Influence of Shape Parameterization on Aerodynamic Shape Optimization

John C. Vassberg *
JetZero, Inc.
Corporate Headquarters
Long Beach, CA 90808, USA

Antony Jameson †
Department of Aerospace Engineering
Texas A&M University
College Station, TX 77843, USA

Von Karman Institute
Brussels, Belgium
17 May 2022

Nomenclature

AR	Wing Aspect Ratio $= \frac{b^2}{S_{ref}}$
b	Wing Span
$\mathcal{B}$	Shape Function Basis
CFD	Computational Fluid Dynamics
C_D	Drag Coefficient $= \frac{Drag}{q_\infty S_{ref}}$
C_L	Lift Coefficient $= \frac{Lift}{q_\infty S_{ref}}$
C_M	Pitching-Moment Coefficient $= \frac{Moment}{q_\infty S_{ref} C_{ref}}$
$Cmax$	Maximum Camber of an Airfoil
C_{ref}	Wing Reference Chord
$count$	Drag Coefficient Unit = 0.0001
DTE	Divergent Trailing Edge
FRP	Fuselage Reference Plane
I	Objective or Cost Function
K	Order of Bezier or B-Spline Curve
LE	Leading-Edge Point per MLL
MAC	Mean Aerodynamic Chord
MLL	Maximum Length Line = Chordline
N, NDV	Number of Design Variables
$RANS$	Reynolds-Averaged Navier-Stokes
Re	Wing Reynolds number based on C_{ref}
S_{ref}	Wing Reference Area
TE	Trailing-Edge Point $= \frac{1}{2}(TE_u + TE_l)$
TE_{Base}	Trailing-Edge Base Height $= yte_u - yte_l$
$Tmax$	Maximum Thickness of an Airfoil
WRP	Wing Reference Plane
x, y, z	Spatial Coordinates
$xcpt$, $ycpt$	X-Y Coordinates of a Control Point
q	Dynamic Pressure $= \frac{1}{2}\rho V^2$
λ	Wing Taper Ratio $= \frac{C_{tip}}{C_{root}}$
$\Lambda_{c/4}$	Wing Quarter-Chord Sweep
π	3.141592654...
∞	Infinity
$\mathcal{O}(*)$	Order of

1 Introduction

This is the third of three lectures prepared by the authors for the von Karman Institute that deal with the subject of aerodynamic shape optimization. In this lecture we briefly discuss several items related to the parameterization of an aerodynamic design space. These items include the advantages and disadvantages of using: 1) absolute vs. perturbed geometry definition, 2) global vs. local shape control, and 3) large vs. small dimensional design spaces. We also review desirable design characteristics that one should consider when formulating the parameterization of a design space. One will find that this formulation is strongly influenced by the basic approach of optimization as well as the cost of function evaluations. Stated differently, in general there is no *one* best approach to design-space parameterization. In fact, the first author employs a very diverse set of optimization methods on a regular basis to address a very diverse set of routine design

*Chief Design Officer
†TEES Eminent Professor

challenges. Aspects of these diverse applications will be touched on in the early sections of this lecture. However, since both authors share a common expertise on detailed aerodynamic shape optimization, this lecture will concentrate primarily on this task. Nonetheless, the ensuing discussions should provide useful insight to help one develop a well-formulated design-space parameterization for a wide variety of problems.

We have stated this before, and we will repeat it again, as it is an important point. In an airplane design environment, there is no need for an optimization based purely on the aerodynamics of the aircraft. The driving force behind (almost) every design change is related to how the modification improves the vehicle, not how it enhances any one of the many disciplines that comprise the design. And although we focus our lectures on the aerodynamics of an airplane, we also include the means by which other disciplines are linked into and affect the aerodynamic shape optimization subtask; some of these will be addressed again in this lecture. Another characteristic of the problems we typically (but not always) work, is that the baseline configuration is itself within 1-2% of what may be possible, given the set of constraints that we are asked to satisfy. This is certainly true for commercial transport jet aircraft whose designs have been constantly evolving for the past half century or more. On the other hand, the first author also develops advanced concepts which are mostly in the embryonic phase of design. These efforts have a completely different set of requirements and expectations that are imposed on the optimization methods employed. Specifically, improvements to an existing commercial aircraft may require high-fidelity flow solutions (RANS) and high-dimensional design spaces, whereas an advanced concept may substantially benefit with low-fidelity analyses and low-dimensional design spaces, but may require that a global optimum is to be located.

Quite often the problem of design is very constrained; this is the case when the shape change is required to be a retrofitable modification that can be applied to aircraft already in service. Occasionally, we can begin with a clean slate, such as in the design of an all-new airplane. And the problems cover the full spectrum of studies in between these two extremes. Let's note a couple of items about this setting. First, in order to realize a true improvement to the baseline configuration, a high-fidelity and very accurate computational fluid dynamics (CFD) method *must* be employed to provide the aerodynamic metrics of lift, drag, pitching moment, spanload, etc. Even with this, measures should be taken to estimate the possible error band of the final analyses; this discussion is beyond the scope of these lectures. The second item to consider is related to the definition of the design space. A common practice is to use a set of basis functions which either describe the absolute shape of the geometry, or define a perturbation relative to the baseline configuration. In order to realize an improvement to the baseline shape, the design space should not be artificially constrained by the choice of the set of basis functions. This can be accomplished with either a relatively small set of very-well-chosen basis functions, or with a large set of reasonably-chosen basis functions. The former approach places the burden on the user to establish an adequate design space; the latter approach places the burden on the optimization software to economically accommodate problems with large degrees of freedom. Over the past two decades, the authors have focused on solving the problem of aerodynamic shape optimization utilizing a design space of very large dimension. The interested reader can find copious examples of the alternative approaches throughout the literature.

Over the past four decades CFD has matured to the level that very accurate aerodynamic performance analyses are now possible for complete aircraft configurations, provided that the flow of the viscous shear layers remain predominately attached to the geometry surfaces. Fortunately, this is usually the case for well-designed aircraft at their intended cruise flight conditions.

Concurrent with the advancement of CFD, aerodynamic shape optimization, and multi-disciplinary optimization have also matured to the stage that they have been successfully incorporated into the aircraft design environment, and now perform crucial roles. However, the costs associated with these optimizations can be quite large, and even prohibitive, for many problems of practical interest. These costs include both computational resources as well as engineering labor hours needed to set up the problems for optimization. Although advancements in computer hardware continue to track Moore's Law, so do the size of our CFD models. As a consequence, research directed towards improving the efficiency of optimization continues.

This paper is organized in the following manner. The first few sections are fairly generic to the formulation of a well-constructed design space. The remaining sections address the influence of parameterization of the design space with a focus on aerodynamic shape optimization. Section 2 describes the optimization techniques employed. Section 3 describes two additional CFD methods applied in this work; one is used by two of the optimization processes, while the other is used for independent cross-analysis evaluations. Section 4 provides a *deep dive* into the anatomy of airfoils and wing geometries, as well as surveys common requirements imposed on an aerodynamic design space. Section 5 describes the set of design spaces utilized herein. Sections 6-8 provide discussions on three example model problems, two of which are Test Cases of

the Aerodynamic Design Optimization (ADO) Discussion Group workshop. Hence, a large amount of data will soon become available on these model problems from a diverse set of research activities. In each of these examples, a statement of optimization, a description of the geometry, and results are provided. Tables of data are embedded within the text, while all figures are appended to the end of the lecture.

2 Aerodynamic Optimization Methods

Several different approaches have been applied to perform various aerodynamic shape optimizations for the sample model problems under discussion. The methods specifically used by the authors include MDOPT, CMA-ES, SYN83 and SYN107. Additional optimization methods are discussed by drawing on the work of others with their permission and with references provided. Descriptions of methods employed by the authors are included below.

MDOPT [1] is a Boeing multidisciplinary design optimization framework for very general air vehicle design and analysis. The system contains a collection of technology modules for performing optimization studies by means of a Graphical User Interface (GUI), and combining robust numerical optimization schemes with higher-order computational analysis. A variety of multidisciplinary objective and constraint functions are available, including aerodynamic, weight, mission performance, and stability and control characteristics. MDOPT's GUI environment helps manage the tasks of: 1) design-space set-up, 2) establishing a design of experiments, 3) fitting the response surface, 4) navigating the response surface to the optimum state, 5) perturbing field grid points, and 6) enforcing a variety of nonlinear constraints. MDOPT can be exercised in two modes, the first being based on response surfaces, and the second being a direct-driven quasi-Newton method. In this lecture, only results from the response-surface approach are discussed. The CFD method utilized herein for MDOPT is OVERFLOW.

CMA-ES [2] is a Covariance Matrix Adaptation Evolution Strategy. The basic idea of the approach is that the design vector is initialized at an arbitrary location in the design space. For the first iteration, a random, isotropic sampling of the local terrain around the mean is performed. Based on the evaluated objective function at these random locations, the mean design vector shifts towards the weighted centroid of the best results from the random sampling. The method also accounts for combinations of individual design-variable displacements by utilizing a covariance matrix to rotate the search from the principle axes of the design vector and introduce anisotropy. The covariance matrix and standard deviation are updated after each iteration and control the random search process. The CFD method utilized herein for CMA-ES is OVERFLOW.

SYN83 [3] utilizes a continuous adjoint to the Euler equations to compute the gradient of the objective function with respect to the design space. Here, a free-surface design space is automatically generated by SYN83. This full parameterization corresponds to the highest dimensional space supported by the discrete points of the grid defining the geometry. Hence, the performance of the resulting optimum airfoil could provide a limit to what is achievable, unless the optimization locates a local optimum that is quite degraded from the globally best design. SYN83 solves both the Euler equations and its adjoint on an internally-generated C-mesh. A typical mesh of (768x128) cells is shown in Figure 1; it provides 513 points on the free-surface to define the airfoil design space. The optimization process begins by solving converged solutions on both the Euler equations and its adjoint for the baseline airfoil shape. Then its design space is navigated in the reverse direction of the gradient projected into an allowable Sobolev space. This continues until the magnitude of the constrained gradient vanishes. Upon completion of the run, a locally-optimum airfoil is found and a converged solution of the Euler equations on this shape is known. In practice, the complete SYN83 optimization process costs only about one-order-of-magnitude more than the cost of a single analysis. However, for the degenerate problem of Section 6, we ran 1,000 design cycles with small steps to converge to the local optimum shapes. In this mode, the cost of optimization was about two-orders-of-magnitude greater than that of a single analysis. Nonetheless, the expense of these optimizations is acceptable as the basic analysis is an inexpensive 2D Euler flow solution.

SYN107 [4] is a RANS-based aerodynamic shape optimization method which utilizes a continuous adjoint to the RANS equations to efficiently compute the gradient of the objective function with respect to the design space. It fully integrates grid generation of a C-H-mesh, the FLO107 CFD code, the ADJ107 adjoint solver, an automatically-generated free-surface design space, mesh perturbation, and gradient-based optimization. SYN107 is capable of handling wing-body and wing-alone configurations. In addition to free-surface designs, SYN107 has been recently enhanced to include a geometry engine based on 3rd-order B-Splines and IGES

output, which provide a bridge to CAD without loss of information. In a similar manner to SYN83, SYN107 uses a smoothed-steepest-descent (Sobolev) approach to maneuver through the allowable design space. A local optimum is found when the constrained gradient vanishes. In practice, a complete SYN107 optimization process is only about one-order-of-magnitude greater in cost than the cost of a single analysis.

The next section provides brief descriptions of two additional CFD methods utilized herein.

3 CFD Analysis Methods

In the present study, two well-validated CFD methods are utilized, namely OVERFLOW and FLO82. The OVERFLOW code is used within the MDOPT and CMA-ES optimization environments as the function call. FLO82 is used to perform extremely-accurate and independent cross-analysis assessments of the NACA0012 designs of Section 6.

We cannot emphasize the importance of conducting independent cross-analyses of any/all designs defined by an optimization process. Optimizers prey on the weaknesses of the CFD method and will exploit these deficiencies to find improvements that are not real. No matter how well one thinks his CFD method has been validated, validation comparisons can benefit from compensating errors, while an optimizer *will* align errors to realize an artificial gain. Hence, perform the cross analyses! Especially if your design is to be used, such as for a wind-tunnel or flight test.

OVERFLOW [5] is a general-purpose CFD method developed by NASA in the early 1990s. OVERFLOW is capable of solving either the three-dimensional Euler or RANS equations using multiple overset structured grids. It can be applied to very complex geometries. For the NACA0012 model problem, OVERFLOW is used to solve the two-dimensional inviscid compressible flow about symmetric non-lifting airfoil sections. Here, only the upper-half plane of the grid is used, and a symmetry boundary condition is applied along the x-axis forward and aft of the airfoil.

FLO82 [6] is a cell-centered Euler method based on an O-mesh. Upwinding is provided by the H-CUSP dissipation scheme of Jameson [7]. FLO82 also has a provision to enforce symmetric flow solutions if an input flag regarding geometric symmetry is enabled and if the angle-of-attack is identically zero. We make use of this feature for the solutions of Section 6. In particular, the analysis process used herein is based on that of Vassberg [8, 9]. Here, an extremely dense and high-quality conformal O-mesh is constructed about the airfoil. The cells of this mesh are unity in aspect ratio. Figure 2 provides a typical O-mesh with cell dimensions of (256x256). Although not shown, the farfield boundary resides about 150 chord-lengths away from the airfoil. A sequence of grid levels is used to establish grid-convergence data, which is then post-processed with Richardson extrapolation to estimate continuum results. The finest grid in this sequence is dimensioned (2,048x2,048) cells. Note that this finest mesh is equivalent to inserting an additional chess-board of (8x8) cells inside each cell of Figure 2. Figures of FLO82 results are presented on this finest mesh. Tables of FLO82 data include drag levels for a grid sequence of $ni = nj = [256\ ,\ 512\ ,\ 1024\ ,\ 2048\ ,\ Continuum]$. Figure 3 provides typical convergence histories for FLO82 for each of these four discrete grids. These plots include convergence of residuals (R), lift (L), drag (D), and number of supersonic points (S). In the case of lift and drag, "Error" is defined as $|C_l - C_{lLAST}|$ and $|C_d - C_{dLAST}|$, respectively. Note that the residuals reach machine-level zero for all grid levels. Further, drag is converged to within 0.01 *counts* of the final value when the drag curve falls below $Log(Error) \leq -6$. Although these convergence histories are representative for normal cases, some of the designed airfoils of Section 6 experienced convergence stall of residuals on the 4-million cell mesh. For the NACA0012 cross-analyses, the FLO82-based process provides a very accurate, independent assessment of the aerodynamic performance of the various airfoils under discussion.

The next section describes in some detail the anatomy of an airfoil and how a stack of airfoils is assembled to define a wing geometry, and provides some practical and aerodynamic considerations for the parameterization of an aerodynamic design space.

4 Anatomy of Airfoil Sections

In order to be effective in architecting the parameterization of a design space, one must fully understand the geometric characteristics which are important to capture. However, this is problem dependent, so one must find his own way for his particular situation. To help illustrate the process, we provide an in-depth review

of the anatomy of an airfoil and how a stack of airfoils is assembled to define a wing geometry. We also address some of the practical issues associated with common practices for geometric definitions of airfoils and wings. Finally, we consider some aerodynamic requirements for good design that we want to build directly into our design space and its parameterization. This awareness of the requirements will ultimately aid us in developing parameterizations of aerodynamic design spaces that will be well suited for our needs.

Airfoil Stack Background

Airfoil sections are the most important building block of aerodynamic geometry. Airfoils are used to define wings, pylons, nacelles, struts, winglets, feathers, horizontal stabilizers, verticals, propellors, turbomachinery blades and stators, cowls, blimps, sailboat sails, keels and ballast-bulbs, cascades, helicopter rotors, fins, chines, strakes, vertical/horizontal-axis wind turbines, flaps, frisbees, and boomerangs. Such broad use of airfoils to define aerodynamic geometry is accompanied with a diverse set of requirements. To start with, an airfoil section is planar; more specifically, it is not a generalized three-dimensional space curve such as that of a wing-body intersection line! It is comprised of a chordline, upper- and lower-surface contours, a leading-edge point and a trailing-edge point. The average of the upper and lower surfaces define the airfoil's camber line. The absolute difference of the upper and lower surfaces define the airfoil's thickness distribution. Airfoils are often characterized by their maximum thickness and maximum camber values. Max-thickness and cross-sectional area affect structural weight and fuel volume. Camber levels influence high-speed performance metrics such as $\frac{ML}{D}$. Leading-edge radius and max-thickness affect low-speed performance metrics such as C_{Lmax}. Surface curvatures might be limited by manufacturing processes. And the list of considerations go on and on.

A wing geometry is defined in the wing reference plane (WRP). In the WRP, the projection of the wing leading-edge line, wing trailing-edge line, and theoretical tip define the wing planform. The wing planform extends to the symmetry plane. It is customary to place the symmetry-plane-leading-edge-point at the origin. Wing planforms are typically characterized by leading-edge sweep, quarter-chord sweep, planform breaks, wing span, wing area, taper ratio, mean-aerodynamic chord (MAC), etc. Some of these quantities are reference values that are defined differently from company to company. For example, the wing area and MAC might be based on the gross planform, or the outboard trapezodial planform extended to the symmetry plane, and other conventions exist. In any case, the detailed geometry is defined by affixing a stack of airfoils to the planform in the WRP. The stack of airfoils are provided at a minimal set of distinct defining stations. It is customary for the first defining station to be at (or near) the symmetry plane, and the last defining station to be at the wing's theoretical tip. These defining stations are constant spanwise cuts in the WRP; the corresponding airfoil defining planes are perpendicular to the WRP. Each nondimensionalized 2D airfoil is rotated by an incidence, translated to the defining station leading edge, then scaled to match the projected planform chord. The airfoil stack is typically sheared vertically (normal to the WRP) to conform to some desired trait, such as to accommodate a straight hinge-line for a control surface, or to minimize spanwise surface wavyness, or some combination thereof. Wing bending can be approximated with a simple shearing of the airfoil stack in the WRP. Surfacing the wing in the spanwise direction of the airfoil stack is handled in a number of manners. A common practice is to linearly loft the surface between each pair of rigged airfoils of neighboring stations. Alternatively, one can perform nonlinear lofts across all rigged airfoils of defining stations between planform breaks. If no planform breaks are present, then a nonlinear loft from root-to-tip is possible. Once the wing surface is fully defined in the WRP, it is then transformed to the fuselage reference plane (FRP) with dihedral rotation and rigging translations. Once rigged into the FRP, the wing-body intersection line is established, and the unexposed wing is trimmed away.

Now that a basic overview of airfoil and wing geometry definitions has been provided, some additional practical issues are now considered.

Practical Considerations

So where do we start? Will a pristine analytic geometry be handed to us? Probably not. Rather, existing geometry comes in all forms and fashions. For example, in the classic book by Abbot and von Doenhoff [10], a number of NACA airfoils are defined by analytic equations, and other analytic geometry definitions can be found in CAD parts, and IGES or STEP files. Yet not all geometry definitions are analytic, and not all analytic definitions are clean. Various AGARD Reports and websites provide airfoil geometries in discrete forms via a table of point coordinates for upper and lower surfaces.

For example, consider the RAE2822 airfoil as defined in table form by the UIUC Airfoil Database [11]. Figure 4 illustrates the discrete set of 65 upper and 65 lower airfoil coordinates as provided by this website. Implied in their table is that the airfoil leading-edge (LE) point resides at the origin, and its trailing-edge (TE) point coincides with $(x, y)_{te} = (1, 0)$. Not all discrete data comes in such clean form. It is common for collaborators to share wing geometry definitions via surface grids which were generated for CFD analysis without explicitly maintaining a grid line along the leading edge. This can become problematic if your design-space parameterization assumes that the leading-edge and trailing-edge points are accurately defined. Conversely, if your parameterization does not make use of any knowledge of where the leading-edge point is, then other issues can arise.

So the first basic element of airfoil anatomy that we will address is how we define the chordline which connects the leading- and trailing-edge points. Firstly, we *do not* assume that the trailing-edge geometry closes to a point, but rather, and in general, that it can terminate with a finite blunt base. Note that this definition accommodates a sharp trailing edge if the base height is zero. It also allows for the possibility of buildable divergent trailing-edge (DTE) shapes. Hence, to allow for a blunt base, we define the trailing-edge point as the average of the upper and lower airfoil surface termination points. Secondly, we define the true leading-edge point as the point on the continuous airfoil contour that maximizes the chordline. However, as noted above, our discrete set of airfoil coordinates may not contain the true leading-edge point, and therefore, will not capture the true max-length-line (MLL) chord. Since it is important to accurately identify the true chordline, we outline a process that reconstructs it from a set of discrete points. This procedure is second-order accurate and seems to be quite adequate in practice.

Figure 5 illustrates a zoomed-in view of the leading-edge region with 9 discrete airfoil-defining points depicted by asterix symbols. We first scan the set of discrete points to locate the discrete point which is farthest from the trailing-edge point. In the figure, this is labeled as Discrete LE; this also defines the Discrete MLL. We then fit a circle through the Discrete LE point and its upper- and lower-surface neighboring points. The farthest point from the trailing-egde point to the resulting 3-point-fit circle is now assumed to be the true leading-edge point for the airfoil. This True LE point is located by constructing a line which connects the trailing-edge point with the center of the fit circle, and then extending this line forward by the radius of the circle. This extended line is now used as the True MLL chord for the discrete set of points. It is important to note that by this definition, where the concept of a MLL is used to define the LE point, an airfoil contour cannot extend beyond its containment circle. This containment circle is centered at the TE and its radius is the airfoil chord-length. Hence, the combined contour of an airfoil with an ice-horn shape cannot be defined in this design space.

Once the true chordline has been identified, it is a simple matter to transform the planar airfoil shape into nondimensional form by translating the LE point to the origin, then rotating it to align the chordline with the x-axis, and then scaling it by the inverse of the chord-length. This puts the TE point at $(x, y)_{te} = (1, 0)$. It is convenient to put the airfoils of a stack into their nondimensionalized system as certain geometric constraints are fairly constant in nondimensional form. For instance, the optimum trade between wing sweep, nondimensionalized thickness/camber and lift coefficient for a cruise Mach number is fairly constant across large variations of aircraft sizes and types. Other constraints can be considered such as the 1-in-10 rule near the trailing edge to preserve structural integrity. Once the set of nondimensional geometric constraints have been enforced on the airfoil geometry, the airfoil stack is transformed back to its rigged position in the WRP system. Here, remaining constraints such as enforcing a straight hinge-line can be adhered to with vertical shearing of the airfoil stack. Finally, the wing is resurfaced and rigged into the FRP system.

Refer to Figure 6 and Eqn (1) as we continue our discussion on the anatomy of airfoils. Specifically, these data are associated with our best-fit B-Splines for the RAE2822 airfoil coordinates as provided by the UIUC website. In the figure, the bold lines depicit the airfoil geometry, the large-radius circular arc at the leading edge is the containment circle, the very-small-radius circle captured inside the airfoil contour at the leading edge is the osculating circle; this conveys leading-edge radius, R_{LE}. The curve above the airfoil upper surface is the thickness distribution. Displayed on this curve is a tic mark which shows location and value of $Tmax$. Just below this tic mark is a vertical line that connects the airfoil upper and lower surfaces, also providing information regarding $Tmax$. The straight horizontal line connecting leading- and trailing-edge points is the chordline. Close to and above the chordline is the camber line; in between these lines is a small vertical line which gives location and value of $Cmax$. Finally, the inflection point of the airfoil lower-surface contour is indicated by a tic mark. Values of these properties of the RAE2822 are provided Eqn (1).

$$
\begin{aligned}
R_{LE} &= 0.008554, \\
(X,T)_{Tmax} &= (0.379526,\ 0.121108), \\
(X,C)_{Cmax} &= (0.757536,\ 0.012641), \\
(X,Y,\theta)_{Inflect} &= (0.65848,\ -0.02927,\ 8.29056^\circ).
\end{aligned} \tag{1}
$$

Recall that we start the design process with some form of geometry definition being provided. No matter the initial state of this definition, once we extend the effort to put the geometry into a nice, clean form, we will want to keep it in a clean and transferable form for the duration of our design work, as well as to share the geometry with others, and for archival purposes. To emphasize, all of this should occur without incurring any loss-of-translation during any of the transfers of geometry. In order to comply with this requirement, for all practical purposes, this essentially requires that the absolute geometry be analytically defined. We note that an accurate representation of an absolute geometry can require a parameterization of relatively-high dimension. This can pose a real dilemma if the cost of optimization adversely scales with design-space dimension; this is the case with many popular optimization methods. Therefore, it may be necessary to utilize a coarse parameterization to perturb the design variables of the dense parameterization, which is needed for the accurate absolute geometry representation. Fortunately, many techniques have been developed to address this issue, and they are well documented in the literature.

With some of the practical considerations now understood, we now turn our attention to understanding some basic aerodynamic requirements.

Aerodynamic Considerations

A good aerodynamic design is usually characterized by smoothly-varying pressure distributions throughout the flowfield domain. For transonic or supersonic flows, it may be impossible to remove all shocks from the field. Nonetheless, beyond the local proximity of shocks, it is still advantageous to achieve flowfields with low pressure gradients wherever possible. In subsonic regions of the flowfield domain, the local surface pressure is strongly dictated by the local streamwise curvature of the surface. In supersonic regions, the local surface pressure is driven primarily by the local streamwise slope of the surface. In both flow types, spanwise slope or curvature has little effect on surface pressures. Hence, to accommodate both types of flows, it is desirable (or even essential) that curvature continuity is explicitly built into the parameterization of an airfoil shape. In addition, it is desirable that the property of local control be built into the parameterization of the design space. These two properties can be achieved with cubic B-Splines, or a string of cubic Bezier curves which preserve curvature across end-points. Use of higher-than-third-order curves is not necessary for aerodynamic reasons, and the extent of control grows with curve order, therefore, cubics provide an optimal capability for aerodynamic geometry representation.

Now that we understand the basics of airfoil and wing geometry, and have considered some practical issues, as well as thought through a few basic aerodynamic requirements, it is time to get into the details of parameterizing the aerodynamic design space.

5 Design Space

This section provides a brief overview of three types of design spaces utilized in the sample cases of this lecture. In order to show why the property of local control is desirable for the parameterization of a design space, we provide a counter example where a parameterization with global control is utilized instead. In the Bezier Family described below, a single Bezier curve defines the airfoil. Degree Elevation is used to increase dimensionality of the design-space parameterization. This ensures that any airfoil geometry in an M-space also exists in the N-space, where $3 \leq M \leq N$. Specifically, this design-space parameterization was developed to challenge the robustness of MDOPT's Response-Surface-based optimization capabilities. The second design space is a full parameterization of the geometry by means of a free surface. The third design-space parameterization discussed herein is based on cubic B-Splines. The last two parameterizations have the property of local control.

Bezier Design-Space Family

Bezier curves are utilized specifically for the NACA0012-ADO model problem of Section 6. Following the approach of Vassberg [12], a baseline Bezier curve for the NACA0012-ADO airfoil is established. Here, the thickness distribution of the NACA0012-ADO airfoil, given by Eqn (12) over the interval $0 \leq x \leq 1$, is optimally approximated by a 4th-order Bezier curve as follows.

Consider a 2D Bezier curve parameterized by $0 \leq u \leq 1$, where $u = 0$ represents the airfoil leading edge (LE), and $u = 1$ its trailing edge (TE). Also constrain the slope of this curve to be vertical at the LE, hence $\frac{dx}{du} = 0$ and $\frac{dy}{du} \neq 0$ at $u = 0$. A 4th-order Bezier curve conforming to these conditions has control-point coordinates:

$$\begin{aligned} xcpt_0 = 0, \quad xcpt_1 = 0, \quad xcpt_4 = 1, \\ ycpt_0 = 0, \quad ycpt_1 \neq 0, \quad ycpt_4 = 0 \end{aligned} \tag{2}$$

This leaves 5 free variables in control-point coordinates that can be manipulated to define a best-fit curve. Now let I be a cost function that provides a measure of the geometric difference between the Bezier curve and the NACA0012-ADO airfoil, defined as:

$$I = \int_0^1 [y_F(u) - y_N(x(u))]^2 \, du. \tag{3}$$

Here, y_F and y_N are the best-fit Bezier curve and NACA0012-ADO equations, respectively. A minimization of this cost function yields a best-fit 4th-order Bezier curve, which we have designated Bez4-0012-ADO. The control points of Bez4-0012-ADO are provided in Table I. Figure 7 illustrates the Bez4-0012-ADO airfoil shape, and its corresponding control points and hull. Note that the y-coordinate is amplified for clarity.

Table I:
Bez4-0012-ADO Control Points.

n	$xcpt_n$	$ycpt_n$-Fit
0	0.0000000	0.0000000
1	0.0000000	0.0256211
2	0.0308069	0.0438166
3	0.1795085	0.1135797
4	1.0000000	0.0000000

The control points of Table I minimize the cost function of Eqn (3) as:

$$I_{min} \doteq 0.9497 * 10^{-8}. \tag{4}$$

A 4th-order Bezier curve defined by the control points of Table I provides a close approximation of the NACA0012-ADO airfoil. Figure 8 provides the geometric difference between this Bezier curve and the NACA0012-ADO airfoils. Since only the 3 interior control points are allowed to vary, this represents a design space of 3 dimensions. However, in order for the design space of the present work to exactly include the NACA0012-ADO airfoil, a perturbation technique is applied instead of an absolute geometry representation. Here, the x coordinate of the control points of Table I are retained, while the y coordinates define a perturbation Bezier curve which is added to Eqn (12) to define an airfoil shape. Hence, airfoil geometries within the Bezier design space are defined as:

$$y_D(u) = y_P(u) + y_N(x(u)), \tag{5}$$

where y_D defines the design shape, y_P is the perturbation Bezier curve, and y_N represents the baseline NACA0012-ADO geometry. Note that the baseline airfoil is recovered when the design vector is zeroed.

In order to enforce a thickness distribution constraint, where the design airfoil is at least as thick as the baseline section, the perturbation Bezier curve, y_P, must be non-negative over the interval $0 \leq u \leq 1$. Hence,

$$y_P(u) \geq 0 \;\; ; \;\; 0 \leq u \leq 1. \tag{6}$$

A fall out of Eqn (6) requires that the first and last design variables are strictly non-negative.

$$ycpt_1 \geq 0 \quad , \quad ycpt_N \geq 0. \tag{7}$$

Here, $ycpt$ is the y value of the Bezier control point, and N is the dimension of the design space, where the order of the Bezier curve is $K = N + 1$. For instance, a 4th-order Bezier curve has $N = 3$ and $K = 4$.

Note that these control points are defined with respect to the absolute geometry of the baseline design. As we will find, the optimum shapes of Section 6 are not efficiently represented by this x-distribution of control points. In retrospect, a much more effective parameterization of this design space would have been based on a more uniformly-distributed set of control points. Since the approach adopted here to define the design geometries is based on the perturbation of Eqn (5), for the 4th-order Bezier curve, this would only require that the $xcpts$ of the interior control points be placed at quarter-chord intervals.

Bezier Degree Elevation

To achieve higher dimensionality, an infinite family of design spaces is constructed. This family has the property that all possible airfoil shapes supported by its M-space are fully contained in its N-space, where $3 \leq M \leq N$. Recall the Bez4-0012-ADO airfoil with control points given in Table I. For the aerodynamic shape optimizations performed in this study, only the internal $ycpts$ ($ycpt_n, 1 \leq n \leq N$) are used as design variables. Here, N is the dimension of the design space, and $K = N + 1$ is the order of the Bezier curve associated with the N-space. For example, a 4th-order Bezier curve is defined by 5 control points ($0 \leq k \leq K = 4$). Since only the internal $ycpts$ ($ycpt_n;\ 1 \leq n \leq N = 3$) are used as design variables, an arbitrary 4th-order Bezier curve pinned at the LE and TE end-points with fixed $xcpts$ locations given by Table I defines our 3-space.

In order to satisfy the property that any M-space is a subset of any N-space, where $3 \leq M \leq N$, we utilize a recursive degree elevation of the Bez4-0012-ADO baseline airfoil. This process is illustrated in Figure 9, where the control points of Bez4-0012-ADO are elevated from its native 3-space to an equivalent airfoil in 4-space. In general, elevating a Kth-order Bezier curve to $(K+1)^{st}$-order has control points given by the following recursive formula.

$$\mathcal{B}_k^{(K+1)} = \left(\frac{k}{K+1}\right)\mathcal{B}_{k-1}^{(K)} + \left(\frac{K+1-k}{K+1}\right)\mathcal{B}_k^{(K)}; \ \ where\ 0 \leq k \leq K+1. \tag{8}$$

Here, $\mathcal{B}^{(K)}$ and $\mathcal{B}^{(K+1)}$ represent the control points of the Kth-order and $(K+1)^{st}$-order Bezier curves, respectively. Note that while $\mathcal{B}_{-1}^{(K)}$ and $\mathcal{B}_{K+1}^{(K)}$ do not exist, their weighting factors per Eqn (8) are zero.

Free-Surface Design-Space

SYN83 and SYN107 perform optimizations on a free surface, where every surface point in the grid is allowed to be independently perturbed normal to the surface geometry. If the airfoil surface is defined by N surface points within the grid, then the free surface has N design variables defining the design space. This represents the highest supported design space possible by the discrete grid, and is commonly referred to as full-parameterization throughout the literature. The SYN83 and SYN107 results presented herein use $N = 513$ and $N = 5,313$, respectively.

There are both advantages and disadvantages to using a free surface for the design space. On the plus side, a full parameterization does not artificially constrain the design, and therefore, can provide the best performance possible in a design optimization. However, this is not always the case, as it can also provide more local minima on which a gradient-based optimization can converge. Other disadvantages include difficulties in the rigorous enforcement of geometric constraints, as well as in the loss-of-translation when transferring the optimum free-surface shapes to CAD or other tools utilized in the design process.

B-Spline Design-Space

In order to provide an interface with CAD systems, without loss due to translation, SYN107 has been recently enhanced to include a high-dimensional B-Spline surface representation. The B-Splines are defined in the airfoil's 2D nondimensionalized coordinate system. Each airfoil section is split into upper- and lower-surface

curves. Third-order B-Splines of 33 control points are utilized to define each surface. The $xcpt$ coordinates of both are preset by a cosine distribution, as per Eqn (9).

$$\begin{aligned} xcpt_0 &= 0, \\ xcpt_n &= \frac{1}{2}\left[1 - cos\left(\frac{n-1}{31}\,\pi\right)\right], \quad 1 \le n \le 32. \end{aligned} \tag{9}$$

Since the leading- and trailing-edge points are pinned, the first and last control points have $ycpt_0 = 0$, and $ycpt_{32} = \pm\frac{1}{2}TE_{Base}$. The remaining $ycpt$ coordinates of each B-Spline are defined with a least-squares fit of their corresponding grid points. These curve-fits are constrained such that the upper-lower B-Spline-pair preserve curvature continuity at the leading edge. Curvature continuity at the LE requires $ycpt_1^u = -ycpt_1^l$. Finally, the original free-surface points are projected to the B-Spline representation and the optimization process continues as normal. This capability provides SYN107 with an internal, mathematically-rigorous representation of the wing geometry. This geometry representation is analytically interrogated for airfoil thickness, camber and curvature distributions, which are then used to enforce any of the typical geometric constraints imposed on wing design. SYN107 results presented herein, based on its B-Spline geometry engine, use $N = 2 \times 31 \times 33 = 2,046$ adjustable control points.

Figure 10 illustrates the B-Spline fit of the RAE2822 airfoil coordinates from Figure 4. The corresponding control points that result from this best-fit are shown in Figure 11. To illustrate the extent of control that a control point has on the curves, Figure 12 depicts a grid overlaying the RAE2822 B-Splines. Each curve segment between grid lines is influenced by only 4 control points, which are located approximately at the intersection of the vertical-grid-lines and the B-Spline curves. Figure 13 provides a close-up view near the RAE2822 leading-edge region and includes the original discrete coordinates (small asterix), the control point (bold dots), the B-Spline curves (bold lines), the curve-segment grid, the chordline, and the osculating circle at the leading-edge point. Figure 14 provides a close-up view near the RAE2822 trailing edge with similar data, plus it includes the thickness distribution, and the camber line with its maximum value and location indicated. Figure 15 provides the same close-up view near the RAE2822 trailing edge, however with only the original airfoil coordinates and the curve-segment grid shown. The reason for showing this last figure is to touch on a point about the least-squares-fit process. Since the cubic B-Splines have the property of local control, it is essential that the discrete coordinate data that is being fit have sufficient coverage. Figure 15 clearly shows that there are about two discrete airfoil coordinates per curve segment per surface, whereas the absolute minimum coverage necessary is one discrete point per surface per curve segment. Hence, the data sampling provided by the UIUC website is more than sufficient for our least-squares fit to be well behaved. If insufficient discrete data is available, an over-sampling of the data may be required for a best-fit to be possible.

This concludes our introductory and backgound discussions on parameterizing an aerodynamic design space. We now turn our attention to reviewing three investigations of aerodynamic shape optimization. The first of the three sample cases is presented next.

6 NACA0012-ADO Invisicd Non-Lifting Airfoil

This section provides the first of three sample cases being discussed within this lecture. At first glance, this sample case would appear to be the simplest of the three. After all, it is a 2D inviscid-flow problem, whereas the other sample cases involve 3D viscous flows. As it turns out, this is far from being a simple test case, and as such, it is the subject of on-going research by members of the AIAA Aerodynamic Design Optimization (ADO) Discussion Group. In addition to our research, please see studies by Bisson [13] and Carrier [14].

Model Problem

The model problem of this optimization is to mimimize the drag of a symmetric airfoil, for an inviscid transonic flow at the condition of $M = 0.85$, and $\alpha = 0°$, subject to the geometric constraint:

$$y_{Optimum}(x) \ge y_{Baseline}(x) \;\; ; \;\; 0 \le x \le 1. \tag{10}$$

Eqn (10) requires that the thickness distribution of the baseline airfoil is maintained at every point along the chord. Note that the flow physics of this model problem is such that the only true source of drag is that associated with any shocks that may arise. The problem is based on one crafted by Vassberg [12], with an anticipation that a shock-free design at these flow conditions and under these geometric constraints is unachievable. This two-dimensional inviscid compressible flow problem is chosen to provide a nonlinear objective function, yet one that only requires moderate computational costs to evaluate. With the inexpensive nature of computing the objective function, it is feasible to survey a wide range of design-space dimensions. However, as we will show in our results, this *simple* optimization problem turns out to be a pathologically difficult test case. This test case was developed by the first author with the intent to challenge (break) aerodynamic optimization methods; more specifically, to expose their weaknesses and identify where further research is required. In this respect, this test case has been quite successful.

The next section describes the baseline NACA0012-ADO geometry.

NACA0012-ADO Baseline Geometry

This section provides a description of the baseline airfoil utilized in this study. This geometry is based on the symmetric NACA0012 airfoil section, however with the closed trailing-edge modification suggested by Nadarajah [15], which is an improvement to that originally proposed by Vassberg [12] in earlier work.

Abbott and von Doenhoff [10] give the analytic equation defining the NACA0012 airfoil as:

$$y_N(x) = \pm\frac{0.12}{0.2}\left(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1015x^4\right), \quad 0 \le x \le 1. \tag{11}$$

The numerator of the lead terms in Eqn (11) (i.e., 0.12) is the maximum thickness of the airfoil. The standard NACA0012 airfoil is defined over the interval: $0 \le x \le 1$. However, at $x = 1$, the y coordinate does not vanish, and therefore, the trailing edge is not sharp, but rather has about a 0.42%-thick blunt base.

In order to avoid issues related to the solution of inviscid flows about aft-facing steps, Vassberg [12] extended the airfoil chord to the local root of Eqn (11) which occurs at $x \simeq 1.0089$. For this ADO test case, Nadarajah [15] suggested instead that the airfoil definition of Eqn (11) be modified by changing the coefficient of the x^4 term such that a sharp trailing-edge is recovered at $x = 1$. The resulting analytic equation which defines the NACA0012-ADO airfoil shape is:

$$y_A(x) = \pm\frac{0.12}{0.2}\left(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1036x^4\right), \quad 0 \le x \le 1. \tag{12}$$

NACA0012-ADO Results

A fairly significant effort has been devoted to this test case, and many optimizations have been performed by a number of researchers. At first impression, this model problem seems trivial. As it turns out, this is instead a very difficult problem. The flow about the resulting optimum airfoils becomes nearly singular, and as such, presents many issues uncommon to our regular applications of aerodynamic shape optimization. In fact, it seems that the better the optimization, the more pathological the problem becomes. To this end, we document optimization runs with MDOPT, CMA-ES evolution strategy, SYN83, as well as include select results from other investigations. Cross-analyses of several of the optimum designs are provided by FLO82.

MDOPT Results

Included in this section are previously-attained results by Vassberg, *et.al.* [12]. Although the model problem of this previous study is not exactly that of the NACA0012-ADO test case, it is so similar that the influence of parameterization on each problem yields very similar characteristics. Furthermore, this prior work described a situation that one should avoid when formulating the design space. This *trap* will be discussed here for the reader's benefit. The original study was conducted in three chronological phases. The first phase was a discovery exercise for the first author (and colleagues) to become familiar with MDOPT. The second phase introduced a SYN83 optimization to provide an optimum airfoil from a very-high-dimension design space. This effort was performed independently by the second author without knowledge of the Phase-I results. The performance of this optimum airfoil was then used as a goal to achieve in Phase-III. The third phase revisited the MDOPT investigations, however, this time with insight of the results of SYN83 in Phase-II. This extra knowledge made a significant difference.

As noted above, the first phase was a discovery exercise. Here, many complete optimizations were conducted (and completely disposed of) before self-consistant results were attained. Note that the optimum drag level obtained in any N-space should be no worse than that achieved in any M-space, if $3 \leq M \leq N$. Initially, to obtain this behavior (or close to it) required much effort. Specifically, the range of the design variables had to be manipulated. If the user-defined range of a design variable (DV) is set too small, or not well centered, then the region of the design space studied may not contain the global optimum. Yet, if the ranges of the DVs are too large, then the response surface from the design of experiments (DOE) can become so inaccurate that only false optimums are pursued. Unfortunately, the pertinent information required to set such ranges is occasionally not known *a priori*; it is accumulated with applied experience on a given class of problems.

As DOEs and Response Surfaces are frequently used thoughout the literature, a note about how their computational expense scales with design-space dimension is in order. The number of coefficients defining a quadratic response surface is: $NCoef = \frac{NDV*(NDV+1)}{2}$. Further, the computational effort required to determine the coefficients for a response surface scales with the number of unknowns cubed. Hence, the build time of the DOE response surfaces is $\mathcal{O}(NDV^6)$. Collected data are tabulated in Table II and illustrated in Figure 16. This figure shows that the asymptotic slope of the trendline is 6.0, and hence, is consistent with that expected. Note that the data of Table II does not include the time required to evaluate the objective functions of the DOE, where the number of cases in a DOE must be greater-than or equal-to the number of the unknown coefficients. Hence, $NDOE = \mathcal{O}(NDV^2)$.

Table II: DOE Response-Surface Build Times.

NDV	$NCoef$	No. DOE Cases	CPU (sec)
6	21	121	7
12	78	256	157
24	300	529	3,341
36	666	1,369	38,042

Figure 17 shows a comparison of the convergence histories of best airfoil shapes for [6,12,24,36] design variables. In this figure, note that the starting point of each curve represents the number of cases in the initializing DOE. As aforementioned, the size of the initial DOE scales as NDV^2. In general, note that as the design-space dimension increases, the number of cases required beyond the initial DOE for convergence also increases, while the drag of the optimum geometries improve, at least until the 36-space result. The interesting characteristic of this trend is that the drag of the optimum in 36-space is worse than that found in 24-space. Since all supported geometries in 24-space are fully contained in 36-space, one explanation for this reversal is that the optimization process simply has not yet found the optimum geometry, even after 2,324 cases have been analyzed. This could also be a consequence of the user-specified range on DVs in 36-space inadequately capturing the pertinent geometries of the 24-space run.

In the second phase, the last author conducted an independent SYN83 optimization on the model problem. This effort was performed without knowledge of the MDOPT results of Phase-I, and therefore, was conducted as a blind test. Since the design space for SYN83 is essentially the highest dimensional space supported by the discrete grid, it was anticipated that the performance of the resulting optimum airfoil could provide a limit to what is achievable. Figure 18 provides a FLO82 solution for the optimum airfoil derived by SYN83. This airfoil has been designated BJ5XE. The shock strength of the optimum airfoil is much diminished relative to the baseline. According to SYN83, the drag coefficient for the BJ5XE optimum airfoil is about 104.4 *counts*, yielding a total reduction of more than 350 *counts* relative to the baseline airfoil. This finding is significantly better than anything discovered in Phase-I.

Due to the large disparity between the results of the first two phases, a third phase was initiated to reopen the MDOPT study of Phase-I. Comparison of the BJ5XE airfoil with optimum geometries of Phase-I uncovered the issue. The TE included-angle of BJ5XE was much larger than that of any of the Phase-I optimum airfoils. While the optimum airfoils of Phase-I came close to the TE included-angle constraint, they did not reside on this constraint boundary. As a consequence, it was not obvious at the time that this constraint was an issue. By relaxing the constraint on the TE included-angle, as well as implementing other lessons learned, the results of Phase-III now align well with those of Phase-II. Although our final results from the last two phases compare well with each other, yielding designs with drag levels of just over 100 *counts*, we will see that more recent optimizations have produced airfoils with much less drag.

Figure 19 shows a comparison of the convergence histories of the best airfoils for [3,6,12,24,36] design variables in Phase-III. In general, as the design-space dimension increases, the drag of the optimum geometries monotonically improves. Further, the number of cases required to achieve convergence increases with NDV.

Table III tabulates the final optimum-drag values obtained in Phase-III as a function of design-space dimension. The baseline airfoil result corresponds to $NDV = 0$.

Table III: Phase-III MDOPT Results.

NDV	No. Cases	Optimum Case	C_{Dopt}
0	1	-	468.9
3	546	359-1	312.5
6	1,349	1281-1	221.5
12	1,787	1729-1	139.5
24	2,377	2315-1	117.6
36	3,034	3020-1	103.8

Figure 20 provides a comparison of the baseline airfoil (solid line, no symbol), with the best airfoil shapes for [3,6,12,24,36] design variables (open symbols), and includes the SYN83 BJ5XE airfoil (solid symbol). The Y coordinate is amplified to enhance visual comparison. It was gratifying to see that the optimum airfoils of Phase-III were approaching BJ5XE as the dimension of the design space increases.

Figure 21 provides a comparison of the pressure distributions of the baseline airfoil with those of the optimum shapes for [3,6,12,24,36] design variables. It is interesting to note that the shock strengths of each of these geometries appear to be essentially the same. The pressure level just upstream of the shock is $C_p \sim -0.9$ and jumps to a level of $C_p \sim +0.1$ just downstream of the shock. Yet the drag levels of these airfoils range from about 100 *counts* to about 470 *counts*. To understand how this can be, one must inspect the flowfield, not just the properties on the airfoil surface. Figures 18 & 22 illustrate the flowfield Mach contours for the BJ5XE and the baseline NACA0012-ADO airfoils, respectively. Note that the shock of the NACA0012-ADO airfoil extends about 75% of a chord-length off the surface into the flowfield. Whereas the shock system of the BJ5XE airfoil is comprised of two parts: 1) a strong normal shock adjacent to and extending from the airfoil to about 5% of a chord-length into the flowfield, and 2) a weak curved shock which extends from about 5%-to-75% of a chord-length off the surface into the flowfield.

CMA-ES Results

For all optimizations based on CMA-ES, the Bezier design space was used and the design (perturbation) vector was initialized as a zero-vector with an isotropic search. Due to wall-clock constraints, higher-dimension optimizations were unable to be completed. Figure 23 illustrates the convergence histories for $N = [3, 6, 9]$. Table IV and Figure 24 provide the convergence of optimum drag levels as the dimension of the Bezier design space is increased. Table V gives the resulting optimum design vectors for $N = [3, 6, 9]$.

Table IV:
CMA-ES/OVERFLOW Results (C_d in *counts*).

N	Iterations	Population	Total Runs	C_{Dopt}	ΔC_d
0	-	-	1	483.70	-
3	161	7	1,127	319.17	-164.53
6	378	9	3,600	212.27	-271.43
9	400	10	4,000	135.74	-347.96

Table V:
CMA-ES Optimum Design Perturbation Vectors in Bezier Space.

N	$ycpt_1$	$ycpt_2$	$ycpt_3$	$ycpt_4$	$ycpt_5$	$ycpt_6$	$ycpt_7$	$ycpt_8$	$ycpt_9$
3	0.054596	-0.050248	0.026009						
6	0.000000	0.111226	-0.168611	0.182871	-0.152965	0.085558			
9	0.000000	0.094056	-0.084422	0.049105	0.085296	-0.237675	0.295685	-0.239163	0.123469

A curious behavior of this optimization method is that running a single iteration does not necessarily improve the design. While sudden increases in the objective function are observed throughout the optimization process, it is most pronounced for the first iterations. This appears to be the result of the random search process being isotropic at this point with the standard deviation of the search being significantly larger than what would be used for gradient estimation. During these first iterations, the algorithm is essentially embarking on a drunkard's walk through the neighborhood around the initial vector until enough information is gathered to meaningfully adapt the covariance matrix and begin the descent. Increases in drag later in the optimization are the result of locally sharp changes in the gradient between iterations, which cannot be efficiently navigated by a relatively large search area. The mean vector then overshoots the desired path, and the algorithm needs to shrink the standard deviation and properly rotate the covariance matrix.

SYN83 Results

Two SYN83 optimizations are presented on the NACA0012-ADO test case. The first initialized the seed airfoil with the baseline NACA0012-ADO section, whereas the second used a well-designed airfoil as the starting point. The resulting SYN83 design airfoils are referred to as SYN-NADOV01 and SYN-NADOV02, respectively. Note that both are local optimum designs. The SYN83 optimizations are performed on a C-mesh with cell dimensions of (768x128). Under this setup, these SYN83 optimizations required about 40 minutes of CPU time on a deskside computer with Intel Core i7 CPU 3.20GHz processors. Figure 1 provides a close-up rendering of this mesh. In the first optimization run, SYN83 reports the initial drag of the NACA0012-ADO airfoil as 456.34 *counts*, and the drag of the SYN-NADOV01 section as 103.71 *counts*. In the second optimization run, SYN83 computes a drag coefficient of 101.79 *counts* for the well-designed seed airfoil, and 79.31 *counts* for the SYN-NADOV02 optimum design. These SYN83 data as well as pertinent deltas are provided in Table VI.

Table VI:
SYN83 Results (C_d in *counts*).

Airfoil		C_d	ΔC_d
Seed	NACA0012-ADO	456.34	-
Design	SYN-NADOV01	103.71	-352.63
Seed	SEED02	101.79	-354.55
Design	SYN-NADOV02	79.31	-377.03

Figures 28-29 provide surface-pressure distributions and field isobars about the SYN-NADOV01 and SYN-NADOV02 designs, respectively. Examining these figures from a global perspective, it is interesting that the resulting optimums are tending towards fore-aft-symmetric designs, where surface pressure distributions and field isobar patterns are roughly mirror images of each other about $x/c = 0.5$. Another interesting observation is that the recompression of the C_P-peak near the leading edge is essentially isentropic; this is accomplished by means of a very weak oblique shock. These characteristics of the optimum designs carry through to the works of others, as will be shown at the end of this section.

FLO82 Drag Assessments

With all of the optimizations and flow solutions performed on this test case, one thing is evident, the flowfield about any of the optimized airfoils is extremely sensitive to just about everything. This includes grid, flow condition, discretization stencil, governing equation, etc. In an attempt to provide independent, grid-converged drag levels for a complete set of airfoils under study, the FLO82 aerodynamic assessment process of Vassberg [9] for inviscid transonic airfoils is utilized. For this test case, FLO82 is run with upper-lower symmetry enforced. Table VII provides FLO82 results for each airfoil on a grid sequence of $ni = nj = [256\ ,\ 512\ ,\ 1024\ ,\ 2048\ ,\ Continuum]$, where the continuum result is a Richardson extrapolation of the other data.

Table VII:
FLO82 Drag Assessment (C_d in *counts*).

Airfoil	N256	N512	N1024	N2048	Continuum	ΔC_d
NACA0012-ADO	470.19	470.09	471.13	471.23	471.27	-
NADOT101	487.50	488.20	488.48	488.58	488.62	+17.35
CMAES03	296.90	294.30	294.66	294.79	294.88	-176.39
CMAES06	194.24	189.00	189.48	189.68	189.82	-281.45
CMAES09	116.85	103.95	101.78	101.77	101.77	-369.50
SYN-NADOV01	153.78	123.24	119.03	118.34	118.21	-353.06
SYN-NADOV02	109.25	86.87	84.68	84.50	84.48	-386.79
SYNT101	122.22	99.20	96.75	96.64	96.63	-374.64
SIVAFOIL	118.60	72.95	51.56	46.56	45.03	-426.24

Under this setup, these FLO82 analysis runs required about 3 hours of CPU time on a deskside computer. Also given in this table is the sensitivity of drag with respect to airfoil thickness for the baseline NACA0012-ADO section. Here, the NADOT101 airfoil is the baseline thickness scaled by a factor of 1.01. Notice that this 1% increase in thickness increases drag by 17.35 *counts* or by 3.68%. Hence, the flowfield about the baseline geometry reacts in a nonlinear manner. A similar sensitivity is computed for the SYN-NADOV02 airfoil using SYNT101. In this case, a 1% increase in thickness increases drag by 12.15 *counts* or by 14.38%. In nondimensional terms, drag for this airfoil is almost 4 times more sensitive than the baseline.

In spite of our best efforts, recent progress by other investigations on the NACA0012-ADO test case have yielded airfoils with much reduced levels of drag. Select results from two of these studies are presented next.

Other Recent Progress

During the 2014 AIAA SciTech Conference, a special session on the ADO test cases was organized. With permission, we present some select results from friends and colleagues based on the NACA0012-ADO test case and pertinent to this lecture on the influence of shape parameterization on aerodynamic shape optimizations.

Figure 30 is a FLO82 cross analysis of an optimum airfoil developed by Bisson and Nadarajah [13]. This cross analysis was performed independently by the first author and is consistent with all of the FLO82 results presented in this lecture. The drag data is included in Table VII above, labeled as SIVAFOIL. Note that this airfoil has almost 50 *counts* less drag than the best design discovered by the authors. Another interesting characteristic of the SIVAFOIL is that it exhibits a tight-double-shock system, as can be seen in Figure 30.

The work of Carrier, *et.al.* [14] is shown in Figures 31-33. The left side of Figure 31 provides the convergence histories of drag for two similar optimizations where the distribution of control points were modeled after Vassberg [12]. In both cases, 6 design variables were used. In one case, the shape parameterization was a 7th-order Bezier curve with 6 adjustable control points. In the other case, the same 6 adjustable control points were used, however for a cubic B-Spline instead. The difference is quite astonishing; the optimum Bezier airfoil converged to about 346 *counts*, whereas the B-Spline airfoil optimized at about 126 *counts*. Although a complete understanding of the root cause for this is pending, we believe that much of it is attributed to the global control of the Bezier curve as opposed to the local control of the B-Spline. The right side of Figures 31 repeats this comparison, however, with evenly-spaced control points. Note that both Bezier and B-Spline based airfoils improved. The Bezier airfoil is now reduced to about 201 *counts*, while the B-Spline drops to about 91 *counts*. Figures 32-33 illustrate another systematic study showing how the dimensionality of the design space affects final results. Here, $N = [6,\ 12,\ 24,\ 36,\ 48,\ 64,\ 96]$. These data indicate that a minimum number of design variables needed to achieve reasonable designs is about 36. This can be seen in the convergence of both pressure distributions as well as geometry, as depicted in Figure 33.

Recently, the authors have revisited this problem. However, this time with an updated version SYN82, which is based on an O-mesh, whereas SYN83 is based on a C-mesh. We wondered if this might have an affect, since the optimum airfoils for the problem tend towards having a nearly rounded trailing edge. As it turns out, this has a dramatic affect on the outcome. In retrospect, the C-mesh topology is reducing both the quality and resolution of the mesh near the trailing edge as the TE-included-angle gets very large. As a consequence, SYN83 optimizations were prematurely stalling in their drag reductions. When we tried this optimization with SYN82, the very first run reduced the drag to about 22 counts, and the only thing required to reduce it further was to increase the number of design cycles. Figure 34 illustrates the result of

a SYN82 optimization based on a (2560 x 512) O-mesh topology after 2500 design cycles. For this run, the drag has been reduced to just under 5 counts. We have run cases out much further and the improvement monotonically approaches zero drag. An interesting artifact of these optimizations is that we have not been able to verify the low-drag-branch solution at $M = 0.85$ with any other mesh than on the mesh used for optimization, and we have tried numerous variations of O-meshes. When we perform a Mach sweep from above, the dragrise trends towards the low-drag value, but it jumps to the high-drag branch just before reaching the design Mach number. The investigations on this problem may never end.

This concludes our discussion on the NACA0012-ADO test case. The next section provides a discussion on an ONERA-M6 sample problem.

7 ONERA-M6 Non-Lifting Viscous Wing

The second example case we present in this lecture is a three-dimensional non-lifting viscous-flow problem based on the benchmark ONERA-M6 wing at flow conditions $M = 0.923$, $\alpha = 0°$, and $Ren = 20x10^6$. Here, we conduct three drag minimizations with varying thickness constraints. In all three optimizations, the maximum thickness of each airfoil section is maintained, however the chordwise thickness distribution is only partially maintained. For the purpose of this demonstration, SYN107 is run with its B-Spline design space and arbitrarily terminated after 50 design cycles. The objective is to minimize total drag, subject to the following geometric constraints.

$$\begin{aligned} Tmax_{opt} &\geq Tmax_{M6}, \\ Tdist_{opt} &\geq [0.95,\ 0.90,\ 0.50] * Tdist_{M6}. \end{aligned} \tag{13}$$

Figure 35 provides the convergence history of total drag as a function of design cycle. Initially, the drag drops in a comparable manner for all three optimizations over the first 5 design cycles, then the $0.95 * Tdist$ optimization departs while the other two remain comparable until the 10th design cycle, after which the three histories assume unique paths. This behavior is caused because the $Tdist$ constraints become active at various stages of the optimization. In fact, the $0.50 * Tdist$ constraint is inactive throughout its optimization, however the $Tmax$ constraint is always active.

Figure 36 provides a comparison of the pressure distributions between the baseline ONERA-M6 and optimized wings. Note that the first and second optimizations yield pressure distributions with double-shock systems. The presence of the forward shock is a consequence of the $Tdist$ constraint. The third optimization is characterized with a fairly-weak single-shock system made possible by the inactive $Tdist$ constraint.

Figure 37 provides drag-loop comparisons as a complement to Figure 36. We note that drag loops relate to pressure drag only, and not to skin-friction drag. Since drag loops are foreign to many readers, we will provide a short discussion on this topic. For this lesson, lets extract the drag loops of the 81.3% semispan station of Figure 37. Figures 38-39 illustrate the drag loops for the ONERA-M6 and $[Tmax + 0.50 * T]$ wings, respectively. However, these loops have been color-coded to distinguish the portions of the loops that integrate drag (red) from the portions that integrate thrust (green). Notice that the optimized wing has substantially less drag (red) than the baseline M6 wing. If you are wondering why it seems as if the optimized wing loop has more thrust (green) than it has drag (red), it is because it does. This is possible because of the influence of the inboard wing. In practice, outboard sections frequently exhibit negative sectional pressure-drag coefficients.

Figure 40 displays a side-by-side comparison of the upper-surface isobars, with the baseline M6 wing on the left and the $[Tmax + 0.50 * T]$ optimum on the right. Referring to the baseline isobars of this figure, if one had to perform an aerodynamic shape optimization on this wing, but with a very small number of design variables, more than likely he would concentrate the DVs along the baseline's shock pattern. This would be partially correct, however, Figure 41 clearly shows that most of the geometric change occurs on the inboard wing near the leading-edge and mid-chord regions. Upon studying the outcome of this optimization, it is clear that SYN107 is primarily manipulating the wing's x-cut area distribution, *aka* area ruling. Figures 42-44 provide chordwise-thickness-distribution comparisons at semispan stations of 6%, 50%, and 87%, respectively. Close inspection of these (especially 6%) shows where the $Tdist$ constraints are active. For completeness, spanwise distributions of thickness and leading-edge radius are provided in Figures 45-46.

This concludes our discussion on the ONERA-M6 test case. We now turn to our third and last sample case which is based on the CRM wing.

8 Common Research Model Lifting Viscous Wing

Model Problem

The model problem for our third test case is based on the NASA Common Research Model (CRM) developed by Vassberg, *et.al.* [16]. Here, the wing of the CRM Wing/Body configuration has been extracted from the DPW-V [17] theoretical geometry definition and transformed by Osusky [18] as follows. The exposed wing of the CRM WB configuration is translated in the negative span direction to place the root airfoil at the symmetry plane, $y = 0$. The wing is also translated in x and z to place the root leading-edge point at the origin. Finally, this geometry has been scaled down by its mean aerodynamic chord (MAC). We refer to this geometry as ADO-CRM-Wing, and in its system, the reference quantities are: $Cref = 1.0$, $Sref/2 = 3.407014$, $Xref = 1.2077$, $Yref = 0.0$, $Zref = 0.007669$, with a semispan of $\frac{b}{2} = 3.75820$.

The objective is to minimize the drag of the ADO-CRM-Wing at the flow condition of $M = 0.85$, and $Re = 5 \times 10^6$, considering a fully-turbulent flow, and subject to the following constraints.

$$\begin{aligned} C_L &= 0.5 \\ C_M &\geq -0.17 \\ Volume &\geq Volume_{initial} \end{aligned} \tag{14}$$

where, $Volume$ refers to the internal volume of the wing.

SYN107 Results

A number of SYN107 optimizations have been performed during this study, however for the sake of brevity, only the two most pertinent ones will be presented. Both of these conform to the problem statement of Eqn (14), however, these optimizations slightly over-constrain the design as it imposes a geometric constraint to maintain the cross-sectional area distribution of the wing with respect to the baseline ADO-CRM-Wing, not just its volume. In addition, these optimizations were performed using an objective function comprised of blending absolute drag and level of pitching-moment violation. The first optimization is a single-point design at $C_L = 0.5$. The resulting geometry of this optimization is referred to as CRMADOV09. This single-point design required about 2.3 hours of elapsed time, running in parallel on 4 cores, on a deskside computer with Intel Core i7 CPU 3.20GHz processors. The second SYN107 optimization is an evenly-weighted multi-point design at $C_L = [0.50, 0.55, 0.45]$. The resulting geometry of the second optimization is referred to as CRMADOV10. This triple-point design required about 7.0 hours of elapsed time, running in parallel on 4 cores, on a deskside computer. In both of these optimizations, the seed wing is the ADO-CRM-Wing.

SYN107 uses an internally-generated C-H-mesh with dimensions (257x65x49). Here, 257 points wrap around the complete airfoil/wake contour, with 161 points defining each airfoil section in the grid. For viscous flow calculations, the grid spacing at the wing in the normal direction is approximately $y^+ = 1$. There are 65 points in this direction from the wing to the farfield. The spanwise dimension of the grid is 49, with 33 K-planes residing on the wing. Hence, the free-surface is comprised of $161 \times 33 = 5,313$ design variables. In these optimizations, the internal B-Spline geometry representation is invoked, which is defined with $2 \times 33 \times 33 = 2,178$ control points.

A by-product of the SYN107 optimization process is a set of force and moment sensitivities, including: $\frac{dC_L}{d\alpha}$, $\frac{dC_D}{d\alpha}$, $\frac{dC_D}{dC_L}$, and $\frac{dC_M}{dC_L}$. In order to provide a more accurate estimate of the force and moments at the true constrained-lift condition of $C_L = 0.5$, the raw data is corrected to this condition using these sensitivities. These data are provided in Table VIII.

Figure 47 illustrates the SYN107 flow solution of the initial seed ADO-CRM-Wing at the design condition of $M = 0.85$, $Re = 5 \times 10^6$, $\alpha = 2.215°$, $C_L = 0.5005$, $C_D = 218.8$ *counts*, and $C_M = -0.1843$. Correcting to a C_L of 0.50, the drag is $C_D = 218.5$ *counts*, and $C_M = -0.18416$ Note that this baseline wing violates the pitching-moment constraint of Eqn (14).

Figure 48 provides the results of the first SYN107 optimization. Here, the CRMADOV09 wing at $\alpha = 2.539°$ has force and moments of $C_L = 0.4989$, $C_D = 207.4$ *counts*, and $C_M = -0.1696$. Correcting to a C_L of 0.50, the drag is $C_D = 207.9$ *counts*, and $C_M = -0.16993$ With the cross-sectional area distribution

maintained, this optimization yields a drag reduction of about 10.6 *counts*. An independent cross-analysis of the CRMADOV09 design using OVERFLOW shows a drag improvement of 10.0 *counts*. These two estimates of the performance improvement are in close agreement.

Figure 49-51 provide the results of the second SYN107 optimization. Here, the CRMADOV10 wing at $\alpha = 2.518°$ has force and moments of $C_L = 0.4993$, $C_D = 207.4$ *counts*, and $C_M = -0.1702$. Correcting to a C_L of 0.50, the drag is $C_D = 209.2$ *counts*, and $C_M = -0.17043$ This optimization provides a drag reduction of about 9.3 *counts* at the design point. Hence, the triple-point design gives back about 1.3 *counts* at the design point relative to the single-point design.

Table VIII:
SYN107 Optimizaions at $C_L = 0.5$, $M = 0.85$, $Re = 5 \times 10^6$.

$WING$	C_L	C_D	C_M	dC_D/dC_L	dC_M/dC_L	$C_D cor$	$C_M cor$
BASELINE	0.5005	0.02188	-0.1843	0.05355	-0.34786	0.02185	-0.18416
CRMADOV09	0.4989	0.02074	-0.1696	0.04375	-0.33373	0.02079	-0.16993
CRMADOV10	0.4993	0.02089	-0.1702	0.04364	-0.30294	0.02092	-0.17043

A comparison of pressure distributions for the three wings is provided in Figure 52. Here, pressure distribution are over-laid at 8 span stations on the wing. Note that due to the pitching-moment constraint of Eqn (14), the optimized wings have reduced aft-loading throughout and increased foreward-loading on the inboard wing. Figure 53 compares the spanload and sectional lift-coefficient distributions for the three wings. Note that the CRMADOV10 multi-point design has slightly migrated the spanload inward. This is most likely to trade a slight increase in induced drag for a larger reduced shock drag at the higher lifting condition of $C_L = 0.55$. Figure 54 illustrates the SYN107 convergence histories of drag for the CRMADOV09 single-point and the CRMADO10 multi-point designs. Although not shown here, monitoring the pitching-moment during the design cycles shows that in the early stages, the optimization aggressively attacks drag until it starts to level out, then it addresses the pitching-moment violation with increasing attention.

To better understand the off-design performance of these wings, a small drag polar is presented in Figure 55. The baseline polar correspond to an α-sweep from 1°-to-3°, at every 0.2°. The polars for the designed wings correspond to an α-sweep from 1°-to-3.25°, at every 0.05°. Notice that the baseline ADO-CRM-Wing polar is well behaved, whereas the CRMADOV09 designed wing exhibits a character typical of single-point optimizations. Finally, the CRMADOV10 three-point design recovers a well behaved drag polar. Per flow condition, each analysis required about 4.2 minutes of elapsed time, running in parallel on 4 cores, on a deskside computer.

Figures 56-57 provide comparisons of lift curves and pitching-moment polars, respectively. Note that the baseline wing violates the pitching-moment constraint, whereas the two optimized wings appropriately conform to it.

9 Acknowledgements

The authors thank James G. Coder (PhD Candadite, Penn State University) for providing the CMA-ES results of Section 6. This is previously unpublished work that Mr. Coder performed in collaboration with the first author during the second half of 2013 after visiting Boeing during that summer as an Intern.

The authors also thank Gerald Carrier (ONERA, Meudon, France) for his kind permission to use excerpts of their work on the NACA0012-ADO model problem. The work performed in their study is extremely thorough and of high quality.

Siva Nadarajah (Professor, McGill University, Montreal, Canada) graciously provided their optimum airfoil shape for the NACA0012-ADO model problem for cross-analysis by the authors with FLO82. This work is also of high quality.

We encourage the interested to read the AIAA Papers from the ADO Session, SciTech Conference, January, 2014. Especially the works of the above two groups.

References

[1] S. T. LeDoux, W. W. Herling, G. J. Fatta, and R. R. Ratcliff. MDOPT - A multidisciplinary design optimization system using higher-order analysis codes. *AIAA Paper 2004-4567*, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, August 2004.

[2] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 2(2):159–195, 2001. .

[3] A. Jameson. Efficient aerodynamic shape optimization. *AIAA Paper 2004-4369*, 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, NY, August 2004.

[4] A. Jameson and J. C. Vassberg. Computational fluid dynamics for aerodynamic design: Its current and future impact. *AIAA Paper 2001-0538*, Reno, NV, January 2001.

[5] R. H. Nichols and P. G. Buning. OVERFLOW User's Manual, Version 2.1t. Technical report, U. of Alabama and NASA, August 2008.

[6] A. Jameson. Solution of the Euler equations for two dimensional transonic flow by a multigrid method. *Applied Mathematics and Computation*, 13:327–356, 1983.

[7] A. Jameson. Artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence in transonic and hypersonic flows. *AIAA Paper 93-3359*, AIAA 11th Computational Fluid Dynamics Conference, Orlando, Florida, July 1993.

[8] J. C. Vassberg and A. Jameson. In pursuit of grid convergence, Part I: Two-dimensional Euler solutions. *AIAA Paper 2009-4114*, 27th AIAA Applied Aerodynamics Conference, San Antonio, TX, June 2009.

[9] J. C. Vassberg and A. Jameson. In pursuit of grid convergence for Two-dimensional Euler solutions. *AIAA Journal of Aircraft*, 47(4):1152–1166, July-August 2010.

[10] I. H. Abbott and A. E. von Doenhoff. *Theory of Wing Sections*. Dover, New York, 1959. ISBN 0-486-60586-8.

[11] RAE2822 airfoil coordinates. http://aerospace.illinois.edu/m-selig/ads/coord/rae2822.dat, July 2013.

[12] J. C. Vassberg, N. A. Harrison, D. L. Roman, and A. Jameson. A systematic study on the impact of dimensionality for a two-dimensional aerodynamic optimization model problem. *AIAA Paper 2011-3176*, 29th AIAA Applied Aerodynamics Conference, Honolulu, HI, June 2011.

[13] F. Bisson, S. Nadarajah, and D. Shi-Dong. Adjoint-based aerodynamic optimization of benchmark problems. *AIAA Paper 2014-0412*, AIAA SciTech Conference, January 2014.

[14] G. Carrier, D. Destarac, A. Dumont, M. Meheut, I. Salah El Din, J. Peter, S. Ben Khelit, J. Brezillon, and M. Pestana. Gradient-based aerodynamic optimization with the *elsa* software. *AIAA Paper 2014-04??*, AIAA SciTech Conference, January 2014.

[15] S. Nadarajah. Drag minimization of the NACA0012 in transonic inviscid flow. ADO DG Sharepoint file: *twodimensional.pdf*, July 2013.

[16] J. C. Vassberg, M. A. DeHaan, S. M. Rivers, and R. A. Wahls. Development of a Common Research Model for applied CFD validation studies. *AIAA Paper 2008-6919*, 26th AIAA Applied Aerodynamics Conference, Honolulu, HI, August 2008.

[17] 5th AIAA CFD Drag Prediction Workshop. http://aaac.larc.nasa.gov/tsab/cfdlarc/aiaa−dpw/, dpw@cessna.textron.com, New Orleans, LA, June 2012.

[18] L. M. Osusky and D. W. Zingg. Lift-constrained drag minimization of a wing allowing section and twist variation with flow governed by the Reynolds-Averaged Navier-Stokes equations. ADO DG Sharepoint file: *crm_study2July2013.pdf*, July 2013.

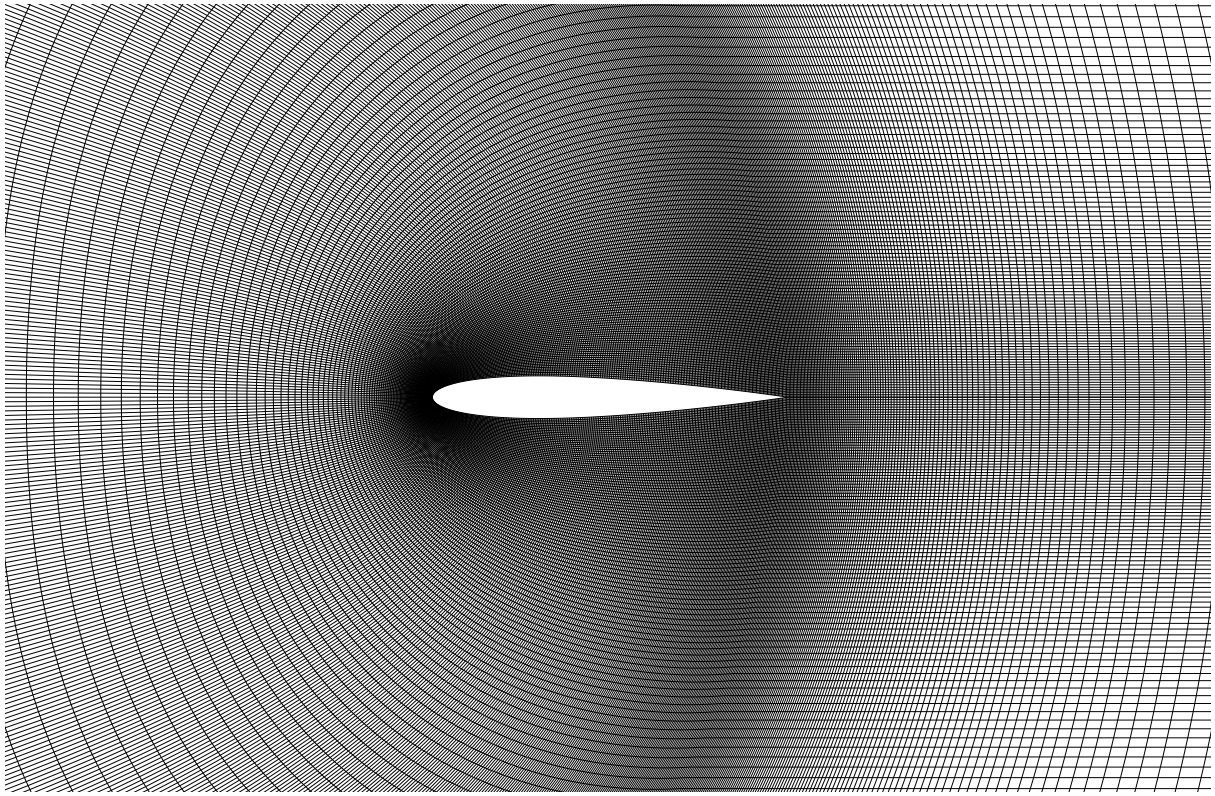


Figure 1: Close-up view of the SYN83 C-mesh about the NACA0012-ADO.

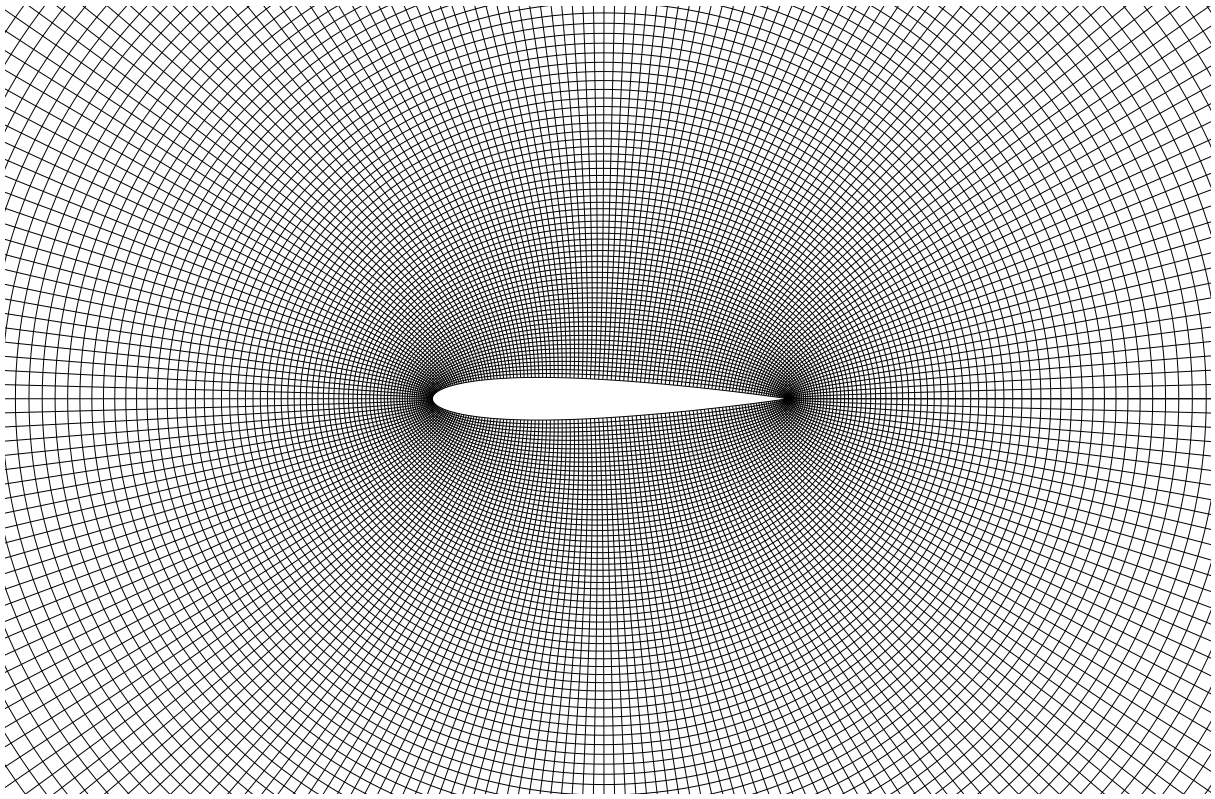


Figure 2: Close-up view of the NACA0012-ADO 256x256 O-mesh.

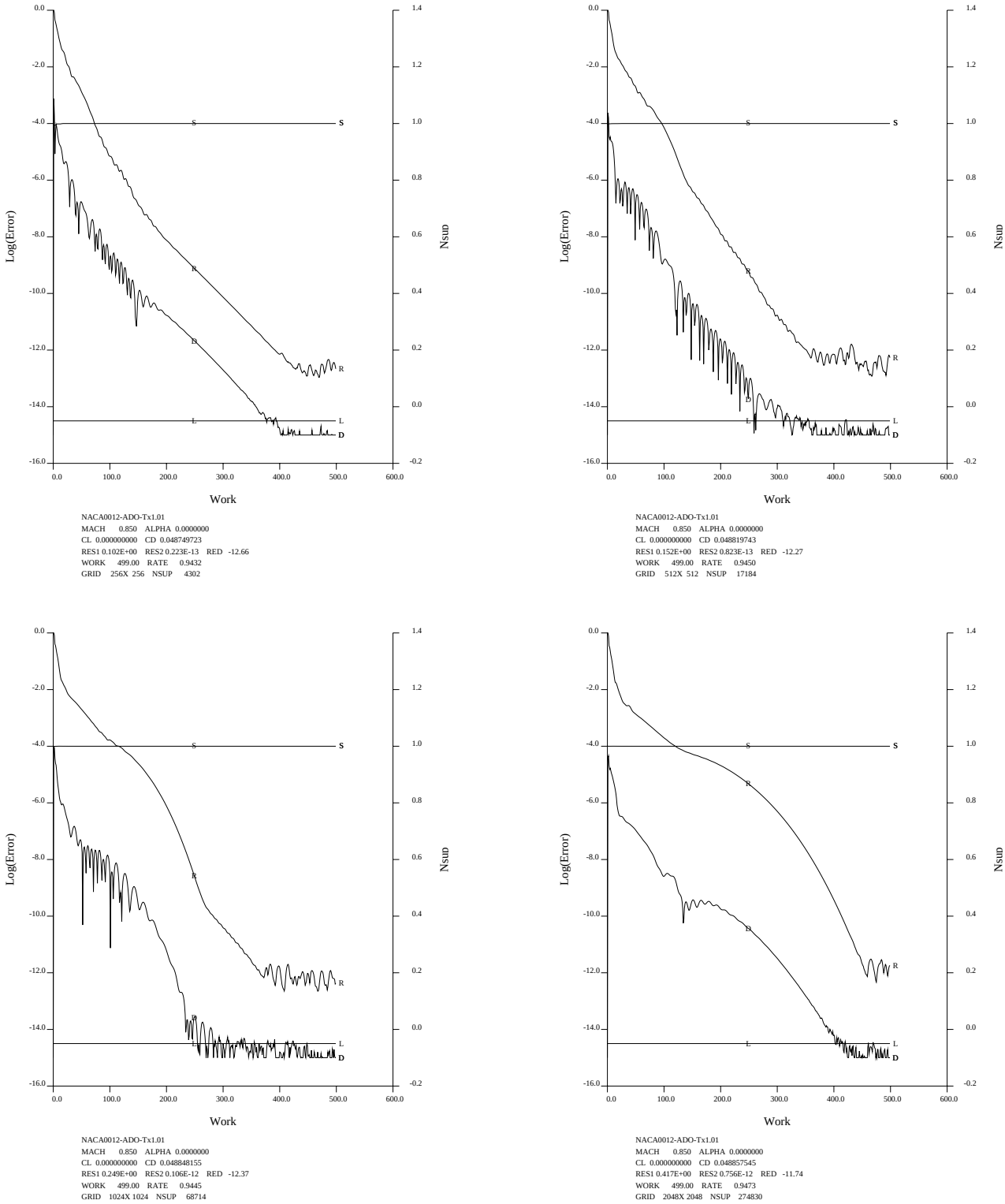


Figure 3: FLO82 Convergence Histories at $M = 0.85$, $\alpha = 0°$.

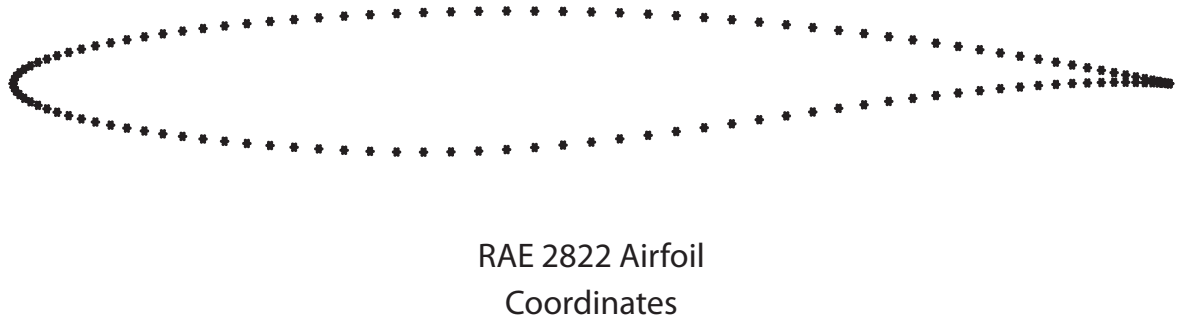


Figure 4: RAE2822 Airfoil Discrete Coordinates.

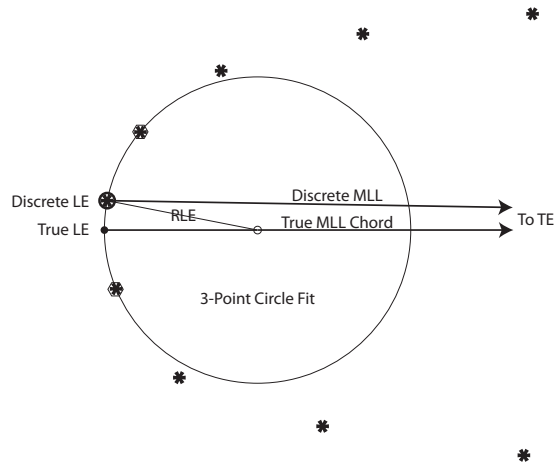


Figure 5: Estimate of an Airfoil's True Leading-Edge Point.

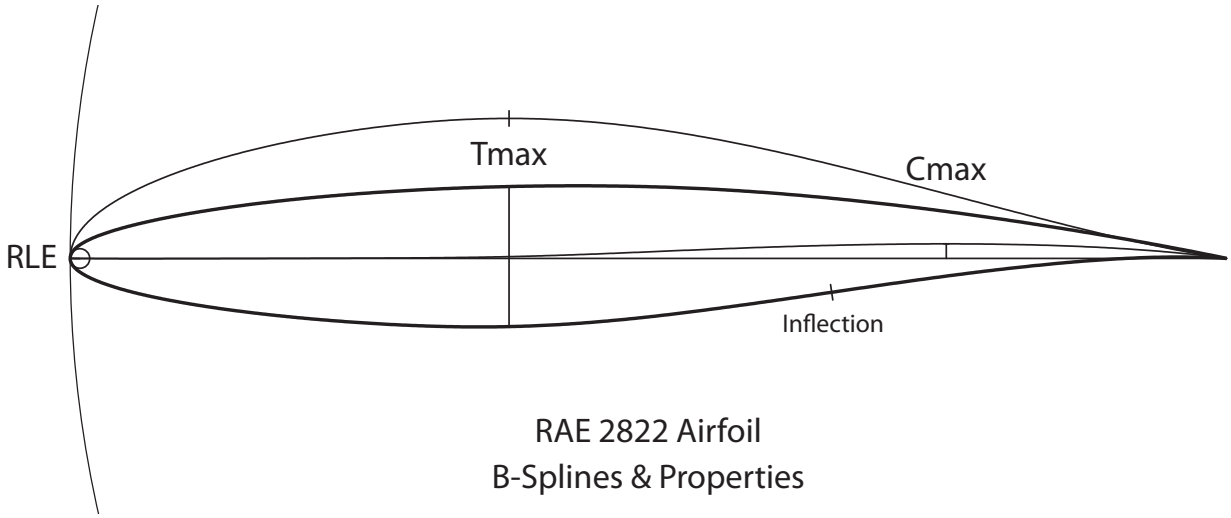


Figure 6: RAE2822 Least-Squares-Fit B-Splines and Geometric Properties.

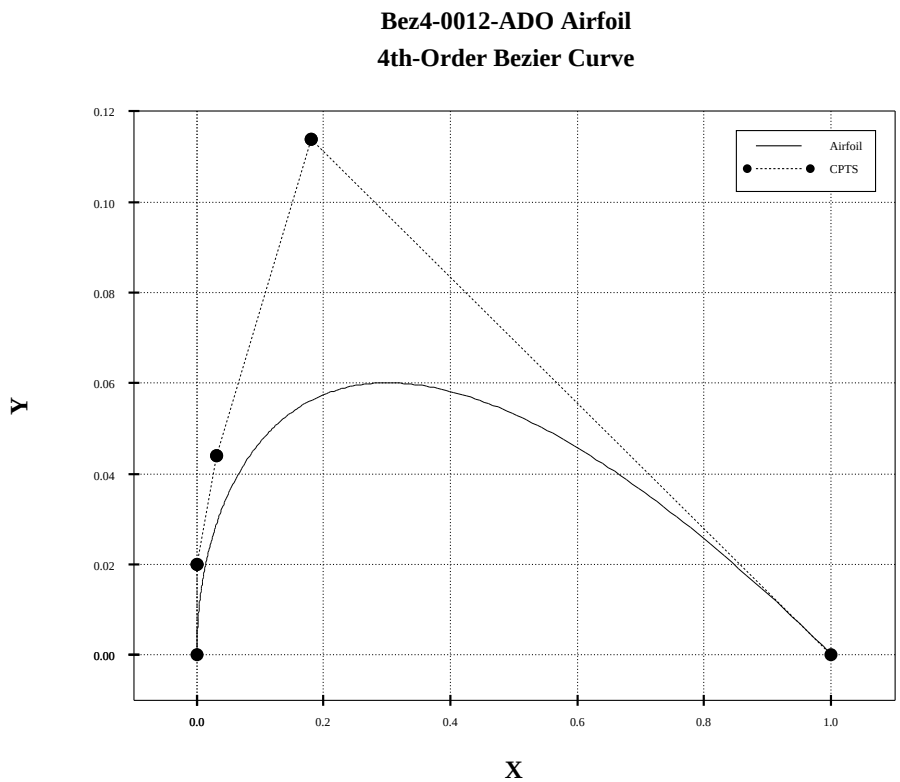


Figure 7: Bez4-0012-ADO Airfoil and 4th-Order Bezier Control Points.

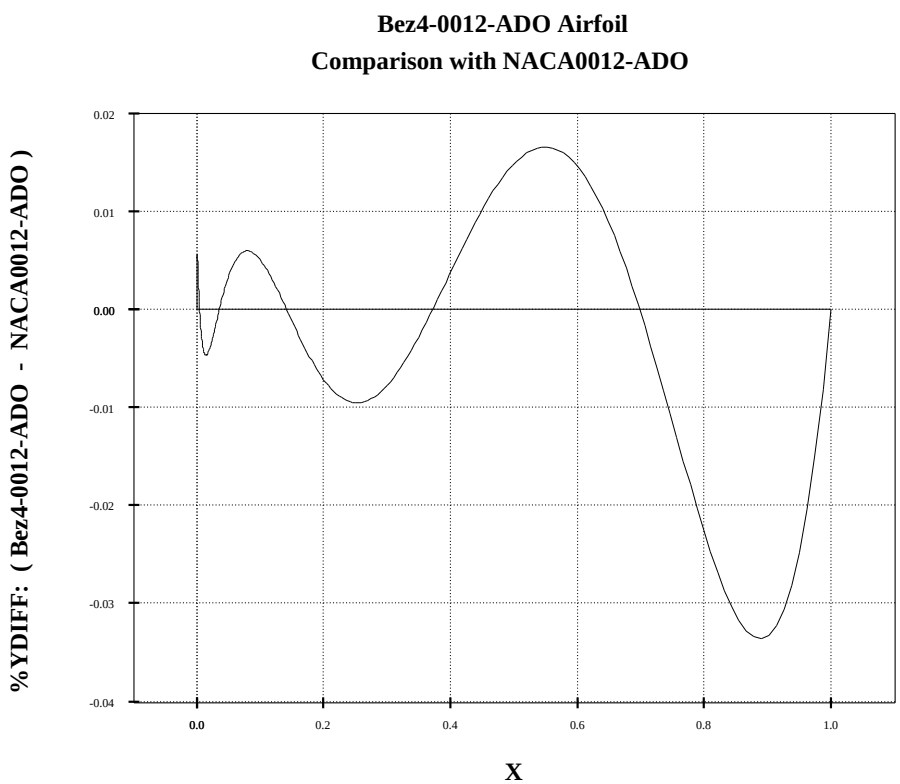


Figure 8: Difference between Bez4-0012-ADO and NACA0012-ADO Airfoils.

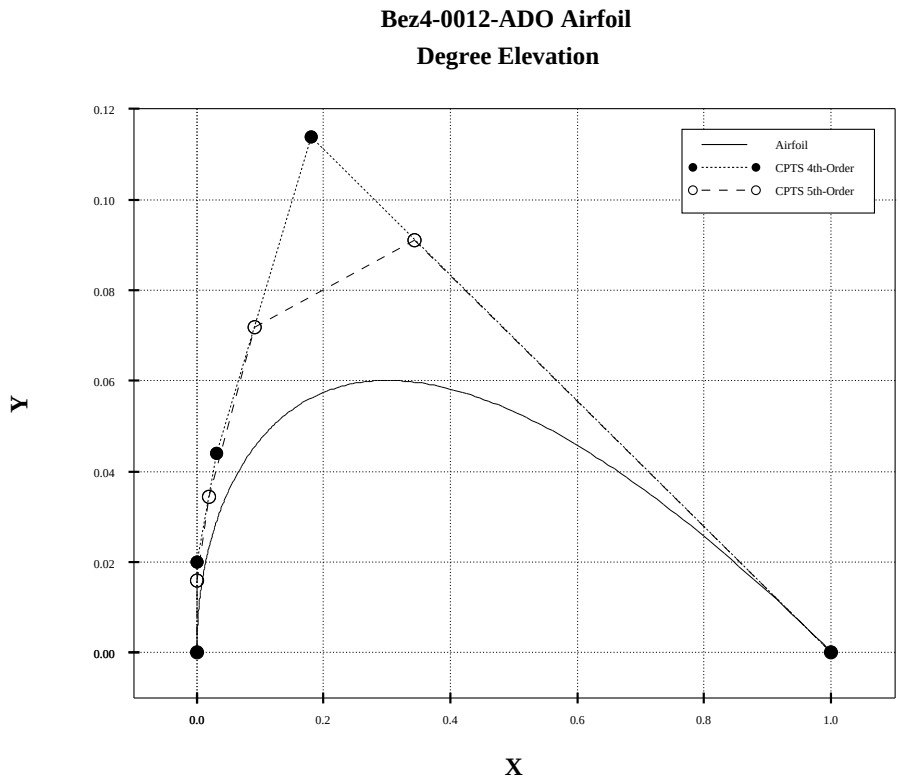


Figure 9: Degree Elevation of Bez4-0012-ADO Curve from 4th to 5th Order.

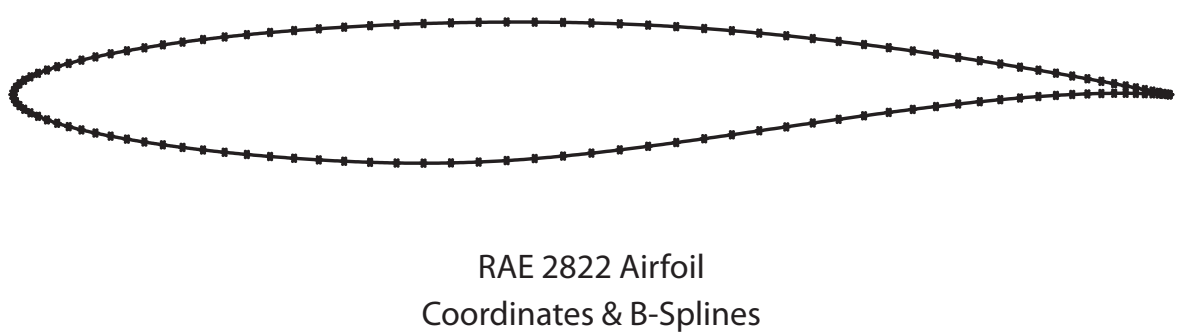
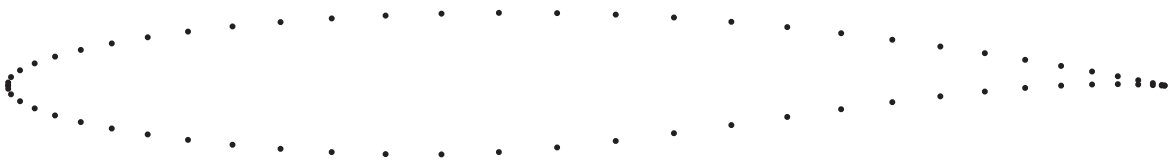
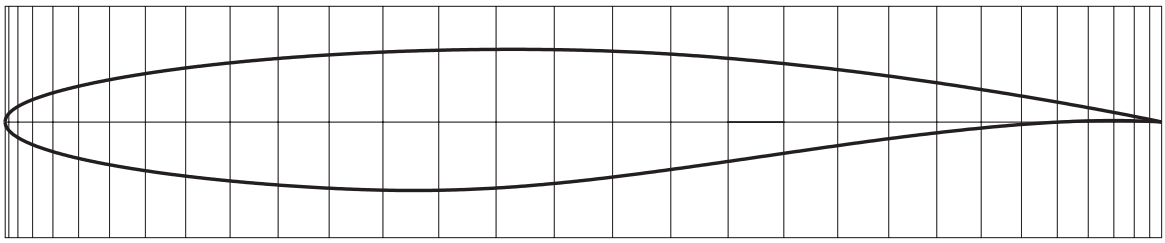


Figure 10: RAE2822 Coordinates with Least-Squares-Fit B-Splines.



RAE 2822 Airfoil
Control Points

Figure 11: RAE2822 Airfoil Control Points.



RAE 2822 Airfoil
Curve Segments

Figure 12: RAE2822 B-Spline Curve Segments.

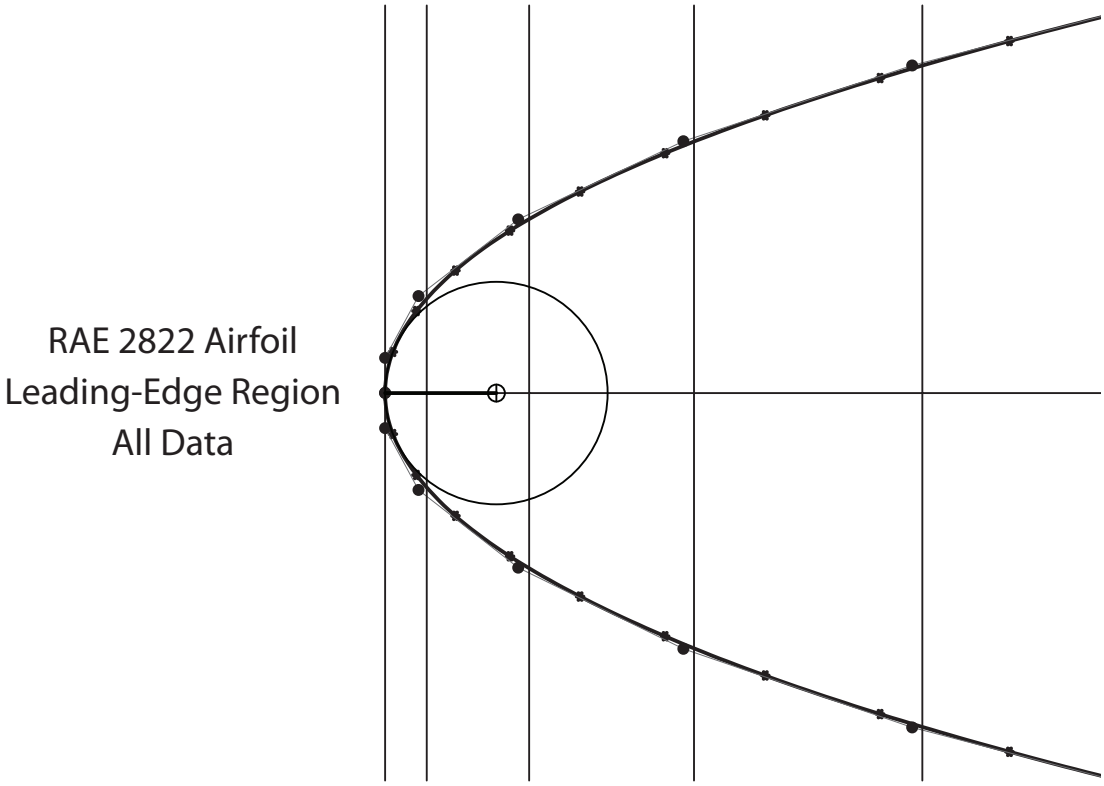


Figure 13: RAE2822 Coordinates, B-Splines and Leading-Edge Radius.

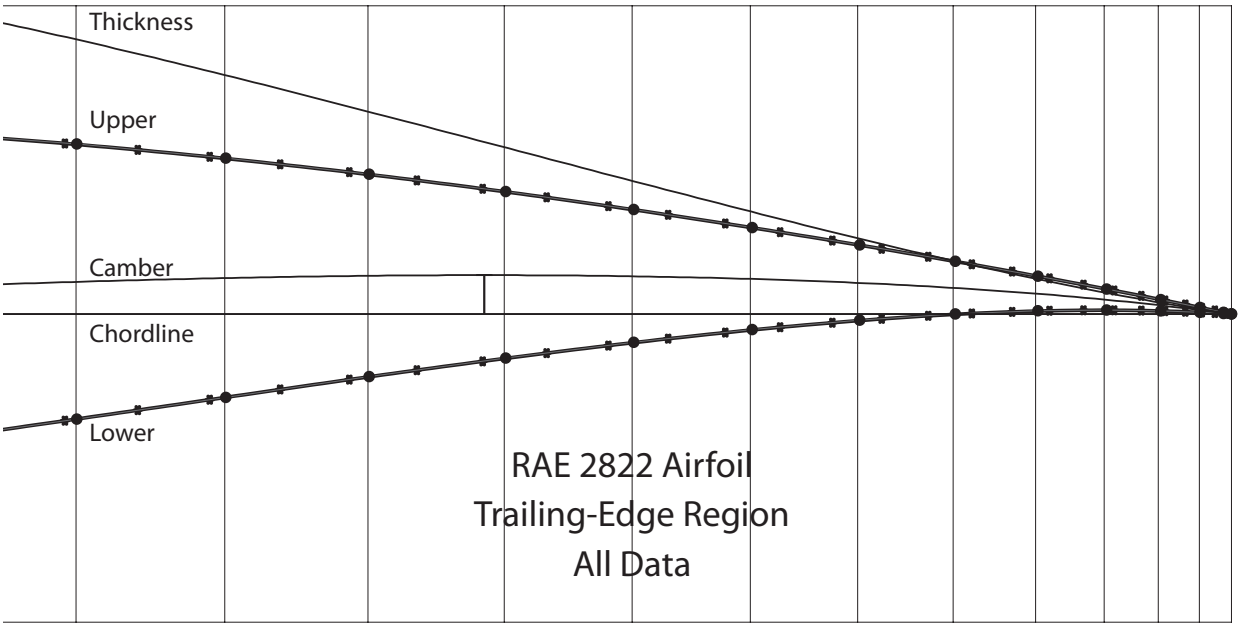


Figure 14: RAE2822 Coordinates, B-Splines, Thickness and Camber Lines near Trailing Edge.

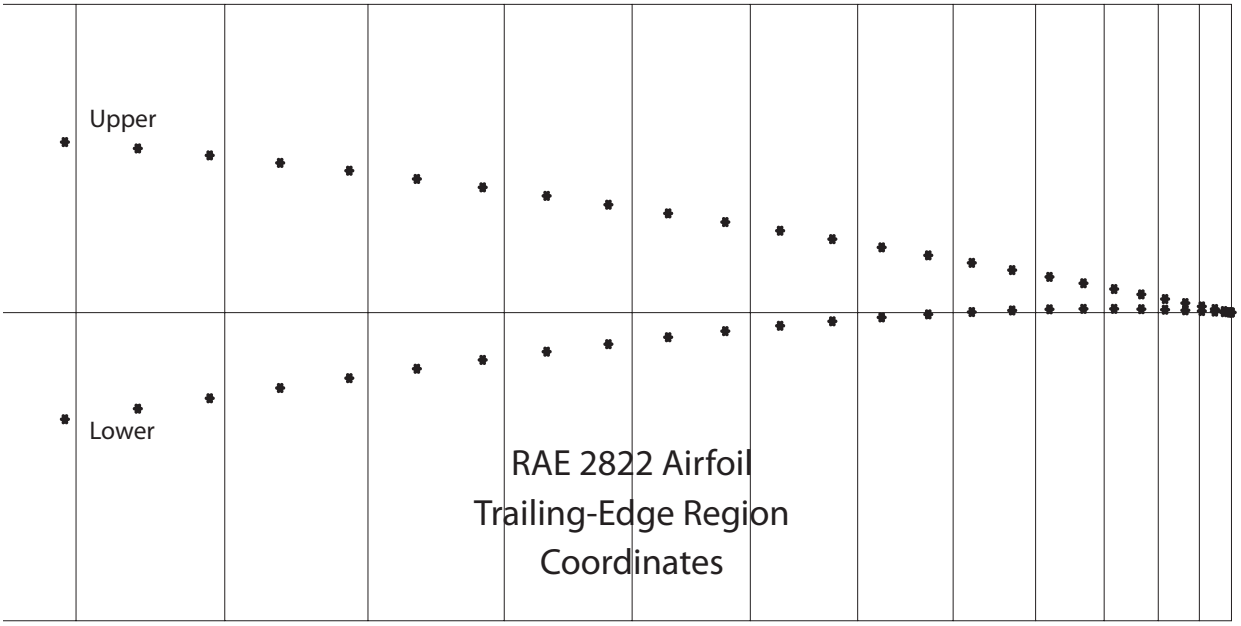


Figure 15: RAE2822 Coordinates and Curve-Segment Grid near Trailing Edge.

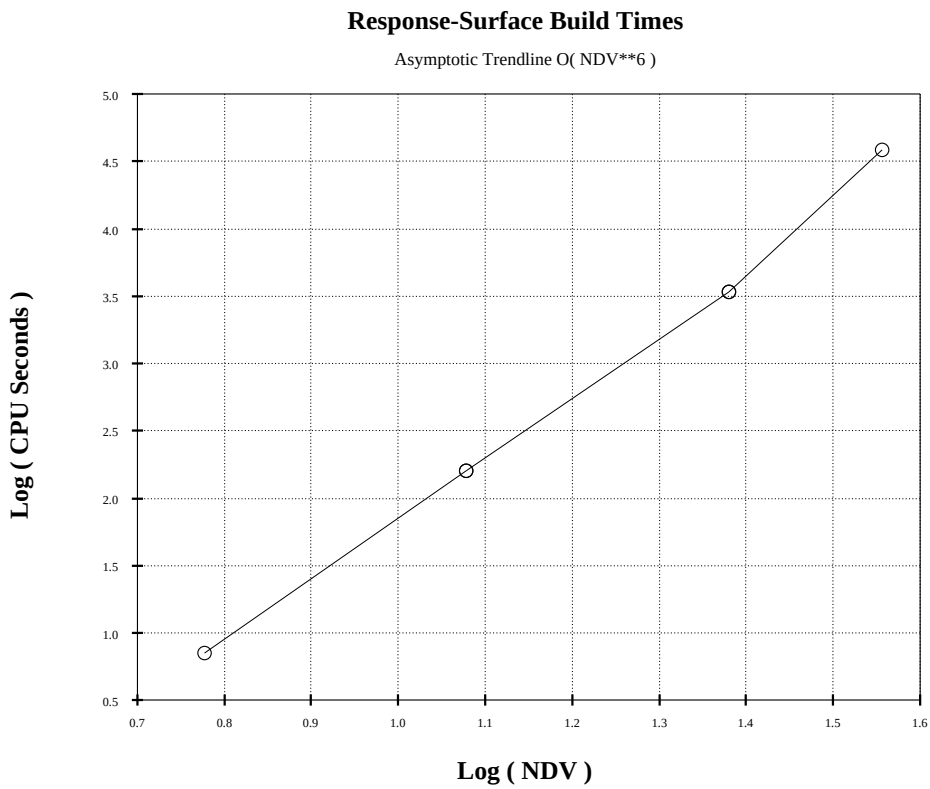


Figure 16: MDOPT Response-Surface Build-Time Trendline.

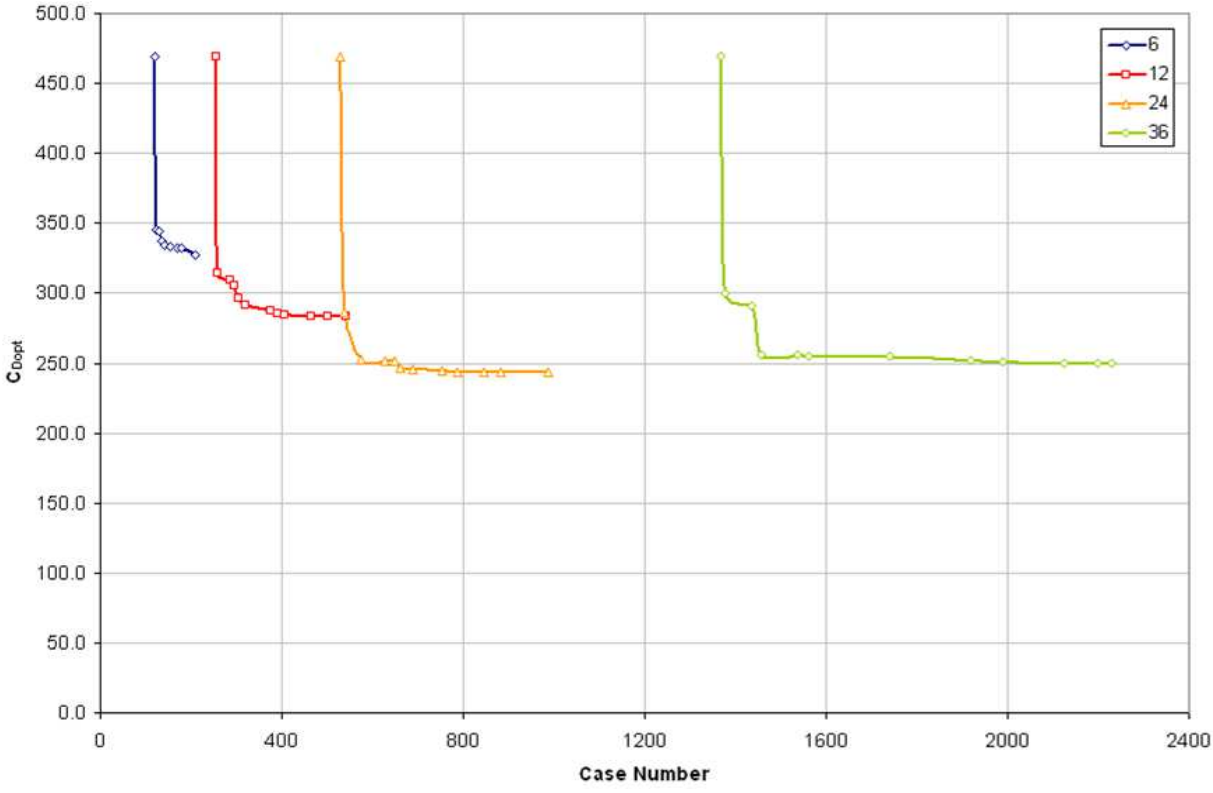


Figure 17: MDOPT Drag Histories in Phase-I.

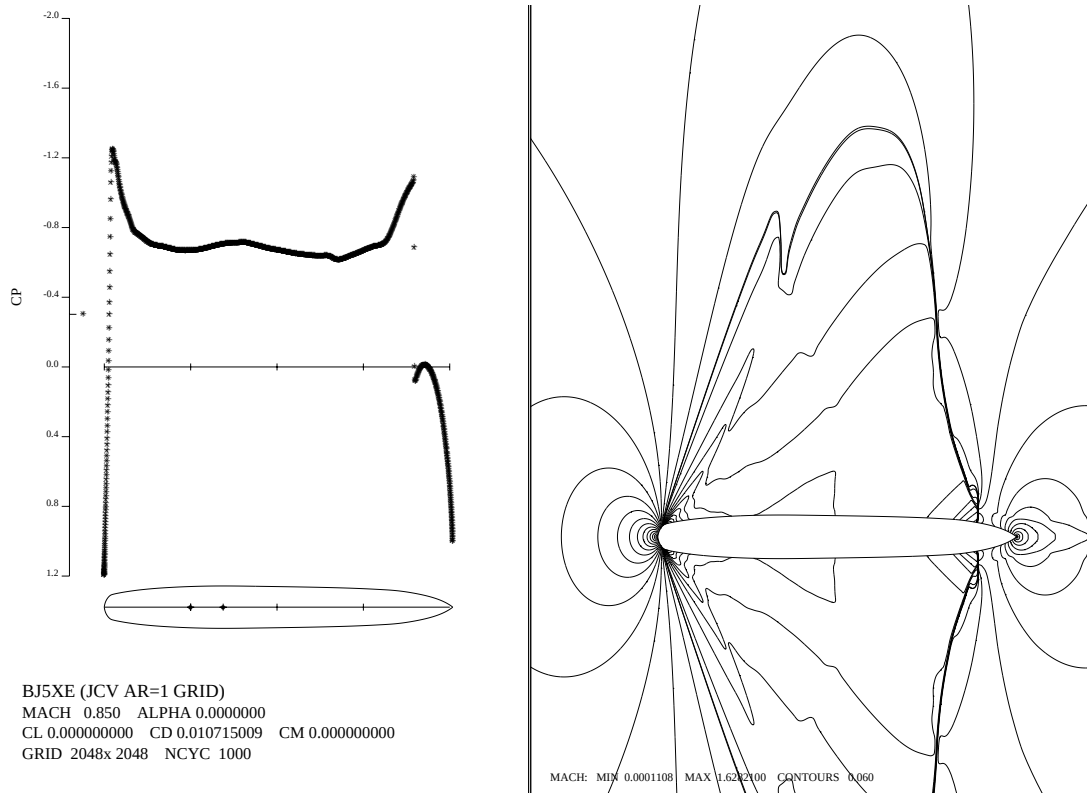


Figure 18: FLO82 Solution for BJ5XE Airfoil at $M = 0.85$, $\alpha = 0°$.

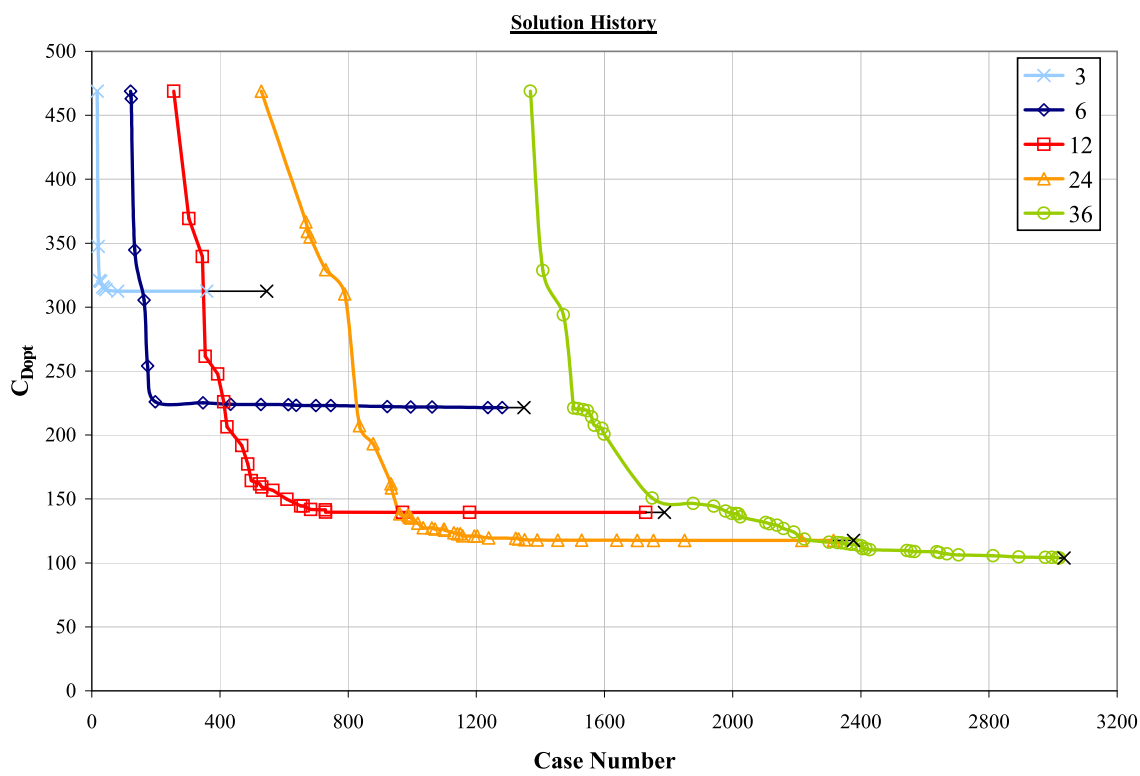


Figure 19: MDOPT Drag Reduction Histories for Various Design Spaces in Phase-III.

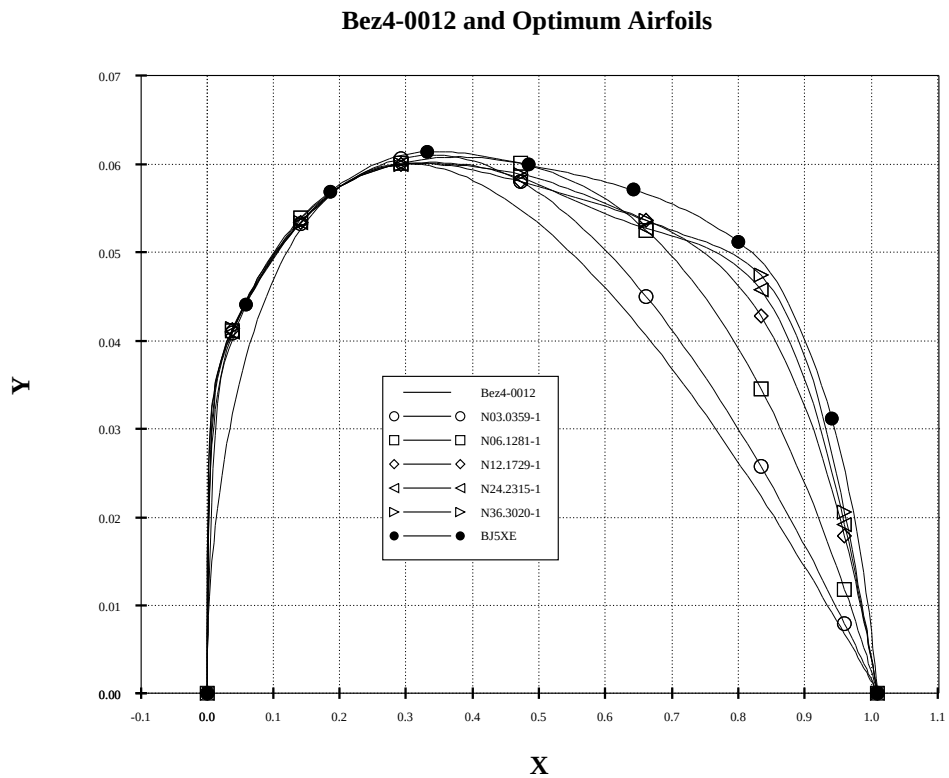


Figure 20: Comparison of Bez4-0012, MDOPT Optimums and BJ5XE Airfoils.

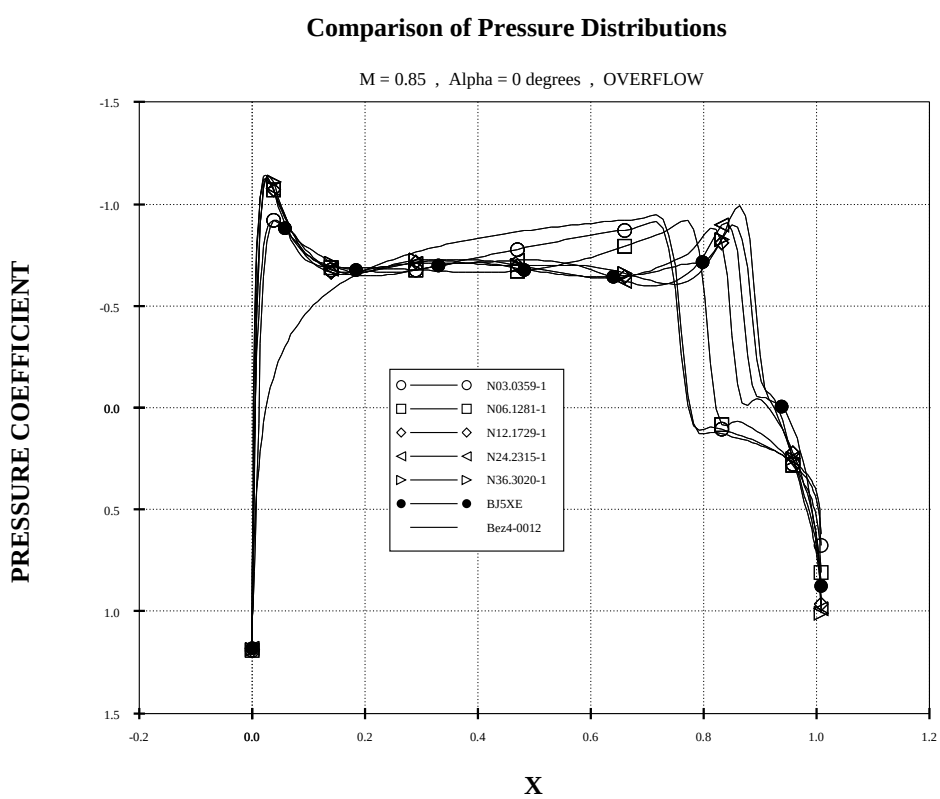


Figure 21: Comparison of Pressure Distributions for Bez4-0012, MDOPT Optimums and BJ5XE Airfoils.

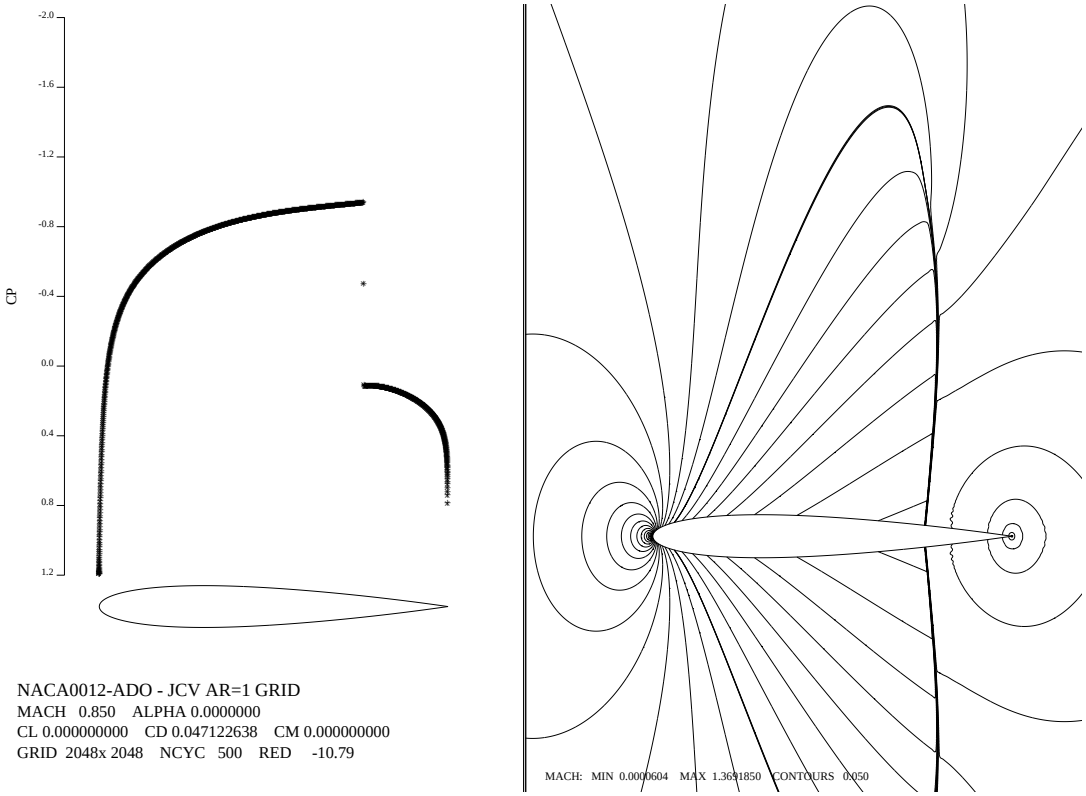


Figure 22: FLO82 Solution for NACA0012-ADO Airfoil at $M = 0.85$, $\alpha = 0°$.

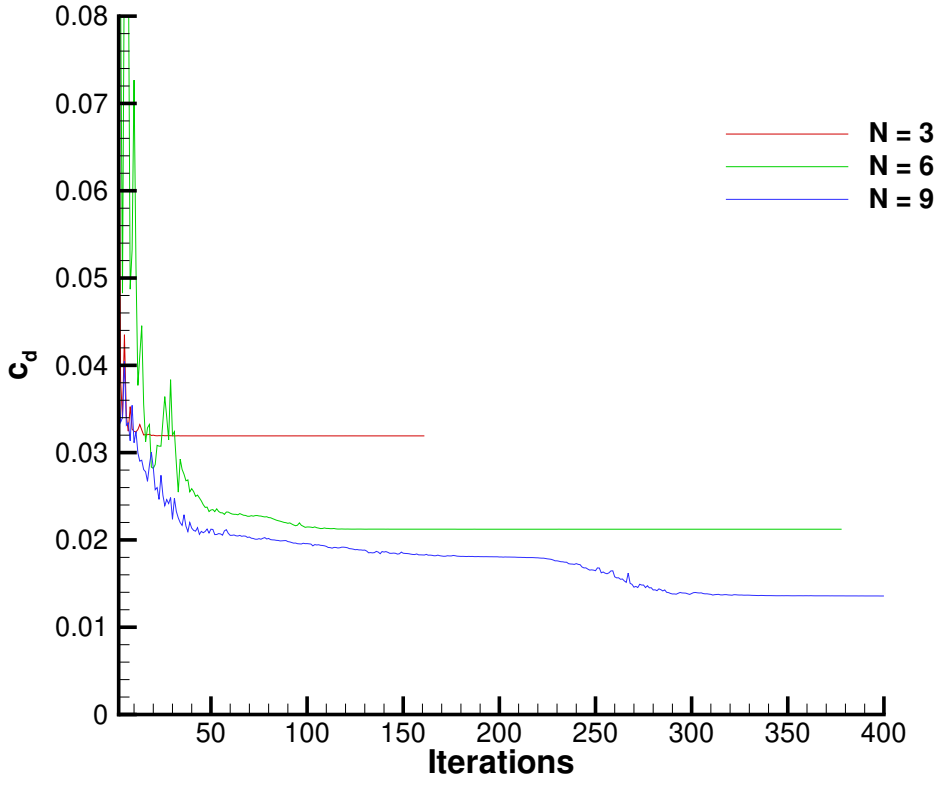


Figure 23: CMA-ES Convergence Histories for [3,6,9]-Spaces.

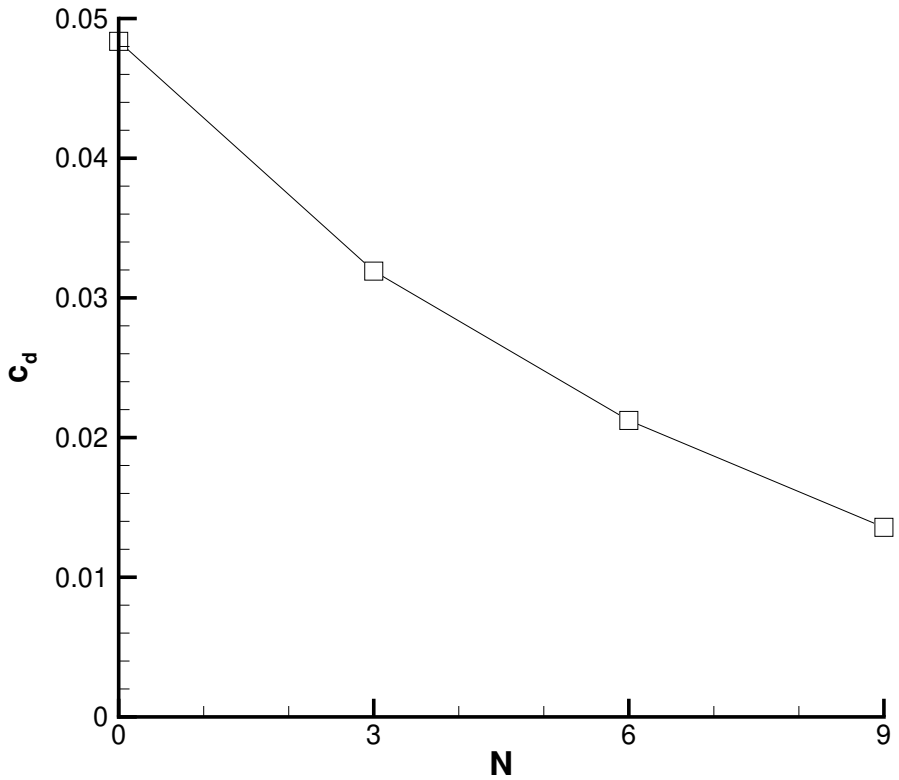


Figure 24: Optimum Drag Levels Found Within Bezier Design-Space Family.

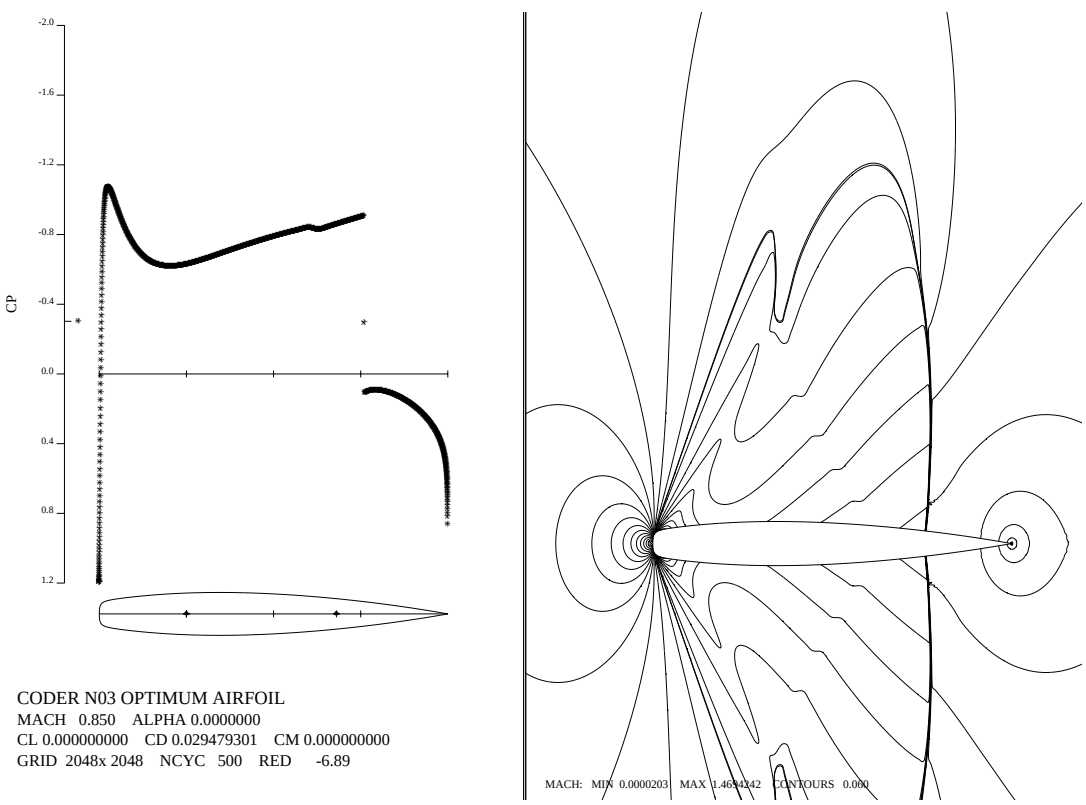


Figure 25: FLO82 Solution for CMAES-N03 Airfoil at $M = 0.85$, $\alpha = 0°$.

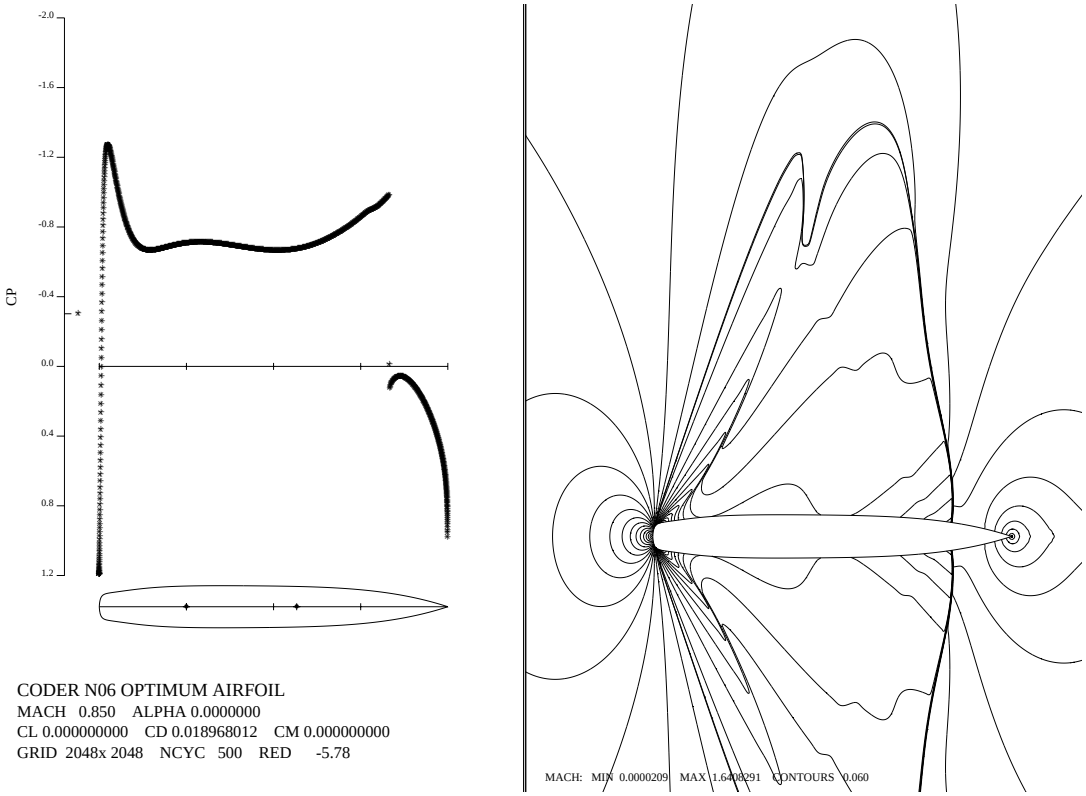


Figure 26: FLO82 Solution for CMAES-N06 Airfoil at $M = 0.85$, $\alpha = 0°$.

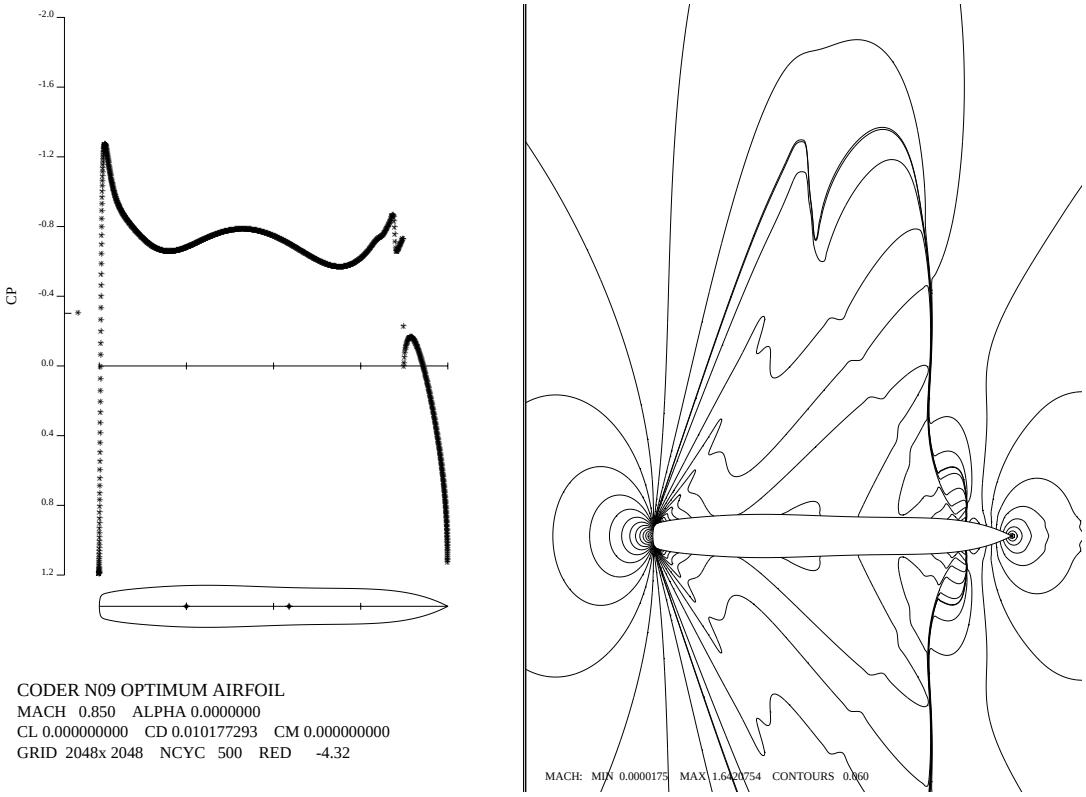


Figure 27: FLO82 Solution for CMAES-N09 Airfoil at $M = 0.85$, $\alpha = 0°$.

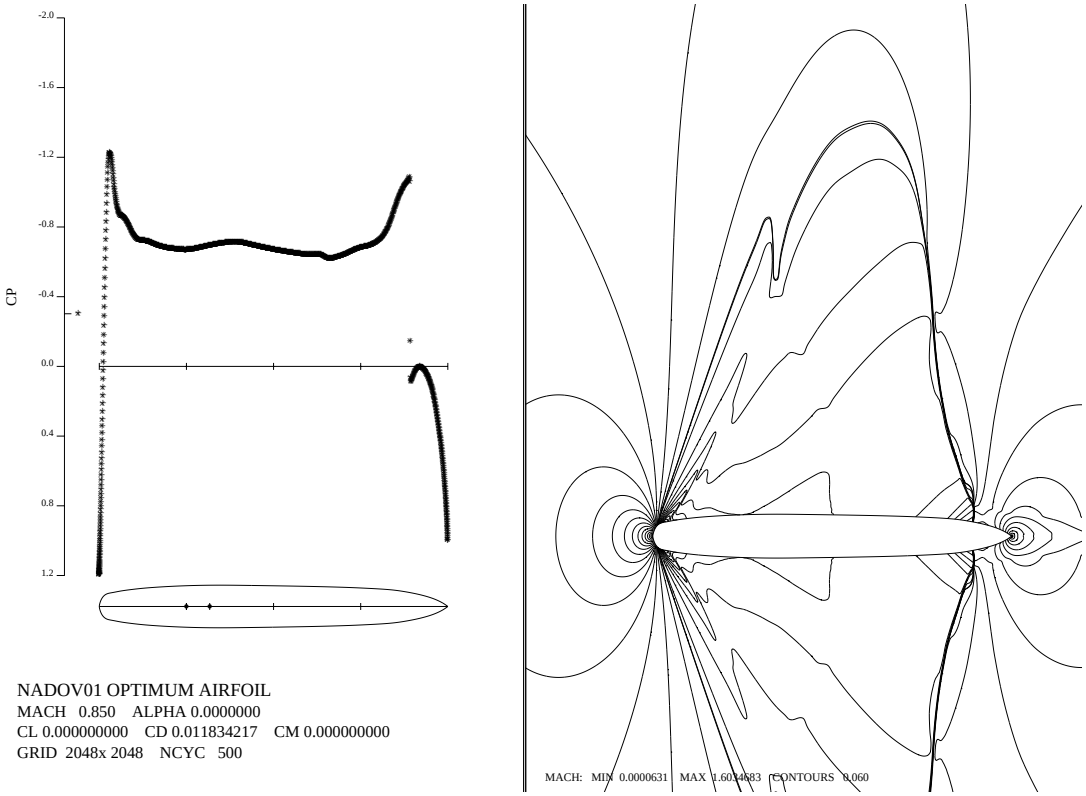


Figure 28: FLO82 Solution for SYN83-NADOV01 Airfoil at $M = 0.85$, $\alpha = 0°$.

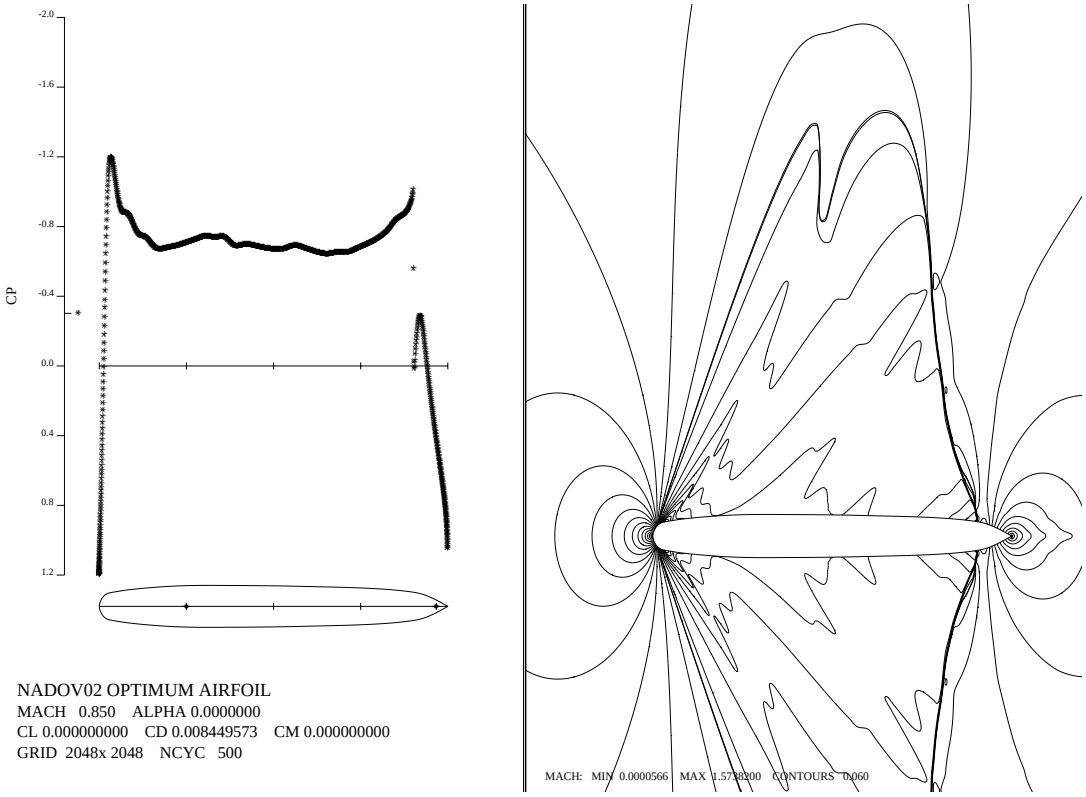


Figure 29: FLO82 Solution for SYN83-NADOV02 Airfoil at $M = 0.85$, $\alpha = 0°$.

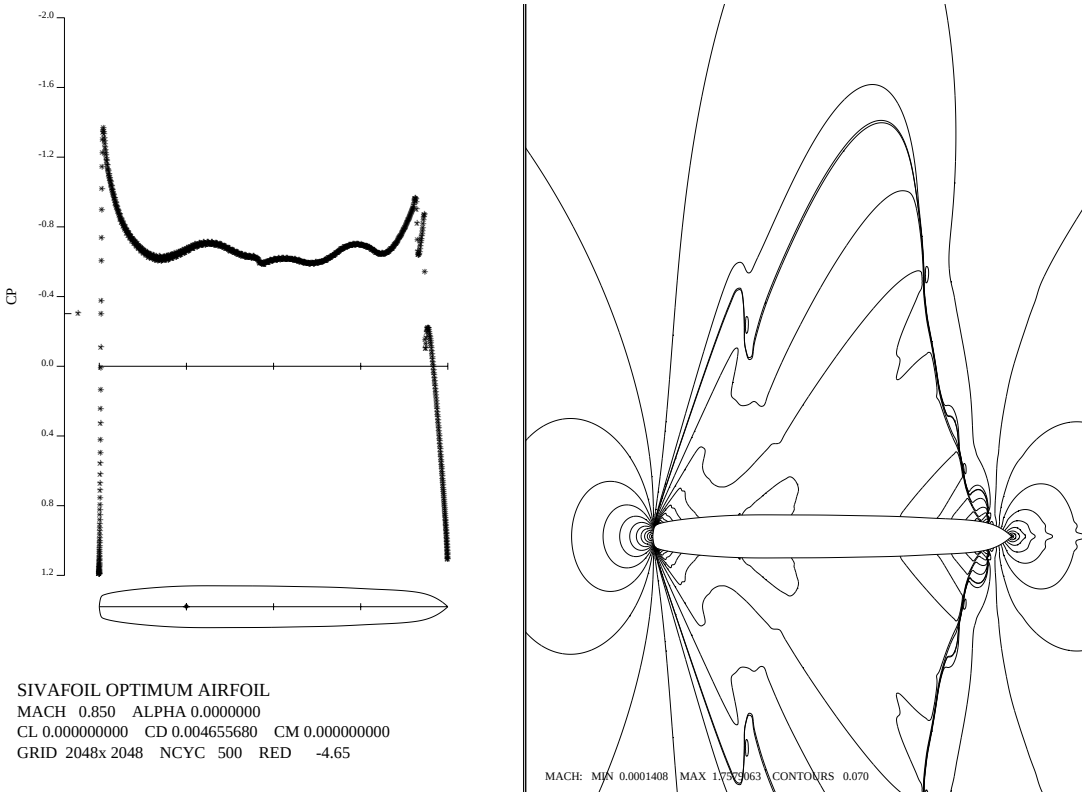


Figure 30: FLO82 Solution for the Bisson-Nadarajah Airfoil at $M = 0.85$, $\alpha = 0°$.

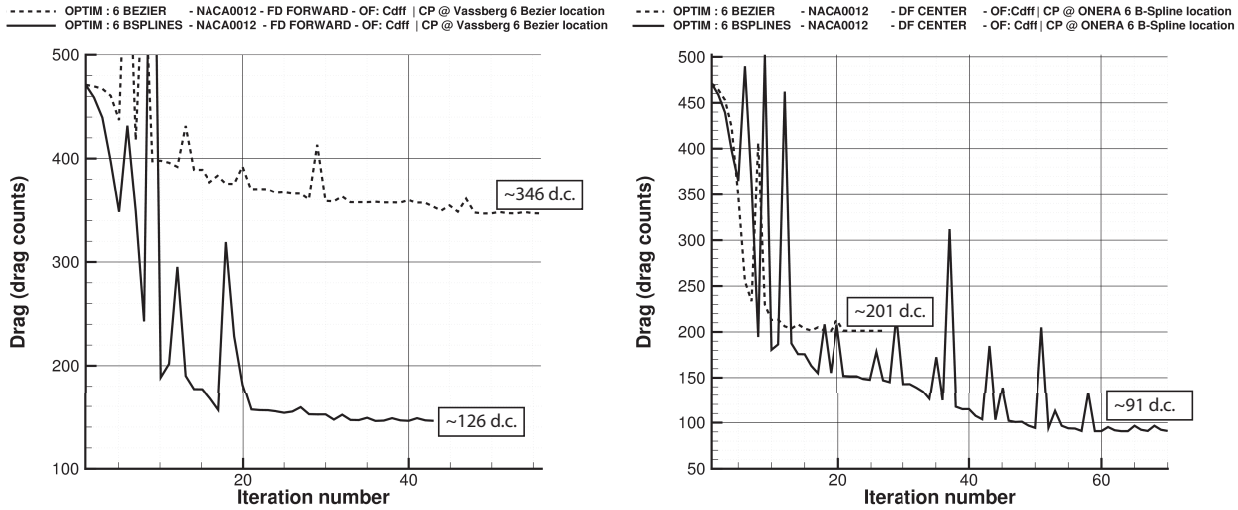


Figure 31: Carrier, et.al Results: Bezier and B-Spline Comparisons.

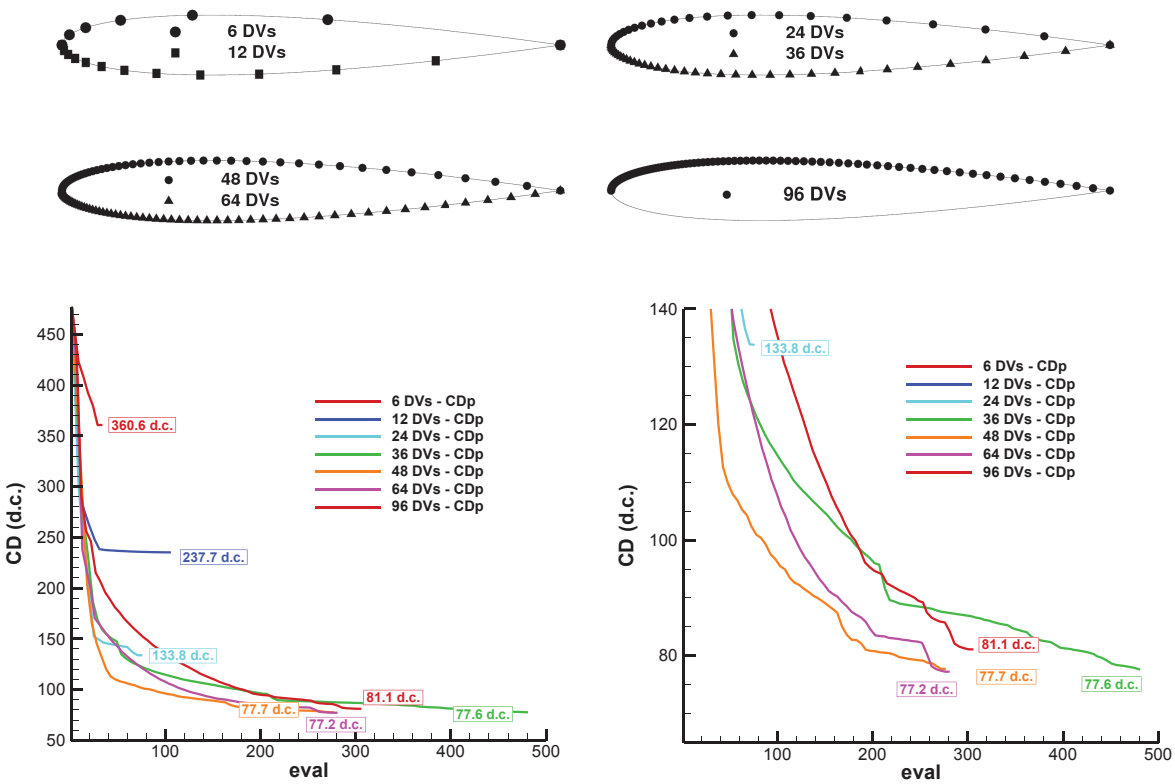


Figure 32: Carrier, et.al Results: Systematic Study showing Drag-Convergence Comparisons.

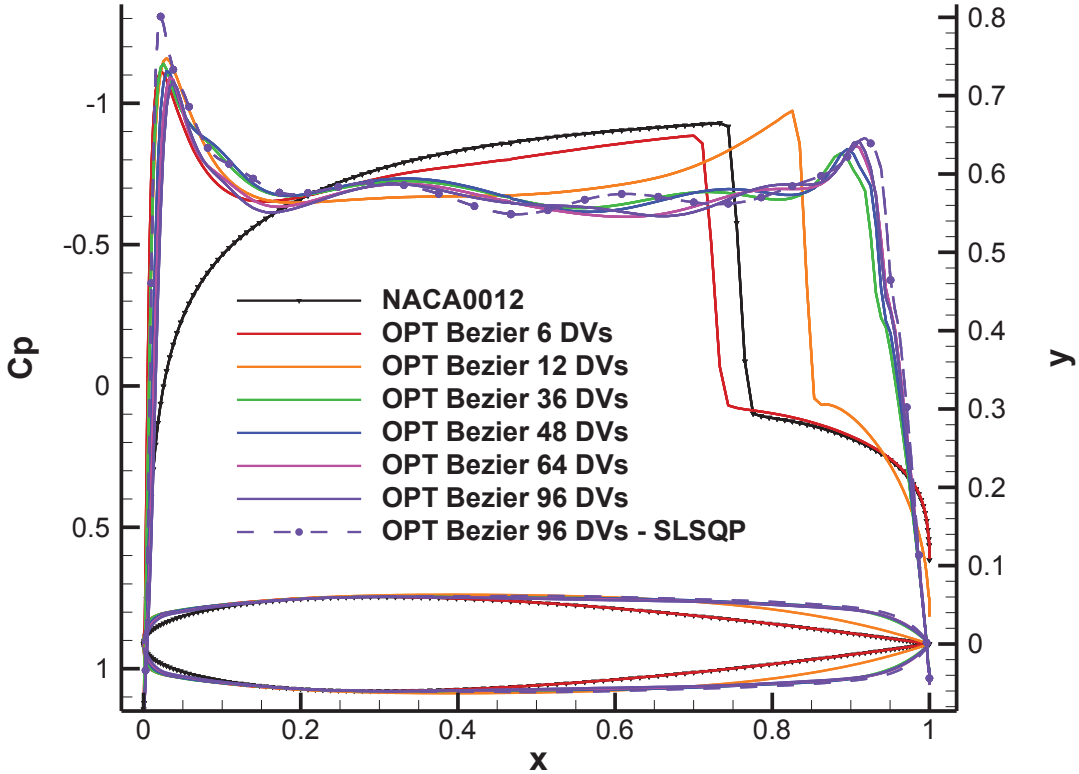


Figure 33: Carrier, et.al Results: Systematic Study showing Geometry and Pressure Comparisons

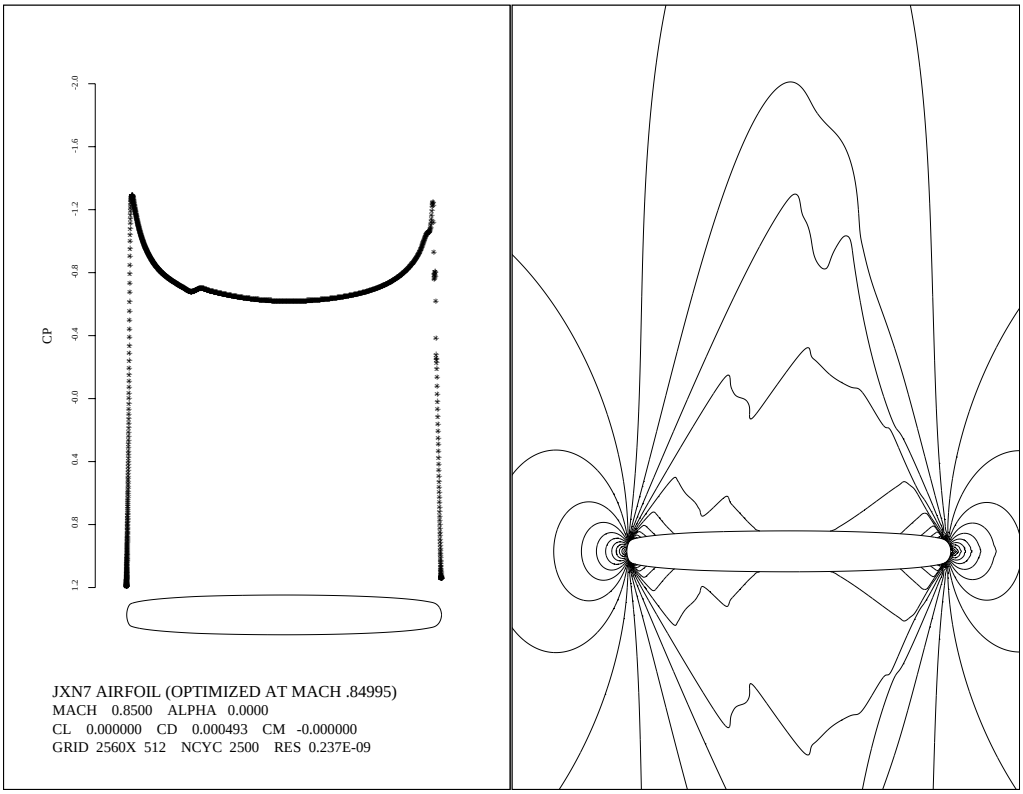


Figure 34: Recent SYN82 O-Mesh Results: Geometry, Pressures and Mach Contours

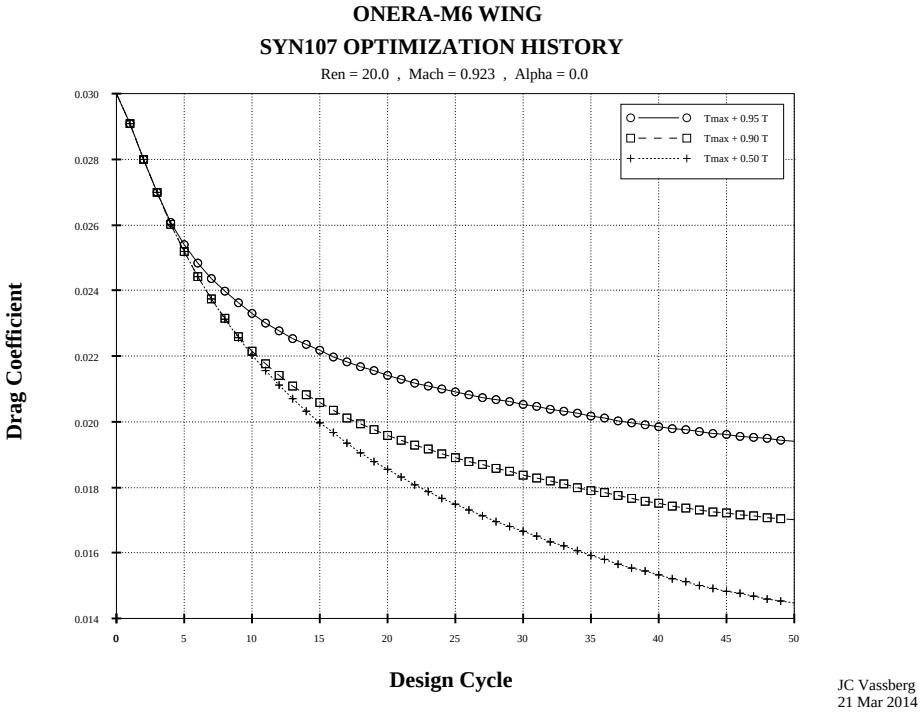


Figure 35: SYN107 Design History of Drag on the ONERA-M6 Wing.

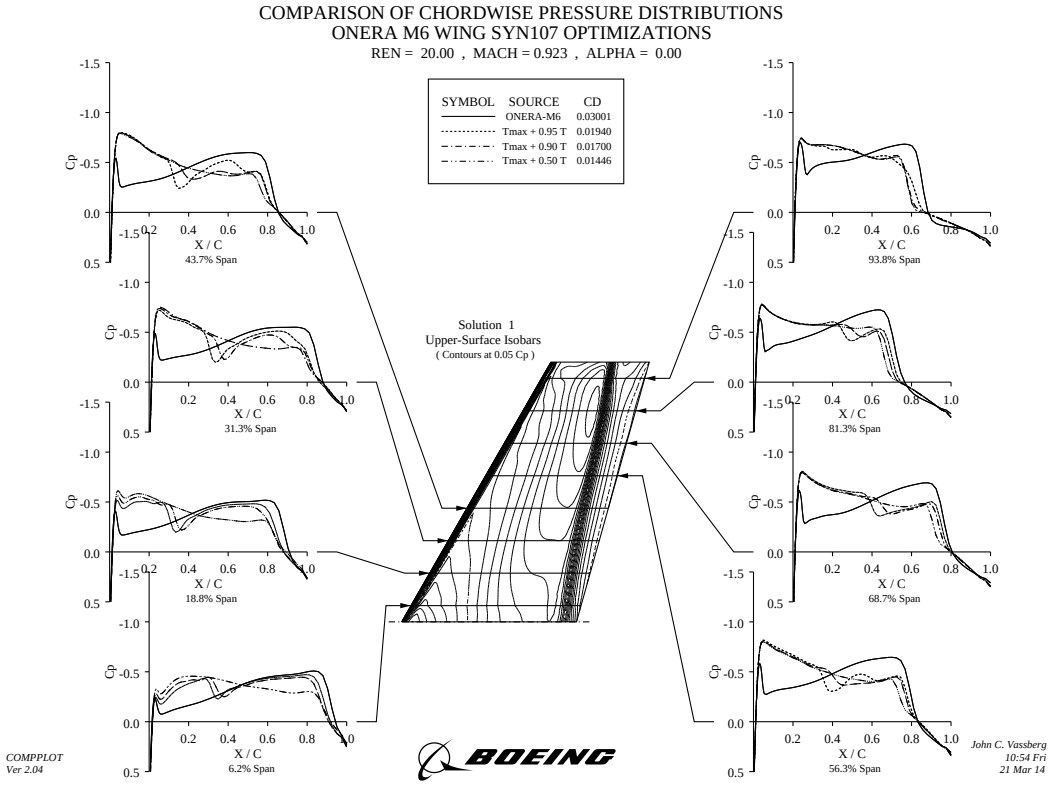


Figure 36: Comparison of ONERA-M6 Baseline and Optimized Wings Surface Pressures.

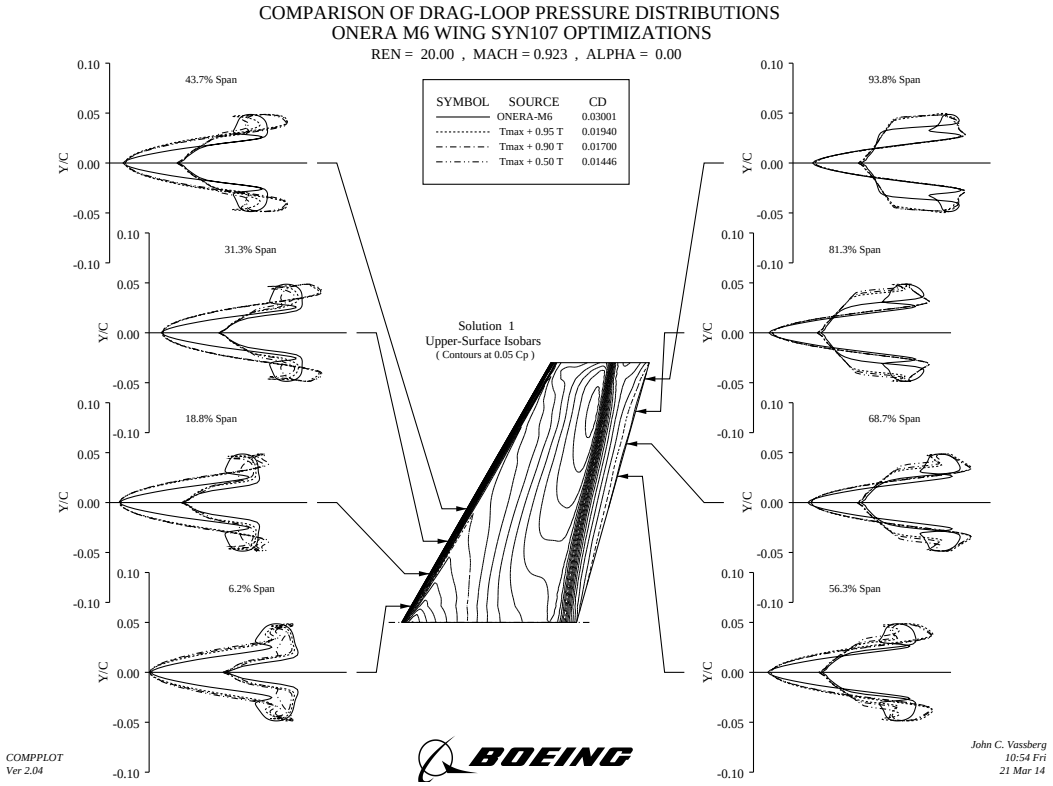


Figure 37: Comparison of ONERA-M6 Baseline and Optimized Wings Drag Loops.

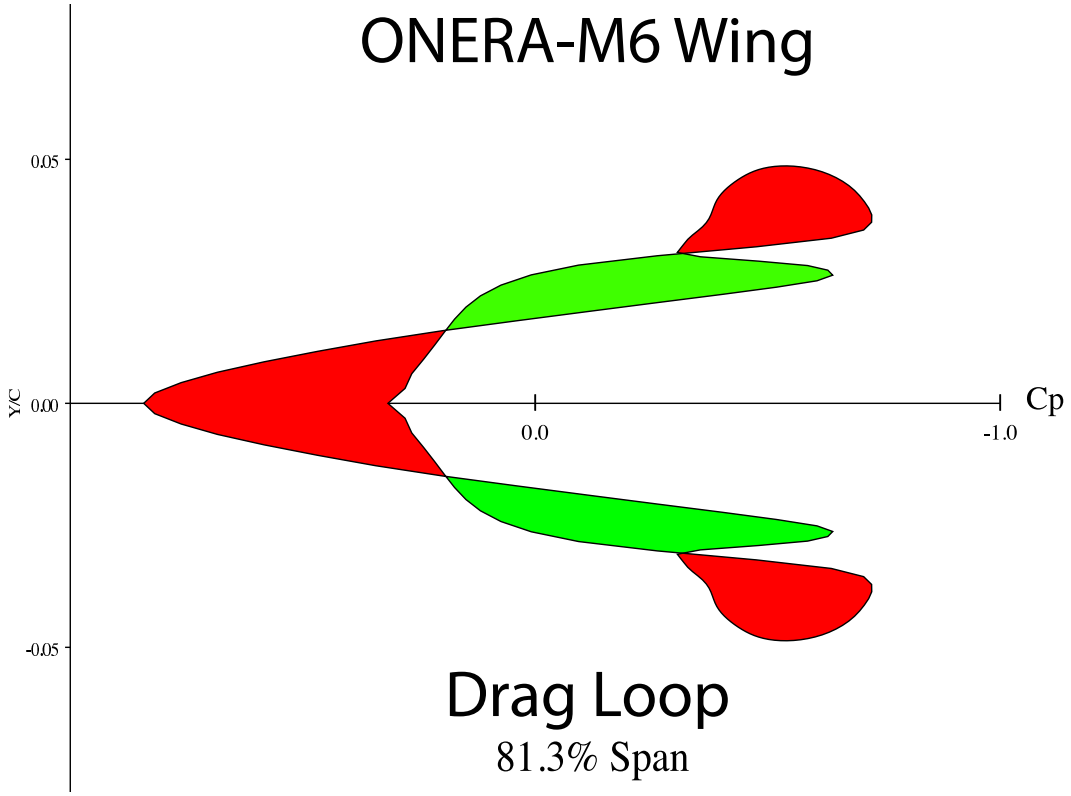


Figure 38: ONERA-M6 Baseline Sectional Drag Loop at $\eta = 0.813$.

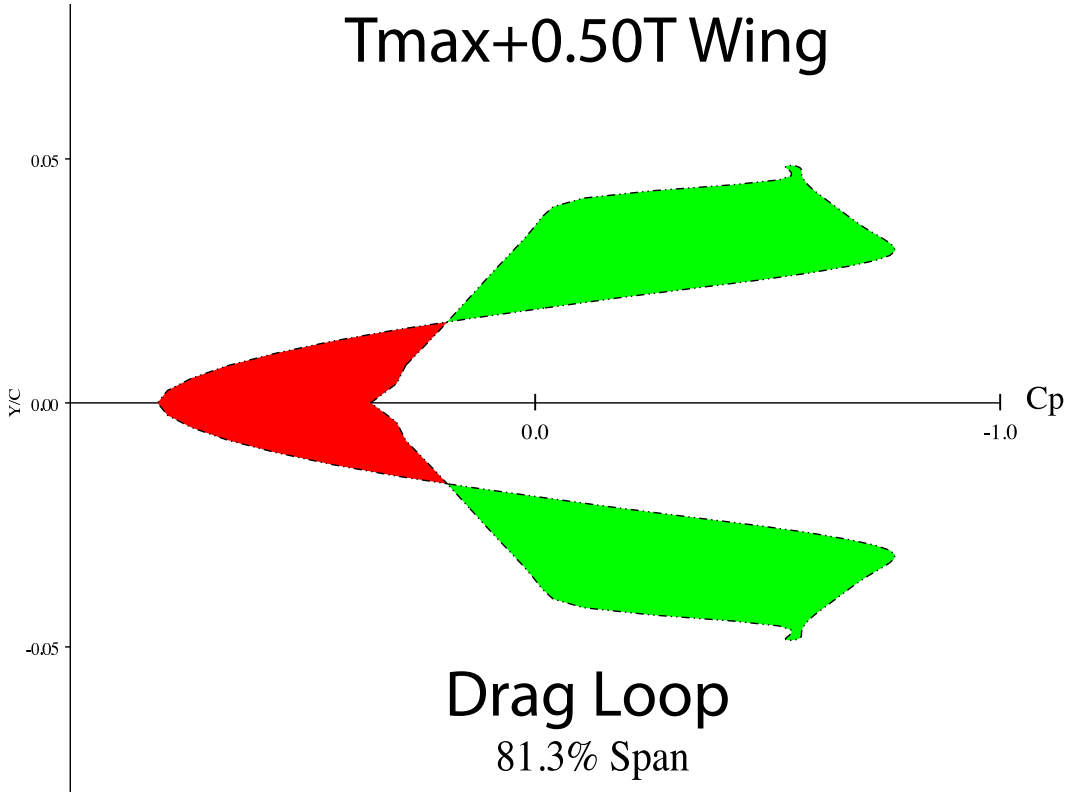


Figure 39: ONERA-M6 Optimum Sectional Drag Loop at $\eta = 0.813$.

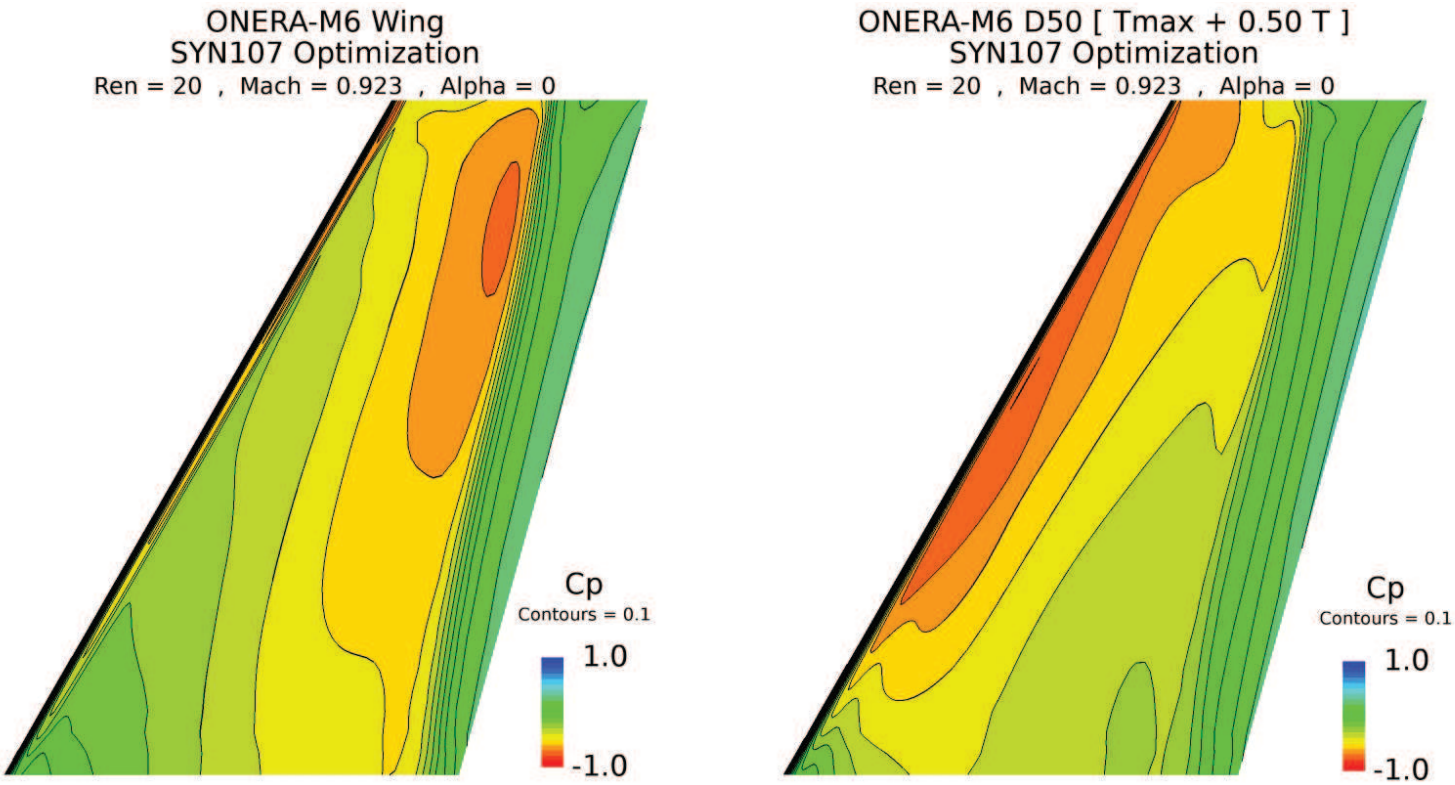


Figure 40: Comparison of ONERA-M6 Baseline and Optimized Wings Isobars.

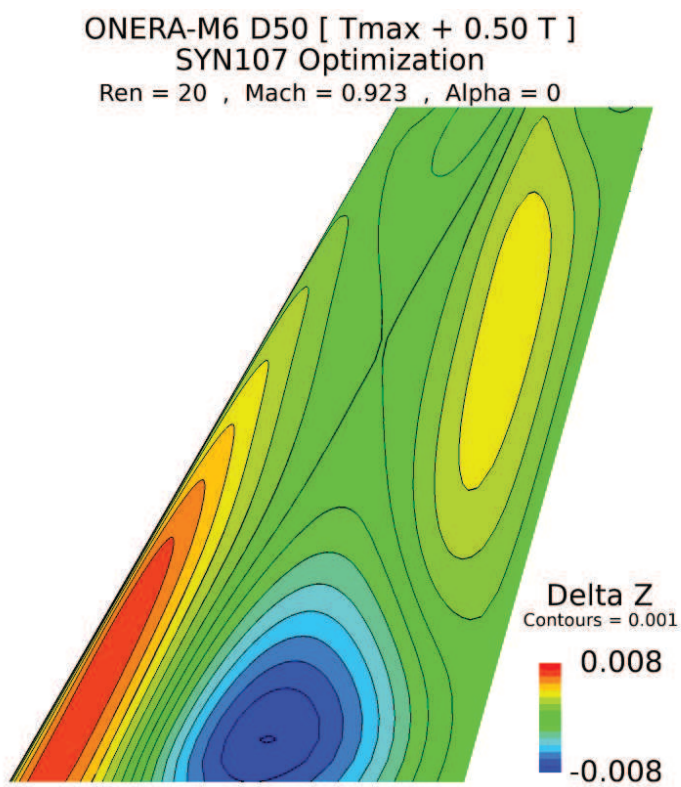


Figure 41: Surface ΔZ of ONERA-M6 Wing (Optimized - Baseline).

Airfoil Technology

Airfoil Geometry -- Camber & Thickness Distributions

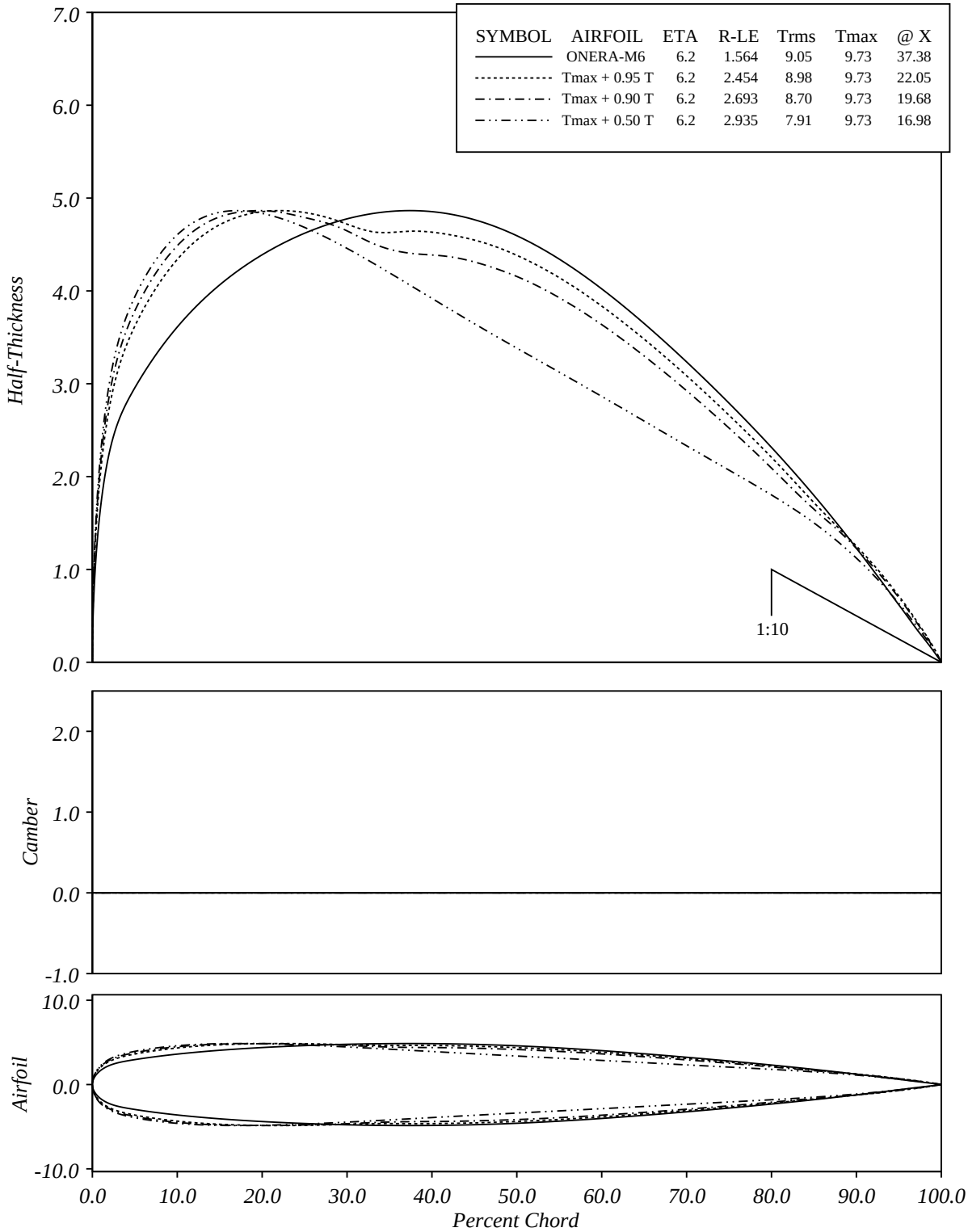


Figure 42: Comparison of ONERA-M6 Airfoil Sections at $\eta = 0.06$.

Airfoil Technology

Airfoil Geometry -- Camber & Thickness Distributions

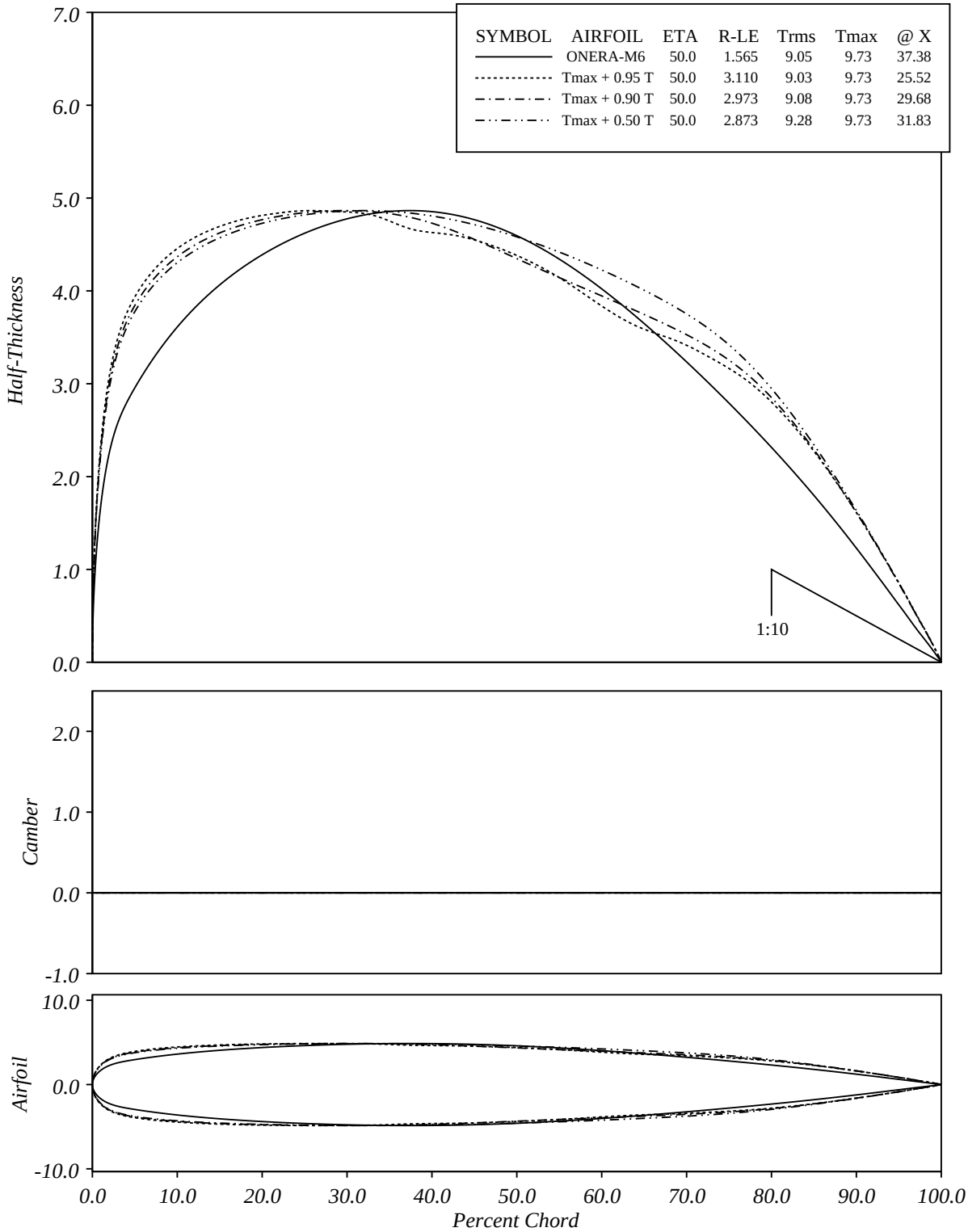


Figure 43: Comparison of ONERA-M6 Airfoil Sections at $\eta = 0.50$.

Airfoil Technology

Airfoil Geometry -- Camber & Thickness Distributions

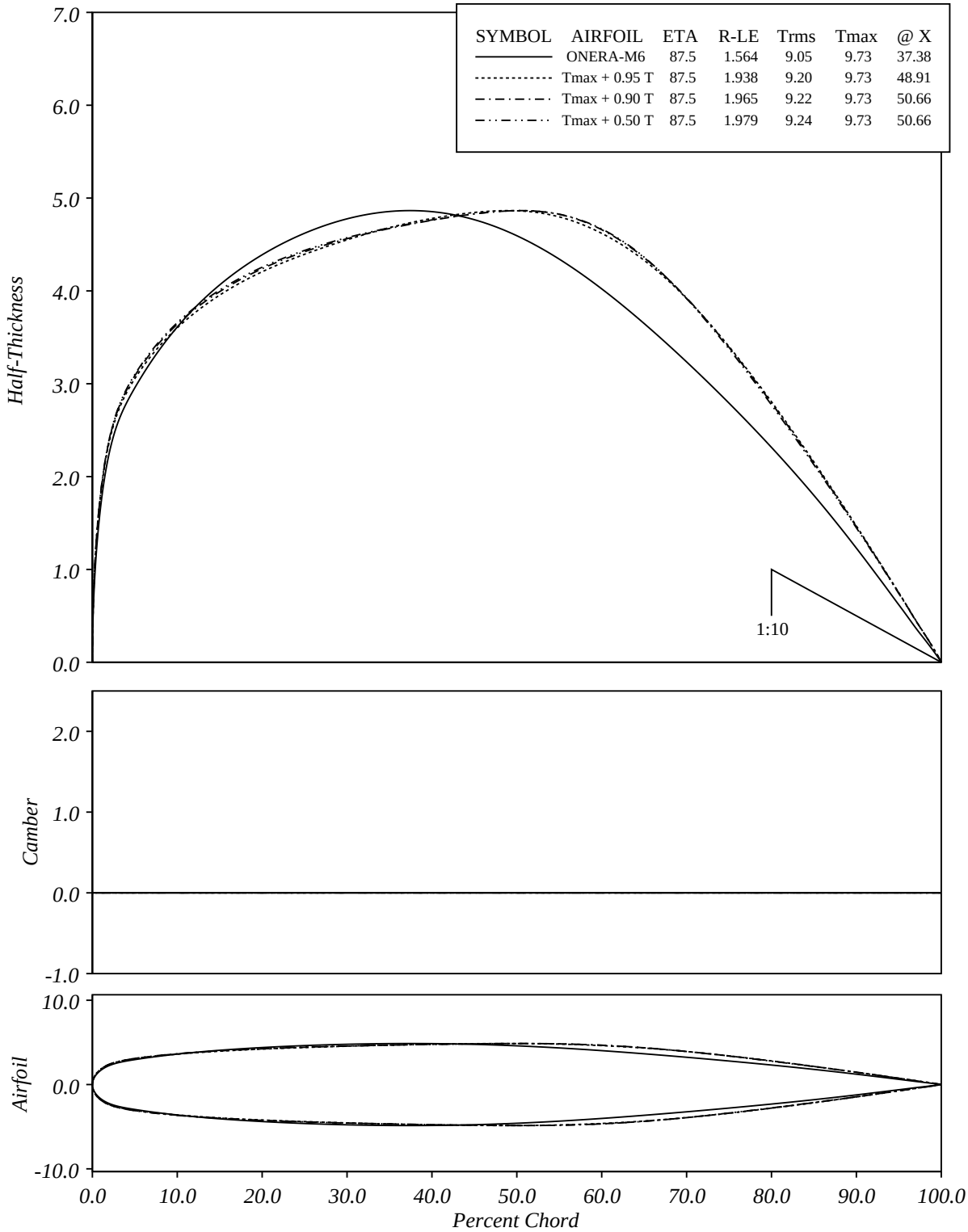


Figure 44: Comparison of ONERA-M6 Airfoil Sections at $\eta = 0.87$.

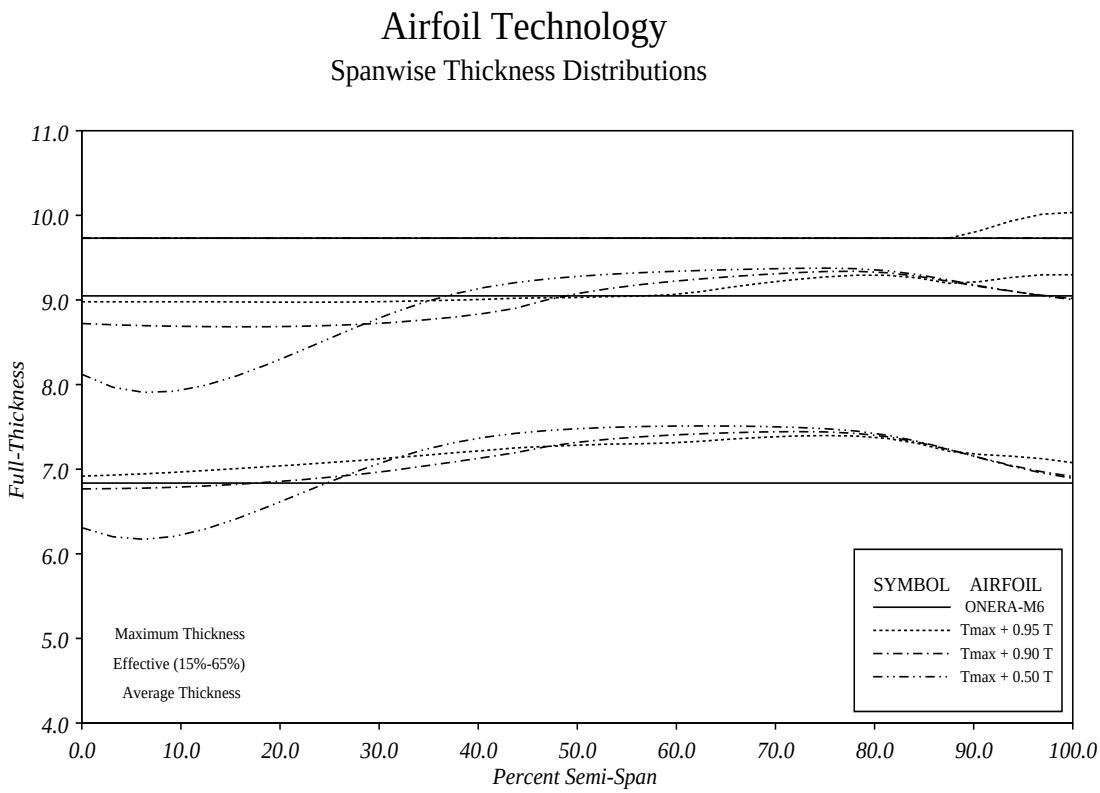


Figure 45: Comparison of ONERA-M6 Baseline and Optimums Thickness Distributions.

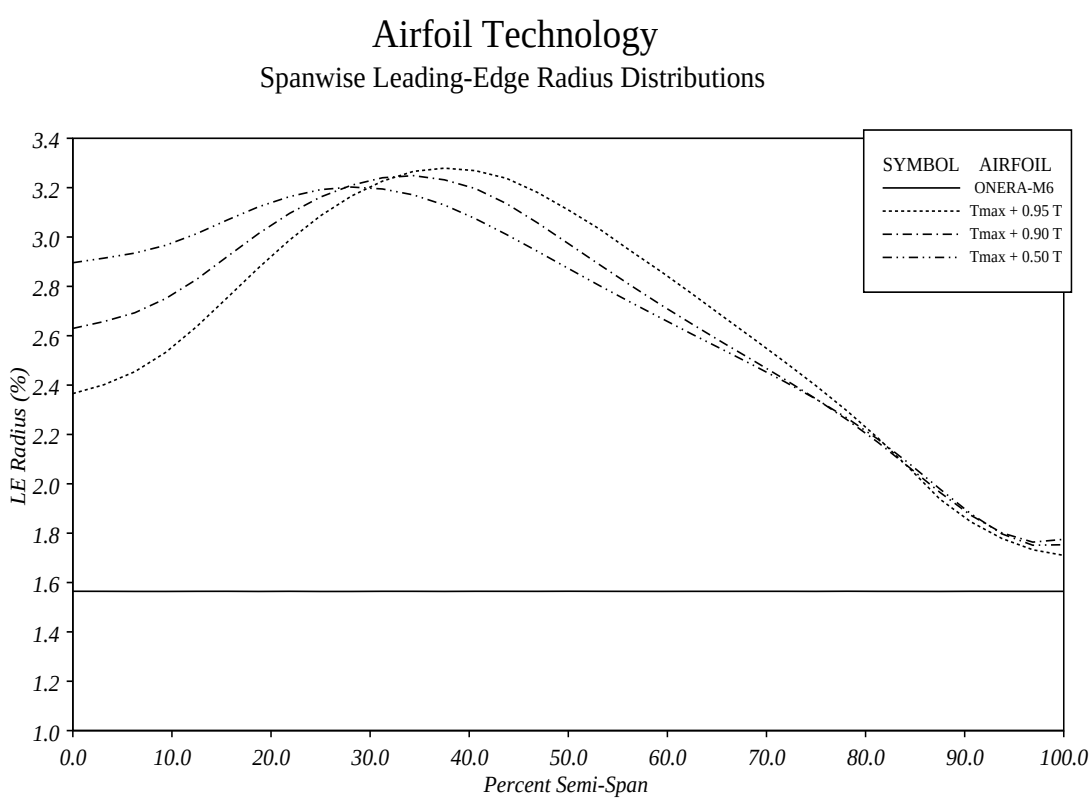


Figure 46: Comparison of ONERA-M6 Baseline and Optimums Leading-Edge Radius Distributions.

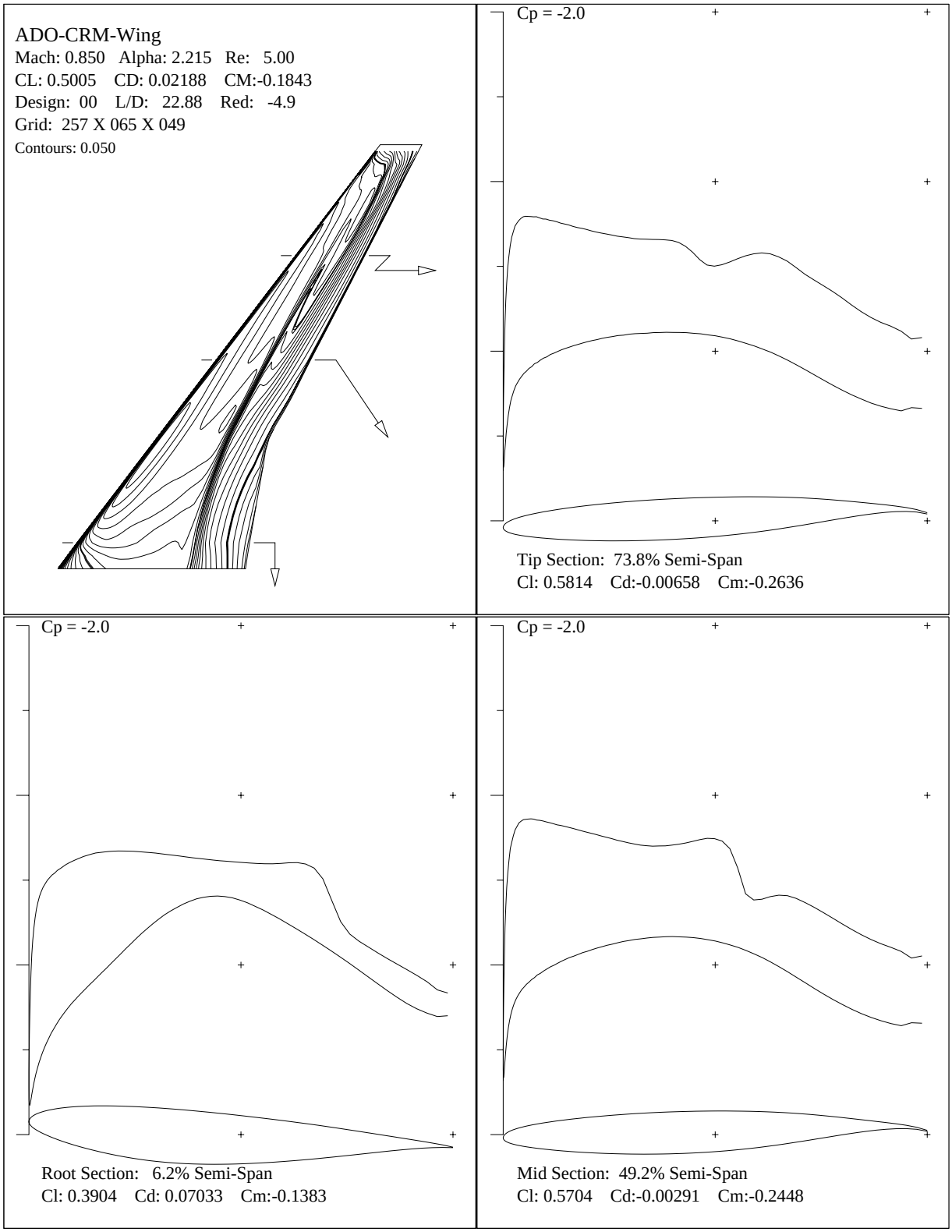


Figure 47: Baseline ADO-CRM-Wing Solution.

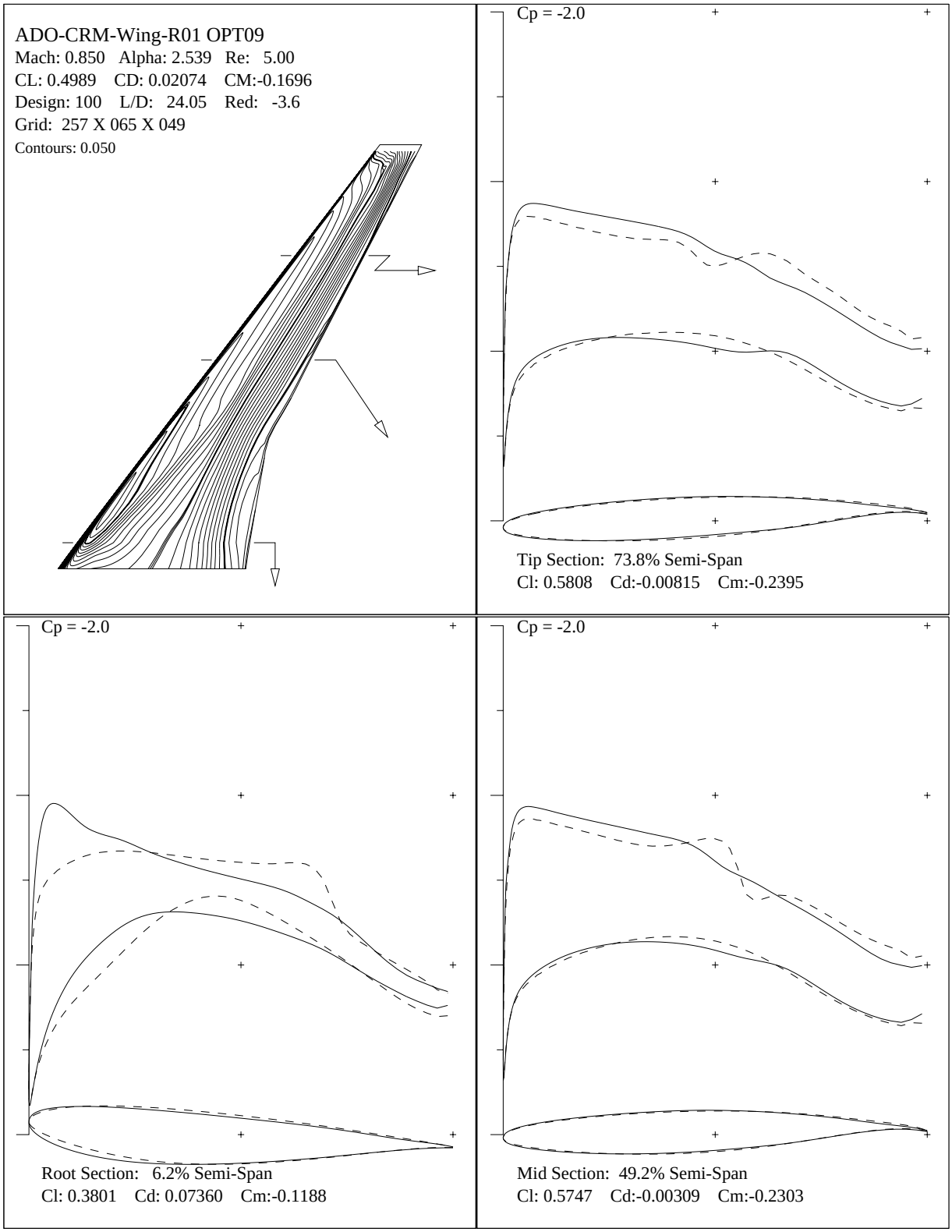


Figure 48: CRMADOV09 Wing Solution.

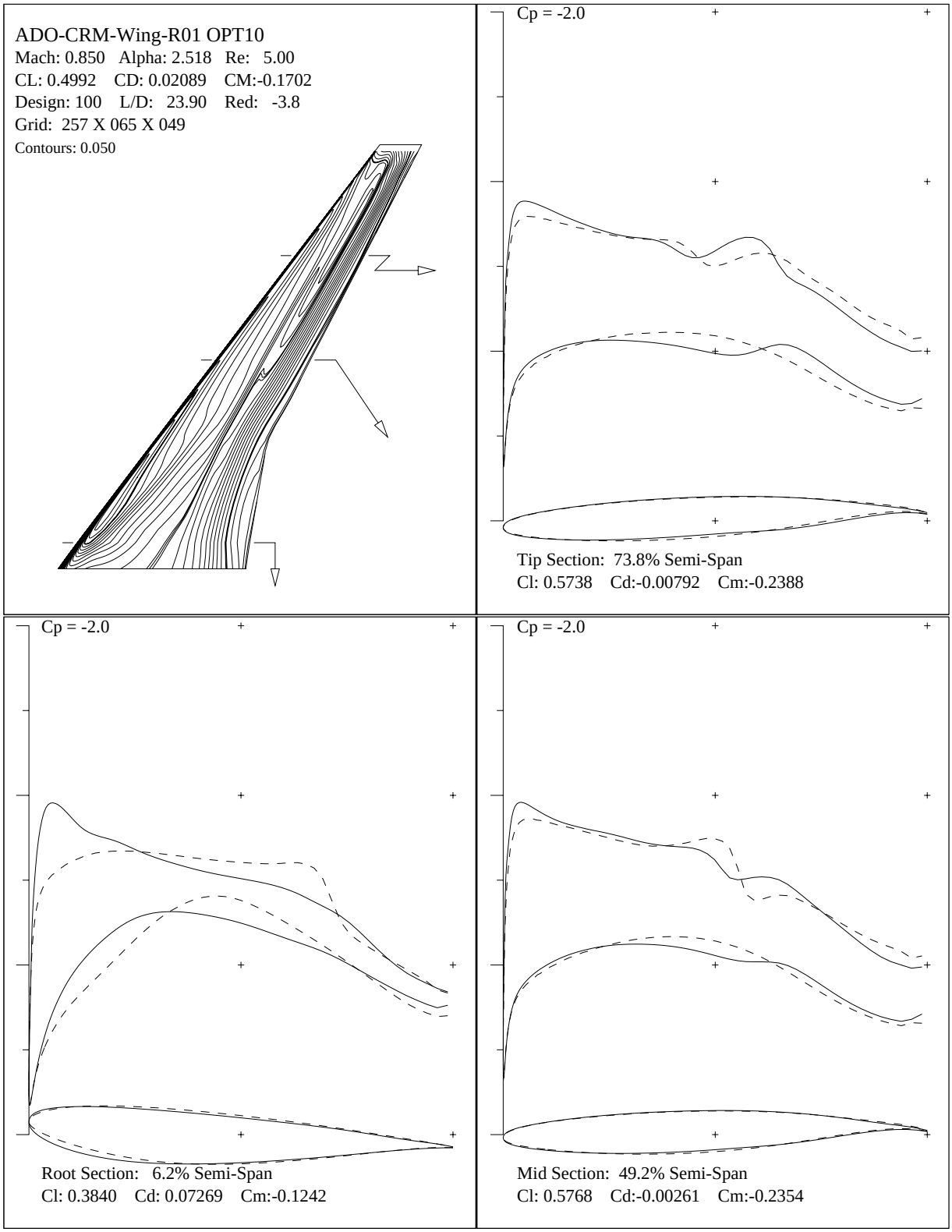


Figure 49: CRMADOV10 Wing Solution, SYN107 Results at $M = 0.85$, $C_L = 0.50$, $Re = 5 \times 10^6$.

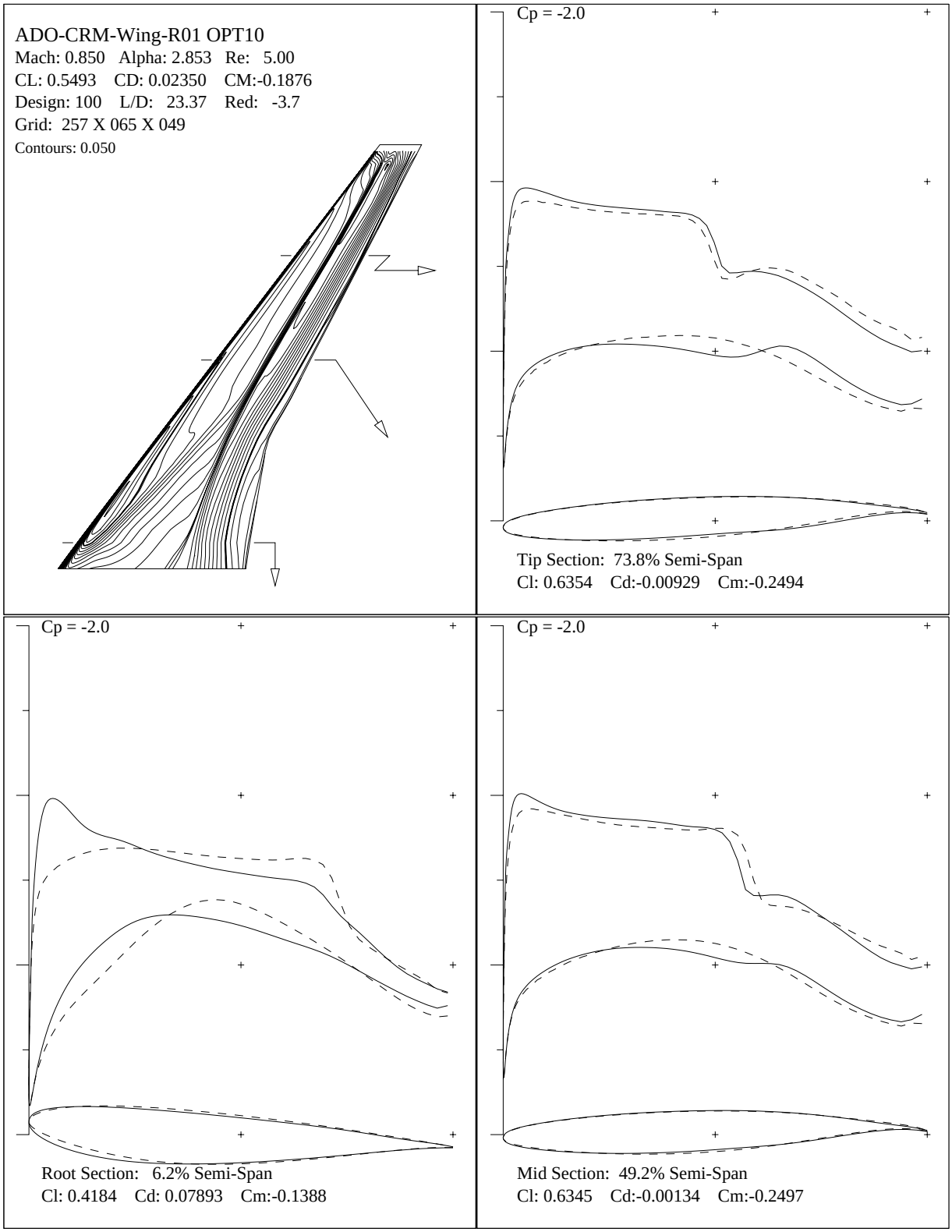


Figure 50: CRMADOV10 Wing Solution, SYN107 Results at $M = 0.85$, $C_L = 0.55$, $Re = 5 \times 10^6$.

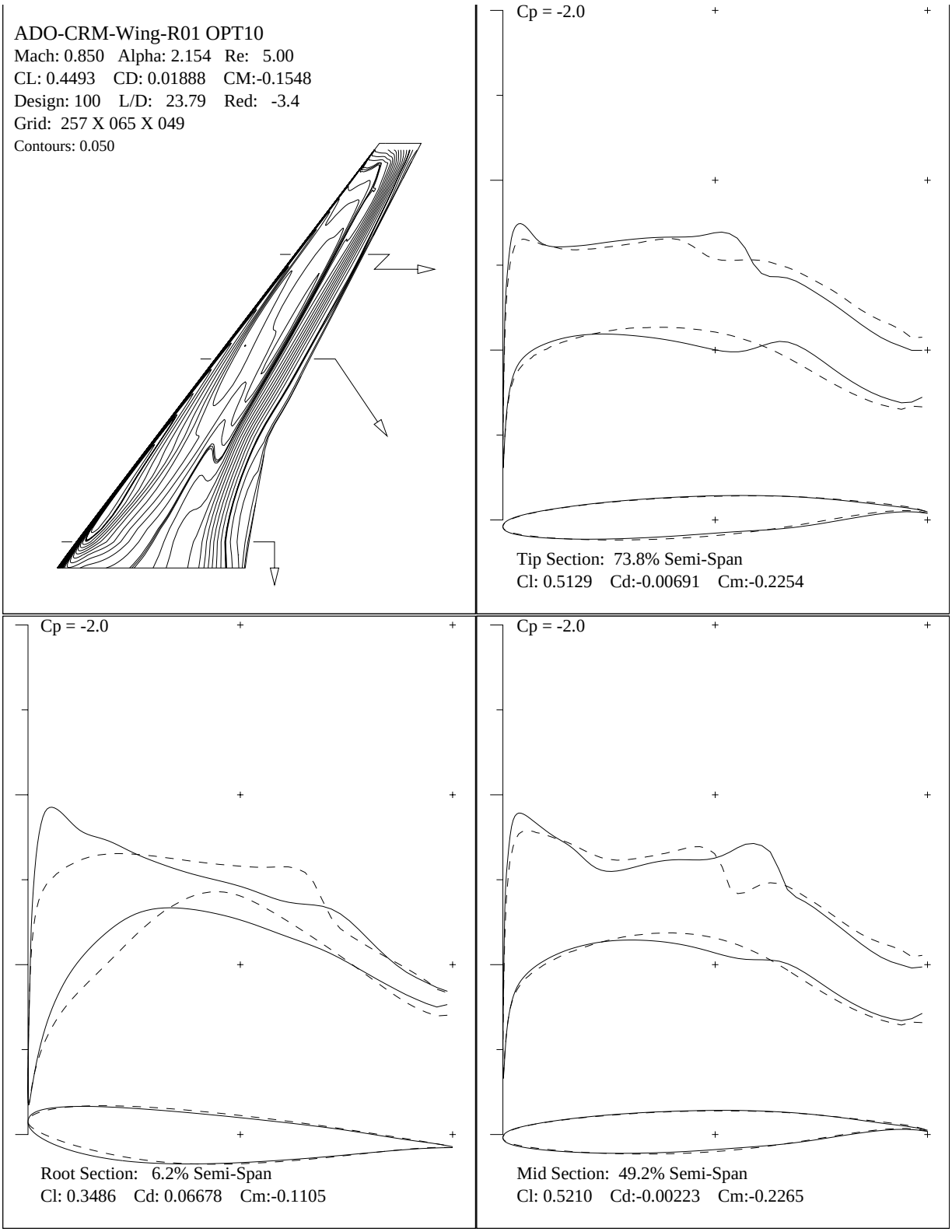


Figure 51: CRMADOV10 Wing Solution, SYN107 Results at $M = 0.85$, $C_L = 0.45$, $Re = 5 \times 10^6$.

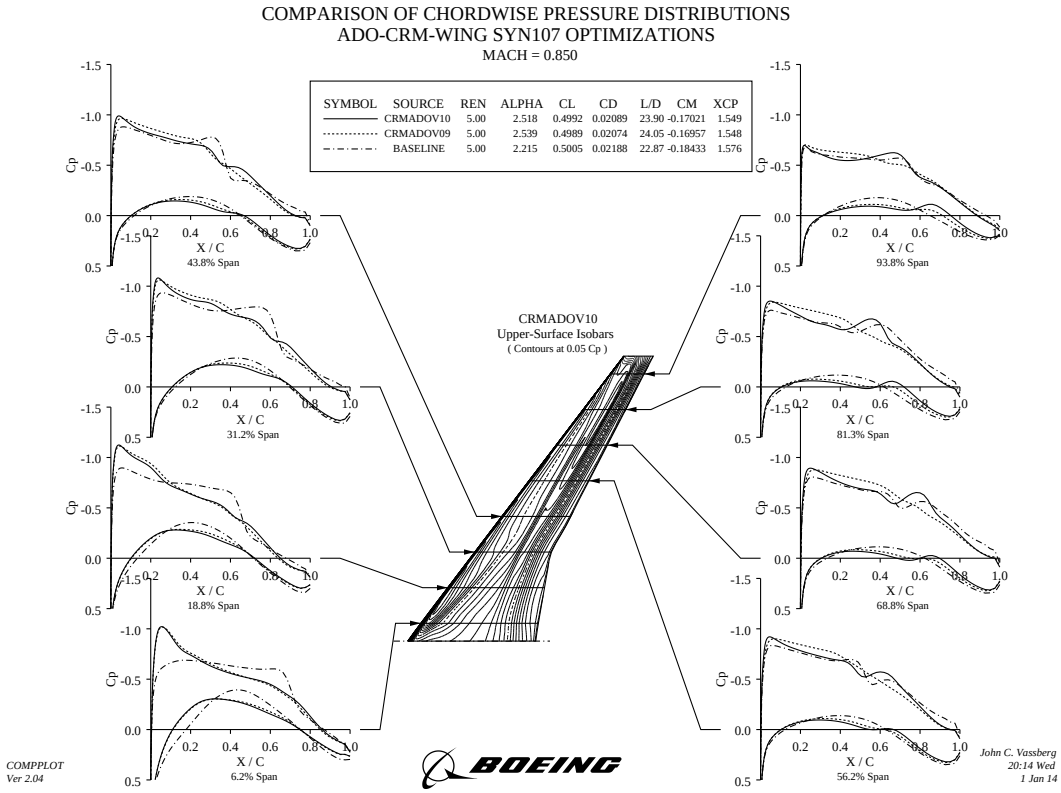


Figure 52: ADO-CRM Pressure Comparisons, SYN107 Results at $M = 0.85$, $C_L = 0.50$, $Re = 5 \times 10^6$.

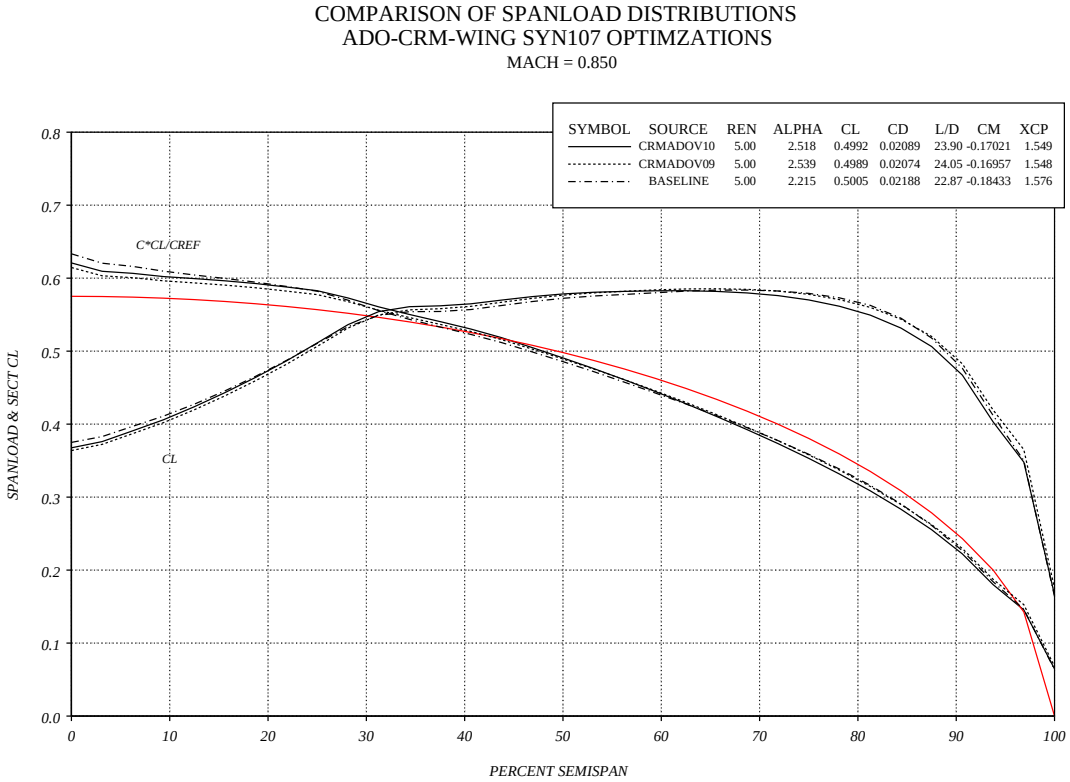


Figure 53: ADO-CRM Spanload Comparisons, SYN107 Results at $M = 0.85$, $C_L = 0.50$, $Re = 5 \times 10^6$.

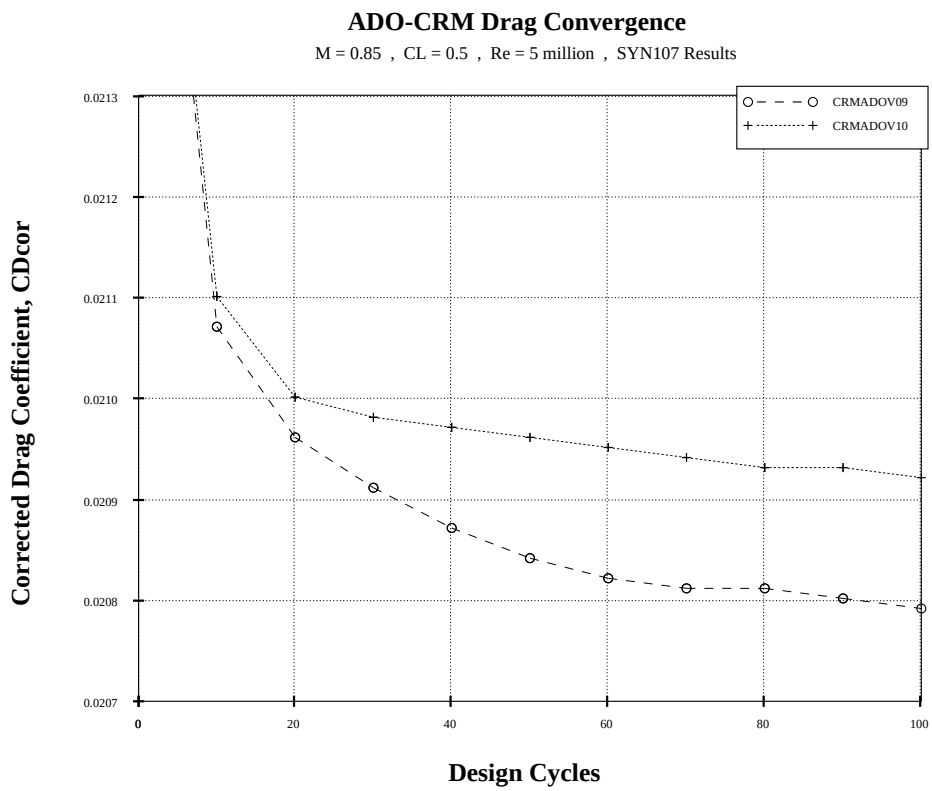


Figure 54: ADO-CRM Drag Reduction Histories, SYN107 Results at $M = 0.85$, $C_L = 0.50$, $Re = 5 \times 10^6$.

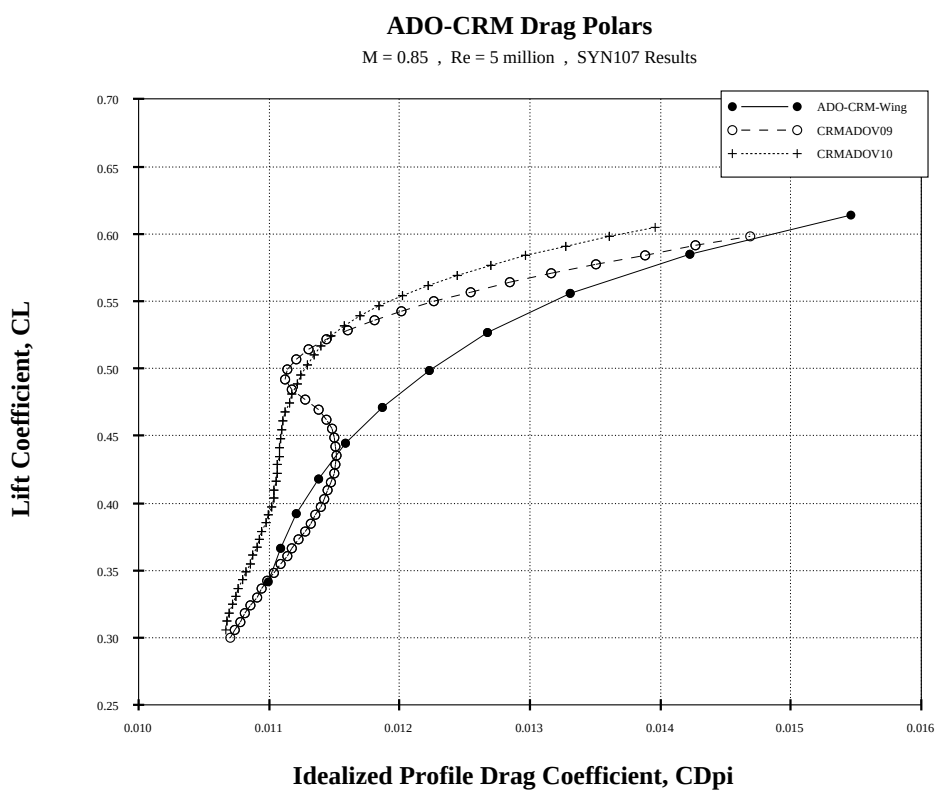


Figure 55: ADO-CRM Drag Polars, SYN107 Results at $M = 0.85$, $Re = 5 \times 10^6$.

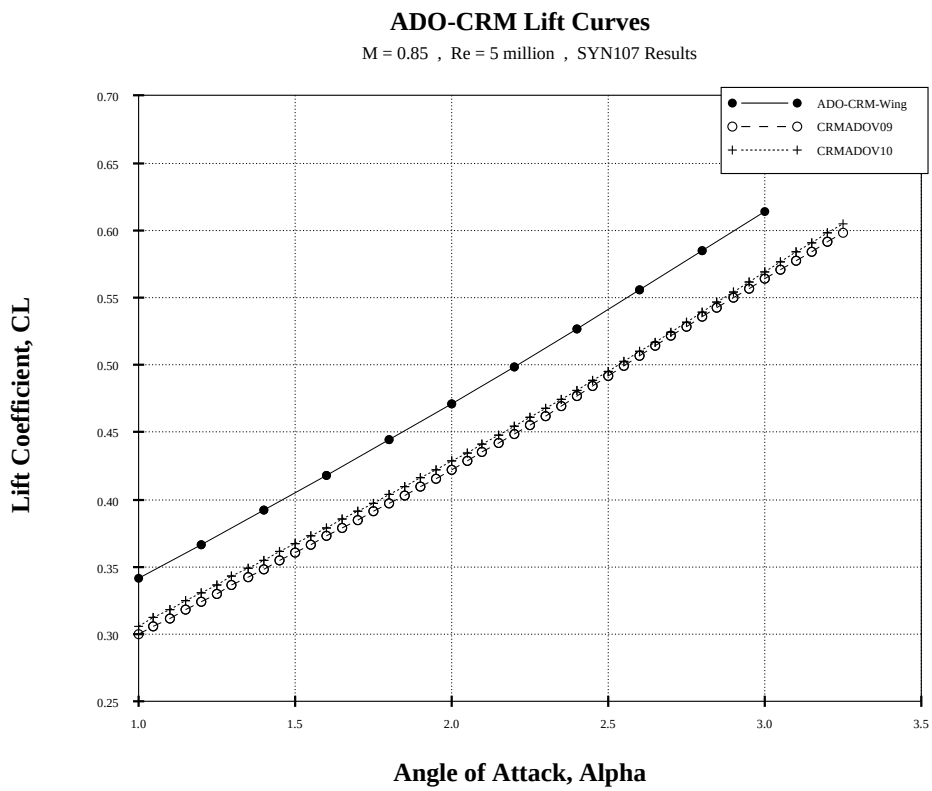


Figure 56: ADO-CRM Lift Curves, SYN107 Results at $M = 0.85$, $Re = 5 \times 10^6$.

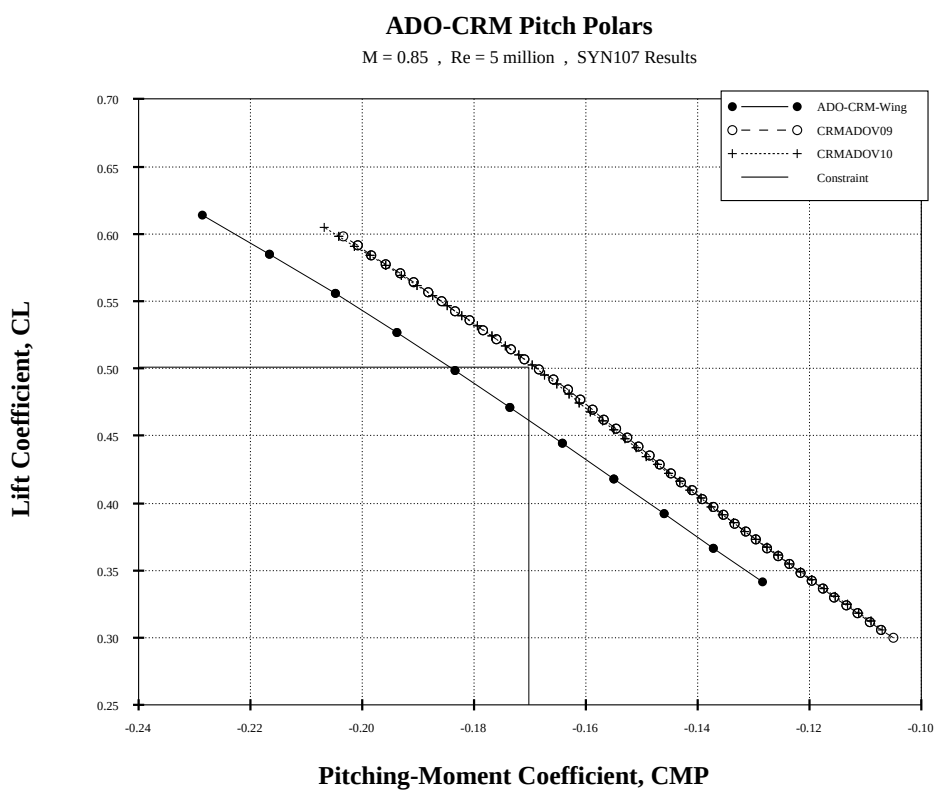


Figure 57: ADO-CRM Pitch Polars, SYN107 Results at $M = 0.85$, $Re = 5 \times 10^6$.

10 Appendix: Notes on Standard Cubic B-Splines

A standard B-Spline is defined with a set of control points, a knot vector, and basis functions. The control-point index range is $k = [0, kmax]$, hence there are $(kmax + 1)$ control points defining the B-Spline. Let $morder$ be the polynomial order of the B-Spline curve segments. The knot vector is parameterized by t. A non-periodic knot vector of $(kmax + morder + 1)$ values begins with the first $morder$ values being set to 0, the next $(kmax + 1 - morder)$ values increasing by 1, and the last $morder$ values set to $(tmax = kmax + 1 - morder)$. The number of curve segments comprising the B-Spline is the interger value of $tmax$. If the knot index begins at 0, then the last value of 0 is located in $knot(morder - 1)$, and the first value of $tmax$ is located in $knot(kmax)$.

For example, the B-Splines described in Section 5 are comprised of 30 cubic curve segments based on 33 control points. Here, $kmax = 32$, $morder = 3$, and $(0 \leq t \leq 30)$. The x-coordinates $(xcpt_k)$ of the control points are prescribed by Eqn (9). The y-coordinates $(ycpt_k)$ of the first and last control points are pinned by the leading-edge and trailing-edge points. The remaining 31 y-coordinates are design variables of the optimizations. The knot vector has values of $(0, 0, 0, 1, 2, 3, ..., 28, 29, 30, 30, 30)$. From begining to end, the B-Spline is $C2$ continuous, and therefore $G2$ continuous as well.

What remains to be discussed is the set of basis functions, $\mathcal{B}_k^m(t)$. A popular method used to compute the basis values is the deBoor algorithm. This method initializes the basis array with the 0th-order values, then recursively elevates the basis array to the polynomial order desired. While this algorithm is useful for computation, it is somewhat difficult to see what is going on. For clarity, we will dive into this a little further. We will restrict this discussion on the cubic B-Spline. However, and in general, what happens near the boundaries is different than in the interior of the B-Spline. For a cubic B-Spline, the interior is spanned by $(2 \leq t < tmax - 2)$. Outside this range on t, special basis functions arise. Note that there are at most $(morder + 1)$ non-zero basis values, and that the summation of them equals 1. One can view these basis values as weighting parameters acting on the control points to define the continuous analytic spline.

Boundary Curve $(0 \leq t < 1)$

For the first curve segment, where $(0 \leq t < 1)$, define $(u = t - 0)$. The corresponding basis functions are then given by:

$$
\begin{aligned}
\mathcal{B}_0^3(u) &= -u^3 + 3u^2 - 3u + 1 = (1-u)^3, \\
\mathcal{B}_1^3(u) &= +\frac{7}{4}u^3 - \frac{9}{2}u^2 + 3u, \\
\mathcal{B}_2^3(u) &= -\frac{11}{12}u^3 + \frac{3}{2}u^2, \\
\mathcal{B}_3^3(u) &= +\frac{1}{6}u^3, \\
\mathcal{B}_k^3(u) &= +0;\ k > 3.
\end{aligned}
$$

Near-Boundary Curve $(1 \leq t < 2)$

For the second curve segment, where $(1 \leq t < 2)$, define $(u = t - 1)$. The corresponding basis functions are then given by:

$$
\begin{aligned}
\mathcal{B}_1^3(u) &= -\frac{1}{4}u^3 + \frac{3}{4}u^2 - \frac{3}{4}u + \frac{1}{4} = \frac{1}{4}(1-u)^3, \\
\mathcal{B}_2^3(u) &= +\frac{7}{12}u^3 - \frac{5}{4}u^2 + \frac{1}{4}u + \frac{7}{12}, \\
\mathcal{B}_3^3(u) &= -\frac{1}{2}u^3 + \frac{1}{2}u^2 + \frac{1}{2}u + \frac{1}{6}, \\
\mathcal{B}_4^3(u) &= +\frac{1}{6}u^3, \\
\mathcal{B}_k^3(u) &= +0;\ k = 0,\ k > 4.
\end{aligned}
$$

Interior Curves ($2 \le t < tmax - 2$)

For all of the interior curve segments, where ($2 \le t < tmax - 2$), define ($u = t - i$), where $i = integer(t)$. The corresponding basis functions are then given by:

$$
\begin{aligned}
\mathcal{B}^3_{i+0}(u) &= -\frac{1}{6}u^3 + \frac{1}{2}u^2 - \frac{1}{2}u + \frac{1}{6} = \frac{1}{6}(1-u)^3, \\
\mathcal{B}^3_{i+1}(u) &= +\frac{1}{2}u^3 - u^2 + \frac{2}{3}, \\
\mathcal{B}^3_{i+2}(u) &= -\frac{1}{2}u^3 + \frac{1}{2}u^2 + \frac{1}{2}u + \frac{1}{6}, \\
\mathcal{B}^3_{i+3}(u) &= +\frac{1}{6}u^3, \\
\mathcal{B}^3_{k}(u) &= +0;\ k < i,\ k > i+3.
\end{aligned}
$$

Near-Boundary Curve ($tmax - 2 \le t < tmax - 1$)

For the second-to-last curve segment, where ($tmax - 2 \le t < tmax - 1$), define ($u = t - i$), where $i = integer(t)$. The corresponding basis functions are then given by:

$$
\begin{aligned}
\mathcal{B}^3_{kmax-4}(u) &= -\frac{1}{6}u^3 + \frac{1}{2}u^2 - \frac{1}{2}u + \frac{1}{6} = \frac{1}{6}(1-u)^3, \\
\mathcal{B}^3_{kmax-3}(u) &= +\frac{1}{2}u^3 - u^2 + \frac{2}{3}, \\
\mathcal{B}^3_{kmax-2}(u) &= -\frac{7}{12}u^3 + \frac{1}{2}u^2 + \frac{1}{2}u + \frac{1}{6}, \\
\mathcal{B}^3_{kmax-1}(u) &= +\frac{1}{4}u^3, \\
\mathcal{B}^3_{k}(u) &= +0;\ k < kmax-4,\ k = kmax.
\end{aligned}
$$

Boundary Curve ($tmax - 1 \le t < tmax$)

For the last curve segment, where ($tmax - 1 \le t < tmax$), define ($u = t - i$), where $i = integer(t)$. The corresponding basis functions are then given by:

$$
\begin{aligned}
\mathcal{B}^3_{kmax-3}(u) &= -\frac{1}{6}u^3 + \frac{1}{2}u^2 - \frac{1}{2}u + \frac{1}{6} = \frac{1}{6}(1-u)^3, \\
\mathcal{B}^3_{kmax-2}(u) &= +\frac{11}{12}u^3 - \frac{5}{4}u^2 - \frac{1}{4}u + \frac{7}{12}, \\
\mathcal{B}^3_{kmax-1}(u) &= -\frac{7}{12}u^3 + \frac{3}{4}u^2 + \frac{3}{4}u + \frac{1}{4}, \\
\mathcal{B}^3_{kmax-0}(u) &= +u^3, \\
\mathcal{B}^3_{k}(u) &= +0;\ k < kmax-3.
\end{aligned}
$$

First and Second Derivatives of a Cubic B-Spline

As it turns out, derivatives of B-Splines can be easily created by taking the deltas of the control points to form a new set of control points, and recognizing that the derivative is also a B-Spline with polynomial order reduced by 1. And, as expected, there is special treatment near the boundaries that need to be applied. Note also that the number of control points of the derivative spline is reduced by 1. However, the number of curve segments remain the same.

Let's define the following.

$$
\begin{aligned}
Q_k &= (xcpt_k, ycpt_k, zcpt_k, ...);\ k = 0, kmax, \\
\Delta Q_k &= Q_{k+1} - Q_k;\ k = 0, kmax - 1.
\end{aligned}
$$

However, near the boundaries, ΔQ_k needs to be adjusted, and this adjustment depends on $morder$.

If $morder = 3$, then:

$$\begin{aligned}\Delta Q_k &= 3\Delta Q_k;\ k = 0,\ k = kmax - 1,\\ \Delta Q_k &= \frac{3}{2}\Delta Q_k;\ k = 1,\ k = kmax - 2.\end{aligned}$$

If $morder = 2$, then:

$$\Delta Q_k = 2\Delta Q_k;\ k = 0,\ k = kmax - 1.$$

If $morder = 1$, then no adjustment is required.

To be clear, *morder* and *kmax* are the values associated with the curve being differentiated. The differentiated curve has values of $morder - 1$ and $kmax - 1$. If a second derivative is desired, then a recursive processing of the first derivative can be performed, however, be sure to send the appropriate values of order and count to your subroutine.

Curvature

In general, the absolute curvature of a 3-dimensional space curve is given by:

$$\mathcal{K}(t) = \frac{||\dot{Q} \times \ddot{Q}||}{||\dot{Q}||^3}.$$

For a planar curve with z-constant, we have the following.

$$\begin{aligned}\dot{Q} \times \ddot{Q} &= \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \dot{x} & \dot{y} & 0 \\ \ddot{x} & \ddot{y} & 0 \end{vmatrix} = (\dot{x}\ddot{y} - \ddot{x}\dot{y})\vec{k},\\ ||\dot{Q}||^3 &= (\dot{x}^2 + \dot{y}^2)^{\frac{3}{2}}.\end{aligned}$$

Note that when $\dot{Q}$ and $\ddot{Q}$ are aligned, the curvature $\mathcal{K}$ vanishes and an inflection point exists in the curve.

Osculating Circle of a Planar Curve

In order to determine the osculating circle of a z-constant planar curve, let's define a local coordinate system (ξ, η, ζ) with the origin at a point on the curve. Here, ξ is tangent to the curve and in the direction of increasing t. Also in the plane of the curve is the normal coordinate η, and perpendicular to the plane is ζ.

$$\begin{aligned}\vec{\xi} &= \frac{(\dot{x}\vec{i} + \dot{y}\vec{j})}{||\dot{Q}||},\\ \vec{\eta} &= \frac{(\dot{x}\vec{j} - \dot{y}\vec{i})}{||\dot{Q}||},\\ \vec{\zeta} &= \vec{k}.\end{aligned}$$

Note that the radius (ρ) of the osculating circle is equal to the inverse of the curvature ($\mathcal{K}$). In 2D, it is convenient to maintain a sign on curvature, and therefore also radius. In this work, we use the convention that a positive curvature points inward to the airfoil contour. Since both the lower and upper airfoil surfaces begin at the leading edge and end at the trailing edge, we introduce a signed unit factor as follows.

$$\rho(t) = sign * \frac{(\dot{x}^2 + \dot{y}^2)^{\frac{3}{2}}}{(\ddot{x}\dot{y} - \dot{x}\ddot{y})}; \qquad sign = \begin{cases} +1\ Upper \\ -1\ Lower \end{cases},$$

with the signed curvature being simply:

$$\mathcal{K}(t) = \frac{1}{\rho(t)}.$$

Curvature at the Leading Edge

In order to enforce $G2$ continuity at the leading edge point, ie. at $t = 0$, let's investigate the first and second derivatives there.

$$\begin{aligned}
\dot{x}_{LE} &= 3(XQ_1 - XQ_0) = 0, \\
\dot{y}_{LE} &= 3(YQ_1 - YQ_0) = 3YQ_1, \\
\ddot{x}_{LE} &= 3(2XQ_0 - 3XQ_1 + XQ_2) = 3XQ_2, \\
\ddot{y}_{LE} &= 3(2YQ_0 - 3YQ_1 + YQ_2) = 3(YQ_2 - 3YQ_1).
\end{aligned}$$

Hence,

$$\begin{aligned}
\rho_{LE} &= \frac{(3YQ_{1U})^3}{(3XQ_{2U} * 3YQ_{1U})} = \frac{(3YQ_{1L})^3}{(3XQ_{2L} * 3YQ_{1L})}, \\
XQ_{2U} &= +XQ_{2L}, \\
YQ_{1U} &= -YQ_{1L}.
\end{aligned}$$

This shows that the only constraint required to maintain $G2$ continuity at the leading edge for our B-Spline setup is the last relationship above. [Aside: In order to maintain $C2$ continuity at the leading edge, we must also enforce that $YQ_{2U} = -YQ_{2L}$.]

Least-Squares Fit

Consider fitting a B-Spline through a set of scatter data, such as a set of discrete points that define an airfoil. Let this set of N points be $P_n = (x_n, y_n); n = 1, N$. For the moment, assume that $N >> K = kmax$. Then, a least-squares fit of these data can be found by minimizing the summed error, defined by:

$$\mathcal{E} = \frac{1}{2} \sum_{n=1}^{N} \left[\left(\sum_{k=0}^{K} \mathcal{B}_{k,n} Q_k \right) - P_n \right]^2 .$$

Here, $\mathcal{B}_{k,n}$ is shorthand for $\mathcal{B}_k(t_n)$, and where t_n is determined from x_n of the scatter data. Differentiating the summed error $\mathcal{E}$ with respect to each control point Q_m gives $(kmax + 1)$ equations which should all be set to 0 for the least-squares fit.

$$\frac{d\mathcal{E}}{dQ_m} = \mathcal{B}_{m,n} \sum_{n=1}^{N} \left[\left(\sum_{k=0}^{K} \mathcal{B}_{k,n} Q_k \right) - P_n \right] = 0; \quad m = 0, kmax.$$

Put into matrix form, this system of equations is given as follows.

$$\begin{bmatrix}
\sum_n \mathcal{B}_{0,n}^2 & \sum_n \mathcal{B}_{0,n}\mathcal{B}_{1,n} & \sum_n \mathcal{B}_{0,n}\mathcal{B}_{2,n} & \dots & \sum_n \mathcal{B}_{0,n}\mathcal{B}_{K,n} \\
\sum_n \mathcal{B}_{1,n}\mathcal{B}_{0,n} & \sum_n \mathcal{B}_{1,n}^2 & \sum_n \mathcal{B}_{1,n}\mathcal{B}_{2,n} & \dots & \sum_n \mathcal{B}_{1,n}\mathcal{B}_{K,n} \\
\sum_n \mathcal{B}_{2,n}\mathcal{B}_{0,n} & \sum_n \mathcal{B}_{2,n}\mathcal{B}_{1,n} & \sum_n \mathcal{B}_{2,n}^2 & \dots & \sum_n \mathcal{B}_{2,n}\mathcal{B}_{K,n} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\sum_n \mathcal{B}_{K,n}\mathcal{B}_{0,n} & \sum_n \mathcal{B}_{K,n}\mathcal{B}_{1,n} & \sum_n \mathcal{B}_{K,n}\mathcal{B}_{2,n} & \dots & \sum_n \mathcal{B}_{K,n}^2
\end{bmatrix}
\begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \\ \vdots \\ Q_K \end{bmatrix}
=
\begin{bmatrix} \sum_n \mathcal{B}_{0,n} P_n \\ \sum_n \mathcal{B}_{1,n} P_n \\ \sum_n \mathcal{B}_{2,n} P_n \\ \vdots \\ \sum_n \mathcal{B}_{K,n} P_n \end{bmatrix}$$

Note that the $\mathcal{B}$ matrix is symmetric and without negative terms.

Obviously, a sufficient sampling of scatter data must be available to yield a valid least-square fit of any kind. However, in the case of a B-Spline, the scatter data must sufficiently support each curve segment. For example, a cubic B-Spline should have at least 4 data points, P_n, influencing each control point, Q_k. Otherwise, the system of equations above could become singular.

To Be Continued...