# Design of Low-Sweep Wings for Maximum Range

Timothy M. Leung* and David W. Zingg†

*Institute for Aerospace Studies, University of Toronto, Toronto, Ontario, M3H 5T6, Canada*

**An efficient Newton-Krylov algorithm for high-fidelity aerodynamic shape optimization is used to design low-sweep wings for maximum range at transonic speeds. In this approach, the steady flow solution is obtained using the Newton method with pseudo-transient continuation. The objective function gradient is computed using the discrete-adjoint method. Linear systems from both the flow and adjoint systems are solved using a preconditioned Krylov method. A quasi-Newton optimizer is used to find the search direction. It is coupled with a line-search algorithm. Our single-point optimization results show that it is possible to design shock-free unswept wings at Mach numbers and lift coefficients comparable to the operating conditions of modern transonic transport aircraft. Robust wing designs for low-sweep and unswept wings under the same operating conditions are obtained through multi-point optimization.**

## I. Introduction

In the design of future generations of civil transport aircraft, optimizing the wing configuration to maximize range is of particular interest to designers. The range maximization problem is analogous to minimizing fuel burn at a fixed range. The amount of fuel consumed in flight has a profound impact on the operating cost for the operator (airline), as fuel cost is currently the largest single expense for airlines globally.[1] Fuel burn also has significant environmental impact, through the production of greenhouse gas emissions, particularly carbon dioxide ($CO_2$) and nitrogen oxides ($NO_x$). In the 1990s, when jet fuel was substantially less expensive, designs favoured operating at higher speeds for passenger comfort, and to reduce labour cost during flight. However, given the rising cost of fuel and increasing environmental regulations in recent years, this philosophy is being revisited.

Advances in high-fidelity computational fluid dynamics (CFD) techniques over the past decades, as well as the development of CFD-based aerodynamic shape optimization techniques in the design process, have allowed engineers to examine the range maximization problem more closely. We are particularly interested in the aerodynamic optimization of unswept and low-sweep wings that can operate within a flight envelope similar to the highly-swept wings currently used by civil transport aircraft. Low-sweep wings are appealing because they can have reduced structural weight, which is another important factor in maximizing range, and the potential for natural laminar flow. The goal is to optimize for robust, shock-free designs at transonic Mach numbers.

Our study is motivated by the work of Jameson *et al.*[2] Their primary results show that even with aerodynamic shape optimization alone, the range of a low-sweep wing operating at lower Mach number is comparable to a highly-swept wing at higher Mach number. In our current work, we revisit the range maximization problem using the Newton-Krylov algorithm based on Nemec and Zingg[3] and Leung and Zingg.[4] Our study includes an improved curve fit to account for the engine's thrust specific fuel consumption as a function of Mach number, as well as using a flexible geometry parameterization method using basis-splines (B-splines). We also examine the effect of robust design. The objective is to assess potential benefits of unswept and low-sweep wings and to find the maximum Mach numbers for robust shock-free operation as a function of sweep angle under a particular set of geometric constraints.

---

*Research Engineer, AIAA Member

†Professor and Director, Tier 1 Canada Research Chair in Computational Aerodynamics, J. Armand Bombardier Foundation Chair in Aerospace Flight, Associate Fellow AIAA

## II. Optimization Problem

### A. Problem Formulation

The goal of aerodynamic shape optimization is to find a set of design variables $\mathcal{X}$ and state variables $\mathbf{Q}$ such that a scalar objective function $\mathcal{J}$ minimized:

$$\min_{\mathcal{X}} \mathcal{J}(\mathcal{X}, \mathbf{Q}) \tag{1}$$

The optimization is subject to both geometric constraints:

$$C_j(\mathcal{X}) \leq 0 \tag{2}$$

as well as the flow constraint:

$$\mathbf{R}(\mathcal{X}, \mathbf{Q}) = 0 \tag{3}$$

In the present study, the flow constraint is defined by the discrete steady Euler equations governing compressible inviscid flow.

### B. Objective Function

We formulate the objective function $\mathcal{J}$ based on the Breguet range equation, which specifies the range $R$ of an aircraft under a cruise-climb flight profile:

$$R = \frac{V}{\mathrm{TSFC}} \frac{L}{D} \ln\left(\frac{W_\mathrm{i}}{W_\mathrm{f}}\right) \tag{4}$$

where $V$ is the speed of the aircraft, $L/D$ is the aircraft's lift-to-drag ratio (aerodynamic efficiency), TSFC is the engine's thrust specific fuel consumption, and $W_\mathrm{i}$ and $W_\mathrm{f}$ are initial and final weights of the aircraft. For high-bypass-ratio turbofan engines used on commercial transport aircraft, TSFC varies roughly linearly with $M$ at transonic speeds. We use the relationship by Mattingly:[5]

$$\mathrm{TSFC} \propto 0.45M + 0.40 \tag{5}$$

The exact relationship between TSFC and $M$ depends on the specific engine configuration. Under the cruise-climb flight profile, the aircraft operates at a constant speed $V$ and lift coefficient $C_\mathrm{L}$. As weight decreases during flight, the altitude increases such that the lift $L$ generated is always equal to the weight. For an aerodynamic shape optimization, the effects of weights $W_\mathrm{i}$ and $W_\mathrm{f}$ are ignored for now. It should be noted that these weights are important factors for a full aero-structural optimization.

Substituting non-dimensional quantities into (4), the range of an aircraft scales with a range factor $\mathcal{R}$:

$$\mathcal{R} = \frac{M}{\mathrm{TSFC}} \frac{C_\mathrm{L}}{C_\mathrm{D}} \tag{6}$$

For optimization at a fixed Mach number, $\mathcal{R}$ can be maximized by maximizing the lift-to-drag ratio at a fixed $C_\mathrm{L}$. For this type of problem, we use the lift-constrained drag minimization objective function:

$$\mathcal{J}_0 = \omega_\mathrm{L}\left(1 - \frac{C_\mathrm{L}}{C_\mathrm{L}^*}\right)^2 + \omega_\mathrm{D}\left(1 - \frac{C_\mathrm{D}}{C_\mathrm{D}^*}\right)^2 \tag{7}$$

The targets in lift and drag ($C_\mathrm{L}^*$, $C_\mathrm{D}^*$) as well as the weights ($\omega_\mathrm{L}$, $\omega_\mathrm{D}$) are specified by the user. We find the values $\omega_\mathrm{L} = 100$ and $\omega_\mathrm{D} = 1.0$ to be effective based on previous experience.[4] If the target lift is attainable and target drag is not, then the lift constraint appears as a penalty term. Therefore, $C_\mathrm{D}^*$ should be a value that is physically unattainable to ensure that the final drag is minimized. Once we have obtained the optimized lift-to-drag ratio, we can compute the range factor $\mathcal{R}$.

### C. Multi-Point Optimization

For multi-point optimization with $N_p$ operating conditions, the objective function is the weighted sum of all operating points:[6]

$$\mathcal{J}_{\mathrm{T}} = \sum_{i=1}^{N_p} \omega_i \mathcal{J}_i \tag{8}$$

This weighted sum is an approximation of a weighted integral $I$ of the objective function over a range of Mach numbers:

$$I = \int_{M_1}^{M_2} P(M)\mathcal{J}\left[\mathcal{X}, \mathbf{Q}(M)\right] dM \tag{9}$$

where the user-specified weighting function $P(M)$ reflects the relative importance attached by the designer to each Mach number in this range. In (8), the weights $\omega_i$ combine the weighting function $P(M)$, as well as the Newton-Cotes rule that is used to approximate (9). For $P(M) = 1$, if we select equally spaced operating points between $M_1$ and $M_2$, with the first and last operating point at $M_1$ and $M_2$ respectively, and apply the trapezoidal rule, we obtain the following weights:

$$\omega_i = \begin{cases} 0.5 & i = 1 \\ 1.0 & i = 2 \ldots N_p - 1 \\ 0.5 & i = N_p \end{cases} \tag{10}$$

### D. Geometric Constraints

We have implemented two geometric constraints: a volume constraint to limit the change in the volume enclosed by the wing, and a thickness constraint to maintain minimum thickness at specified locations of the wing. Both are expressed as penalty terms in the objective function:

$$\mathcal{J} = \mathcal{J}_0 + \mathcal{J}_{\mathrm{p,V}} + \mathcal{J}_{\mathrm{p,T}} \tag{11}$$

For volume constraint, the penalty term $\mathcal{J}_{\mathrm{p,V}}$ is added when the volume $V$ deviates from the initial volume $V_0$:

$$\mathcal{J}_{\mathrm{p,V}} = \omega_{\mathrm{V}} \left(1 - \frac{V}{V_0}\right)^2 \tag{12}$$

The penalty weight $\omega_{\mathrm{V}}$ is supplied by the user. We choose a value of $\omega_{\mathrm{V}} = 50.0$. For thickness constraints, we specify a minimum thickness at fixed relative positions along the chord $(x/c)$ and semi-span $(y/(b/2))$ on the wing. The penalty term is added if thickness at the $i$-th constraint location $t_i$ is below the minimum thickness $t_i^*$. The contributions from all thickness constraints are summed and multiplied by a user supplied weight $\omega_{\mathrm{T}}$:

$$\mathcal{J}_{\mathrm{p,T}} = \omega_{\mathrm{T}} \sum_i \left(1 - \frac{t_i}{t_i^*}\right)^2 \tag{13}$$

We use a penalty weight of $\omega_{\mathrm{T}} = 40.0$.

## III. Algorithm Description

### A. Flow Analysis

The governing equations for the optimization are the Euler equations, which are discretized on multi-block structured grids. In our parallel strategy, each block in the grid and the corresponding component of $Q$ is distributed to a separate processor. Second-order centred differencing is applied at interior nodes, while first-order one-sided differencing is used at boundaries and block interfaces. For numerical stability, we use a scalar dissipation model based on the JST scheme.[7,8] Boundary conditions and the coupling between blocks at the interfaces are enforced using simultaneous approximation terms (SATs).[9]

Discretization of the steady Euler equations produces a set of nonlinear algebraic equations, which can be written as:

$$\mathbf{R}(\mathbf{Q}) = \mathbf{0} \tag{14}$$

Beginning with an initial guess $\mathbf{Q}^{(0)}$ based on free-stream properties, and applying the Newton method, we solve a linear system in the form:

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right)^{(n)} \Delta \mathbf{Q}^{(n)} = -\mathbf{R}(\mathbf{Q}^{(n)}) \tag{15}$$

The left-hand-side matrix is the flow Jacobian. The flow vector $\mathbf{Q}$ is updated after each iteration, and the linear system is solved again until $\|\mathbf{R}(\mathbf{Q})\|_2$ is reduced by more than 10 orders of magnitude. At each iteration, (15) is solved using the Krylov method Flexible Generalized Minimal Residual (FGMRES).[10,11] Note that when using a Krylov method, only matrix-vector products with the flow Jacobian are required. These can be approximated by one-sided differencing:

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right) \mathbf{v} \approx \frac{\mathbf{R}(\mathbf{Q} + \epsilon \mathbf{v}) - \mathbf{R}(\mathbf{Q})}{\epsilon} \tag{16}$$

leading to a Jacobian-free approach. The linear system is right-preconditioned using an approximate-Schur preconditioner based on Refs. 9 and 12.

To improve the stability of the Newton method during the start-up phase, the flow solver uses an approximate-Newton method, where a first-order Jacobian $\mathbf{A}_1$ replaces the flow Jacobian in (15). A pseudo-transient time step is also added for globalization:

$$\Delta t_i^{(n)} = \frac{\Delta t_{\text{ref}}^{(n)}}{J_i(1 + \sqrt[3]{J_i})} \tag{17}$$

where the reference time step for iteration $n$ is defined as

$$\Delta t_{\text{ref}}^{(n)} = A(B)^n \tag{18}$$

Values of $A = 0.1$ and $B = 1.5$ are used. In summary, during the approximate-Newton start-up phase, the linear system solved at each iteration $n$ is given by:

$$\left[\mathbf{T}^{(n)} + \mathbf{A}_1^{(n)}\right] \Delta \mathbf{Q}^{(n)} = \mathbf{R}^{(n)} \tag{19}$$

where $\mathbf{T}^{(n)}$ is a diagonal matrix containing the reciprocal of the local time steps. The flow solver switches to the Newton method when the residual has been reduced by one order of magnitude:

$$\frac{\|\mathbf{R}^{(n)}\|_2}{\|\mathbf{R}^{(0)}\|_2} < 0.10 \tag{20}$$

During the Newton phase, the reference time step is based on Mulder and van Leer:[13]

$$\Delta t_{\text{ref}}^{(n)} = \max\left[\alpha \left(\frac{\|\mathbf{R}^{(n)}\|_2}{\|\mathbf{R}^{(0)}\|_2}\right)^{-\beta}, \Delta t_{\text{ref}}^{(n-1)}\right] \tag{21}$$

We find $\alpha = 1.0$ and $\beta = 1.75$ found to be satisfactory values for a wide range of problems. As the residual decreases, the time step approaches infinity, and the full Newton step is recovered.

## B. Gradient Computation

A fast and accurate evaluation of the objective function gradient $\mathcal{G}$ is necessary for an effective gradient-based optimizer. The gradient can be expressed in terms of the design variables $\mathcal{X}$ and adjoint variables $\mathbf{\Psi}$:

$$\mathcal{G} = \frac{\partial \mathcal{J}}{\partial \mathcal{X}} - \mathbf{\Psi}^T \frac{\partial \mathbf{R}}{\partial \mathcal{X}} \tag{22}$$

where $\mathbf{\Psi}$ is the solution to the discrete adjoint equation:

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right)^T \mathbf{\Psi} = \frac{\partial \mathcal{J}}{\partial \mathbf{Q}}^T \tag{23}$$

The adjoint equation is solved using FGMRES, right-preconditioned using the approximate-Schur preconditioner. To obtain an accurate gradient, we specify a linear tolerance of $10^{-8}$. We obtain the Jacobian matrix on the left-hand-side of (23) by hand differentiation of the inviscid fluxes and dissipation terms. The complex step method[14] is used to linearize the SATs at block boundaries. In addition, the right-hand-side $\partial \mathcal{J}/\partial \mathbf{Q}$ is hand differentiated. Finally, partial derivatives with respect to design variables ($\partial \mathcal{J}/\partial \mathcal{X}$, $\partial \mathbf{R}/\partial \mathcal{X}$) are evaluated using second-order centred differencing. Note that evaluating the partial derivatives with respect to design variables does not require additional flow solves.

### C. Geometry Parameterization

We parameterize the geometry of the wing using B-spline control surfaces.[4,15] In this method, the $k$-th order B-spline representation of a surface in 3D space using $M \times N$ control points and basis functions is given by:

$$\vec{a}(s,t) = \sum_{j=1}^{N} \sum_{i=1}^{M} (\mathcal{X}_{\mathrm{B}})_{i,j} \, \mathcal{M}_{i,k}(s) \mathcal{N}_{j,k}(t) \tag{24}$$

where $\vec{a}$ is the position vector along the curve at parametric distances $s$ and $t$ from the origin, $(\mathcal{X}_{\mathrm{B}})_{i,j}$ are the locations of the control points, and $\mathcal{M}_{i,k}(s)$ and $\mathcal{N}_{j,k}(t)$ are the basis functions of order $k$, defined by the Cox-deBoor relationships:[16]

$$\mathcal{M}_{i,1}(t) = \begin{cases} 1 & \text{if } \; d_i \le t < d_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

$$\mathcal{M}_{i,k}(t) = \left[ \frac{t - d_i}{d_{i+k-1} - d_i} \right] \mathcal{M}_{i,k-1}(t) + \left[ \frac{d_{i+k} - t}{d_{i+k} - d_{i+1}} \right] \mathcal{M}_{i+1,k-1}(t) \tag{26}$$

where $d_i$ represents the elements of a uniform open knot vector given by:

$$d_i = \begin{cases} 0 & 1 \le i \le k \\ i - k & k+1 \le i \le M \\ M - k + 1 & M+1 \le i \le M + k \end{cases} \tag{27}$$

$\mathcal{N}$ is similarly defined. At the start of the optimization cycle, the surface grid is first parameterized using B-spline control surfaces. For a structured surface grid with $I$ and $J$ nodes in the parametric directions $s$ and $t$, the B-spline surface can be described in discrete matrix form as:

$$\begin{aligned} \mathbf{G}_{\mathrm{S}} &= \mathcal{U}\mathcal{D} \\ \mathcal{D} &= \mathcal{X}_{\mathrm{B}}\mathcal{V} \end{aligned} \tag{28}$$

where $\mathbf{G}_{\mathrm{S}}$ contains either the $x$, $y$ or $z$ coordinates for each surface grid node $(j,\, k)$, $\mathcal{U}$ and $\mathcal{V}$ store the basis function values at parametric distances $s$ and $t$ from the grid origin, $\mathcal{D}$ is an intermediate matrix of size $M \times J$, and $\mathcal{X}_{\mathrm{B}}$ is a matrix containing the $x$, $y$ or $z$ coordinates of the control points:

$$\begin{aligned} \mathbf{G}_{\mathrm{S}} &= \begin{bmatrix} x_{11} & \cdots & x_{1J} \\ \vdots & & \vdots \\ x_{I1} & \cdots & x_{IJ} \end{bmatrix} \qquad \mathcal{U} = \begin{bmatrix} \mathcal{N}_1(s_1) & \cdots & \mathcal{N}_M(s_1) \\ \vdots & & \vdots \\ \mathcal{N}_1(s_I) & \cdots & \mathcal{N}_M(s_I) \end{bmatrix} \\ \mathcal{X}_{\mathrm{B}} &= \begin{bmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & & \vdots \\ x_{M1} & \cdots & x_{MN} \end{bmatrix} \qquad \mathcal{V} = \begin{bmatrix} \mathcal{M}_1(t_1) & \cdots & \mathcal{M}_1(t_J) \\ \vdots & & \vdots \\ \mathcal{M}_N(t_1) & \cdots & \mathcal{M}_N(t_J) \end{bmatrix} \end{aligned} \tag{29}$$

The distances $s$ and $t$ are calculated based on the nodal indices:

$$\begin{aligned} s_i &= \frac{i-1}{I-1}(m - k + 2) \\ t_j &= \frac{j-1}{J-1}(n - k + 2) \end{aligned} \tag{30}$$

The control point locations are found by first solving for $\mathcal{D}$, and then $\mathcal{X}_\text{B}$ in the least-squares problems in (28). This process is repeated for each of the three coordinates. To generate a new surface grid in response to changes in the location of the control points, the intermediate matrix $\mathcal{D}$ in (28) is first generated based on the new control point locations $\mathcal{X}_\text{B}$, and then the new surface grid $\mathbf{G}_\text{S}$ is generated.

Using this parameterization strategy, we can use two levels of design variables. At the planform level, control points are grouped into a reduced set of planform variables, such as semi-span ($b/2$), chord ($c$), leading-edge and trailing-edge sweep ($\Lambda_\text{LE}$, $\Lambda_\text{TE}$), dihedral ($\Gamma$) and twist ($\Omega$) angles. Other planform parameters such as taper ratio ($\lambda$) and aspect ratio ($\mathit{AR}$) are extracted from the above planform variables. At the wing section level, each individual control point may move vertically to adjust the wing section shape. Increasing the number of control points improves the flexibility of the parameterization.

### D. Grid Movement

We use a fast and robust algebraic grid movement method to generate a new volume grid after each iteration, and to evaluate the partial derivatives $\partial\mathcal{J}/\partial\mathcal{X}$ and $\partial\mathbf{R}/\partial\mathcal{X}$ in (22). Our algorithm modifies the nodal coordinates along a grid line using the algebraic equation:

$$\vec{x}_k^{\text{new}} = \vec{x}_k^{\text{old}} + \frac{\Delta\vec{x}_1}{2}\left[1+\cos\left(\pi S_k\right)\right] \quad \text{for} \;\; k = 2\ldots k_\text{max} \tag{31}$$

where $\Delta\vec{x}_1$ is the displacement of the surface node, $k_\text{max}$ is the number of nodes along the grid line from the surface to the far-field boundary, and

$$S_k = \frac{\sum_{i=2}^{k}\left|\vec{x}_i - \vec{x}_{i-1}\right|}{\sum_{i=2}^{k_\text{max}}\left|\vec{x}_i - \vec{x}_{i-1}\right|} \tag{32}$$

is the normalized arc-length distance along the grid line. Although this grid movement algorithm does not explicitly guarantee the new grid to be of good quality, we have found it to be effective for most aerodynamic shape optimization applications when the geometry change is relatively small, and also when the far-field boundary is sufficiently far from the body surface.

## IV. Results and Discussion

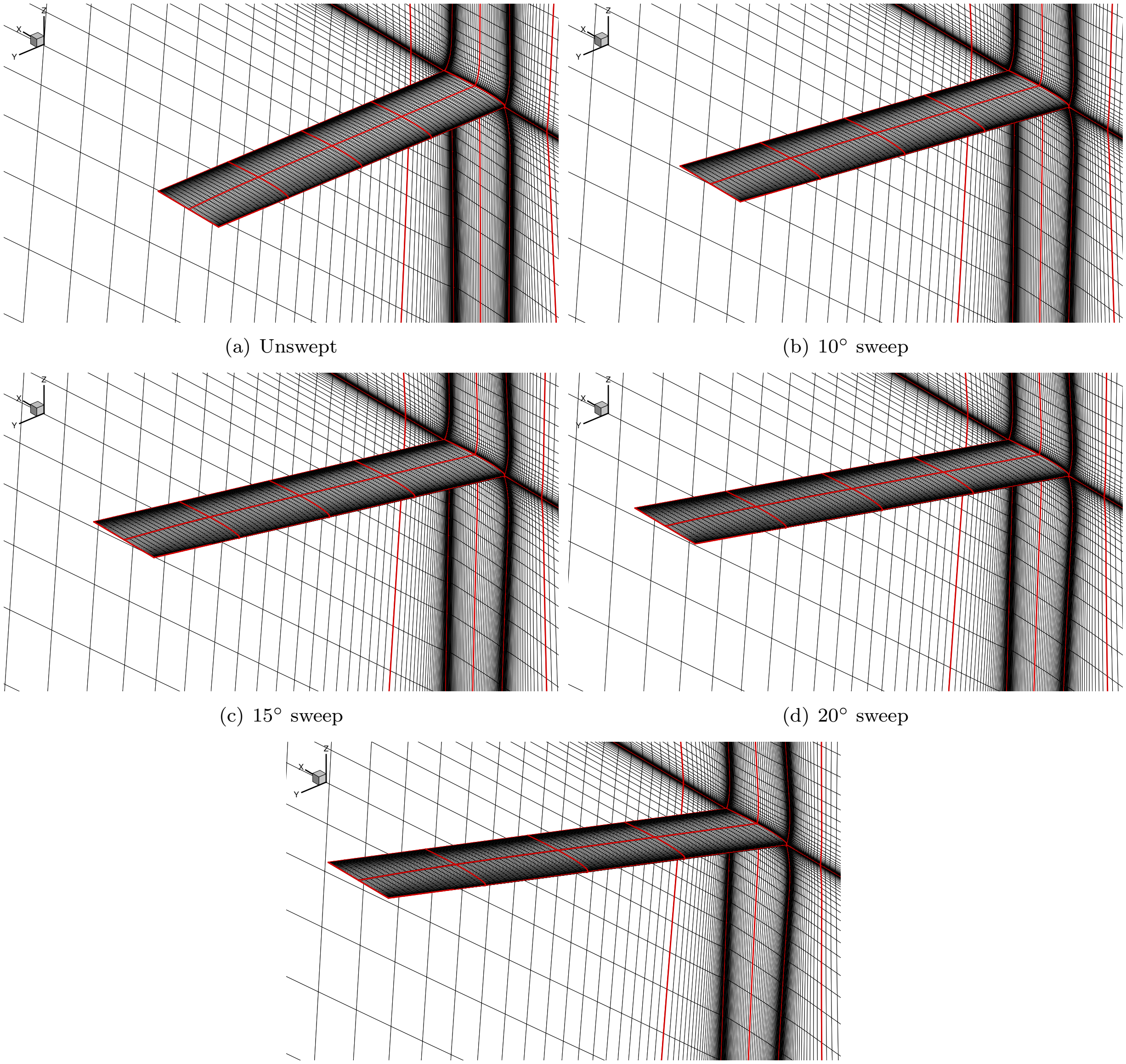
In order to find the maximum Mach number for robust shock-free operation, and to examine the trade-offs between operating Mach number and sweep angle for a transonic wing, we perform both single-point and multi-point optimization of untapered wings with various sweep angles. Our optimization results are obtained on a distributed memory cluster. The cluster uses Intel Xeon 5500 (Nehalem) processors with a CPU speed of 2.53GHz, with 16GB of shared memory per computation node (8 processors). The computational nodes are connected by a non-blocking 4x-DDR Infiniband network. Communication between processors is done using the message passing library MPICH.

### A. Single-Point Wing Optimization

We perform single-point drag minimization on untapered wings at transonic speeds between $M = 0.60$ and $M = 0.86$. Each optimization is performed at a fixed Mach number. Once the optimization has converged, the range factor $\mathcal{R}$ is computed for each optimized wing. The initial geometries are untapered wings with sweeps angles ranging from $0°$ to $25°$. The sweep angles of the wings are fixed during optimization. All the wings have the section geometry of the NACA 0012 airfoil, with an aspect ratio of $\mathit{AR} = 8.0$. The volume grids are 96-block H-H topology grids with 2.77-million nodes. The surface grids are shown in Fig. 1. Off-wall spacing is $2.0 \times 10^{-3}c$, and far-field boundaries are at least $22c$ from any point on the wing surface.

Our goal is to maximize the lift-to-drag ratio of this wing, while maintaining the same lift at each operating Mach number. That is, target $C_\text{L}^*$ is set such that $M^2 C_\text{L}^*$ is constant. The target lift coefficients $C_\text{L}^*$ are shown in Table 1. A constant drag coefficient of $C_{\text{D}_0} = 0.0150$ is added to the computed $C_\text{D}$ as an estimate of viscous effects and fuselage and engine interaction.[2]

The top and bottom surfaces of each wing are parameterized with a cubic B-spline surface with 13 control points in the spanwise direction and 11 in the chordwise direction. There are 200 B-spline control point design variables, which include $z$-coordinates of every control point, except near the leading edge and at the trailing edge, where the control points are fixed. The change in the twist angle ($\Omega$) and the angle



(a) Unswept (b) 10° sweep (c) 15° sweep (d) 20° sweep (e) 25° sweep

**Figure 1. H-H grids over the initial wings**

of attack ($\alpha$) are also design variables, for a total of 202 design variables. Note that the wing's sweep angle, taper ratio, aspect ratio and planform are fixed throughout the optimization. A volume constraint (12) is implemented. In addition, to prevent grid crossover, thickness constraints (13) are also implemented at $x/c = 0.95$ to maintain a 0.05% minimum thickness. In every optimization case, both the volume and thickness constraints are active at the end of the optimization run.

As an example of optimizer convergence, Fig. 2 shows the convergence for the $\Lambda = 10°$ case at $M = 0.74$. In every case presented, the optimizer is able to reduce the gradient $L_2$-norm ($\|\mathcal{G}\|_2$) by more than six orders of magnitude. Furthermore, in the cases where we are able to completely eliminate wave drag, the objective function is reduced to below $10^{-6}$, and the final drag coefficient is within 2% of the value predicted by lifting-line theory for an elliptical lift distribution. About 85% of the the improvement in $L/D$ ratio occurs within the first 20 iterations, consistent with our previous results.[4] Using 96 processors, each flow solve requires about six minutes to complete, while the cost of a gradient evaluation is about 75% that of a flow solve.

The range factors for single-point optimization of all the cases are shown in Fig. 3. Each data point along

| $M$ | $C_L^*$ |
|---|---|
| 0.60 | 0.802 |
| 0.64 | 0.705 |
| 0.68 | 0.625 |
| 0.72 | 0.557 |
| 0.74 | 0.527 |
| 0.76 | 0.500 |
| 0.78 | 0.475 |
| 0.82 | 0.430 |
| 0.84 | 0.409 |

**Table 1. Target lift coefficients for optimization cases**

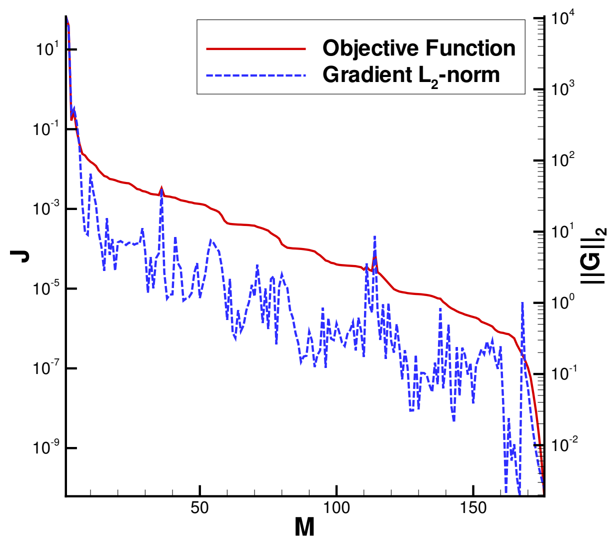



**Figure 2. Convergence history for single-point optimization at** $M = 0.74$

the curves represents the optimal shape from a single-point optimization, at the given sweep angle and Mach number. The range factors shown are at each optimized wing's design operating condition. The maximum Mach numbers for which shock-free flow solutions exist are shown in Table 2 for each sweep angle. It is important to recognize that these results are specific to the particular value of $C_L$ and geometric constraints used.

## B. Multi-Point Wing Optimization

Fig. 3 shows that with a single-point optimization, an unswept wing optimized for $M = 0.78$ has a performance comparable to a wing with higher sweep at this Mach number. However, Fig. 4 shows that at Mach numbers slightly lower or higher than $M = 0.78$, the optimized wing's performance deteriorates significantly, i.e. the design is not robust. In comparison, the wings optimized for the same Mach number with $\Lambda = 15°$ and $\Lambda = 25°$ are robust against changes in operating Mach number. Multi-point optimization is used to prevent this "point optimization" phenomenon in the unswept wing, and to ensure a robust design.

In the multi-point optimization, we consider a range of Mach numbers at the same lift. We consider three-point optimization cases using (8). In these cases, we introduce two additional operating points at Mach numbers that are $\pm 0.015$ from the design operating point. For example, the wing designed to operate at $M = 0.78$ will also include operating conditions at $M = 0.765$ and $M = 0.795$. The weighting on these operating conditions is based on approximating the integral (9) with $P(M) = 1$ using the trapezoidal rule.

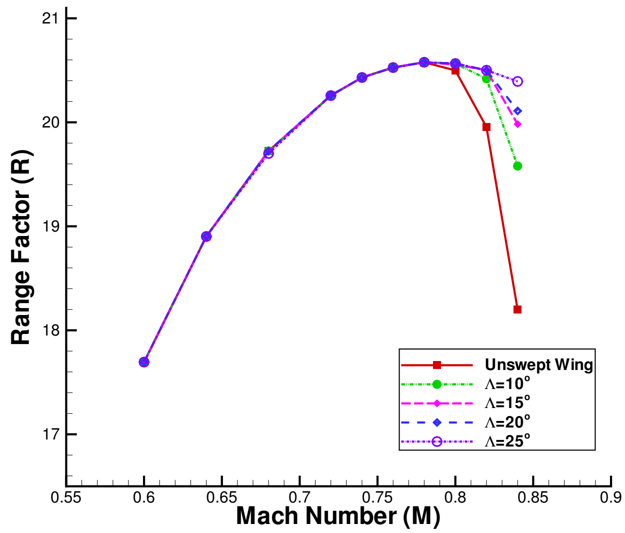



**Figure 3. Range factors for single-point optimized unswept wings**

| Sweep angle ($\Lambda$) | Max $M$ for shock-free flow |
|---|---|
| 0° | 0.78 |
| 10° | 0.80 |
| 15° | 0.82 |
| 20° | 0.82 |
| 25° | 0.84 |

**Table 2. Maximum Mach number for shock-free flow solution for each wing**

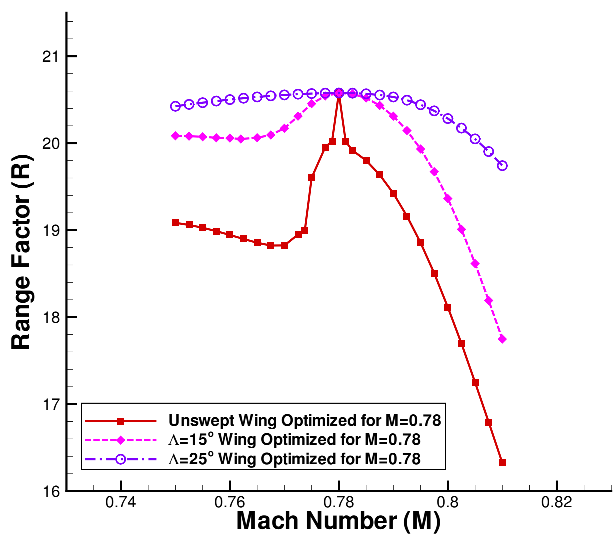



**Figure 4. Range factors for wings optimized for $M = 0.78$**

In these three-point cases, therefore, $\omega_i = 0.5$ is used for the lower and higher operating Mach numbers, and $\omega_i = 1.0$ for the design Mach number. The same geometric design variables and constraints are used, and the angles of attack at the two new operating points are also considered as design variables. The multi-point optimization begins with the optimized geometry from the single-point case. The targets and weights in lift

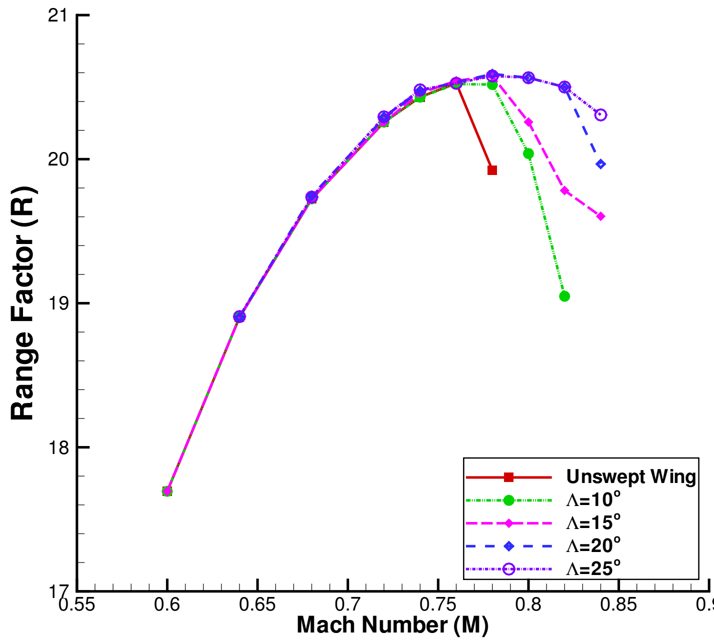



**Figure 5. Range factors for three-point optimized unswept wings**

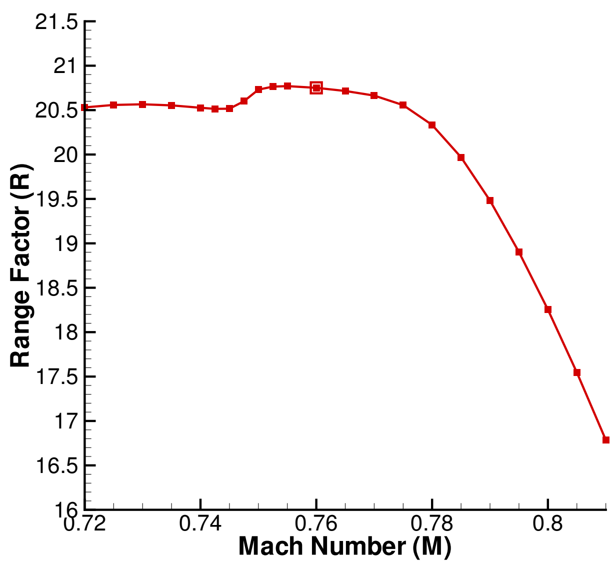



**Figure 6. Range factors for an unswept wing optimized for $M = 0.76$**

and drag ($\omega_{\mathrm{L}}$, $C^*_{\mathrm{L}}$, $\omega_{\mathrm{D}}$, $C^*_{\mathrm{D}}$) are selected in the same way as as the single-point cases, i.e. the lift generated at these Mach numbers is the same as a wing operating at $M = 0.76$ with $C_{\mathrm{L}} = 0.50$.

The resulting range plot is shown in Fig. 5. Each data point represents the range factor $\mathcal{R}$ of the optimized wing at its design operating Mach number. We see that the unswept wing is still competitive against swept wings up to $M = 0.76$, while the wing with $\Lambda = 10°$ remains competitive up to $M = 0.78$, with only a slight decrease in $\mathcal{R}$ compared to wings with higher sweep angles. Efficient operation at $M = 0.82$ or above requires a sweep angle of greater than $10°$. The performance of the wings shown in Fig. 5 is more robust against perturbations in Mach number compared to those shown in Fig 3. An example for the unswept wing found based on a multi-point optimization with a design Mach number of 0.76 is shown in Fig 6. For these robust designs, the maximum Mach numbers for which a shock-free flow exists are shown for each sweep angle in Table 3.

| Sweep angle ($\Lambda$) | Max $M$ for shock-free flow |
|---|---|
| 0° | 0.76 |
| 10° | 0.78 |
| 15° | 0.78 |
| 20° | 0.82 |
| 25° | 0.84 |

**Table 3. Maximum Mach number for shock-free flow and robust design at each sweep angle**

## V. Conclusion

We have applied a Newton-Krylov algorithm for aerodynamic shape optimization to the design of low-sweep wings for maximum range. An improved curve fitting for thrust-specific fuel consumption is used to approximate engine performance and its effect on range. The point optimization phenomenon is observed for unswept wings at higher Mach numbers, and a multi-point optimization is used to obtain a robust design. Our results show that an optimized unswept or low-sweep wing can have a competitive range factor against highly-swept wings at transonic speeds, but at the cost of operating at slightly slower speeds. However, the results presented are specific to the lift coefficient and geometric constraints used. Future work will include further examine the potential of low-sweep wings by performing aero-structural optimization incorporating laminar-turbulent transition prediction.

## References

[1] "Airline Fuel and Labour Cost Share," IATA economic briefing, International Air Transportation Association, February 2010.

[2] Jameson, A., Vassberg, J. C., and Shankaran, S., "Aerodynamic-structural design studies of low-sweep transonic wings," AIAA Paper 2008-145, American Institute of Aeronautics and Astronautics, January 2008.

[3] Nemec, M. and Zingg, D. W., "Newton-Krylov algorithm for aerodynamic design using the Navier-Stokes equations," *AIAA Journal*, Vol. 40, No. 6, June 2002, pp. 1146–1154.

[4] Leung, T. M. and Zingg, D. W., "Single- and multi-point aerodynamic shape optimization using a parallel Newton-Krylov approach," AIAA Paper 2009-3803, American Institute of Aeronautics and Astronautics, June 2009.

[5] Mattingly, J. D., *Elements of Propulsion: Gas Turbines and Rockets*, American Institute of Aeronautics and Astronautics, Reston, VA, 2006.

[6] Buckley, H. and Zingg, D. W., "On aerodynamic design through multipoint numerical optimization," AIAA Paper 2011–1244, American Institute of Aeronautics and Astronautics, 2011.

[7] Jameson, A., Schmidt, W., and Turkel, E., "Numerical solution of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes," *14th Fluid and Plasma Dynamics Conference*, Palo Alto, CA, 1981, AIAA Paper 81–1259.

[8] Pulliam, T. H., "Efficient solution methods for the Navier-Stokes equations," Tech. rep., Lecture Notes for the von Kármán Inst. for Fluid Dynamics Lecture Series: Numerical Techniques for Viscous Flow Computation in Turbomachinery Bladings, Brussels, Belgium, Jan. 1986.

[9] Hicken, J. E. and Zingg, D. W., "Parallel Newton-Krylov solver for the Euler equations discretized using simultaneous-approximation terms," *AIAA Journal*, Vol. 46, No. 11, 2008, pp. 2773–2786.

[10] Saad, Y. and Schultz, M. H., "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear problems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, 1986, pp. 856–869.

[11] Saad, Y., "A Flexible inner-outer preconditioned GMRES algorithm," *SIAM Journal on Scientific Computing*, Vol. 14, No. 2, 1993, pp. 461–469.

[12] Saad, Y. and Sosonkina, M., "Distributed Schur complement preconditioning," *SIAM Journal on Scientific Computing*, Vol. 21, No. 4, 1999, pp. 1337–1356.

[13] Mulder, W. A. and van Leer, B., "Experiments with implicit upwind methods for the Euler equations," *Journal of Computational Physics*, Vol. 59, 1985, pp. 232–246.

[14] Squire, W. and Trapp, G., "Using complex variables to estimate derivative of real functions," *SIAM Review*, Vol. 40, No. 1, March 1998.

[15] Fudge, D., Zingg, D. W., and Haimes, R., "A CAD-free and a CAD-based geometry control system for aerodynamic shape optimization," AIAA Paper 2005-0451, American Institute of Aeronautics and Astronautics, January 2005.

[16] Rogers, D. F. and Adams, J. A., *Mathematical Elements for Computer Graphics*, McGraw-Hill, 2nd ed., 1990.