise the algorithm may become sub-optimally trapped. The challenge is that an aerodynamic shape analysis throughout a geometrically varying search space will encounter both non-continuous topological and local flow changes, each providing local optima.[71,96,97] Gradient-dependant algorithms' robustness significantly decreases in the presence of discontinuity and lack of convergence, usually related to turbulence modelling, making the objective function noisy.[98] Kenway[99] encountered such a problem with aerodynamic shape optimisation with a separation-based constraint formulation to mitigate buffet-onset behaviour at a series of operating conditions. The discontinuity from the 'separation sensor' function arose from monitoring the wing local surface for separated flow; this resulting in locally negative skin friction coefficients. To address this issue blending functions were to be implemented to smooth the discontinuity, smearing the separation sensor value around the separated flow region.

Kenway and Martins,[99] among several others,[13,64,100,101] have used multi-point optimisation strategies in order to consider several operating conditions simultaneously. For more realistic and robust design it is crucial to take into account more than one operating condition, especially off-design conditions, which form additional multi-objective requirements into the optimisation. Figure 11 shows the results for both single-point and multi-point optimisation of the Common Research Model configuration presented by Kenway and Martins;[99] results are shown for the nominal operating condition. The single-point optimisation achieved an

8.6 drag count reduction and the shock-wave over the upper surface of the wing is almost entirely eliminated. Drag divergence curves in this work show the nature of the single-point optimisation presenting a significant dip in the drag at the design condition, but the performance is significantly deteriorated at off-design conditions relative to the baseline condition. The multi-point optimisation, accounting for 3 design conditions, found that drag at the nominal operating condition increased by 2.8 counts and produced double shocks on the upper surface of the wing visible in Figure 11. However, at the sacrifice of performance at the nominal operating condition, off-design conditions for the multi-point optimisation design was found to perform substantially better over the entire range of Mach numbers. Though biasing the optimisation toward certain operating conditions the authors show that multi-point optimisation with all conditions near the on-design condition is not sufficient for an overall robust design when considering operational envelopes.

4. APPLICATION OF GRADIENT-FREE METHODS TO AERODYNAMIC OPTIMISATIONS

The principal source of difficulty in the application of gradient-based optimisers is the requirement for having a non-discontinuous and mathematically predictable design space. Non-gradient based methods can prove more complex to implement than GBM, but they do not require continuity or predictability over the design space, and usually increase the likelihood of finding a global optimum.[102]

Methods of optimisation known as metaheuristics can offer robust methods of finding a solution, and increases the likelihood of converging onto a solution at the global optimum. These gradient-free methods are known to be capable of engaging with numerically noisy optimisation problems that be difficult for GBM. This is because metaheuristics operate from a completely different paradigms usually based on the some naturally occurring phenomenon.[103] Unlike gradient methods, derivatives of the cost functions are not necessary, allowing metaheuristics to easily cope with non-continuous or numerically noisy cost functions. Furthermore, no pre-defined baseline design or knowledge of the design space is required and gradient-free methods typically optimise several solutions in parallel.

A. Genetic Algorithms

Evolutionary computation is founded upon biological evolutionary theory of which there are four historical paradigms motivating activity within the field: Evolutionary Programming, Evolutionary Strategies (ES), Genetic Algorithms (GAs), and Genetic

Programming.[103] The basic differences between these lie in the nature of the representation schemes, the reproduction operators, and selection methods. Focus will be placed upon genetic algorithms as they have successfully been applied to a wide-base of aerodynamic design optimisations due to their ease of use, broad applicability, and global perspective of the search domain.

GAs are a population-based optimisation technique based on the Darwinian theory of survival of the fittest: a primary aspect of evolution.[104] These algorithms are often praised for their ability to explore and exploit solutions simultaneously due to their inherent multi-start capability. They are easily capable of constructing insightful design trade-off relationships, referred to as Pareto fronts, between objectives as shown in Figure 12. Figure 12 shows the resulting Pareto optimal solutions found by Yamazaki et al.[11] in a winglet design problem illustrating the trade-off between pure drag (induced+wave+profile) and root bending moment.

GAs are well-suited to complex optimisation tasks as they can easily use both discrete and continuous variables, and can easily handle non-linear, non-convex, and non-continuous objective functions.[51] Chiba et al.[105] suggest that GAs have four distinct advantages which encourage their use in aerodynamic/aerostructural optimisations: 1) GAs have the ability to find multiple optimal solutions and design trade-offs; 2) GAs process information in parallel, optimising from multiple points within the design space; 3) high-fidelity CFD codes can be adapted to GAs without any modifications; and 4) GAs are insensitive to numerical noise that may be present in the computation. The main drawbacks associated with these algorithms are high computational cost, poor constraints handling abilities, requirement for problem specific tuning and limitations in how many variables are feasible to handle. Studies have shown that GAs are very fast at identifying regions of optimality within a design space but demonstrate slow convergence as they moves nearer optimal solutions.[18] Some studies have tried to build on the classical GA to enhance its applications to aerodynamic optimisation.[106]

GA optimisation is, in its most basic form, an iterative process which can be summarised as:[103,107]

1. **Random generation** of individuals to form the initial population.
2. **Evaluation** of the fitness/survivability of each individual in the population to the given environment. This would be done with the aerodynamic solver.
3. **Selection** of individuals to take part in genetic operations.
4. Apply genetic operations which mimic **reproduction** to define a new population.

5. **Iterate** over steps 2-4 are over multiple generations until some convergence criterion is met.

Figure 13 illustrates this iterative process. Key influencers, each posing characteristic difficulties, in the construction of GAs suitable for aerodynamic optimisation problems include: 1) GA population size; 2) selection methods; 3) genetic operations (namely crossover[103,108] and mutation[103,109–111]); 4) genotype-phenotype mapping; 6) sufficient design constraints for adequate problem definition; and 7) computational resource. Insufficient selection of these factors can delay, if not prohibit, the performance of a GA in finding optimal solutions with the desired precision. Many variants of GAs exist in many innovative and abstract forms due to the array of different operators possible within the 'selection' and 'reproduction' stages.

Population size is very much related to the complexity of the problem and the number of design variables considered; understanding how the population size influences a particular problem is not a trivial task. A good selection of population size will improve both computation time and solution quality. It is generally argued that small populations can lead to premature convergence and poor solution optimisation and that larger populations may unnecessarily expend computational resources. Various generalised guidelines exist regarding appropriate population size and methods for tuning can be found in the literature.[112–114] Pandey et al.[115] present a detailed comparative review of approaches to prevent premature convergence based on several different factors affecting the GAs behaviour: initial population; population diversity; fitness function; search space scope Vs. selection pressure; problem difficulty Vs. number of individuals.

Selection methods and genetic operators exist in different forms and hold no strict rules on how they are implemented and is often down to user preferences. The most common selection methods for optimisations include: elitism;[97,111] roulette wheel;[103,107] and tournament selection.[107,116,117] These are somewhat independent of the genotype-phenotype mapping simply presenting alternative ways to select candidate solutions. After selecting candidate solutions the genetic operators breakdown and re-combination of schema (crossover operators) and perturb (mutation operations) random candidate variables. Genetic operations are dependant on genotype-phenotype mapping. The exact combination of genetic operations are often omitted from relevant literature however common examples of crossover include: single-point crossover;[97,110] two point crossover;[109–111] multi-point crossover;[118] gene-lottery;[119] uniform crossover;[110] and blended crossover (BLX-α)[120]. Mutation methods, and GAs as a whole, are covered in detail by Deb.[120]

Perhaps one of the most common GA variants across all disciplines is the non-dominated sorting-based multi-objective evolutionary algorithm-II (NSGA-II).[121] It is capable of both single and multi-objective optimisation, typically enforcing constraints through tournament selection and elitism. Lyu et al.[71] show that NSGA-II, for RANS-based optimisation, with a population of 24 for 200 generations struggled to handle more that eight variables incurring great computational expense and failed to meet the optimisation tolerance. Kim et al.[122] found NSGA-II to be sufficient for the aerodynamic and aeroacoustic optimisations of an axial-flow fan however only employed two design variables. Droandi and Gibertini[123] successfully applied the NSGA-II to the aerodynamic optimisation of tilt-rotor aircraft blade using a compressible Navier-Stokes solver. They define nine sections along the blade span for 27 variables and consider both single-point (helicopter mode or aeroplane mode operation) and multi-point (combined helicopter mode and aeroplane mode operations) optimisation. This study suggests NSGA-II can be expected to perform well in terms of Pareto optimisation tolerances due to the non-dominated organisation of the problem. This helps move the population toward Pareto optimality without suffering from convexity within the problem.

Genotype-phenotype mapping is a major influencer in a GAs performance. The genotype space presents the genetic blueprints or the DNA of the geometric search space which dictate the rules and traits of candidate solutions. The genotype is then translated through genotype-phenotype mapping to express a solution's attributes which add to or detract from its performance relative to the aerodynamic objective function. These are the solutions observable traits, referred to as the solutions phenotype, which feed into the numerical analysis.

Traditional GAs use binary encoding[124] to describe candidate solution genotype. In binary encoding, the number of exploitable schema is maximised[125] and the algorithm is able to converge quickly to a solution,[126] but can yield low-quality solutions when applied with many variables resulting. In binary encoded GAs, string length must be assigned *a priori* giving the algorithm discretised precision; the higher the precision required, the longer the string length and considerably slows the algorithm and memory requirements.[120,127] Avenues for decreasing the length of large binary chromosomes are covered by Mcgookin.[111]

A particular drawback in binary encoding is the existence of Hamming cliffs in the genotype search space.[128] This is a limitation caused by the discretisation of variables in genotype space. It requires the GA to simultaneously change the genotype bit representation in a very small and precise manner to achieve a more optimal solution. This is obviously an issue in consideration of aerodynamic optimisation as often small changes in variables can have large impact on performance, but its inevitable presence is often ignored. The probability that genetic operations will achieve this is unlikely while also the binary code does not preserve the locality of points in the phenotype space.[129] Additionally binary GAs often suffer from bias towards superior solutions in the early population stages due to genetic drift. Genetic drift will encourage premature convergence to sub-optimal solutions and is discussed in detail by Lim.[130] Work towards maintaining a heightened population genetic diversity ensures avoiding premature convergence,[131] where diversity is the volume of dissimilarity between individuals within a given population. Non-random mating has been shown to maintain genetic diversity.[132] Massaro and Benini[133] discuss a variety of relevant diversity preserving techniques and compare several binary encoded GA variants.

Benini,[134] Zhao et al.[135] and Skinner and Zare-Behtash[67] present different binary encoded GAs for aerodynamic design optimisation each maintaining diversity differently. Benini[134] use the Genetic Diversity Evolutionary Algorithm (GDEA) developed by Toffolo and Benini[136] in the optimisation of a transonic compressor in CFD. This algorithm uses a diversity preserving mechanism which treats the genetic diversity as an optimisation objective to establish a criterion for fitness assignments. The dissimilarity between two individuals would be measured mathematically in phenotype space by Euclidean distance; this then increases the chance of aerodynamically superior but geometrically different solutions moving into genetic operations. This enhanced exploitation and exploration within the design space involving 23 design variables optimising a population of 20 individuals for 100 generations with a run time of 2000 processing hours, with four-processor parallel computing.

Skinner and Zare-Behtash[67] developed a population structured genetic algorithm (sGA) with dynamically structured binary genotypes enabling alternative non-planar wing configurations of to exist in parallel. The sGA facilitated gene comparability by means of a genetic hierarchy that enables diverse configurations to be maintained simultaneously through the inclusion or exclusion of certain characteristics.[137] Additionally, the application of an adaptive mutation rate was used to balance global and local searching ranges of the design space. This enhanced the search in the initial stages of the optimisation and then reduced exploration in the later stages as the optimisation process matured. This enabled the sGA to identify solutions with higher potential early in the optimisation process, and in later stages focus on the betterment of solutions identified taking emphasis away from exploring for entirely new solutions. The design space involving 28 design variables optimising a population of 100 individuals for 600 generations with a run time of 3-4 processing hours using a mid-fidelity aerodynamic analysis code.

Zhao et al.[135] used a Multi-Island GA (MIGA)[138,139] to optimise laminar flow regions over an aerofoil. The MIGA operates by dividing the population into sub-populations isolating them in different regions of the design space. Solutions thus develop into independent groups under different parametrisation and are systematically allowed to migrate between 'islands' and interact after every few generations.[139] A population of 50 solutions divided over 5 sub-populations were evaluated over 50 generations with solution migration required every 4 generations. The high frequency of migrations required to maintain diversity and the low sub populations suggest that the algorithm was very prone to genetic drift.

Real-number encoding is more commonly used to resolve limitations of binary encoding, and is widely confirmed to be more efficient than binary.[140–143] In real-number encoding the genotype and phenotype spaces share an identical topological structure,[120] and has the advantage of dynamic coding and floating point representation to tackle large design spaces that require a continuous design space that would otherwise be discretised.[144] Real-encoding gives a robust search while keeping the string length small. In addition, real-encoding eliminates the need to define search boundaries for the algorithm as the distribution of solution candidates continuously moves in the direction of more promising regions within the design space.[145] If there are strict design variable limits then well-defined constraints which limit the algorithm variable boundaries must be used.

It is worthwhile indicating that ES are essentially the same as real-encoded GAs with the crossover operation removed and a less-random mutation methodology applied to influence parent solutions. Hence, ES are a perturbation based optimisation algorithm without the schema maintaining operations of a GA. ES are often deemed better for handling vast design spaces as demonstrated by Jones[146] and have been adopted in some instances for aerodynamic optimisations. Olhofer et al.[147], in the optimisation of turbine blades, indicate that ES can utilise smaller populations than GAs and achieve faster convergence thus reducing the computation time. However, it is indicated that subsequent optimisations are required to gradually increase the dimensionality of the problem to allow the ES to cope with a complex design spaces. Periaux et al.[148] employ a similar ES method in which they layer the optimisation algorithm's the population size, the mutation rate and CFD mesh refinement. It was found that the ES algorithm employed was unable to cope with multiple objectives or multi-disciplinary design problems.

Hashimoto et al.[28] explored the maximisation of fuselage/wing lift generation for high-wing aircraft configurations using RANS with design variables. They employ a real-encoded, adaptive range multi-objective genetic algorithm (ARMOGA)[149] to reduce the computational burden and the total number of function evaluations needed. ARMOGA was developed for integration with computationally demanding CFD-based optimisation to find multiple Pareto-optimal solutions more efficiently than conventional multi-objective GAs, such as NSGA-II[121]. This is due to their ability to perform concentrated searches, adapting their search regions based on population statistics as shown in Figure 14. In comparison, the search region of traditional GAs remain constant. When tackling large aerodynamic optimisations, particularly when using CFD, it is detrimental to the optimiser performance to generate many infeasible solutions. Modifying the search range of the algorithm makes it possible to estimate candidate solutions efficiently and prevent wasting computational resource on the generation and evaluation of poor solutions.[149] A clear conflict here is the preservation of population diversity. Figure 14 highlights that the population becomes confined into a reduced search space. This lowers computational expense, however it encourages a lack of genetic diversity. To counter this the ARMOGA uses population re-initialisation every few generations to randomly increase diversity. This preservation technique was adapted from micro-genetic algorithm (μGA)[150] which was designed to maintain small populations. The ultimate problem with this method is that even though there is a constant infusion of new schema at regular intervals, increasing the dissimilarity between aerodynamic configurations, facets of the schema are not allowed to propagate and develop; this can become a fundamental limitation.

Chiba et al.[59] also employed the ARMOGA and present one of the few optimisation studies which consider a GA for high-fidelity multi-disciplinary optimisation. In order to meet the computational demands in a reasonable time two super computers were employed. The ARMOGA used 35 design variables with 5 constraints to optimise an aeroelastic wing by varying the wing aerofoil shape, twist and dihedral distributions while maintaining wing topology. A population of 8 individuals for 19 generations with a population refresh rate and search region range adjustment every 5 generations. It is reported that the algorithm did not converged but sufficient optimality was reached. A single generation composed of approximately 70 Euler computations to perform static aeroelastic deformation, and 90 RANS computations for the aerodynamic evaluation of the deformed wing; equating to roughly 880 hours CPU time. Furthermore, multi-point optimisation for 3 flight conditions were considered which is notably unique for optimisation studies of this type. Solutions evaluated are shown in Figure 15, projecting all solutions evaluated in a three-dimensional space between the design objectives specified: minimisation of the block fuel, maximise take-off weight and maximise drag divergence. It is clear that the non-dominated solutions did not comprise a Pareto front and so no globally optimal trade-off was found.

Sasaki et al.[151] applied the ARMOGA to a transonic and supersonic wing design, optimising 72 design variables describing the wing shape and planform subject to four-objectives, and geometric constraints. The GA used a population of 64 due to the definition of a large search space. It was found that by tightening the definition of the thickness distribution constraints, more realistic wing solutions were identified. This highlights the capability of GAs in taking advantage of loose problem definitions, especially in multi-disciplinary optimisations. Sasaki et al.[152] point out that the GA solutions optimised for a cruising supersonic wing, with 105 variables and 15 constraints, satisfied all defined constraints but essentially over optimised the wing aerodynamics, thus compromising the structural integrity of the wing. This highlights the need for precise definitions of the problem via constraint handling to carefully reach optimised results. Optimal solutions are often found near, or on constraint boundaries, therefore, more thorough constraint definition will result in a more efficient algorithm and superior results.

Finally, a resource based problem, almost exclusively a problem for heuristic optimisers and of particular importance to GAs, is the need to ensure careful distribution of computational resources; this is reflected by the use of parallelism of GA in all applications. Typically aerodynamic evaluations are executed in parallel with the grid generations and associated flow calculations distributed over available processing elements.[145,153] The selection of topology connecting the processors is sensitive and will significantly effect the calculation efficiency. Wang et al.[153] present a study on the performance of GAs operating in parallel for the optimisation of aerofoils with high-lift devices and convergent-divergent nozzles. Tse and Chan[154] and Holst and Pulliam[155] provide aerodynamic optimisations where the order 10^3 function evaluations were required to reach convergence making the employment of parallel computing mandatory.

B. Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a stochastic population based (swarm) optimisation technique developed by Kennedy and Eberhart[156,157] in the 1990s. Mathematically, swarming is the collective decentralised motion of a large number of self-propelled entities as a collective animal behaviour and is exhibited by many living creatures such as birds, fish, and insects. Studies have praised the algorithm for easy implementation to aerodynamic solvers, its computationally inexpensive memory requirements, and simple mathematical operators.[158] PSO is developed from an agent orientated paradigm, meaning that particles are semi-autonomous agents capable of communicating and updating their status. This autonomy and ability for particles to sense their surrounding environment makes PSO an attractive gradient-free optimiser. However, the PSO is inherently unconstrained which can make handling variables, and enforcing design constraints difficult.[51] Zhang et al.[159] have carried out a comprehensive survey of several PSO variants outside of aerodynamic optimisation.

A swarm of particles represents a group of potential solutions moving in an n-dimensional space of design variables.[160] In the multi-dimensional design space, particles are assigned positions, representing the candidate solution, and velocity components which determines how the solution variables are updated. PSO algorithms, in their most basic form, can be summarised as:[160,161]

1. **Random generation** of particles in space to form the initial population with position and velocity vectors.

2. **Evaluation** of the fitness of each particle in the given environment.

3. **Update** each particle's velocity and position within the design space, adapting with regards

to its personal best position (*pbest*) and the global best swarm position (*gbest*).

4. **Iterate** over steps 2 to 3 until some convergence criterion is met.

Particles typically update their search direction based on their current position ($\mathbf{x}_k^i$), the particles best known position ($\mathbf{p}^i$), and swarm best known position ($\mathbf{p}^g$). Figure 16 illustrates this iterative process. The particles are typically manipulated according to equation (2), originally defined by Kennedy and Eberhart,[157] with three weighting factors: the inertia weighting factor, w, the cognitive acceleration factor, c_1, and the social acceleration factor, c_2, where R is uniform random numbers from interval $[0, 1]$. The original PSO algorithm uses constant values of 1, 2, and 2 for w, c_1, and c_2 respectively.[157]

$$\mathbf{v}_{k+1}^i = \underbrace{w\mathbf{v}_k^i}_{Current\ motion} + \underbrace{c_1\,\mathbf{R}(\mathbf{p}^i - \mathbf{x}_k^i)}_{Particle\ memory\ influence} + \underbrace{c_2\,\mathbf{R}(\mathbf{p}^g - \mathbf{x}_k^i)}_{Swarm\ influence} \tag{2}$$

The example for a particle searching with PSO is illustrated in Figure 17, where $\mathbf{x}_k^i$ is the position of particle i at iteration k, $\mathbf{v}_k^i$ is the current velocity of particle i at iteration k, $\mathbf{v}_{k+1}^i$ is its search direction, leading to an updated position $\mathbf{x}_{k+1}^i$ of the particle i for iteration $k+1$. Movement of a particle is governed by interactions of inertial current motion, cognitive influence and social influence as shown. With a strong emphasis on the social parameter, with a low to zero cognitive parameter, the algorithm will converge quickly to an initial best solution. When there is more emphasis on the cognitive parameter, with a low or zero social parameter, no global information is shared and the algorithm will not converge properly. From these interactions the velocity vector $\mathbf{v}_{k+1}^i$ is defined so that the particle moves to new position $\mathbf{x}_{k+1}^i$.

The basic algorithm described above has an undesirable property when $\mathbf{x}^i = \mathbf{p}^i = \mathbf{p}^g$ for any particle i. If this scenario arises, the velocity update equation (2) reduces to $w\mathbf{v}^i$. Therefore, when the velocity is close to zero, all particles will stop moving and converge prematurely. Key influences on the PSO algorithm performance in aerodynamic optimisation problems include: 1) swarm size and initialisation; 2) appropriate weighting of the inertial weight, and cognitive/social acceleration factors; 3) constraint-handling 4) computational resource.

Swarm size is directly related to the scale of the problem; with increasing shape dimensionality, more particles will be required to adequately search the design space for optimality. Shi and Eberhart[162] performed comprehensive studies on the effects of swarm size on the convergence properties and solution quality of the PSO algorithm. They concluded, in general purpose application, that swarm size had little effect on overall performance as long as the swarm falls within a reasonable range; this range typically being 30 to 40 particles.

Unfortunately, most aerodynamic optimisation studies provide few details on swarm size. Additionally, poor performance may be attributed to the initial population distribution. If the population does not adequately cover a search region efficiently, it may not be possible to locate regions of optimality. Grosan et al.[163] discuss population behaviours and distribution in detail for general-purpose PSO application.

Many aerodynamic optimisation studies directly employ the classical equation (2) with no modifications.[24,61,164–167] Hence, PSO is often considered as a general-purpose optimiser, which is capable of handling different types of variables and functions with no adaptation necessary. Ouissa et al.[167] use the classical PSO algorithm, with 20 particles for 100 iterations, to optimise the energy capture by a wind turbine through real-time control of the blade pitch. They found that the original PSO algorithm was capable of dealing with strong non-linearity occurring from dynamically varying wind-speeds. Therefore, reflecting the capacity for adaptability and fast convergence for single objective, low search space dimensionality optimisation. The inertial weighting factor, w, was set to 0.73 which is a common factor among aerodynamic optimisation studies.[168]

Venter and Sobieszczanski-Sobieski,[164] also employing the classic PSO algorithm, showed that it can simultaneously handle continuous and discrete multi-disciplinary bi-level optimisations coping with severe numerical noise. The algorithm's weighting are modified however ($w = 1.4$, $c_1 = 1.5$, $c_2 = 2.5$) in order to bias convergence towards the globally best solution. No problem specific tuning was performed as emphasis of the work was placed on general application of the PSO algorithm in a multi-disciplinary aerodynamic environment; a swarm size of 100 particles was used. The algorithm was given an unconstrained problem which aimed to maximise the range of the wing utilising 210 geometric and structural design variables. Although the problem presented by the authors was unconstrained the problem still required variable bounds. The approach used by the authors was to restrict the velocity vector of a particle forcing it to move back towards the feasible design space. Dealing with constraints in this way is discussed in depth by Perez and Behdinan.[169] Forcing particles to meet some condition may begin to artificially drive a solution. Alternatively, penalty-based constraints are known to work well with gradient-free optimisers but can have a significant influence on performance and can lead to numerical ill-conditioning.

Performing a statistical analysis of the PSO solutions Venter and Sobieszczanski-Sobieski[164] show that the standard deviation over ten independent optimisations was 0.17% relative to the mean objective function which improved by 6.43%. This robustness did however come at a high computational cost, requiring an average of 9660 function calls per optimisation. It is identified that this high cost is due to the PSO algorithm's un-tuned weightings and the fact that the objective space contains severe numerical noise. Studies focused solely on tuning PSO performance such as that by Pant et al.[170] highlight that poor selection of the weighting parameters can lead to premature, very slow, or no convergence at all; the literature reflects that there is no standard way to balance cognitive or social weightings and problem specific tuning is necessary.

Jansen et al.[24] used a medium-fidelity aerostructural panel code approach and an augmented Lagrange multiplier particle swarm optimisation (ALPSO) algorithm to show that a winglet is optimal when span constraints are present. The algorithm was additionally capable of accurately quantifying optimal configurations and raked wing and box-wing under different conditions. When the algorithm compensated for the effects of viscous drag a C-wing configuration was optimal, however with the inclusion of structural considerations the C-wing and box-wing configurations were found to add more structural weight than their respective drag reduction potential could compensate for. The ALPSO parameters used, defined by Jansen[171], a swarm size of 40 for 12 iterations using: $w = 2$; $c_1 = 2.5$; and, $c_2 = 0.5$. Despite the seemingly good performance, the optimiser was found to converge prematurely forcing several optimisations to be considered for each problem. The ALPSO algorithm was tuned to also reject infeasible solutions which was shown to help move the algorithm towards more feasible designs. The same algorithm was also employed by Haghighat et al.[166] with a similar physics solver for the design optimisation of an active load alleviation control system for flexible wings. The performance was not commented on by the authors. Lyu et al.[71] use the ALPSO algorithm for the lift-constrained drag minimisation of the NASA CRM under RANS equations. Using a swarm size of eight for eight wing twist variables they find that the optimiser is sufficiently capable however ill-suited to the problem and so incurred excessive computational expense.

The social and cognitive weightings can also be defined by constriction models, as implemented by Azab and Ollivier-Gooch.[168] In this constrained aerodynamic optimisation of aerofoil design, it was found that this approach was susceptible premature convergence but is overcome by randomly re-scattering particles if premature convergence was detected. In addition the use of scalar quantities for the acceleration factors in aerodynamic optimisations was expanded upon by introducing non-linear mathematical programming and additional weightings.[172]

Using a fixed inertial weight can have detrimental effects on the convergence behaviour of the algorithm. Praveen and Duvigneau[165] present a dynamically decreasing inertia weighting factor (from 0.9 to 0.4) for the optimisation of supersonic and transonic wing designs using high-fidelity CFD. The dynamic update of the inertial weighting for the betterment of the PSO algorithm for shape optimisation was proposed by Fourie and Groenwold.[172] This ensured a more exploratory search for optimal configurations is followed by the promotion of convergence towards the best known solutions. It is common practice in aerospace focused optimisations that these weightings are not constant scalars, and instead are implemented as some vector of weights. Appropriate inertia weighting balances global and local searching to either increase solution diversity and search the whole design space or alternatively encourage intensified searches in local regions. The idea being to terminate the PSO algorithm with a more local search.

Wang et al.[158] and Nejat et al.[173] each present CFD-based aerofoil optimisations adopting dynamically varying weightings. Both studies argueing that scalar implementation is more susceptible to becoming trapped in a single search direction towards a local optima. The inertial weighting linearly decreases in the same manner described for Praveen and Duvigneau.[165] The cognitive and social weighting, for both studies, vary randomly between 1.5 and 2.5. Wang et al.[158] found this to give the algorithm good global search ability while maintaining good convergence characteristics for 19 design variables achieving an average population drag reduction of 25%. The algorithm also seems to have been able to successfully identify regions within this design space that corresponded to supercritical and high-lift aerofoils with supercritical characteristics.

Nejat et al.[173] used dynamically varying weights to mitigate premature convergence. They modified the classic PSO algorithm to increase stability and address premature clustering in the early stages of the search process which PSO studies have found to be a common issue among the algorithm variants.[174,175] Modifications they employ include: 1) addition of a constriction coefficient to limit the maximum velocity of particles in space (divergence); 2) non-dominated sorting; 3) neighbour density estimation to encourage exploration of less crowded regions; and 4) addition of a collision operator to the velocity update equation (equation (2)). Detection of less-crowded regions of the design space coupled with the collision operation introduces areas of attraction and repulsion to the decentralised search. Nejat et al.[173] found that this gave high diversification in early stages and avoided premature convergence; however, as the tenancy for convergence increased the regions of repulsion had to decrease providing sufficient diversity at different stages of swarm optimality. The algorithm was shown to perform well when optimising an aerofoil stall characteristics based on four conflicting objectives, able to find optimal trade-off characteristics for the aerofoil maintaining suitable performance metrics. Other forms of repulsive PSO algorithms are present in aerodynamic optimisation problems, but are generally criticised for slowing the algorithm performance and poor ability for local search. Guglieri,[126] for example, found that a repulsive PSO algorithm in the optimisation of helicopter rotor blades gave sufficient optimisation tolerances but performance was very slow likely due to solution diversity. Alternative methods to mitigate premature convergence may include craziness operators.[165,176]

In contrast to repulsive PSO algorithms that have been used in aerodynamic design applications in an attempt to prevent swarm clustering, there are PSO algorithms which encourage swarm clustering. Li et al.,[61] among others,[168,177] used clustering/grouping to improve the PSO exploration and exploitation of design variables and/or objective space. Using CFD-based optimisation, Li et al.[61] perform the optimisation of the engine nacelle/pylon position on an aircraft in order to minimise interference drag subject to vertical and horizontal position, shown in Figure 18, and 12 constraints restricting maximum movement due to mesh limitations. The method used here to form sub-groups within a design space uses the classic PSO algorithm and simply develops different groupings with a shared objective due to the arrangement of unequal weightings. Therefore, the swarm develops simultaneously in various ways but maintains swarm social learning. By this means it ensures a global optimisation ability of the entire group while accounting for local search ability. The algorithm was able to communicate design features between the sub-groupings manoeuvring the engine nacelle in such a way to reduce the local velocity over the pylon, thus reducing the strength of the interference drag experienced reducing the aircraft drag coefficient by 1.1%.

A different method of swarm clustering is presented by Hu et al.[177] in optimising the air supply to maintain air quality inside the aircraft cabin. Sub-swarms maintain equal weightings, and alternatively are assigned different objectives. This essentially defines single-objectives sub-swarms with bias to different search regions which, through social interaction, form a multi-objective swarm. This represents a strong design strategy for the optimisation of problems where the relative importance of many conflicting objectives is unclear.[161]

C. Simulated Annealing

Simulated Annealing (SA) is a stochastic point-by-point optimisation algorithm method based on the physical cooling process of molten metal.[178–181] The physical process of annealing provides the inspiration for finding optimal solutions to combinatorial problems based on statistical mechanics. SA is loosely related to the Hill Climbing[178] optimisation algorithm in that both algorithms perform a point-by-point local search around the current candidate solution. The main significant difference is that Hill Climbing will only accept a new solution if it is better than the current one: SA incorporates the possibility of accepting poorer candidate solutions, thus meaning the algorithm can escape local optima. This procedure of accepting poorer solutions is referred to as the Metropolis Procedure/Criterion,[179] and it acts to mimic the probability of atoms jumping between higher and lower energy levels as the material cools.[182]

The main iterative steps which the SA algorithm operates is as follows:[183,184]

1. **Initial selection** of candidate solution and a high initial temperature (material state).
2. **Second selection** of a candidate solution is generated at random in the vicinity of the initial point.

3. **Comparison** is made calculating the difference between the aforementioned two solutions.

4. In the next iterations, another solution is created at random in the local neighbourhood of the current solution, and the **Metropolis Criterion** either accepts or rejects the new solution.

5. **Annealing schedule** is updated.

6. **Iterate** over steps 2 to 5 until convergence criteria is met.

In the literature, there are various suggestions on how to implement SA, however classical SA is expressed as equation (3).[111] Typically, the temperature is decreased from the initial temperature, T_o, to the current temperature, T_n, based on some reduction constant, γ^n, assigned by the annealing schedule at iteration n. Figure 19 illustrates this procedure.

$$T_n = \gamma^n T_o \tag{3}$$

The Metropolis Criterion performs a probabilistic check so that sub-optimal solutions are allowed to replace parameters that have better cost values. By analogy of the process of annealing, the probability (P) of the new candidate solution's cost (C_{new}) relative to the previous solution's cost ($C_{previous}$) is determined using Boltzmann's equation:[179]

$$P = exp(\frac{C_{previous} - C_{new}}{T_n}) \tag{4}$$

P is then compared to a randomly generated number in the range $[0, 1]$ to determine acceptance or rejection of the new solution. A more in-depth discussion is provided by Rafferty.[109] Eglese[184] emphasises that application of the SA algorithm is sensitive to: 1) the initial temperature (T_o); 2) the annealing schedule/temperature function, to determine reduction constant γ at each iteration; 3) number of iterations to be carried out at each temperature; and 4) a stopping criterion for any application. The initial temperature is considered a significant control parameter such that initially, when the system is molten and the temperature is high, search perturbations are large.[109] As more iterations are performed, the temperature is decreased according to the so-called annealing/cooling schedule every n iterations; this is the so-called cooling cycle.

Tiow et al.[185] used the SA algorithm to optimise two separate turbomachinery cascades with the application of inverse design and CFD. The constraints, based on existing knowledge of the design space, were implemented to restrict the search to more optimal regions within the design space to increase the algorithm performance. Additional performance was incurred by coupling the CFD solver with a database of previously generated solutions to accelerate subsequent calculations. The algorithm used a fixed annealing schedule of $\gamma = 0.75$ which operated in conjunction with the Metropolis Criterion every 10 iterations for a total optimisation run of 300 iterations. Iterations used a purely random walk in all search directions and step sizes were taken from uncorrelated uniform distributions.[186] The initial temperature is set low at 0.5^oC; typical initialisation temperatures are much higher as low temperature initialisation can restrict the search of all design variables equally.[187] The low starting temperature suggests that parameters were finely tuned to the problem in order to remove unnecessary calculation.[188] Increasing the initial temperature above the optimal starting temperature would only increase computational expense without betterment of the solution. The objective of the optimisation was to minimise entropy losses through the cascade; this would provide a highly non-linear search space as the cost function, related to aerodynamic losses, is a function of blade shape therefore presenting many local optima. The blade profiles were parametrised with 58 control variables. Loss reductions on the order of 20% were achieved in both test cases reflecting the ability of the algorithm to find superior results.

Leylek et al.[189] use the classical SA algorithm for a bat-inspired wing morphing aerodynamic optimisation in an attempt to expand the wing flight envelope; details of the SA algorithm are presented by Manzo et al.[190] The algorithm is reported to have taken between 500 to 2000 iterations (equating to 3 to 12 hours total run time) terminated by a convergence criterion for which 10 successive function evaluations must deviate less than 1% from one another, thus saving unnecessary computation time. The algorithm determined the starting temperature by taking 30 random samples from each variable in order to assess the average cost increase associated with poorer solutions which is then divided by the initial Metropolis Criterion, 0.95. This ensured that the initialised temperature held an element of randomness and was very high, allowing the algorithm to avoid local optimal in early iterations. The annealing schedule used a reduction constant of 0.85, which implemented a temperature decay every 160 iterations. The authors point out that all 5 design variables describing 3 wing topologies with varying aerofoil section, shown in Figure 20, were presented in a discretised space which was found to decrease the accuracy of the algorithm but also decrease the search space size and the total run time. Figure 20 illustrates the capability of the wing morphology even though a discretised design space is used. The algorithm, using a panel code for aerodynamic analysis, was able to adequately capture the wing active camber balancing 4 performance metrics, each one posing conflicting performance characteristics. Comparing the three different bat-wing topologies the algorithm isolated and exploited the wing shape characteristics such as aspect ratio, camber distribution and wing tip deformation to improve the planform performance dependant on the performance metric in interest. This indicates that the algorithm was able to use bat-like morphology optimisation of a wing to provide significant performance gains and adaptability in the flight envelope.

The classical SA strategy lacks the inherent ability and mechanisms for co-operation between individual points due to its stochastic point-by-point behaviour. Thus, as the search space dimensionality increases, the classical SA algorithm will become less efficient in searching for the optimal solution. Motivated by this, Liu[191] modify the search behaviour of the classical SA algorithm by incorporating fractional factorial analysis based on Taguchi orthogonal arrays. This enhanced numerical convergence and accuracy of the optimiser for higher-order problems while addressing the algorithm's poor performance in highly-dimensional design spaces. Figure 21a illustrates the search behaviour for the classical SA and the Taguchi-SA methods; Figures 21b and 21c show the resulting search paths determined by the different search behaviour for the classical SA and Taguchi-SA for a non-linear multi-modal function respectively. The classical SA algorithm assesses new candidate solutions, x^p, randomly. The Taguchi-SA assesses new candidate solutions according to structured orthogonal tables which determines systematic and uniform sampling from the neighbourhood of the current candidate solution. This methodically improves the combination of design variables to the best candidate solution, x^{best}, within a defined search radius. Comparing Figures 21b and 21c, the increased efficiency of the search behaviour is evident. Liu[191] also show the algorithm capable of handling functions with up to 100 dimensions.

Liu[191] applied the Taguchi-SA to aerodynamic design optimisation of high lift aerofoils and wing planforms, using a finite volume solver, were the enhanced SA algorithm was able to further optimise an already optimised high lift aerofoil. The classical Metropolis Criterion, and a fixed cooling rate of 0.85 was used. The primary objective was reduction of the nose-down pitching moment with secondary objectives of increasing the lift-to-drag ratio. A reduction of 9.11% in the pitching moment was achieved, and the lift-to-drag ratio increased by 32.74%. Furthermore, the Taguchi-SA was used to optimise a supersonic wing planform with variables of sweep, aspect ratio, and taper ratio to minimise the drag rise experiences when transitioning through transonic flows. The algorithm was found to increase the sweep angle and span (increasing the aspect ratio), whereas the taper ratio of the wing was reduced; this reduced the average drag force between Mach 0.2 to 2 by 2.74%.

Mukesh and Lingadurai[183] apply the simplex simulated annealing (SIMPSA), established by Press and Teukolsky,[192] to maximise the aerofoil lift coefficient at a fixed angle of attack, with respect to 12 design variables and subject to 5 constraints. Modifications to the SA algorithm enable it to make more appropriate moves to new candidate solutions by combining the Metropolis Criterion algorithm with the non-linear simplex algorithm. The classical SA and SIMPSA algorithms increased the lift coefficient by 22.8% and 27.4% respectively, showing robustness for global optimisation of non-convex, highly constrained aerodynamic optimisation. Mukesh and Lingadurai[183] provide no detail of the algorithm annealing schedule or other optimiser variables but state that the algorithm was sensitive to the number of iterations for each cooling cycle and initial temperature.

Wang and Damodaran[193] offer a computational brute force methodology to improve the SA algorithm through parallel-processing. They argue that stochastic methods such as SA are very appropriate for multi-disciplinary aerodynamic optimisations however incorporating modifications to the classical SA algorithm can only result in moderate computational savings. Thus, the performance of a serial and parallel SA algorithm are compared in a generic wing optimisation problem adopting the Breguet-Range equation. This combines aerodynamic and structural considerations in the maximisation of range with respect to 4 design variables which are subject to 6 constraints. All SA settings are constant between the serial and parallel optimisations. A key point, which the authors do not address, is that applying parallel computing to the SA algorithm is analogous to a genetic algorithm with zero crossover operation probability. Results showed that the parallel implementation of the SA optimiser over eight processors achieved a 4.4 speed up time and achieved the exact same solution quality as the serial SA. The serial SA algorithm required over 600 objective function evaluations, whereas the parallel SA only required 148 function evaluations. Hence, the number of reduced function calls in this aerodynamic design application is either chance or by design. In parallel, eight random initialisation (compared to one in serial) of candidate solutions in the design space increases the likelihood of one of those point being sufficiently closer to an optimal design. Alternatively, the SA algorithms initialised in parallel may have been modified to use some method for co-operation to navigate the search space.

Subsequently, Wang and Damodaran[194] explore broader application of parallel SA optimisation with CFD for different aerodynamic optimisations, including: 1) diffuser shape design; 2) convergent nozzle; 3) supersonic axi-symmetric nozzle. Based on this work, among others such as Aly et al.,[195] for aerodynamic optimisation, there are loose guidelines for the selection of annealing schemes and selection of termination criteria.

5. APPLICATION OF HYBRIDISED ALGORITHMS TO AERODYNAMIC OPTIMISATIONS

Numerous hybrid algorithms which incorporate elements from different optimisation algorithms exist and have shown successful application. Here we will only consider hybrid algorithm schemes applied to aerodynamic design problems; in an optimisation context, hybridisation can be defined as mixing two or more algorithms, with possibly, further complementary features.

Deterministic approaches take advantage of the analytical properties of the search space to generate a sequence of candidate solutions with systematic improvements, usually resulting in the number of design iterations required to be small; a major short-coming is the dependency of a compatible design space and sufficient baseline geometry. Heuristic approaches do not take into account properties of the search space, but instead seek improvements to candidate solutions on the basis of experience and judgement with a probabilistic approach which can lead to large numbers of design iterations and long computing times. Therefore, hybrid approaches try to combine methods from these approaches in an attempt to mitigate the weaknesses each hold. Hybridisation does not exclusively require deterministic-heuristic combinations as heuristic-heuristic hybrid algorithms also exist.

The various methods of hybridisation between different types of algorithms can be classified into three main groups: 1) pre-hybridisation, where, for example, the population of the GA is pre-optimised using the GBM; 2) organic-hybridisation, in which the GBM is used as an operator within the GAs for improving each population member in each generation; and 3) post-hybridisation, in which the GAs final population is used to provide an initial design for the GBM. It is should be noted that these classifications are not limited to the hybridisation of GAs and GBMs, however, the most frequent hybrid algorithm found in aerospace application have been designed in attempt to combine the best characteristics of GAs and GBM.[23,98,122,196–203]

Gage et al.[196] present the post-hybridisation of a classical GA with sequential quadratic programming for the topological design of wings and trusses. This work is notable for hybridisation as it is one of the first hybrid methods (HM) employed in aerodynamic design optimisation. They demonstrated that post-hybridisation is effective for final refinement of the GA's candidate solutions. By switching to a GBM once the GAs population is sufficiently mature, computational demands can be reduced and superior solutions can be found relative to allowing the GA to continue. In more recent work, Kim et al.[122] also use post-optimisation, to improve the aerodynamic and acoustic performance of a axial-flow fan, by combining the multi-objective real-encoded NSGA-II from which the Pareto-optimal solutions are further optimised using sequential quadratic programming. The specific difficulty in this method of hybridisation is the transition from a multi-objective problem to a single-objective problem. There are typically two ways to transition: 1) combine all of the objectives into a composite objective using a weighted-sum approach for example; and 2) sequential optimisation, optimising one objective at a time while treating all other objectives as equality constraints. Kim et al.[122] adopted the latter method, pointing out that it did not preserve Pareto optimality and created a set of optimal solutions for each objective with many duplications forming.

A pre-optimisation strategy is proposed by Chernukhin and Zingg[23] which takes advantage of the GA's stochastic search capability and the gradient-based optimiser SNOPT's ability to efficiently identify local optima and enforce constraints directly. They implement the developed HM into a series of different aerodynamic optimisation problems including: aerofoil optimisation, transonic wing-section optimisation, subsonic wing-section optimisation, and blended-wing-body optimisation. The HM's initial population is optimised by SNOPT for a limited number of design iterations on each candidate solution in order to balance biased and insufficient solution development. The improved candidate solutions are then stochastically perturbed by the GA's crossover operations to define a new population. Perturbing-mutation operations (perturbs random solution variables) had to be avoided as it was found to conflicted with the refinement capability of the GBM. Pure-mutation operations (random replacement of solution variables) were included to improve the stochastic search. Similar pre-optimisation strategies have been employed by Xing and Damodaran[204,205] which combine GA stochastic searching and GBMs analytical optimisation procedures in the optimisation of nozzle shapes.

Compiled optimisation results from Chernukhin and Zingg[23] for a series of optimisation problems is shown in Figure 22. From Figure 22(a) the HM was found to significantly outperform other algorithms for highly-multi-modal problems; namely the Griewank function which offers hundreds of locally optimal solutions. Both the aerofoil and transonic wing optimisation are expected to be uni-modal base on the author's results. It is seen from Figure 22(b) & (c) that the HM, achieving equivalent solution quality, added unnecessary computational expense relative to the GBMs which were several orders-of-magnitude more efficient; the HM and GA methods are useful to prove that no additional local optima exist. Both the subsonic wing and blended-wing body optimisation present mildly multi-modal problems in which 7 and 8 local optima were found respectively. The 8 local optima for the blended-wing-body optimisation can be seen in Figure 1. For the subsonic wing design, Figure 22(d), the GBM initialising with a single starting position within the design space could not find any globally optimal solution; however employment of the GBM with multi-start initial design points (GB-MS) found optimal designs with fewer function evaluations than the HM. Finally, in the convergence plot for the blended-wing-body, Figure 22(e), the gradient method prove most effective. The GB-MS and the HM are seen to have similar convergence rates and are both expected to be able to find globally optimal solutions but were terminated due to time constraints. These results suggest that, for aerodynamic optimisation problems which are generally mildly multi-modal, the HM implemented here adds unnecessary complication and computational demand.

The classical GA has been extended for the purpose of more efficient aerodynamic shape design optimisation by Catalano et al.[98] to include two new operations based on gradient search in a hybrid organic-optimisation algorithm. Figure 23 shows how the classical genetic algorithm has been modified to include such operators. The activation of each gradient operator is controlled probabilistically, inspired by classical crossover and mutation operation use. The first gradient-based operator, which acts on the whole population, has two further probabilistic controls determining behaviour with each candidate solution. The first determines the maximum number of iterations and the second determines the sensitivity analysis method the gradient optimiser uses. The second gradient operator is relatively simpler, applying only one optimisation iteration to the current best solution according to the steepest descent rule with a random step size. Finally there is an elitist mechanism which preserves the fittest solutions from each generation and reintroduces the best known solution from all generations into the current population.

For the single-objective optimisation of an aerofoil with 24 variables, Catalano et al.[98] find that the complex HM developed is comparable to the standard GBM in terms of the number of function evaluations only achieving a 0.4% better objective. Under different settings (population, number of iterations, sensitivity analysis, etc.), they find that the hybrid algorithm slightly outperforms the gradient method, reducing the objective function further by 2.2%, but requires many more function evaluations. The classical GA is outperformed by all other algorithms. Therefore, the hybridised GA-GBM showed accelerated performance in aerodynamic optimisation relative to a standard GA and capable of matching, and in one case able to outperform, the GBM. The GBM (with a suitable sensitivity analysis) is shown overall more effective, the hybrid algorithm performed well however added unnecessary complexity to the optimisation framework.

PSO and GBM have also been combined for aerodynamic design. Jansen et al.[24] used a PSO algorithm post-optimised with the gradient optimiser, SNOPT, in the aerostructural optimisation on non-planar wings using a panel method and potential flow theory. Little information is given on the hybridisation however the reasoning behind the method employed was to compensate for instances when the PSO converged prematurely. This ensured the PSO solutions were at a local minimum and offered further solution refinement where possible.

Azab and Ollivier-Gooch[168] propose a alternative technique which implements pre-optimisation of a swarm using SQP with adjoint gradients to find an initially high quality global swarm position before using the PSO algorithm. The objective is to define a high quality focal point in the design space that influences the swarm with the aim of reducing the total computational demand to find the global optima. In transonic drag optimisation with thickness and lift constraints, the pre-optimisation using the SQP method was not able to find a shock free optima but did reduce the drag. The SQP adjoint gradients tend to minimise the drag through poor mechanisms, thus becoming stuck in a local optima. The PSO algorithm essentially found the initial point provided by the SQP irrelevant as the SQP performance was limited in the given design space. They identify that the influence of numerical noise (partly due to the shock wave appearing and vanishing with geometric changes) coupled with the limitations of the Euler flow physics model are what prevented the GBM finding the appropriate gradient information. Additionally, when the objective space was very nearly flat, representing incremental improvements, the shallow gradients available in the objective space were overwhelmed by numerical noise meaning that pre-optimisation procedures did nothing. The computational expense of the hybrid scheme is reported at 4 to 9 times more than the SQP algorithm alone, but for all optimisation cases the HM found far superior optimal solutions. The authors did not compare the computational expense of the HM to the un-hybridised PSO algorithm.

Alternative hybrid methods based on hybridising two heuristic approaches also exist in aerodynamic optimisation literature. A modified binary genetic algorithm (GA) and SA have been used together for organic-hybridisation by Herbert-Acero et al.[206] in the optimisation of wind turbine rotors. A schematic diagram of the hybrid algorithm procedure is shown in Figure 24. Figure 24 shows that the hybrid procedure optimised the radial geometric distributions of the blade, using the SA algorithm, and the selection of aerofoil shape along the blade discretised span used the GA genetic operators. The GA did not optimise the combination of sectional aerofoil shape over the span, it acted to perturb the combination of sectional aerofoil (specifically the camber distribution) along the blade for which there were 100^{17} possible combinations. To demonstrate the hybrid framework developed, a SA algorithm was used to optimise radial geometric of different blades with fixed camber distributions from which the results were compared to hybrid procedure. Figure 25 compares the different geometries and aerodynamic performance curves for different optimised NACA 4-digit wind turbine blades. The genetic operations in the hybrid algorithm gave the optimisation process the ability to leverage the sectional aerofoil distribution to allow a nearly constant chord and thickness distribution. The obtained chord length attributed to aerodynamic benefits of operating at higher Reynolds numbers and the low thickness mitigated performance deterioration near stall. Furthermore, the results showed that the hybrid optimiser held distinct improvements in the blade design including: a reduction of the cut-in wind speed, increase aerodynamic efficiency, and an overall reduction of material used to manufacture the blades.

Other heuristic hybrids that exist include hybridisations of operations between GAs and PSO,[207–209] or SA and PSO.[210] These studies typically attempt to handle highly multi-modal design spaces requiring large degrees of modification while addressing strongly stipulated aerodynamic constraints. For example, Khurana et al.[207] show that for aerofoil shape optimisation, the classical PSO algorithm was unable to optimise to a feasible optimal which met all constraints once the dimensionality of the problem exceeded 10 variables. Additionally, excessive computational resource was required for the evaluation of aerodynamically infeasible designs. Part of the problem was associated with the lack of search diversity attributing to sub-optimal solutions. To overcome this they introduced systematic mutation operators from GAs and observed significant convergence improvements in fewer iterations.

6. APPLICATION OF SURROGATE MODELLING TO AERODYNAMIC OPTIMISATION

Surrogate modelling can be viewed as a non-linear inverse problem with the aim of determining a continuous function that relates design variables to output responses from finite data. Surrogate assisted optimisation aims to alleviate the computational burden of the aerodynamic optimisation process by defining a simplified mathematical relationship allowing for fewer numerical simulations to be required. To interrogate the surrogate an optimisation algorithm is needed to perform a global search of the design space relating to the response surface. It is seen from the literature that surrogates are almost exclusively coupled with gradient-free population based optimisation methods, such as genetic or particle swarm algorithms.[20,61,211] With surrogate assisted stochastic optimisation Madsen et al.[212] optimised a diffuser shape, Li et al.[61] optimised the nacelle/pylon position for a wing, and Lundberg et al.[213] optimised ground vehicle aerodynamics.

Key steps are outlined by Hashimoto et al.[28] for a surrogate assisted aerodynamic optimisation which is summarised in Figure 26. The construction of the surrogate generally consists of three steps: 1) design of experiment sample plan to generate initial sample points in the design space - point selection is discussed by Boopathy and Rumpfkeil[214]; 2) numerical simulations are performed to compute the output/performance of each sample point; 3) sample point data (input & output) are used by an approximation model to construct the surrogate. Replacing a particular problem analysis with a surrogate analysis does not affect the problem formulation, but it will strongly influence the solutions identified. Therefore, once the surrogate model is constructed it must be validated (sometimes considered a 4th step). This has the purpose of establishing the predictive capabilities of the surrogate model in design regions away from known sample data. Keane[215] used both empirical and CFD data to create a surrogate for drag variations with gross changes in wing topology. This study suggested updating the initial data set iteratively by adding new points where good designs were found; with this update the initial design from the original response surface was greatly improved.

There are both parametric and non-parametric alternatives in constructing a surrogate model. Parametric approaches (such as kriging or polynomial regression) are model dependant forming a functional relationship between the response variables and the design variable samples that are known. Non-parametric approaches (such as radial-basis functions or neural networks) use local models in different regions of the sample data to build-up an overfall frame work of the model. Furthermore, surrogates can also be classified into regression type (polynomial regression, radial basis functions) which tend to be better suited to noisy functions, and interpolation type (kriging) creating best-fit response models. Usage of any of these models is not straight forward as the quantity and quality of information the user has to provide in the construction of the surrogate is not known a priori. Furthermore, the efficient exploitation of training data can be restricted by inherent problem complexity, constraints, design variable dimensionality, and accuracy Vs. computational cost. Hence selection of the most appropriate surrogate is considered problem dependant, as it will directly influence the optimisation algorithms' decision making capability. There are no general rules leading to the choice of type of surrogate, generation of sample data for training and validation, and indeed the combination of surrogate model and optimisation algorithm. Different surrogate models will be better suited to different data sets and care must be taken to not over-generalise the problem or false optimisation my occur. A review and comprehensive discussion of surrogate modelling is presented by: Queipo et al.,[216] Simpson et al.,[217] Jin et al.[218] and Forrester and Keane.[219]

Parametric surrogates have been widely applied to aerodynamic optimisations due to their flexibility and ease of use. The resulting model will form a response surface that fits exactly to the sample data points. Therefore parametric models are very well suited to conditions where a design space is poorly/sparsely sampled, clustered, or noisy. These approaches do not rely on any specific model structure and are a successful statistical tool for modelling globally dispersed spatial observations. Perhaps the most straight-forward parametric surrogate is one formulated through polynomial regression. Lian and Liou[220] use 2nd-order polynomial response surface equations to construct the functional relationship between design variables and objectives enabling GAs to re-design centrifugal compressors and transonic compressor blades. As the surrogate greatly reduces the computational cost of function evaluations, the GA population and generation sizes were increased due to the freed computational resources. This facilitated increased exploration and identification of superior solutions.

Other studies have used polynomial regression response surfaces for the optimisation of helicopter rotor designs to increase efficiency at a reduced vibration over different operating conditions. Collins et al.[221] report development of these methods and show that 4th-order polynomials, requiring over 300 simulations to construct, are capable of accurately approximating performance metrics achieving a regression coefficient (R^2) of 0.99. This surrogate maintained robustness of the high fidelity numerical simulations at a greatly reduced computational time, however, scaling functions were required to map low-fidelity results to the higher-fidelity domain. Similar approximation models have been used by Leisink et al.[222] for the multi-point aerodynamic design of helicopter blades. The authors consider three design points to be optimised by a GA. In the initial stage of the optimisation, the GA operates at a low-fidelity level in a series of reduced design space. Following this, a high-fidelity surrogate model is created and the optimisation is continued. Their results compare the GA surrogate assisted optimisation with the un-assisted GA, the resulting Pareto optimal fronts are shown in Figure 27. The multi-point optimisation for the surrogate assisted GA required a total of 180 simulations (80 simulation set up the initial surrogate with 5 subsequent updates of additional 20 simulations) while the lone GA required 1600 simulations. Pareto Results in Figure 27 show reasonable agreement, however the surrogate assisted optimisation does not capture the entire extent of favourable solutions across the front. Hence better trade-off solutions and other favourable configurations that are found by the GA are completely missed by the surrogate optimisation. A common flaw with scaling low-fidelity results to a higher-fidelity domain is that improvement of a design predicted by a low-fidelity model does not guarantee an improvement in the high-fidelity problem.[223] Zadeh et al.[224] present methods for tuning design variables to cope with discrepancies between high and low-fidelity models.

Kriging surrogates (discussed in detail for aerodynamic surrogate modelling by Rosenbaum and Schulz[225]), are perhaps one of the most effective meta-models due to their ability to model complicated responses through either interpolation or regression. Kriging additionally provides an indication of prediction uncertainty, can be used to overcome some of the contrasting performance of low and high fidelity samples. The specific approach for this is referred to as co-kriging, in which a low fidelity kriging surrogate is used to tune the hyperparameters of a higher fidelity surrogate model. Additionally, for kriging based algorithms, the quantity of initial sample information is independent of the number of design variables typically resulting in efficient algorithms for highly dimensional domains. Toal et al.[211] discuss effective hyper parameter tuning for co-kriging, and consider several methodologies. The co-kriging surrogate developed is later used by Toal et al.[226] for aerodynamic optimisations with a PSO algorithm, with which they also consider the difference in performance of kriging and co-kriging. The co-kriging demonstrates the capability of reducing the number of high-fidelity CFD analysis required by 40% for a multi-point optimisation in a highly dimensional multi-modal design space. This computational saving enabled the consideration of additional design conditions or fewer overall analyses to be carried out. The authors also highlight the necessity for care if the design space contains high non-linearity or discontinuity as more complex response curves will need to be developed. Elsewhere, co-kriging regression surrogates have been shown to utilise wind tunnel to calibrate the meta-model constructed from computational data.[227,228]

Laurenceau et al.[229] present a comprehensive study comparing kriging-based and co-kriging-based optimisations for increasingly dimensional and multi-modal aerodynamic problems. They highlight that with increasing dimensionality, the number of computational analyses used to construct the surrogate is likely to become restricted, dependant upon the computational resources available. In limiting the sample size to 200 CFD analyses in a wing-drag optimisation it was found that the kriging based optimisation lacked sufficient accuracy leading to the optimiser wasting computational cost from excessive exploration of the design space. In comparison, the co-kriging-based optimisation performed $\approx 70\%$ less expiration of the design space while enabling a more optimal solution to be identified. Chung and Alonso[230], in the optimisation of a supersonic aircraft, also find that co-kriging-based approximation models are of higher accuracy than kriging-based ones. This again greatly improves the efficiency of the optimisation by reducing computational cost in a large multi-modal design space. Furthermore, Koziel er al.[26] adopt co-kriging for multi-objective aerofoil optimisations to improve the surrogate's flexibility resulting in stronger convergence properties of the optimisation procedure.

There are several examples within the literature which do not adopt multi-fidelity co-kriging models. Takenaka et al.[231] use a kriging assisted genetic algorithm for the design exploration of winglet designs. The selection of the initial 32 sample points for the surrogate model were made by the standard Latin Hypercube Sampling (LHS)[232] model; the initial samples are shown in Figure 28. Based on this study the kriging surrogate model is very efficient for design problems with multiple conflicting objectives and a small number of design variables; six variables are used here. Their optimised winglet resulted in a total drag reduction of approximately 22 counts for an increase in root bending moment of 5.3%. The total drag reduction provided by the winglet was validated using wind tunnel data.

Hashimoto et al.[28] demonstrated that kriging is capable of fitting complex design spaces and reducing computational demands in the optimisation of high-wing aircraft. A particular trait which makes kriging attractive in complex design optimisations is that statistical improvement criteria, and related uncertainties, are readily available at any point in the design space without additional expense. This enables a more robust exploration of the surrogate by accounting for both the predicted value and its uncertainty simultaneously, transforming the objective function into the corresponding Expected Improvement (EI) function.[233] The EI function indicates the probability of a point being optimum in the design space and also gives an indication to regions with high uncertainty. Design points with high EIs represent a balance between finding promising regions in the design space based on the surrogate prediction (local search) and finding regions of high uncertainty in the surrogate prediction (global search). Through selecting the best EI points to perform additional numerical simulations, the kriging model can be iteratively re-calibrated, as in Figure 26, improving the model and searching for superior solution simultaneously. For the optimisation of helicopter blade design to reduce vibrations due to both blade-vortex interaction and dynamic stall, Glaz et al.[234] also use an iteratively updated kriging assisted GA to identify Pareto-optimal designs. The kriging model was capable of capturing fundamental trade-offs among different flight conditions and associated vibrational loading. In this work it is suggested that the application of kriging to aerodynamic optimisations is only appropriate when the time needed to generate the interpolation points is much greater than the time needed to interpolate the data.[217]

Non-parametric surrogates are becoming more popular in aerodynamic optimisations, but can be more complex to implement. The increasing popularity is based on their capability to approximate any continuous behaviour with arbitrary precision of the host computing environment. For instance radial basis functions (RBF), although inferior to kriging models regarding interpolating accuracy[235], are easier to characterise and modify, and its superior smoothness can make it more suitable for many design spaces. Morris et al.[56] is seen to use RBF for CFD-based optimisations to provide both a framework to deform the mesh discretising the computational domain and also a way to interpolate forces and moments between them, as the two meshes are likely not coincident. Additionally, they can easily extend from two- to three-dimensional problems. Fincham and Friswell[33] use the RBF in a similar manner optimise morphing aerofoils decomposing the RBF into three coupled data clusters representing the aerodynamic surface, changes to the aerodynamic surface, and the morphing actuation system. One of the advantages the authors found was that once the initial computation cost of evaluating the design points was established, the computational cost through-out the optimisation was constant (therefore predictable) and significantly reduced.

Bevan et al.[236,237] construct a radial basis function based surrogate model to optimise vortex-generators to alleviate separation and buffet onset for thick and highly loaded aerofoils. The authors state that the advantage of radial basis functions is their ability to fit scattered multivariate data exactly, thus preserving the CFD data at sample points. Lower-fidelity approximations of the higher-fidelity design space were necessary to preserve important flow physics required for the design to increase the accuracy of interpolating between known design points. Ong et al.[238] hybridised a GA and a sequential quadratic programming optimiser. Coupling this method with radial basis functions allowed them to decompose the full-scale problem into a sequence of sub-problems confined to regions defined by the surrogate. The series of local surrogate feed into the global system and is shown capable of coping with fundamental difficulties associated with dimensionality for a single-objective optimisation. The main purpose of constructing the surrogate in this way was to increase the predictability of new design points. In this work it is seen that neighbouring solutions had more influence on the design than remote ones.

Neural networks (multi-layered radial basis functions) form a more sophisticated behavioural modelling technique which may be unnecessary for most aerodynamic optimisations. The persistent difficulties associated with these methods is the need to choose the structure of the surrogate network and a suitable training sequence. Khurana et al.[207] suggest that neural networks work well when the relationship between the design space and the objective function is complex, less intuitive, and highly-dimensional. They develop a neural network, on a trial-and-error basis, to duplicate the performance of RANS simulations which a PSO algorithm searches for optimal aerofoil shapes. Massaro et al.[239] coupled a GA with a Neural Network for the multi-point optimisation of helicopter rotor performance. The Neural Network modelled the hover condition for the rotor in addition to the two different flight conditions enabling a crossover of active constraints.

In a situation where one may not know which surrogate model would perform best, the use of multiple or hybrid surrogates may be suitable and can offer several advantages to the optimisation decision making capability.[216] Viana et al.[240] study the application of several open source surrogate modelling tool boxes demonstrating that hybridising up to ten surrogates (one of which kriging) out performed the stand alone kriging surrogate by substantially reducing the number cycles required for convergence. In certain instances however while the number of cycles for the optimisation was reduced the number of functions calls required increased, i.e. computational demand increased as wall-clock time decreased, indicating parallel computation could be beneficial. Additionally results show that the rate of convergence did not scale with the number of surrogates in the hybrid model - 5 and 10 surrogate hybrid models achieved comparable results. The particular aspect of this study attractive to aerodynamic optimisation is that the use of multiple surrogates are shown to have a distinct advantage in highly dimensional design spaces with multiple design targets. Hybrid surrogate modelling under different techniques aid in design exploration and generate diversity as well as improve the surrogates prediction thus reducing the model uncertainty.

Tianyuan and Xiongqing[241] recognised in the aerostructural optimisation of an unmanned combat air vehicle that different output responses did not all coincide with the same point in the search domain. Thus the application of different surrogate models were used to map out different aspects of the response output. This can help minimise each surrogates generalisation error which is used to assess the quality of the approximation model for prediction and establish its capability (validation) for use in analysis and optimisation. A kriging model was found most suitable for predicting the drag coefficient, low radar cross section constraints and the structural weight while second-order polynomial regression models the internal volume of the structure. This implies that the internal volume varied in a much simpler manner with possibly fewer optima and complexity - this makes sense as the internal volume is likely not to vary too much. The surrogates were constructed from the same 200 samples and are not iteratively updated as the relative error of the surrogates and experiments is less than 5% for the considered design space.

Namura et al.[14] follow similar reasoning and combine the multi-objective GA with two surrogate models in the optimisation of vortex generators for super-critical aerofoils. The first surrogate used a kriging model which was identified as more accurate for modelling the lift coefficient and the chord-wise separation location; the second was a hybrid radial basis function/kriging surrogate showed higher accuracy in estimating the lift-to-drag-ratio. The lift-to-drag ratio is a result of many factors and so can be expected to be complex with regions of discontinuity due to presence of separation. Both kriging and radial basis functions can adapt well to complex spaces, but by hybridising them the resulting approximation surface will be better for capturing both macro and micro design trends.[242,243]

7. CONCLUSION

A Survey of several important aerodynamic optimisation strategies are reported here. There is however, significant scope for further work to be carried out on the optimisation algorithms discussed and the strategies used to integrate them with aerodynamic based problems.

It is common knowledge that appropriateness of geometric parametrisation, problem definition and optimisation algorithm definitively depends on the nature of the problem at hand; i.e. number of variables, their scope, required fidelity, and ultimately how the designer embeds and tunes the algorithm. Although relative cost is only one of several important considerations in choosing an optimisation algorithm, for an effective aerodynamic optimisation process deep understanding and consideration must be given to: 1) the level of information fidelity required from the flow solver, dependant on the type of problem; 2) scope of parametrised design space; 3) types of design variables, e.g. discrete and/or continuous; 4) single or multi-objective optimisation; 5) constraints handling; 6) properties of the design space, e.g. number of local optima, discontinuities.

Gradient-based approaches using the discrete adjoint sensitivity analysis scales approximately linearly with the number of design variables and is very capable of handling thousands of design variables and constraints. They inherently require a geometrically compact set of design variables and an initial user defined baseline geometry and enable systematic design capability in which high-fidelity simulations may be more useful if the analytical properties of the design space are continuous and exhibit low-modality. The gradient-based approach may be more appropriate for detailed aerodynamic design as they are only capable of offering a narrow range of solutions. The cost of gradient-free relative to gradient-based methods increases dramatically and they often require tighter convergence tolerances to be prescribed as they can take advantage of loose constraint definitions. In aerodynamic optimisations, overcoming the computational cost related to these algorithms often use methods that facilitate the progressive containment of shape parametrisation to refine the search space; this tries to reduce the number of wasteful numerical simulations. The benefit of these algorithms is in their simplicity and their parallel nature which is capable of trade-off multi-disciplinary performance metrics over a range of solutions.

Hybrid methods have become more popular in aerodynamic optimisation. Such hybridisations can be used to take advantage of the explicit strengths of certain algorithms in order to account for fundamental difficulties of other algorithm; this improving the overall algorithm performance for more effective and efficient problem solving. For aerodynamic optimisation hybrid algorithms in some instances have been found capable, but for most applications are found to over complicate the optimisation process. From a practical perspective in aerodynamic optimisation, much work remains to be done in benchmarking existing stand-alone architectures to conclude decisively if hybrid methods enhance optimisation capability or just create more complicated optimisation infrastructures.

In many circumstances the relationship between the design parameters and the objective function is highly non-linear, however the objective function relationship with the design variables if often found to follow predictable trends. Consequently, surrogates can help in understanding this relationship which moves the optimisation towards understanding the behaviour of different search components. This ultimately can lead to a design structure that incorporates more problem-specific knowledge and helps to overcome dependency on the user.

For all aerodynamic optimisations regardless of the overall architecture there are several general issues which must be addressed in-order to exploit the optimisation algorithm to its full potential. Despite the widespread success of many algorithms in different contexts, there will always be the persistent question related to the usefulness of a particular algorithm solving for a wide range of problems. The *No Free Lunch Theorem* for optimisation by Wolpert and Macready[244] states that the ultimate optimisation algorithm does not exist, and that all optimisation algorithms have the same average performance over a set of optimisation problems; this basically suggesting that different algorithms are better than others for particular classes of problems.

Finally, it is important to highlight that other optimisation schemes are available than those presented in this paper: they may not yet be popularised, to the authors knowledge, into geometry dependant aerodynamic optimisation design problems (e.g. Mixed Integer Optimisation or gravity flow optimisation). The architectures in this literature survey represent significant progress in aerodynamic optimisation, which has been an ongoing endeavour for over 60 years. Major advances have been found from developments in computational resources and high-fidelity modelling. Newest algorithms and implementation strategies are often decades old, where new algorithms are simply permutations of older concepts.

References

[1] European Commission. Flightpath 2050 Europe's Vision for Aviation.

[2] J.E. Green. Civil Aviation and the Environment - The Next Frontier for the Aerodynamicist. *Aeronautical Journal*, 110(1110):469–486, 2006.

[3] J. Nocedal, S.J. Wright, and S.M. Robinson. *Numerical Optimization*. Springer, New York., 1999.

[4] G. Anderson and M. Aftosmis. Adaptive Shape Parameterization for Aerodynamic Design. *NASA Technical Report: NAS-2015-02*, (May), 2015.

[5] S. Wakayama and I. Kroo. Subsonic Wing Planform Design Using Multidisciplinary Optimization. *Journal of Aircraft*, 32:746–753, 1995.

[6] N.P.M. Werter and R.D. Breuker. Aeroelastic Tailoring and Structural Optimisation Using An Advanced Dynamic Aeroelastic Framework. In *International Forum on Aeroelasticity and Structural Dynamics*, pages 1–20, Saint Petersburg, Russia, 2015.

[7] J. Sobieszczanski-Sobieski and R.T. Haftka. Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments. In *34th Aerospace Sciences Meeting and Exhibit, AIAA 96-0711*, volume 70, pages 1–33, Reno, Navada, 1996.

[8] T.F. Wunderlich. Multidisciplinary Wing Optimization of Commercial Aircraft With Consideration of Static Aeroelasticity. *CEAS Aeronautical Journal*, 6(3):407–427, 2015.

[9] D.W. Levy, K.R. Laflin, E.N. Tinoco, J.C. Vassberg, M. Mani, B. Rider, C. Rumsey, R.A. Wahls, J.H. Morrison, O.P. Broderson, S. Crippa, D. J. Mavriplis, and M. Murayama. Summary of data from the first AIAA CFD Drag Prediction Workshop. *51th AIAA Aerospace Sciences Meeting*, pages 1–31, 2013.

[10] C.J. Roy and E.N. Tinoco. Summary of Data from the Sixth AIAA CFD Drag Prediction Workshop: Case 1 Code Verification. *55th AIAA Aerospace Sciences Meeting*, (January):1–19, 2017.

[11] W. Yamazaki, K. Matsushima, and K. Nakahashi. Aerodynamic Design Optimization Using the Drag-Decomposition Method. *AIAA Journal*, 46(5):1096–1106, 2008.

[12] Z. Lyu, G. Kenway, and J.R.R.A. Martins. Aerodynamic Shape Optimization Investigations of the Common Research Model Wing Benchmark. *AIAA Journal*, 53(4):968–985, 2014.

[13] G. Kenway and J.R.R.A. Martins. Multipoint High-Fidelity Aerostructural Optimization of a Transport Aircraft Configuration. *Journal of Aircraft*, 51(1):144–160, 2014.

[14] N. Namura, S. Obayashi, and S. Jeong. Efficient Global Optimization of Vortex Generators

on a Super Critical Infinite-Wing Using Kriging-Based Surrogate Models. In *52nd Aerospace Sciences Meeting, AIAA 2014-0904*, pages 1–12, 2014.

[15] G.R. Anderson, M. Nemec, and MJ. Aftosmis. Aerodynamic Shape Optimization Benchmarks with Error Control and Automatic Parameterization. *53rd AIAA Aerospace Sciences Meeting, AIAA 2015-1719*, pages 1–18, 2015.

[16] L.M. Rios and N.V. Sahinidis. Derivative-Free Optimization: A Review of Algorithms and Comparison of Software Implementations. *Journal of Global Optimization*, 56(3):1247–1293, 2013.

[17] S. Amaran, N.V. Sahinidis, B. Sharda, and S.J. Bury. Simulation Optimization: A review of Algorithms and Applications. *Annals of Operations Research*, 240(1):351–380, 2016.

[18] D.W. Zingg, M. Nemec, and T.H. Pulliam. A Comparative Evaluation of Genetic and Gradient-Based Algorithms Applied to Aerodynamic Optimization. *European Journal of Computational Mechanics*, 17(1-2):103–126, 2008.

[19] G.N. Vanderplaats. *Numerical Optimization Techniques for Engineering Design: With Applications.* McGraw-Hill Book Company, 1999.

[20] Y. Zhang, Z.H. Han, L. Shi, and W.P. Song. Multi-Round Surrogate-based Optimization for Benchmark Aerodynamic Design Problems. In *54th AIAA Aerospace Sciences Meeting, AIAA 2016-1545*, number 127, pages 1–17, San Diego, California, 2016.

[21] J. Sobieszczanski-Sobieski. A Linear Decomposition Method For Large Optimisation Problems - Blueprint For Development. *NASA TM-83248*, 1982.

[22] J. Sobieszczanski-Sobieski. Optimisation by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems. *NASA Technical Report: N89-25149*, 1988.

[23] O. Chernukhin and D.W. Zingg. Multimodality and Global Optimization in Aerodynamic Design. *AIAA Journal*, 51(6):1342–1354, 2013.

[24] P.W. Jansen, R.E. Perez, and J.R.R.A. Martins. Aerostructural Optimization of Nonplanar Lifting Surfaces. *Journal of Aircraft*, 47(5):1490–1503, sep 2010.

[25] S. Peigin and B. Epstein. Multiconstrained Aerodynamic Design of Business Jet By CFD Driven Optimization Tool. *Aerospace Science and Technology*, 12(2):125–134, mar 2008.

[26] S. Koziel, Y. Tesfahunegn, A. Amrit, and L.T. Leifsson. Rapid Multi-Objective Aerodynamic Design Using Co-Kriging and Space Mapping. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2016-0418*, number January, pages 1–10, San Antonio, TX, USA, 2016.

[27] P.D. Vecchia and F. Nicolosi. Aerodynamic Guidelines in The Design and Optimization of New Regional Turboprop Aircraft. *Aerospace Science and Technology*, 38:88–104, oct 2014.

[28] A.H. Hashimoto, S.J. Jeong, and S.O. Obayashi. Aerodynamic Optimization of Near-Future

High-Wing Aircraft. *Transactions of The Japan Society for Aeronautical and Space Sciences*, 58(2):73–82, 2015.

[29] M. Athanasopoulos, H. Ugail, and G.G. Castro. Parametric Design of Aircraft Geometry Using Partial Differential Equations. *Advances in Engineering Software*, 40(7):479–486, 2009.

[30] N.E. Sevant, M.I. Bloor, and M.J. Wilson. Aerodynamic Design of a Flying Wing Using Response Surface Methodology. *Journal of Aircraft*, 37(4):562–569, 2000.

[31] A. Jameson, L. Martinelli, and N.A. Pierce. Optimum Aerodynamic Design Using the Navier-Stokes Equations. *Theoretical and Computational Fluid Dynamics*, 10(1-4):213–237, 1998.

[32] G. Kenway, G. Kennedy, and J.R.R.A. Martins. A CAD-Free Approach to High-Fidelity Aerostructural Optimization. In *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, AIAA 2010-9231*, pages 1–18, Fort Worth, TX, 2010.

[33] J.H.S. Fincham and M.I. Friswell. Aerodynamic Optimisation of a Camber Morphing Aerofoil. *Aerospace Science and Technology*, 43:245–255, 2015.

[34] J. Hall, D.J. Poole, T.C.S. Rendall, and C.B. Allen. Volumetric Shape Parameterisation for Combined Aerodynamic Geometry and Topology Optimisation. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA 2015-3354*, number June, pages 1–29, Dallas, TX, 2015.

[35] S. Koziel and Z. Michalewicz. Evolutionary Algorithms, Homomorphous Mappings, and Constrained Parameter Optimization. *Evolutionary Computation*, 7(1):19–44, 1999.

[36] J. Siens and M.S. Innocente. Particle Swarm Optimisation: Fundamental Study and its Application to Optimisation and to Jetty Scheduling Problems. *Trends in Engineering Computational Technology*, pages 103–126, 2008.

[37] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Multi-Objective Optimization Test Problems. *CEC-2002*, 2:825–830, 2002.

[38] T. Kato, K. Shimoyama, and S. Obayashi. Evolutionary Algorithm with Parallel Evaluation Strategy of Feasible and Infeasible Solutions Considering Total Constraint Violation. *IEEE*, 1 (978):986–993, 2015.

[39] T.D. Robinson, K.E. Willcox, M.S. Eldred, and R. Haimes. Multifidelity Optimization for Variable-Complexity Design. In *Proceedings of the 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 1–18, Portsmouth, VA, 2006. AIAA 2006-7114.

[40] S. Gemma and F. Mastroddi. Multi-Disciplinary and Multi-Objective Optimization of an Unconventional Aircraft Concept. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA 2015-2327*, pages 1–20, Dallas, TX, 2015.

[41] M.J.D. Powell. A Direct Search Optimization Method that Models the Objective and Constraint Functions by Linear Interpolation. *Advances in Optimization and Numerical Analysis*,

7:51–67, 1994.

42 M.D.J. Powell. Direct Search Algorithms for Optimization. *Acta Numerica*, 7:287–336, 1998.

43 A. March and K. Willcox. Constrained Multifidelity Optimization Using Model Calibration. *Structural and Multidisciplinary Optimization*, 46:93–109, 2012.

44 D.R. Jones, C.D. Perttunen, and B.E. Stuckman. Lipschitzian Optimization Without the Lipschitz Constant. *Journal of Optimization Theory and Application*, 79(1):157–181, 1993.

45 D.R. Jones. Encyclopedia of Optimization, 2001.

46 K.W. Lee, W.H. Moase, and S.Z. Khong. Aerodynamic Shape Optimization via Global Extremum Seeking. *IEEE Transactions on Control System Technology*, pages 1–8, 2015.

47 F. Deng, Z. Jiao, and J. Chen. Overall Performance Analysis Oriented Aerodynamic Configuration Optimization Design for Hypersonic Vehicles. *Journal of Spacecraft and Rockets*, pages 1–12, 2017.

48 S. Ebastien and L.E. Digabel. Algorithm 909 : NOMAD : Nonlinear Optimization with the. *ACM Transactions on Mathematical Software*, 37(4):1–15, 2011.

49 M.D. Ardema, J.V. Bowles, and T. Whittaker. Near-Optimal Propulsion-System Operation for an Air-Breathing Launch Vehicle. *Journal of Spacecraft and Rockets*, 32(6):951–956, 1995.

50 Y. Liu, L. Zhou, and Y. Lu. Research on Multidisciplinary Analysis Tool based on Parametric Model of Hypersonic Morphing Waverider. *Applied Mathematics and Information Sciences*, 9 (4):2047–2056, 2015.

51 R. Hassan, B. Cohanim, O. Weck, and G. Venter. A Comparison of Particle Swarm Optimisation and The Genetic Algorithm. In *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, pages 1–13. AIAA 2005-1897, 2005.

52 P. Dellla Vecchia. *Development of Methodologies for the Aerodynamic Design and Optimization of New Regional Turboprop Aircraft.* PhD thesis, University of Naples, 2013.

53 J. Ren, A.S. Thelen, A. Amrit, X. Du, L.T. Leifsson, Y. Tesfahunegn, and S. Koziel. Application of Multifidelity Optimization Techniques to Benchmark Aerodynamic Design Problems. *54th AIAA Aerospace Sciences Meeting*, (January), 2016.

54 Z. Zhang, P.C. Chen, Z. Zhou, S. Yang, Z. Wang, and Q. Wang. Adjoint Based Structure and Shape Optimization with Flutter Constraints. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2016-1176*, pages 1–23, San Diego, California, 2016.

55 M. Nemec and M.J. Aftosmis. Output Error Estimates and Mesh Refinement in Aerodynamic Shape Optimization. In *51st AIAA Serospace Sciences Meeting*, pages 1–18, Grapevine, TX, 2013.

56 A.M. Morris, C.B. Allen, and T.C.S. Rendall. CFD-Based Optimization of Aerofoils Using

Radial Basis Functions For Domain Element Parameterization and Mesh Deformation. *International Journal for Numerical Methods in Fluids*, (October 2007):601–629, 2008.

[57] J.E. Hicken and D.W. Zingg. Aerodynamic Optimization Algorithm with Integrated Geometry Parameterization and Mesh Movement. *AIAA Journal*, 48(2):400–413, 2010.

[58] D. Li and R. Hartmann. Adjoint-Based Airfoil Optimisation With Discretization Error Control. *International Journal for Numerical Methods in Fluids*, 77:1–17, 2015.

[59] K. Chiba, A. Oyama, S. Obayashi, K. Nakahashi, and H. Morino. Multidisciplinary Design Optimization and Data Mining for Transonic Regional-Jet Wing. *Journal of Aircraft*, 44(4): 1100–1112, 2007.

[60] M. Murayama, K. Nakahashi, and K. Matsushima. Unstructured Dynamic Mesh for Large Movement and Deformation. *AIAA Paper 2002-0122*, 2002.

[61] J. Li, Z. Gao, J. Huang, and K. Zhao. Aerodynamic Design Optimization of Nacelle/Pylon Position On An Aircraft. *Chinese Journal of Aeronautics*, 26(4):850–857, 2013.

[62] Z. Lyu and J.R.R.A. Martins. Aerodynamic Design Optimization Studies of a Blended-Wing-Body Aircraft. *Journal of Aircraft*, 51(5):1604–1617, 2014.

[63] T.R. Brooks, G. Kennedy, and J.R.R.A. Martins. High-fidelity Aerostructural Optimization of a High Aspect Ratio Tow-steered Wing. *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2016-1179*, (January):1–16, 2016.

[64] D.A. Burdette, G. Kenway, and J.R.R.A. Martins. Performance Evaluation of a Morphing Trailing Edge Using Multipoint Aerostructural Design Optimization. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2016-0159*, pages 1–14, San Diego, California, 2016.

[65] M. Biava, M. Carrion, G.N. Barakos, and D. Stewart. Adjoint-Based Optimisation of Ducted Propellers for Hybrid Air Vehicles. In *54th AIAA Aerospace Sciences Meeting, AIAA 2016-1784*, pages 1–13, San Diego, California, 2016.

[66] S.A. Ning and I. Kroo. Multidisciplinary Considerations in the Design of Wings and Wing Tip Devices. *Journal of Aircraft*, 47(2):534–543, 2010.

[67] S.N. Skinner and H. Zare-Behtash. Aerodynamic Optimisation of Non-planar Lifting Surfaces. In *57th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2016-0164*, pages 1–21, San Diego, California, 2016.

[68] D. Goldberg. Genetic Algorithms in Searching Optimisation and Machine Learning. *Addison-Wesley, Reading, MA*, 1989.

[69] N.F. Foster and G.S. Dulikravich. Three-Dimensional Aerodynamic Shape Optimisation Using Genetic and Gradient Search Algorithms. *Journal of Spacecraft and Rockets*, 34(1), 1997.

[70] M. Secanell and A. Suleman. Numerical Evaluation of Optimization Algorithms for Low-

Reynolds-Number Aerodynamic Shape Optimization. *AIAA Journal*, 43(10):2262–2267, 2005.

[71] Z. Lyu, Z. Xu, and J.R.R.A. Martins. Benchmarking Optimization Algorithms for Wing Aerodynamic Design Optimization. In *The Eighth International Conference on Computational Fluid Dynamics, ICCFD8-2014-0203*, pages 1–18, Chengdu, Sichuan, China, 2014.

[72] J.R.R.A. Martins, J.J. Alonso, and J.J. Reuther. Complete Configuration Aero-Structural Optimization Using a Coupled Sensitivity Analysis Method Complete Configuration Aero-Structural Optimization Using a Coupled Sensitivity Analysis Method. In *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimisation Conference, AIAA Paper 2002-5402*, pages 1–14, Stanford, CA, 2002.

[73] J.E.V. Peter and R.P. Dwight. Numerical Sensitivity Analysis for Aerodynamic Optimization: A Survey of Approaches. *Computers and Fluids*, 39(3):373–391, 2010.

[74] J.R.R.A. Martins and J.T. Hwang. Review and Unification of Discrete Methods for Computing Derivatives of Single- and Multi-disciplinary Computational Models. *AIAA Journal*, 51(11): 2582–2599, 2013.

[75] J.R.R.A. Martins, P. Sturdza, and J.J. Alonso. The complex-step derivative approximation. *ACM Transactions on Mathematical Software*, 29(3):245–262, 2003.

[76] G. Kenway and J.R.R.A. Martins. Aerostructural Shape Optimization of Wind Turbine Blades Considering Site-Specific Winds. In *12th AIAA\ISSMO Multidisciplinary Analysis and Optimisation Conference, AIAA 2008-6025*, pages 1–12, Victoria, British Columbia, Canada, 2008.

[77] P.E. Gill, W. Murray, and M.A. Saunders. SNOPT: An SQP Algorithm for Large-Scale Constrained Optimization. *SIAM Journal on Optimization*, 47(1):99–131, 2005.

[78] O. Pironneau. On Optimum Design In Fluid Mechanics. *Journal of Fluid Mechanics*, 64(1): 97–110, 1974.

[79] D. Kraft. A Software Package for Sequential Quadratic Programming. *Technical report, Tech. Rep. DFVLR-FB 88-28, German Aerospace Center*, 1988.

[80] D. Kraft. Algorithm 733: TOMPFortran Modules for Optimal Control Calculations. *ACM Trans- actions on Mathematical Software (TOMS)*, 20(3):262–281, 1994.

[81] P.D. Frank and G.R. Shubin. A Comparison of Optimization-Based Approaches For a Model Computational Aerodynamics Design Problem. *Journal of Computational Physics*, 98(1):74–89, 1992.

[82] A. Jameson, N.A. Pierce, and L. Martinelli. Optimum Aerodynamic Design Using The Navier-Stokes Equations. In *35th Aerospace Sciences Meeting and Exhibit, AIAA Paper 97-0101*, number January, 1997.

[83] C.A. Mader, J.R.R.A. Martins, J.J. Alonso, and E.V. Der Weide. ADjoint: An Approach for the Rapid Development of Discrete Adjoint Solvers. *AIAA Journal*, 46(4):863–873, 2008.

[84] Z. Lyu, G. Kenway, C. Paige, and J.R.R.A. Martins. Automatic Differentiation Adjoint of the Reynolds-Averaged Navier-Stokes Equations with a Turbulence Model. In *21st AIAA Computational Fluid Dynamics Conference, AIAA 2013-2581*, pages 1–24, San Diego, California, 2013.

[85] J.E. Hicken and D.W. Zingg. Induced-Drag Minimization of Nonplanar Geometries Based on the Euler Equations. *AIAA Journal*, 48(11):2564–2575, 2010.

[86] H. Gagnon and D.W. Zingg. High-Fidelity Aerodynamic Shape Optimization of Unconventional Aircraft through Axial Deformation. In *52nd Aerospace Sciences Meeting, AIAA 2014-0908*, pages 1–18, National Harbor, Maryland, 2014.

[87] J.R.R.A. Martins, J.J. Alonso, and J.J. Reuther. A Coupled-Adjoint Sensitivity Analysis Method for High-fidelity aero-structural design. *Optimization and Engineering*, 6(1):33–62, 2005.

[88] G. Kenway, G.J. Kennedy, and J.R.R.A. Martins. Scalable Parallel Approach for High-Fidelity Steady-State Aeroelastic Analysis and Adjoint Derivative Computations. *AIAA Journal*, 52 (5):935–951, 2014.

[89] G. Kennedy and J.R.R.A. Martins. High Aspect Ratio Wing Design: Optimal Aerostructural Tradeoffs for the Next Generation of Materials. In *52nd Aerospace Sciences Meeting*, pages 1–24, 2014.

[90] D.A. Burdette, G. Kenway, Z. Lyu, and J.R.R.A. Martins. Aerostructural Design Optimization of an Adaptive Morphing Trailing Edge Wing. In *15th AIAA/ISSMO Multidosciplinary Analysis and optimisation Conference, AIAA paper 2014-3275*, pages 1–13, 2014.

[91] A. Elham and M.J.L. van Tooren. Coupled Adjoint Aerostructural Wing Optimization Using Quasi-Three-Dimensional Aerodynamic Analysis. In *16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA 2015-2487*, pages 1–20, Dallas, TX, 2015.

[92] C.A. Mader and J.R.R.A. Martins. Stability-Constrained Aerodynamic Shape Optimization of Flying Wings. *Journal of Aircraft*, 50(5):1431–1449, 2013.

[93] B. Grossman, Z. Gurdal, G.J. Strauch, W.M. Eppard, and R.T. Haftka. Integrated Aerodynamic/Structural Design of a Sailplane Wing. *Journal of Aircraft*, 25(9):855–860, 1988.

[94] B. Grossman, R.T. Haftka, P.J. Kao, D.M. Polen, and M. Rais-Rohani. Integrated Aerodynamic-Structural Design of a Transport Wing. *Journal of Aircraft*, 27(12):1050–1056, 1990.

[95] I.R. Chittick and J.R.R.A. Martins. An Asymmetric Suboptimization Approach to Aerostructural Optimization. *Optimization and Engineering*, 10(1):133–152, 2009.

[96] I. Kroo. Non-planar Wing Concepts For Increased Aircraft Efficiency. *VKI lectire series on Innivative Configurations and Advanced Concepts for Future Civil Aircraft*, pages 1–29, 2005.

[97] S.J. Miller. *Adaptive Wing Structures For Aeroelastic Drag Reduction and Loads Alleviation.* PhD thesis, University of Manchester, 2010.

[98] L.A. Catalano, D. Quagliarella, and P.L. Vitagliano. Aerodynamic Shape Design Using Hybrid Evolutionary Computing and Multigrid-Aided Finite-Difference Evaluation of Flow Sensitivities. *International Journal for Computer Aided Engineering and Software*, 32(2):178–210, 2014.

[99] G. Kenway and J.R.R.A. Martins. Aerodynamic Shape Optimization of the CRM Configuration Including Buffet-Onset Conditions. In *54th AIAA Aerospace Sciences Meeting, AIAA 2016-1294*, pages 1–23, San Diego, California, 2016.

[100] M. Nemec, D.W. Zingg, and T.H. Pulliam. Multi-Point and Multi-Objective Aerodynamic Shape Optimisation. In *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimisation Conference, AIAA Paper 2002-5548*, Atlanta, Georgia, 2002.

[101] S. Peigin and B. Epstein. Multipoint Aerodynamic Design of Wing-Body Configurations for Minimum Drag. *Journal of Aircraft*, 44(3):971–980, 2007.

[102] G. Venter. Particle Swarm Optimization. In *43rd AIAA / ASME / ASCE / AHS / ASC Structures , Structural Dynamics , and Materials Conference, AIAA 2002-1235*, pages 1–9, Denver, Colorado, 2002.

[103] S.N. Sivanandam and S.N. Deepa. *Introduction to Genetic Algorithms.* Springer, 2008.

[104] O. Castillo, P. Melin, and J. Kacprzyk. *Soft Computing for Hybrid Intelligent Systems.* Springer, Berlin, Germany, 2008.

[105] K. Chiba, S. Obayashi, and K. Nakahashi. High-Fidelity Multidisciplinary Design Optimization of Aerostructural Wing Shape for Regional Jet. In *23rd AIAA Applied Aerodynamics Conference, AIAA 2005-5080*, pages 621–635, Toronto, Canada, 2005.

[106] Z. Jinhuan and Z. Zhenggui. An Improved Genetic Algorithm and its Applications to the Optimisation Design of an Aspirated Compressor Profile. *International Journal for Numerical Methods in Fluids*, 2015.

[107] R.L. Haupt and S.E. Haupt. *Practical Genetic Algorithms.* Wiley-Interscience, 2nd edition, 2004.

[108] M. Mitchell. *An Introduction to Genetic Algorithms.* MIT Press, first edition, 1998.

[109] K.J. Rafferty. *A Comparison Study of Search Heuristics for an Autonomous Multi-Vehicle Air-Sea Rescue System.* PhD thesis, University of Glasgow, 2014.

[110] K.G. Khoo and P.N. Suganthan. Evaluation of Genetic Operators and Solution Representations for Shape Recognition by Genetic Algorithms. *Pattern Recognition Letters*, 23(13):1589–1597, 2002.

[111] E.W. Mcgookin. *Optimisation of Sliding Mode Controllers for Marine Applications : A study*

of methods and implementation issues. PhD thesis, University of Glasgow, 1997.

[112] C.R. Reeves. Using Genetic Algorithms With Small Populations. *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 92–99, 1993.

[113] O. Roeva. Improvement of Genetic Algorithm Performance for Identification of Cultivation Process Models. In *9th WSEAS International Conference on Evolutionary Computing*, pages 34–39, Sofia, Bulgaria, 2008.

[114] O. Roeva, S. Fidanova, and M. Paprzycki. Influence of the Population Size on the Genetic Algorithm Performance In Case of Cultivation Process Modelling. In *Proceedings of the 2013 Federated Conference on Computer Science and Information Systems*, pages 371–376, 2013.

[115] H.M. Pandey, A. Chaudhary, and D. Mehrotra. A Comparative Review Of Approaches To Prevent Premature Convergence In GA. *Applied Soft Computing Journal*, 24:1047–1077, 2014.

[116] M.E. Alfaro Cid. *Optimisation of Time Domain Controllers For Supply Ships Using Genetic Algorithms and Genetic Programming.* PhD thesis, Ph.D Thesis, University of Glasgow, 2003.

[117] B.L. Miller and D.E. Goldberg. *'Genetic Algorithms, Tournament Selection, and the Effect of Noise,' Complex Systems 9.* 1995.

[118] K.A. De Jong and W.M. Spears. A Formal Analysis of the Role of Multi-Point Crossover in Genetic Algorithms. *Annals of Mathematics and Artificial Intelligence*, 5(1):1–26, 1992.

[119] L.M. Schmitt. Theory of Genetic Algorithms II: Models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling. *Theoretical Computer Science*, 310(1-3):181–231, 2004.

[120] K. Deb. Multi-objective Optimization using Evolutionary Algorithms, 2001.

[121] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[122] J. Kim, B. Ovgor, K. Cha, J. Kim, S. Lee, and K. Kim. Optimization of the Aerodynamic and Aeroacoustic Performance of an Axial-Flow Fan. *AIAA Journal*, 52(9):2032–2044, 2014.

[123] G. Droandi and G. Gibertini. Aerodynamic Blade Design with Multi-Objective Optimization for a Tiltrotor Aircraft. *Aircraft Engineering and Aerospace Technology*, 87:19–29, 2015.

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

[125] A. Kumar. Encoding Schemes in Genetic Algorithm. *International Journal of Advanced Research in IT and Engineering*, 2(3):1–7, 2013.

[126] G. Guglieri. Using of Particle Swarm for Performance Optimization of Helicopter Rotor Blades. *Appied Mathematics*, 3:1403–1408, 2012.

[127] Z Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer, 3rd edition, 1995.

[128] R. Poli, W.B. Langdon, and N.F. Mcphee. *A Field Guide to Genetic Programming*. Number March. 2008.

[129] M. Gen and R. Cheng. *Genetic Algorithms and Engineering Optimisation*. John Wiley & Sons, Inc., 2000.

[130] T.Y. Lim. Structured Population Genetic Algorithms: A Literature Survey. *Artificial Intelligence Review*, 41(3):385–399, feb 2012.

[131] D. Popvic. Retaining Diversity of Search Point Distrobution Through aBreeder Genetic Algorithm For Neural Network Learning. *IEEE Int. Conf. Neural Netw.*, pages 495–498, 1997.

[132] C. Fernandes and A. Rosa. A Study On Non-Random Mating and Varying Population Size in Genetic Algorithms Using a Royal Road Function. *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, 1:60–66, 2001.

[133] A. Massaro and E. Benini. A Surrogate-Assisted Evolutionary Algorithm Based on the Genetic Diversity Objective. *Applied Soft Computing*, 36:87–100, 2015.

[134] E. Benini. Three-Dimensional Multi-Objective Design Optimization of a Transonic Compressor Rotor. *Journal of Propulsion and Power*, 20(3):559–565, 2004.

[135] D.J. Zhao, Y.K. Wang, W.W. Cao, and P. Zhou. Optimization of Suction Control on an Airfoil Using Multi-island Genetic Algorithm. *Procedia Engineering*, 99:696–702, 2015.

[136] A. Toffolo and E. Benini. Genetic Diversity as an Objective in Multi-Objective Evolutionary Algorithms. *Evolutionary Computation*, 11(2):151–167, 2003.

[137] D. Dasgupta and D.R. Mcgregor. sGA : A Structured Genetic Algorithm . Technical report, University of Strathclyde, Glasgow, U.K.

[138] B. Artyushenko. Analysis of Global Exploration of Island Model Genetic Algorithm. *CADSM 2009. 10th international conference*, (2):280–281, 2009.

[139] J. Zhang, L. Xu, and R. Gao. Multi-Island Genetic Algorithm Optimization of Suspension System. *Telkomnika*, 10(7):1685–1691, 2012.

[140] L. Eshelman and J. Schaffer. Real-Coded Genetic Algorithms and Interval-Schemata. In *Foundations of Genetic Algorithms*, volume 2, pages 187–202. 1993.

[141] W.T. Mccormick, P.J. Schweitzer, T.W. White, and WT. Mccormick. Problem Decomposition and Data Reorganization by a Clustering Technique. *Operations Research*, 20(5):993–1009, 1972.

[142] G.A. Walters and D.K. Smith. Evolutionary Design Algorithm for Optimal Layout of Tree Networks. *Engineering Optimisation*, 24:261–281, 1995.

[143] J.M. Badyrka, R.M. Jenkins, and R.J. Hartfield. Aerospace Design : A Comparative Study of Optimizers. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, AIAA 2010-1311*, Orlando, Florida, 2010.

[144] A. Oyama, S. Obayashi, and K. Nakahashi. Real-Coded Adaptive Range Genetic Algorithm and Its Application to Aerodynamic Design. *JSME International Journal Series A*, 43(2): 124–129, 2000.

[145] A. Oyama and M. Liou. Transonic Axial-Flow Blade Optimization : Evolutionary Algorithms / Three-Dimensional Navier Stokes Solver. *Journal of Propulsion and Power*, 20(4):612–619, 2004.

[146] D.R. Jones. Large-scales Multi-disciplinary Mass Optimization in the Auto Industry. In *MOPTA08 Modelling and Optimization: Theory and Applications*, number August, Ontario, Canada, 2008.

[147] M. Olhofer, Y. Jin, and B. Sendhoff. Adaptive Encoding for Aerodynamic Shape Optimization Using Evolution Strategies. In *Proceedings of the 2001 Congress on Evolutionary Computation*, pages 576–583, 2001.

[148] J. Periaux, D.S. Lee, L.F. Gonzalez, and K. Srinivas. Fast Reconstruction of Aerodynamic Shapes Using Evolutionary Algorithms and Virtual Nash Strategies in a CFD Design Environment. *Journal of Computational and Applied Mathematics*, 232:61–71, 2009.

[149] D. Sasuki and S. Obayashi. Efficient Search for Trade-Offs by Adaptive Range Multi-Objective Genetic Algorithms. *Journal of Aerospace Computing, Information, and Communication*, 2 (April):44–64, 2005.

[150] K. Krishnakumar. Micro-Genetic Algorithms For Stationary and Non-Stationary Function Optimisation. In *Proceedings of Intelligent Control and Adaptive Systems*, pages 289–296, 1989.

[151] D. Sasaki, M. Morikawa, S. Obayashi, and K. Nakahashi. Aerodynamic Shape Optimization of Supersonic Wings by Adaptive Range Multiobjective Genetic Algorithms. In *First International Conference on Evolutionary Multi-Criterion Optimization*, pages 639–652, 2001.

[152] D. Sasaki, S. Obayashi, and H. Kim. Evolutionary Algorithm vs. Adjoint Method Applied to SST Shape Optimization. In *Proceedings of the Ninth Annual Conference of the CFD Society of Canada*, Waterloo, 2001.

[153] J.F. Wang, J. Periaux, and M. Sefrioui. Parallel evolutionary algorithms for optimization problems in aerospace engineering. *Journal of Computational and Applied Mathematics*, 149 (1):155–169, 2002.

[154] D. Tse and L. Chan. Transonic Airfoil Design Optimization Using Soft Computing Methods. *Canadian Aeronautics and Space Journal*, 46:65–73, 2000.

[155] T.L. Holst and T.H. Pilliam. Aerodynamic Shape Optimisation Using Real-Number-Encoded Genetic Algorithm. In *19th Applied Aerodynamics Conference, AIAA Paper 20012473*, Anaheim, CA, 2001.

[156] R. Eberhart and J. Kennedy. A New Optimizer Using Particle Swarm Theory. In *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995.

[157] J. Kennedy and R.C. Eberhart. Particle swarm optimization. In *Proceedings., IEEE International Conference on Neural Networks*, pages 1942–1948, Perth, Australia, 1995.

[158] Y. Wang, B. Zhang, and Y. Chen. Robust Airfoil Optimization Based On Improved Particle Swarm Optimization Method. *Applied Mathematics and Mechanics*, 32(10):1245–1254, 2011.

[159] Y. Zhang, S. Wang, and G. Ji. A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications. *Mathematical Problems in Engineering*, 2015:1–38, 2015.

[160] J. Kennedy and R.C. Eberhart. *Swarm Intelligence*. Morgan Kaufmann Publishers, 1st edition, 2001.

[161] A. Campos, A. Pozo, and E.P. Duarte. Evaluation Of Asynchronous Multi-Swarm Particle Optimisation On Several Topologies. *Concurrency Computation Practice and Experience*, 25 (6):1057–1071, 2013.

[162] Y. Shi and R. Eberhart. A Modified Particle Swarm Optimizer. In *International Conference on Evolutionary Computing*, pages 69–73, Piscataway, NJ, 1998.

[163] C. Grosan, A. Abraham, and M. Nicoara. Search Optimization Using Hybrid Particle Sub-Swarms and Evolutionary Algorithms. *International Journal of Simulation Systems, Science & Technology*, 6(10):60–79, 2005.

[164] G. Venter and J. Sobieszczanski-Sobieski. Multidisciplinary Optimization Of A Transport Aircraft Wing Using Particle Swarm Optimization. *Structural and Multidisciplinary Optimization*, 26(1-2):121–131, 2004.

[165] C. Praveen and R. Duvigneau. Low Cost PSO Using Metamodels and Inexact Pre-Evaluation : Application to Aerodynamic Shape design. *Computer Methods in Applied Mechanics and Engineering*, 198:1087–1096, 2009.

[166] S. Haghighat, J.R.R.A. Martins, and H.H.T. Lui. Aeroservoelastic Design Optimization of a Flexible Wing. *Journal of Aircraft*, 49(2):432–443, mar 2012.

[167] B. Ouissam, D. Moulay, and R. Mohamed. Title Control of Pitch Angles to Optimize the Aerodynamic Using Particle Swarm Optimization. *WSEAS Transactions of Systems and Control*, 10:22–28, 2015.

[168] M. Azab and C. Ollivier-Gooch. High Order Aerodynamic Optimization Using New Hybrid Sequential Quadratic Programing-Particle Swarm Intelligence Technique. In *50th AIAA Aerospace Sciences Meeting, AIAA Paper 2012-0730*, pages 1–19, 2012.

[169] R.E. Perez and K. Behdinan. Particle Swarm Optimization in Structural Design. In *Swarm Intelligence: Focus on Ant and Particle Swarm Optimisation*, pages 373–394. I-Tech Education

and Publishing, Vienna, 2007.

170 M. Pant, R. Thangaraj, and A. Abraham. Particle Swarm Optimization : Performance Tuning and Empirical Analysis. *Foundations of Computer Intelligence*, 3:101–128, 2009.

171 P.W. Jansen. *Aerostructural Optimization of Non-Planar Lifting Surfaces.* PhD thesis, University of Toronto, 2009.

172 P.C. Fourie and A.A. Groenwold. The Particle Swarm Optimization Algorithm in Size and Shape Optimization. *Structural and Multidisciplinary Optimization*, 23:259–267, 2002.

173 A. Nejat, P. Mirzabeygi, and M.S. Panahi. Airfoil shape optimization using improved Multi-objective Territorial Particle Swarm algorithm with the objective of improving stall characteristics. *Structural and Multidisciplinary Optimization*, 49(6):953–967, 2014.

174 R. Mendes. *Population Topologies and Their In uence in Particle Swarm Performance.* PhD thesis, Universidade doMinho, 2004.

175 M.B. Ghalia. Particle swarm optimization with an improved exploration-exploitation balance. *IEEE*, 978:759–762, 2008.

176 R. Duvigneau. A Multi-Level Particle Swarm Optimization Strategy for Aerodynamic Shape Optimization. *Evolutionary Methods for Design, Optimisation, and Control*, 2007.

177 X. Hu and X. You. Determination of the optimal control parameter range of air supply in an aircraft cabin. *Building Simulation*, 8(4):465–476, 2015.

178 J. Johnson and P. Picton. *Mechatronics: Concepts in Artificial Intelligence: Designing Intelligent Macheines Volume 2*, volume 2. Oxford, Butterworth Heinemann and The Open University, 1996.

179 S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by Simulated Annealing. *SCIENCE*, 220(4598):671–680, 1983.

180 S. Kirkpatrick. Optimization by Simulated Annealing : Quantitative Studies. *Journal of Statistical Physics*, 34:975–986, 1984.

181 S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach.* Prentice Hall, New Jersey, 2nd edition, 1995.

182 N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6): 1087–1092, 1953.

183 R. Mukesh and K. Lingadurai. Aerodynamic Optimization Using Simulated Annealing and its Variants. *International Journal of Engineering Trends and Technology*, 2(3):73–77, 2011.

184 R.W. Eglese. Simulated Annealing : A tool for Operational Research. *European Journal of Operational Research*, 46:271–281, 1990.

185 W.T. Tiow, K.F.C. Yiu, and M. Zangeneh. Application of simulated annealing to inverse

design of transonic turbomachinery cascades. *Journal of Power and Energy*, 216:59–73, 2002.

186 H.E. Romeijin and R.L. Smith. Simulated Annealing for Constrained Global Optimization. *Journal of Global Optimization*, 5:101–126, 1994.

187 W.L. Goffe, G.D. Ferrier, and J Rogers. Global Optimization of Statistical Functions With Simulated Annealing. *Journal of Econometrics*, 60:65–99, 1994.

188 B. Behzadi, C. Ghotbi, and A. Galindo. Application of The Simplex Simulated Annealing Technique to Nonlinear Parameter Optimization for the SAFT-VR Equation of State. *Chemical Engineering Science*, 60:6607–6621, 2005.

189 E. Leylek, J. Manzo, and E. Garcia. Engineering Notes: Bat-inspired Wing Aerodynamics and Optimisation. *Journal of Aircraft*, 47(1):323–328, 2010.

190 J.E. Manzo, E.A. Leylek, and E. Garcia. Drawing Insight From Nature: A Bat Wing For Morphing Aircraft. In *Proceedings of ASMA Conference on Smart Materials, Adaptive Structures, and Intelligent Systems, SMASIS08*, pages 671–678, Ellicott City, MD, 2008.

191 J. Liu. Novel Taguchi-Simulated Annealing Method Applied to Airfoil and Wing Planform Optimization. *Journal of Aircraft*, 43(1):102–109, 2006.

192 W.H. Press and S.A. Teukolsky. Simulated Annealing Optimization Over Continuous Spaces. *Computers in Physics*, 5(426):426–429, 1991.

193 X. Wang and M. Damodaran. Comparison of Deterministic and Stochastic Optimisation Algorithms for Generic Wing Design Problems. *Journal of Aircraft*, 37(5):929–932, 2000.

194 X. Wang and M. Damodaran. Aerodynamic Shape Optimization Using Computational Fluid Dynamics and Parallel Simulated Annealing Algorithms. *AIAA Journal*, 39(8):1500–1508, 2001.

195 S. Aly, M. Ogot, and R. Pelz. Stochastic Approach to Optimal Aerodynamic Shape Design. *Journal of Aircraft*, 33(5):956–961, 1996.

196 P. Gage, I. Kroo, and P. Sobieski. Variable-Complexity Genetic Algorithm for Topological Design. *AIAA Journal*, 33(11):2212–2217, 1995.

197 A.H.W. Bos. Aircraft Conceptual Design by Genetic/Gradient-Guided Optimization. *Engineering Applications of Artificial Intelligence*, 11(3):377–382, 1998.

198 A. Vicini and D. Quagliarella. Airfoil and Wing Design Through Hybrid Optimization Strategies. In *AIAA Paper 1998-2729*, 1998.

199 F. Muyl, L. Dumas, and V. Herbert. Hybrid Method For Aerodynamic Shape Optimization in Automotive Andustry. *Computers & Fluids*, 33(5-6):849–858, 2004. ISSN 00457930.

200 P. Gudla and R. Ganguli. An Automated Hybrid Genetic-Conjugate Gradient Algorithm for Multimodal Optimization Problems. *Applied Mathematics and Computation*, 167:1457–1474, 2005.

[201] T. Goel, R. Vaidyanathan, R.T. Haftka, W. Shyy, N.V. Queipo, and K. Tucker. Response Surface Approximation of Pareto Optimal Front in Multi-Objective Optimization. *Computer Methods in Applied Mechanics and Engineering*, 196:879–893, 2007.

[202] Y.S. Ong, K.Y. Lum, and P.B. Nair. Hybrid evolutionary algorithm with Hermite radial basis function interpolants for computationally expensive adjoint solvers. *Computational Optimization and Applications*, 39(1):97–119, 2008.

[203] S. Lehner and W.A. Crossley. Hybrid Optimization for a Combinatorial Aircraft Design Problem. In *9th AIAA Aviation Technology, Integration, and Operations Conference (ATIO), AIAA 2009-7116*, pages 1–15, South Carolina, 2009.

[204] X. Xing and M. Damodaran. Design of Three Dimentional Nozzle Shapes Using Hybrid Optimisation Techniques. In *42nd AIAA Aerospace Sciences Meeting and Exhibit, AIAA 2004-26*, 2004.

[205] X. Xing and M. Damodaran. Design of Three-Dimansional Nozzle Shape using NURBS, CFD and Hybrid Optimisation Strategies. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA-4368*, 2004.

[206] J. Herbert-Acero, J. Martínez-Lauranchet, O. Probst, S. Méndez-Díaz, K. Castillo-Villar, M. Valenzuela-Rendón, and P. Réthoré. A Hybrid Metaheuristic-Based Approach for the Aerodynamic Optimization of Small Hybrid Wind Turbine Rotors. *Mathematical Problems in Engineering*, pages 1–18, 2014.

[207] M.S. Khurana, H. Winarto, and A.K. Sinha. Airfoil Optimisation by Swarm Algorithm With Mutation and Artificial Neural Networks. *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, (January):1–19, 2009.

[208] L. Cao and D. Zhang. Aerodynamic Configuration Optimization for Hypersonic Gliding Vehicle Based on Improved Hybrid Multi-Objective PSO Algorithm. *IEEE: ICSPCC2015*, pages 1–5, 2015.

[209] M.A. Sahnehsaraei, M.J. Mahmoodabadi, M. Taherkhorsandi, and S.M.M. Yazdi. A Hybrid Global Optimisation Algorithm: Particle Swatm Optimisation in Association with a Genetic Algorithm. *Complex System Modelling and Control Through Intelligent Soft Computations*, 9: 45–86, 2015.

[210] Y. Wang and W. Li. Aircraft Aerodynamic Configuration design based on Hybrid Multi-objective Particle Swarm Optimisation Algorithm. *Chinese Journal of Aeronautics*, 29(5): 1202–1206, 2008.

[211] D.J.J. Toal, N.W. Bressloff, A.J. Keane, and C.M.E. Holden. The Development Of A Hybridized Particle Swarm For Kriging Hyperparameter Tuning. *Engineering Optimization*, 43 (6):675–699, 2011.

212 J.I. Madsen, W. Shyy, and R.T. Haftka. Response Surface Techniques For Diffuser Shape Optimization. *AIAA Journal*, 38(9):1512–1518, 2000.

213 A. Lundberg, P. Hamlin, D. Shankar, A. Broniewicz, T. Walker, and C. Landström. Automated Aerodynamic Vehicle Shape Optimization Using Neural Networks and Evolutionary Optimization. *SAE International Journal of Passenger Cars - Mechanical Systems*, 8(1):242–251, 2015.

214 K. Boopathy and M. Rumpfkeil. A Multivariate Interpolation and Regression Enhanced Kriging Surrogate Model. In *21st AIAA Computational Fluid Dynamics Conference, AIAA 2013-2964*, pages 1–18, 2013.

215 A.J. Keane. Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods Introduction. *Journal of Aircraft*, 40(4), 2003.

216 N.V. Queipo, R. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P.K. Tucker. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41(1):1–28, 2005.

217 T.W. Simpson, J. Peplinski, P.N. Kock, and J.K. Allen. Metamodels for Computer-Based Engineering Design: Survey and Recommendations. *Engineering with Computers*, 17(2):129–150, 2001.

218 R. Jin, W. Chen, and T.W. Simpson. Comparative Studies of Metamodeling Techniques Under Multiple Modeling Criteria. *Journal of Structural Optimization*, 23(1):1–13, 2001.

219 A.I.J. Forrester and A.J. Keane. Recent Advances in Surrogate-Based Optimization. *Progress in Aerospace Sciences*, 45:50–79, 2009.

220 Y. Lian and M. Liou. Multiobjective Optimisation Using Coupled Response Surface Model and Evolutionary Algorithm. *AIAA Journal*, 43(6):1316–1325, 2005.

221 K.B Collins and D.N. Mavris. Application of Low- and High-Fidelity Simulation Tools to Helicopter Rotor Blade Optimization. *Journal of the American Helicopter Society*, 042003: 1–11, 2013.

222 D. Leusink, D. Alfano, and P. Cinnella. Multi-Fidelity Optimization Strategy for the Industrial Aerodynamic Design of Helicopter Rotor Blades. *Aerospace Science and Technology*, 42:136–147, 2015.

223 T.D. Robinson, M.S. Eldred, K.E. Willcox, and R. Haimes. Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping. *AIAA Journal*, 46(11):2814–2822, 2008.

224 P.M. Zadeh, V.V. Toropov, and A.S. Wood. Metamodel-Based Collaborative Optimization Framework. *Structural and Multidisciplinary Optimization*, 38(2):103–115, 2009.

225 B. Rosenbaum and V. Schulz. Response Surface Methods for Efficient Aerodynamic Surrogate Models. *Computational Flight Testing*, 123:113–129, 2013.

226 D.J.J. Toal and A.J. Keane. Efficient Multipoint Aerodynamic Design Optimization. *Journal*

of Aircraft, 48(5):1685–1695, 2011.

227 A.I.J. Forrester. Black-Box Calibration for Complex-System. *Philosophical Transactions of The Royal Society*, pages 3567–3579, 2010.

228 Y. Kuya, K. Takeda, X. Zhang, and A.I.J. Forrester. Multi fidelity Surrogate Modeling of Experimental and Computational Aerodynamic Data Sets. *AIAA Journal*, 49(2):289–298, 2011.

229 J. Laurenceau, M. Meaux, M. Montagnac, and P. Sagaut. Comparison of Gradient-Based and Gradient-Enhanced Response-Surface-Based Optimizers. *AIAA Journal*, 48(5):981–994, 2010.

230 H. Chung and J.J. Alonso. Using Gradients to Construct Cokriging Approximation Models for High-Dimensional Design Optimization Problems. In *401th AIAA Aerospace Sciences Meeting and Exhibit, AIAA 2002-0317*, pages 1–15, 2002.

231 K. Takenaka, K. Hatanaka, W. Yamazaki, and K. Nakahashi. Multidisciplinary Design Exploration for a Winglet. *Journal of Aircraft*, 45(5):1601–1611, 2008.

232 M.D. McKay, R.J. Beckman, and W.J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code. *American Statistical Association and American Society for Quality*, 42(1):55–61, 1979.

233 D.R. Jones, M. Schonlau, and J. William. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimisation*, 13(4):455–492, 1998.

234 B. Glaz, P. Friedmann, and L. Liu. Helicopter Vibration Reduction throughout the Entire Flight Envelope Using Surrogate-Based Optimization. *Jounral of The American Helicopter Society*, 54(012007):1–15, 2009.

235 W Jacquet, B Truyen, P. De Groen, I Lemahieu, and J Cornelis. Global Optimization In Inverse Problems: A Comparison of Kriging and Radial Basis Functions. Technical report, 2005.

236 R.L.T. Bevan, D.J. Poole, C.B. Allen, and T.C.S. Rendall. Simulation and Surrogate-Based Design of Rectangular Vortex Generators for Tiltrotor Aircraft Wings. In *33rd AIAA Applied Aerodynamics Conference, AIAA 2015-2725*, pages 1–18, Dallas, TX, 2015.

237 R.L.T. Bevan, D.J. Poole, C.B. Allen, and T.C.S. Rendall. Optimization of Vane-Type Vortex Generators for Tiltrotor Wings using Computational Fluid Dynamics. In *54th AIAA Aerospace Sciences Meeting, AIAA 2016-1304*, pages 1–19, San Diego, California, 2016.

238 Y.S. Ong, P.B. Nair, and A.J. Keane. Evolutionary Optimization of Computationally Expensive Problems Via Surrogate Modeling. *AIAA Journal*, 41(4):687–696, 2003.

239 A. Massaro, A. D'Andrea, and B. Ernesto. Multiobjective-Multipoint Rotor Blade Optimization in Forward Flight Conditions Using Surrogate-Assisted Memetic Algorithms. In *Conference: 37th European Rotorcraft Forum*, 2011.

[240] F.A.C. Viana, R.T. Haftka, and L.T. Watson. Efficient Global Optimization Algorithm Assisted by Multiple Surrogate Techniques. *Journal of Global Optimisation*, 56:669–689, 2013.

[241] H. Tianyuan and Y. Xiongqing. Aerodynamic/Stealthy/Structural Multidisciplinary Design Optimization of Unmanned Combat Air Vehicle. *Chinese Journal of Aeronautics*, 22(4):380–386, 2009.

[242] V .R. Joseph, Y. Hung, and A. Sudijianto. Blind Kriging : A New Method for Developing Metamodels. *ASMA Journal of Mechanical Design*, 130:1–8, 2008.

[243] N. Namura, K. Shimoyama, S. Jeong, and S. Obayashi. Kriging/RBF-Hybrid Response Surface Method for Highly Nonlinear Functions. *Journal of Computational Science and Technology*, 6: 81–96, 2012.

[244] D.H. Wolpert and W.G. Macready. No Free Lunch Theorems for Search. *Technical Report SFI-TR-95-02-101*, 1996.

8. AUTHOR BIOGRAPHY

Shaun N. Skinner is a PhD candidate in the Aerospace Sciences research division in the School of Engineering at the University of Glasgow. His Ph.D. focuses on the numerical and experimental aerodynamic optimisation of non-planar wing configurations for drag, vibration, and load alleviation. Other research experience and interests in fluid dynamics includes: (i) supersonic film cooling; (ii) passive and active flow control for low and high speed flows; (iii) aero-elastics; (iv) experience in the design and usage of wing tunnel models and flow diagnostic equipment in experimental facilities ranging from low subsonic, incompressible flows, to supersonic flows, upto Mach 3.

Hossein Zare-Behtash is a lecturer in the Aerospace Sciences research division in the School of Engineering University of Glasgow. He is a Chartered Engineer, Member of the Royal Aeronautical Society, Member of the Institution of Mechanical Engineers, and Senior Member of the American Institute of Aeronautics and Astronautics. His research interests cover a spectrum of fluid dynamics challenges and applications related to fluid-structure interactions, such as; (i) active/passive flow control technologies for low-speed and transonic, applications such as plasma actuators, 3D bumps, and vortex generators, (ii) development/application of advanced flow diagnostic and instrumentation methods such as thermographic phosphors, and unsteady PSPs, (iii) extensive experience in the design and usage of a variety of experimental facilities ranging from low subsonic, incompressible, Mach numbers to hypersonics, Mach numbers 5 and above, (iv) unsteady aerodynamics of non-planar wing planforms optimised using structured genetic algorithms.

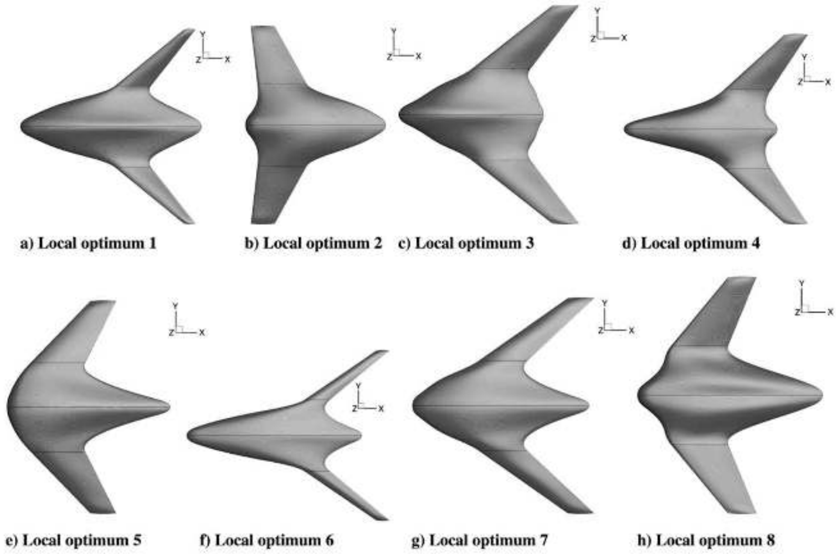


Figure 1: Local optima found in a blended wing optimisation.[23]

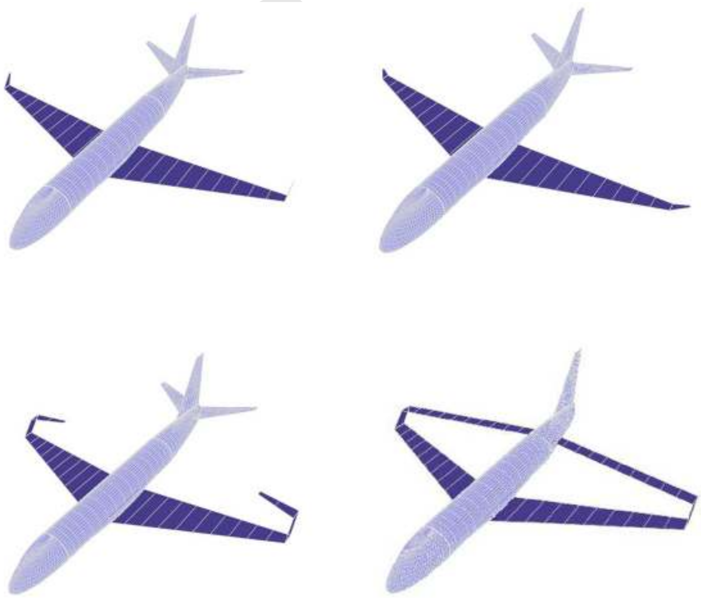


Figure 2: Global variables parametrising wing topology.[171]

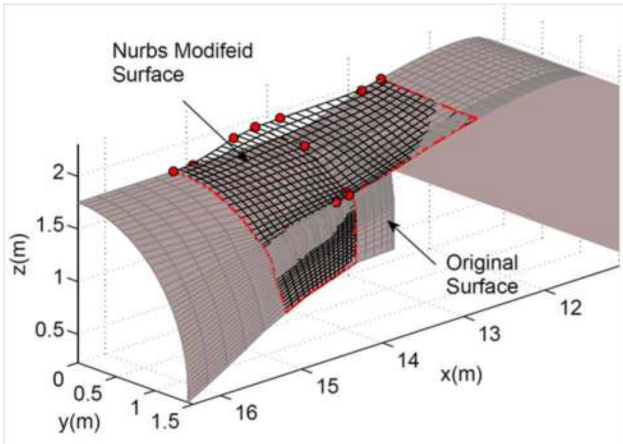


Figure 3: NURBS surfaces parametrising surface blend on fuselage.[27]

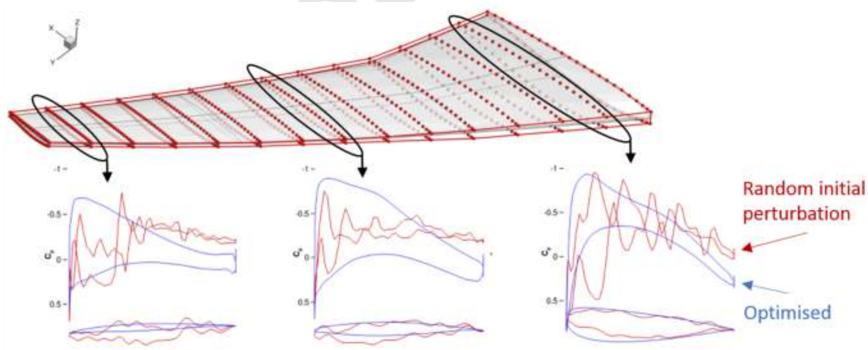


Figure 4: Free-form deformation parametrising wing with 720 control points. Selected wing cross-sections are highlighted to show initial perturbed (red), and final optimised (blue) wing cross-sections. Associated C_P distributions shown.[12]

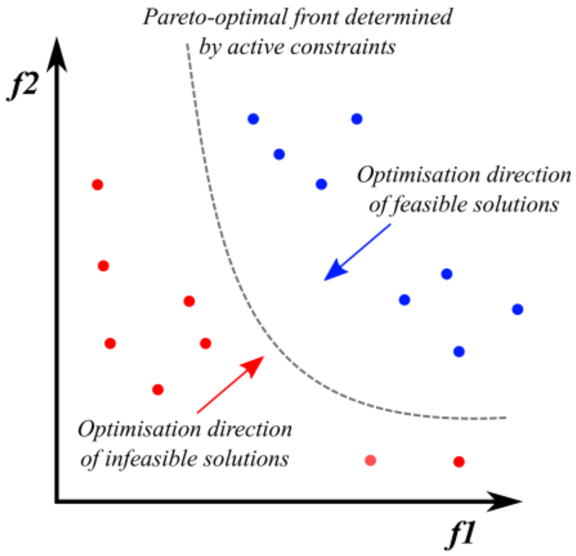


Figure 5: Concept of using parallel evaluation strategy of feasible and infeasible solutions to guide optimisation direction in a GA.

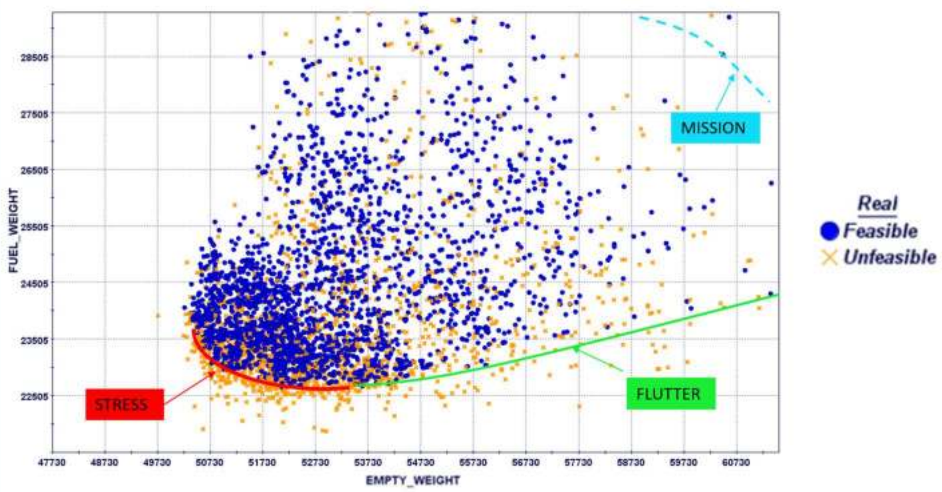


Figure 6: Overlapping boundaries for feasible and infeasible solutions objective space.[40]

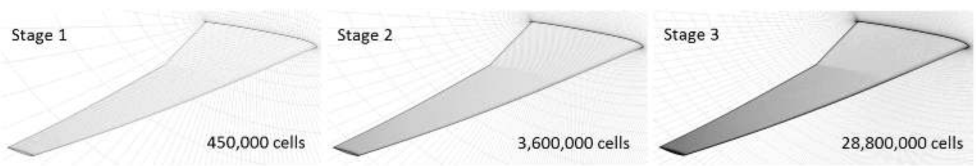


Figure 7: Multi-level optimisation stages for the systematic increase of solver fidelity as different stages of solution optimality is achieved.[12]

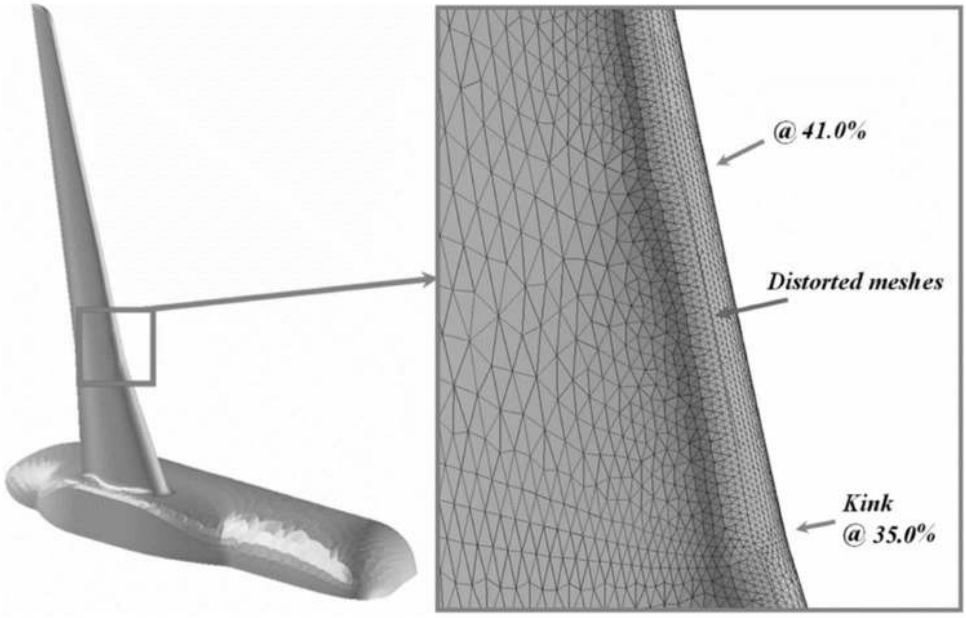


Figure 8: Example of mesh distortion at the wing leading edge.[59]

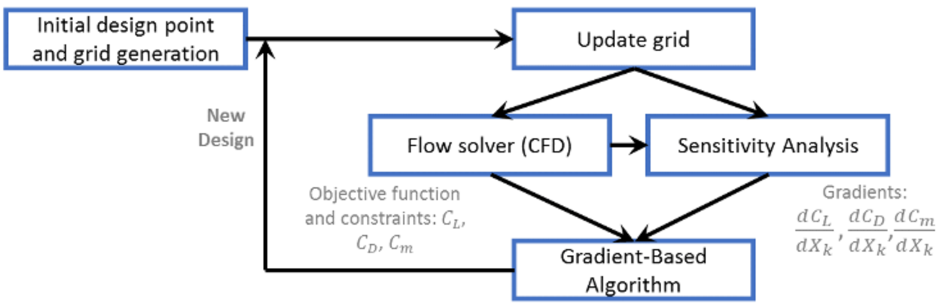


Figure 9: Schematic diagram of a gradient-based aerodynamic optimisation process.

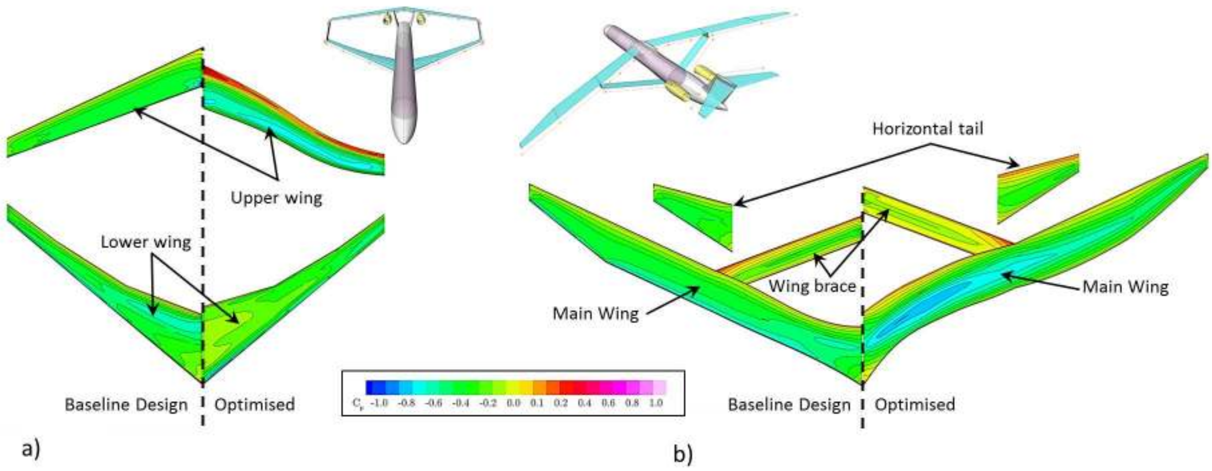


Figure 10: Upper surface pressure coefficient contours over initial and optimised un-conventional aircraft configurations.[86]

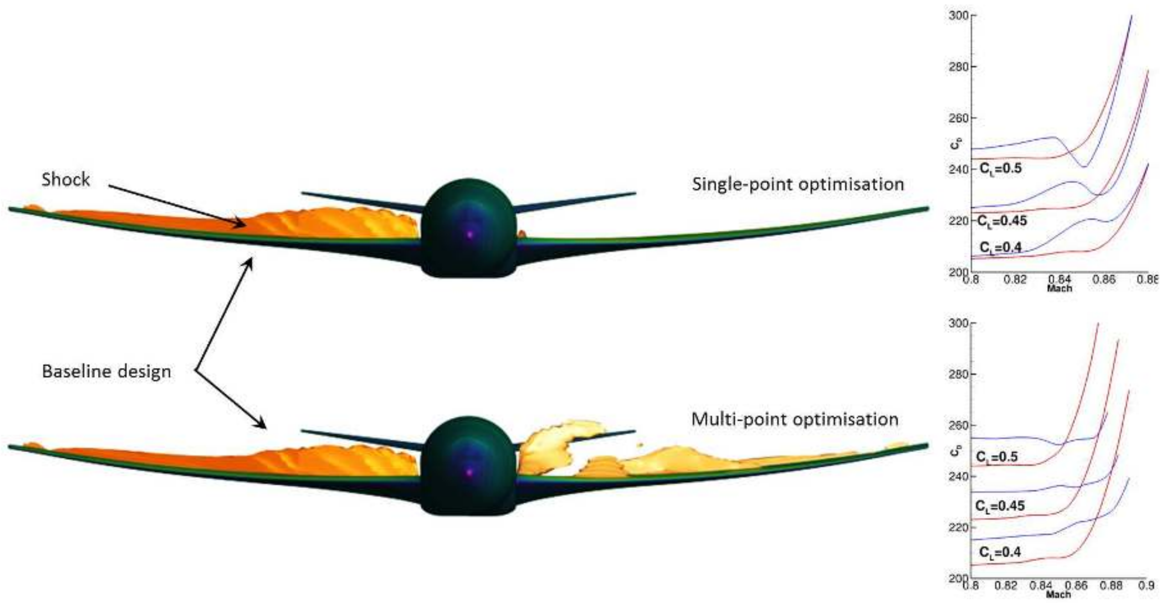


Figure 11: High performance low drag solutions found for single design point at nominal operating conditions. For multi-point optimisation performance at the nominal operating condition is sacrificed.[99]

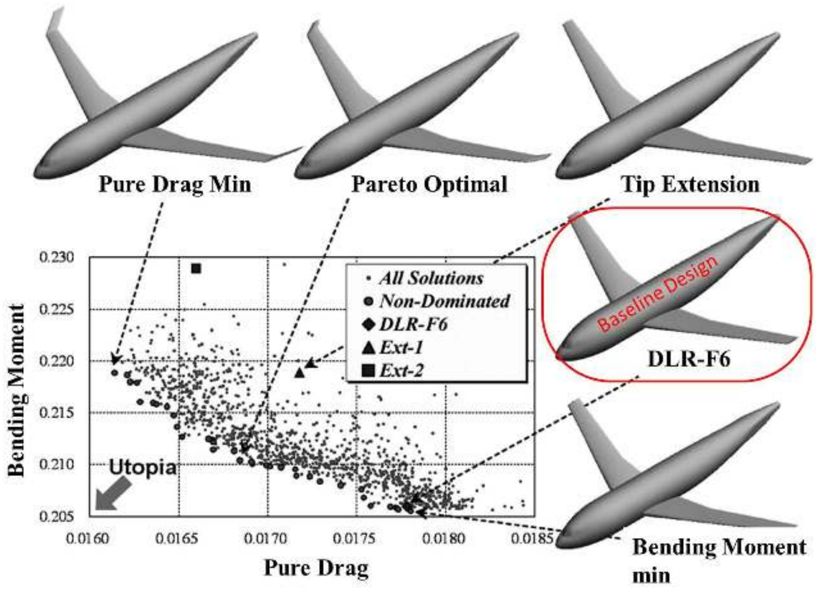


Figure 12: Yamazaki et al.[11] non-dominated solution information for a winglet design problem using a genetic algorithm.

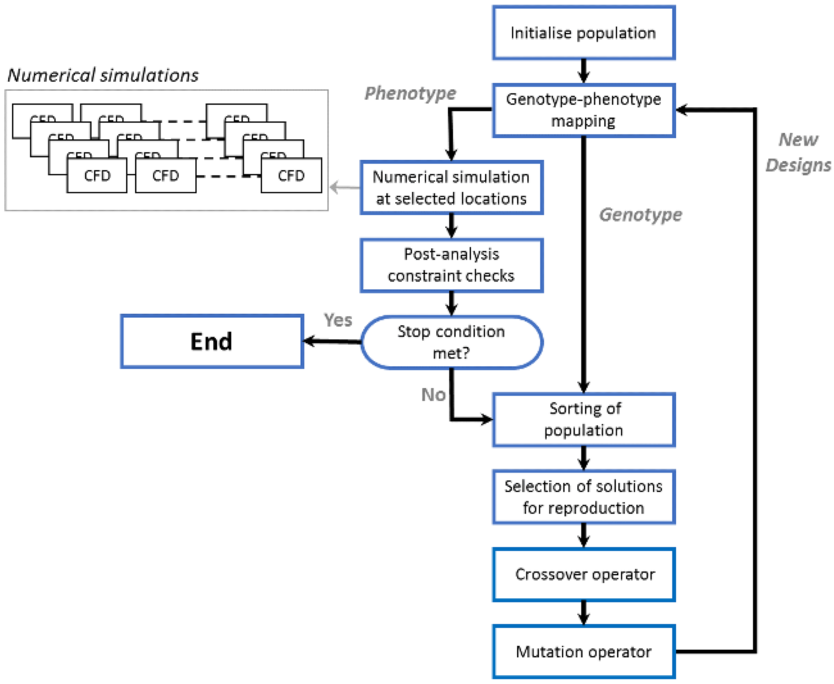


Figure 13: Schematic diagram of typical genetic algorithm structure for aerodynamic optimisation.

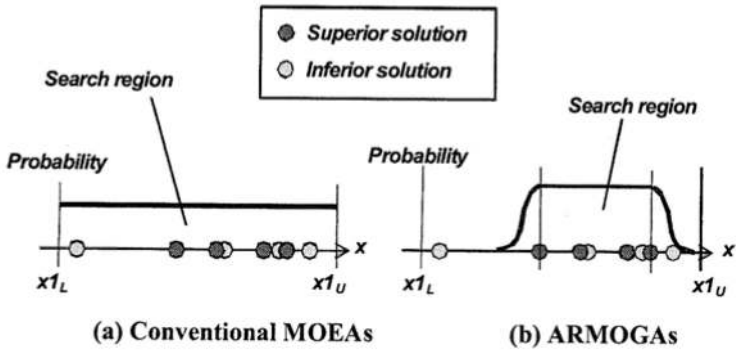


Figure 14: Sketch of search regions.[149]

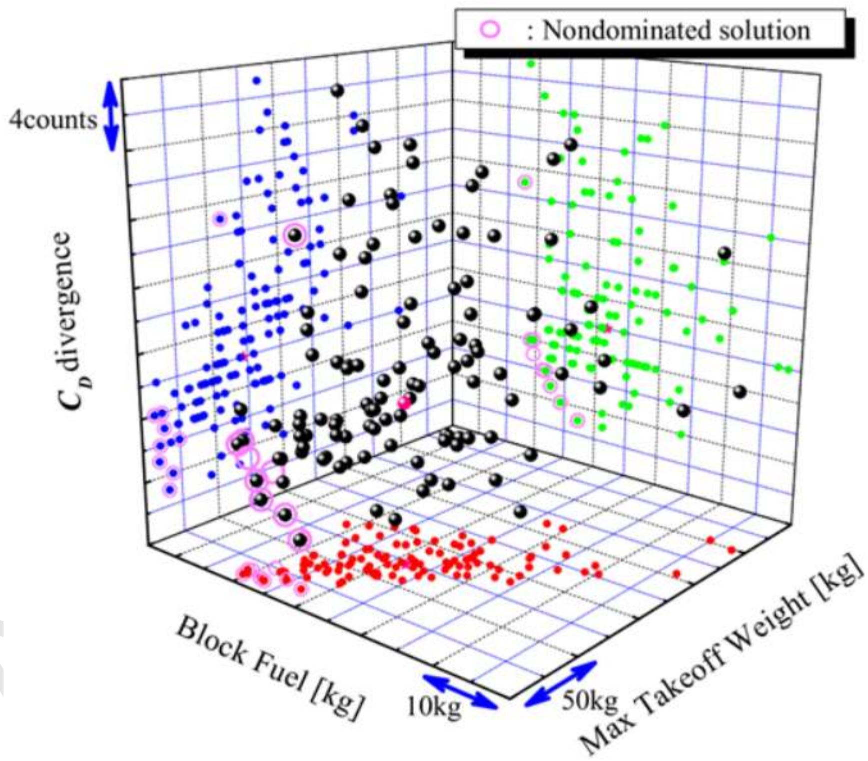


Figure 15: All solutions identified by ARMOGA in three-dimensional space of all objective functions.[59]

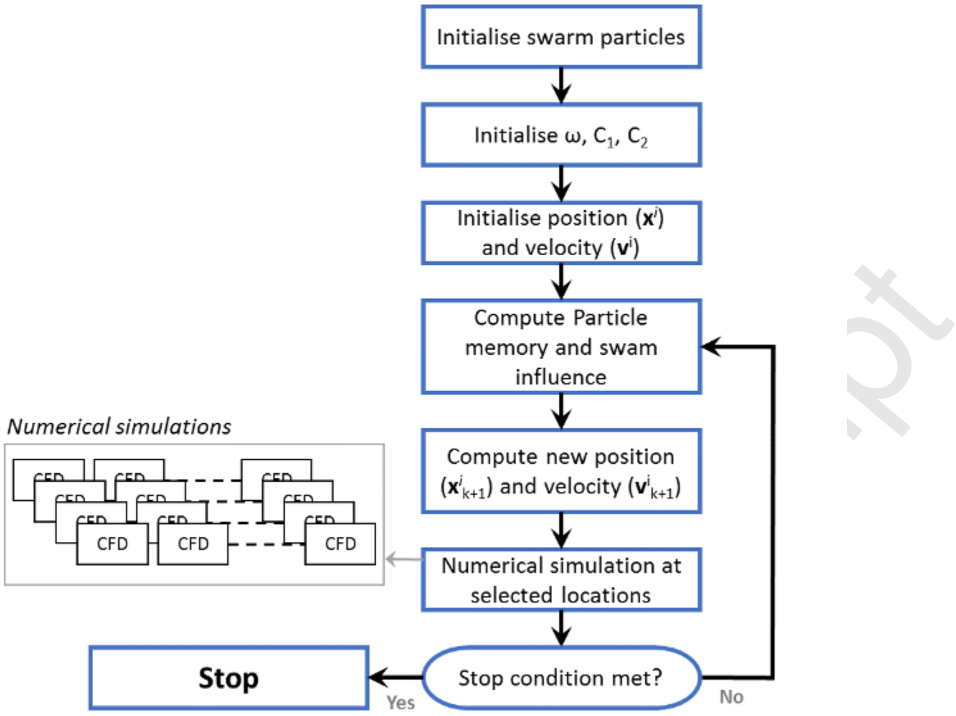


Figure 16: Schematic diagram of typical particle swarm optimisation structure.

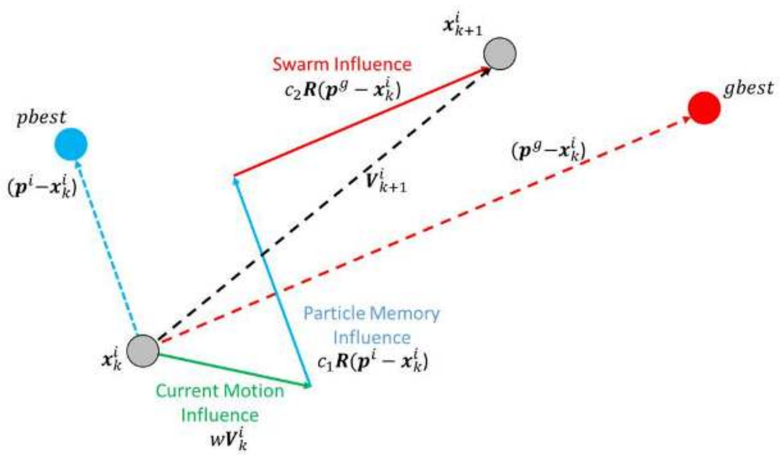


Figure 17: Depiction of the velocity and position updates in Particle Swarm Optimisation.

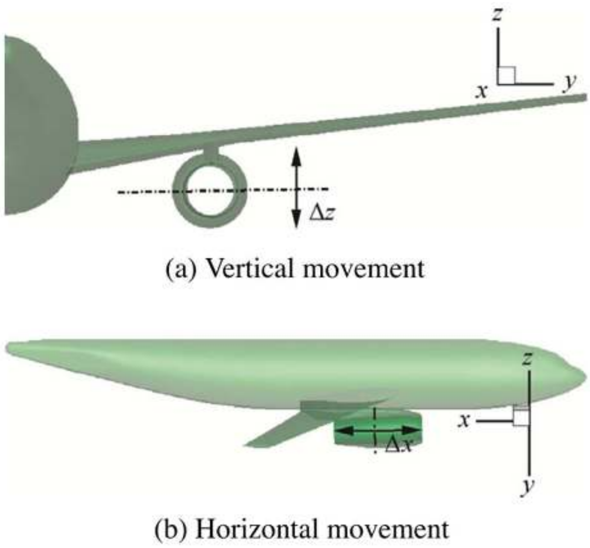


Figure 18: Nacelle design variables used by Li et al.[61]

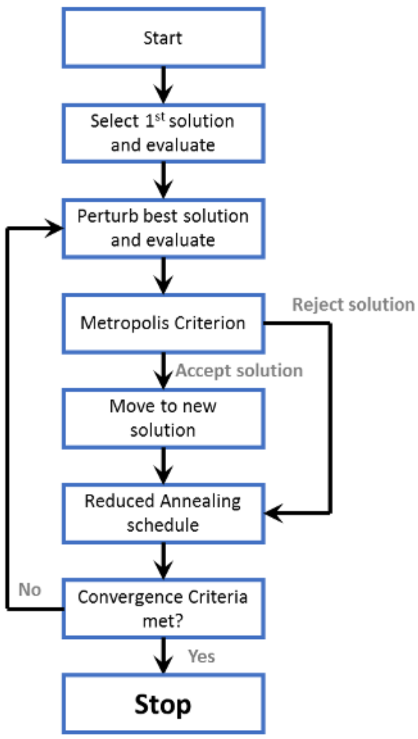


Figure 19: Simulated Annealing flowchart.

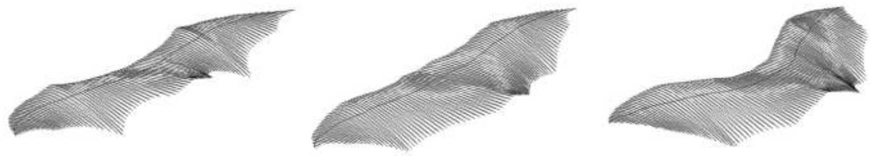


Figure 20: Different bat-wing topologies for wing morphology optimisation using simulated annealing.[189]

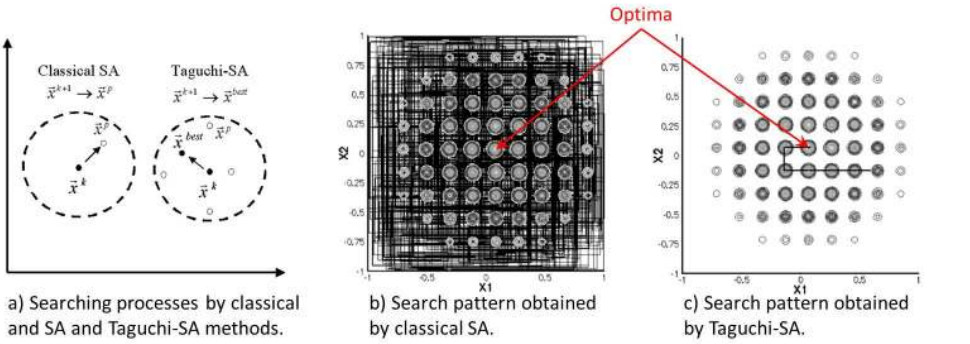


Figure 21: Comparison of search processes and resulting search patterns obtained using classical SA and Tauchi-SA methods for a non-linear multi-modal function.[191]

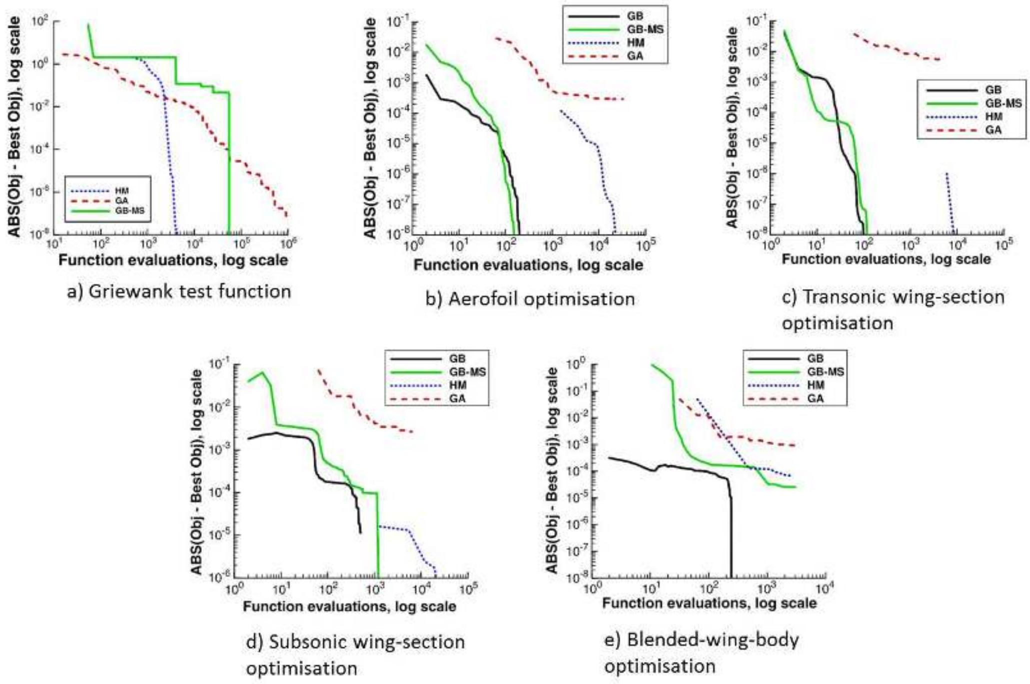


Figure 22: Compiled optimisation results, from Chernukhin and Zingg,[23] comparing optimiser convergence plots for a gradient-based (GB), a multi-start gradient-based (GB-MS), a genetic algorithm (GA), and a developed genetic algorithm/gradient-based hybrid method (HM) for different optimisation problems.

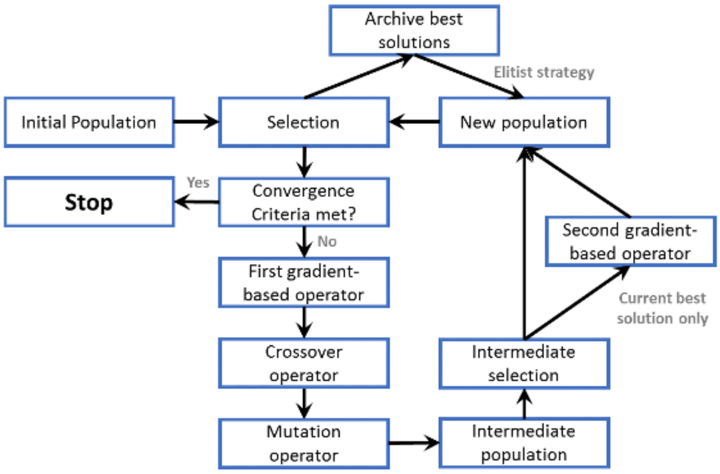


Figure 23: Hybrid organic-optimisation algorithm.[98]

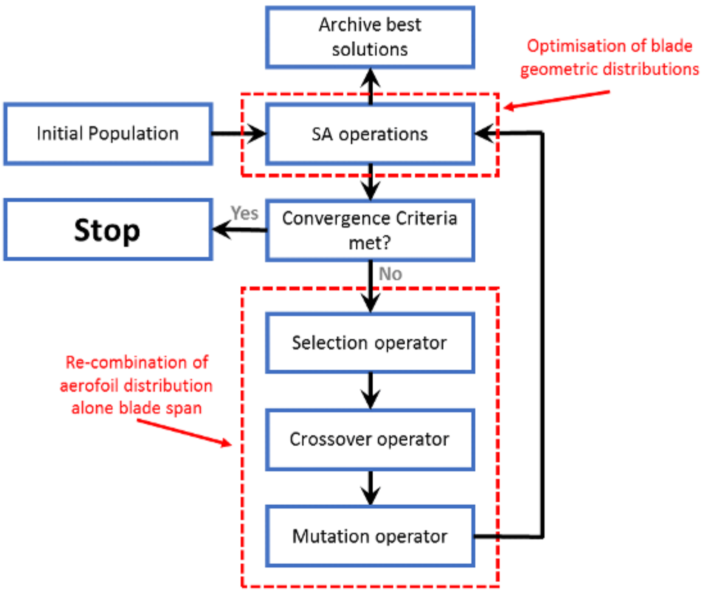


Figure 24: Hybrid GA-SA organic-optimisation algorithm.[206]

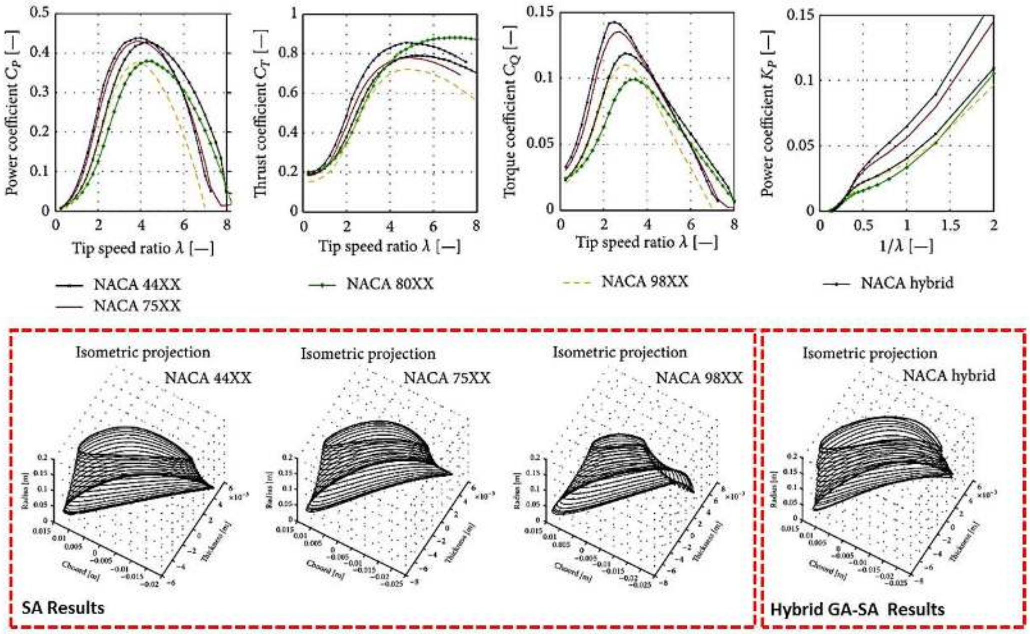


Figure 25: Geometry and aerodynamic performance of optimised wind turbine blades, optimised by the SA and hybrid GA-SA algorithms.[206]

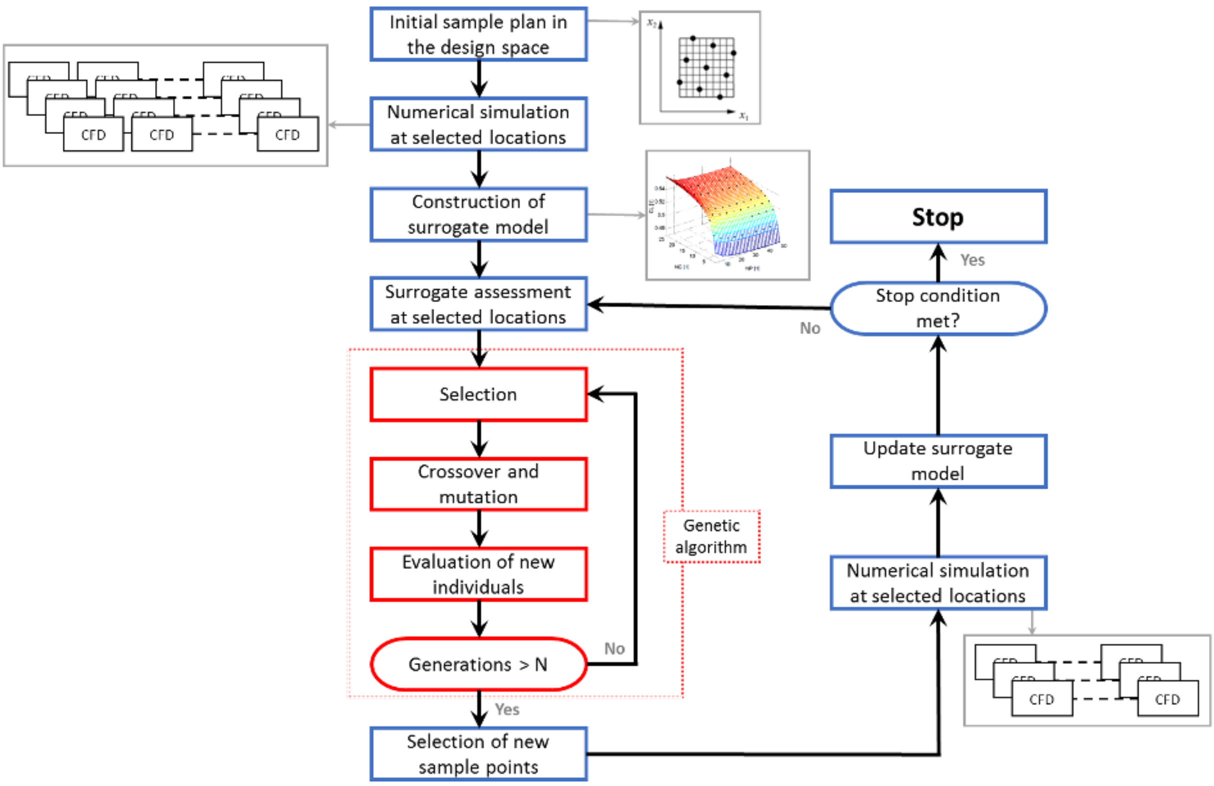


Figure 26: Example structure for surrogate based optimisation with a standard genetic algorithm.

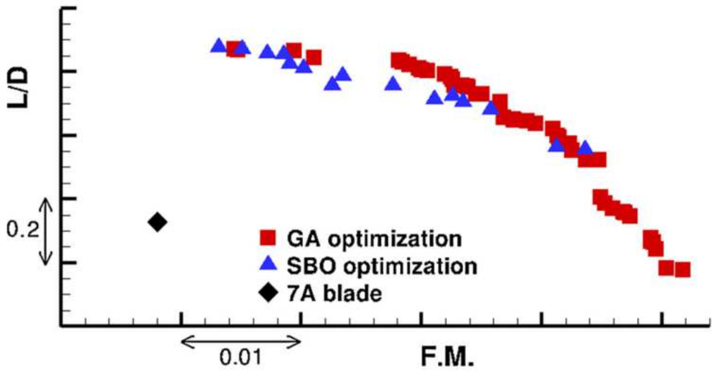


Figure 27: Pareto optimal fronts for the GA and the surrogate assisted GA compared to the 7A helicopter blade baseline design.[222]

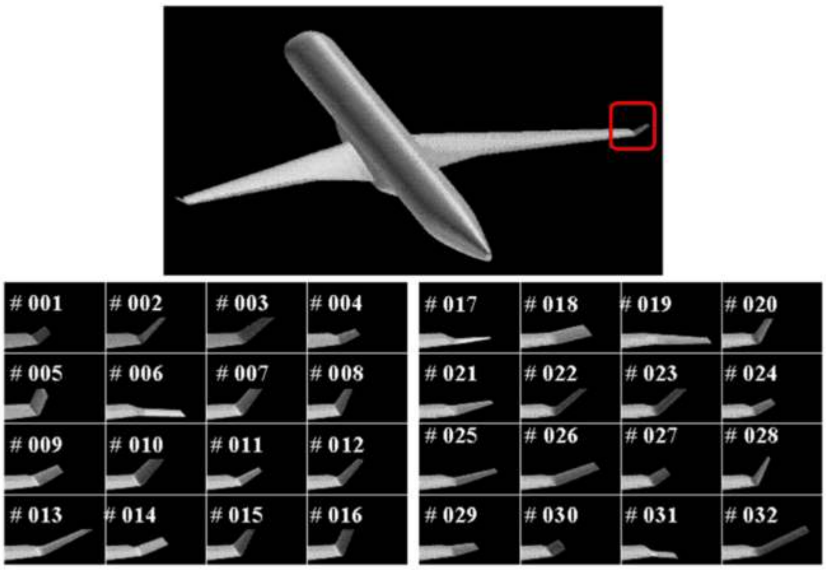


Figure 28: Takenaka et al.[231] 32 initial Latin Hypercube sample sites for the development of a kriging surrogate in the optimisation of a commercial aircraft winglet.