1 Introduction

This is the first of two lectures prepared by the authors for the von Karman Institute that deal with the subject of aerodynamic shape optimization. In this lecture we introduce some theoretical background on optimization techniques commonly used in the industry, apply some of the approaches to a couple of very simple model problems, and compare the results of these schemes. We also discuss their merits and defficiencies as they relate to the class of aerodynamic shape optimization problems the authors deal with on a daily basis. In the second lecture, we provide a set of sample applications. Before we continue with the simple model problems, let's first review some properties of the aerodynamic shape optimization studies the authors are regularly concerned with.

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; these will be addressed in detail in the second 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. This class of problems is much more demanding than those in which the baseline is far from the optimum design. Frequently, it is easier to show a 25% improvement relative to a baseline that is 30% off optimum than it is to realize a 1% gain on a starting configuration that only has 2% to give.

Quite often the problem 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 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 decade, 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.

With some understanding of where we are headed, let's now return to the simple model problems included herein, review various aspects of the optimization process, and discuss how these relate to the aerodynamic shape optimization problem at hand. The first model problem introduces some of the basics; the second one is a classic example in mathematical history.

2 The Spider & The Fly

In our first model problem, we will discuss how to set up a design space, and how to navigate this design space from an initial state towards a local optimum using gradient-based search methods. We will also talk about some *traps* to avoid when setting up a problem of optimization.

The original *spider and fly* problem was first introduced in Reference [1]. In our version of the spider-fly problem, we have a wooden block with dimensions of 4 *in* wide, by 4 *in* tall, by 12 *in* long; the bottom of this block is resting on a solid flat surface. See Figure 1. On one of the square ends sits a spider, located 1 *in* from the top and centered left-to-right. On the opposite side a fly is trapped in the spider's web; the fly is located 1 *in* from the bottom and centered left-to-right. The spider considers the path where he would initially travel 1 *in* straight up to the top, then 12 *in* axially across the top face, then 3 *in* downward to the fly; the length of this path is 16 *in*.

As it turns out, the spider was a mathematician in a former lifetime, so he wonders if this is the minimum-length path possible. In order to solve this enigma, the spider sets up a problem of optimization. To cast this optimization problem into a mathematical formulation, the spider must some how constrain his motion to the surface of the wooden block, and furthermore, he knows he cannot traverse the bottom side of the block as it is resting on the solid flat surface. It is clear that the aforementioned 16 *in* path is a local optimum, and due to the symmetry of the problem, there are really only two other types of paths that need to be studied. In one type, the spider moves laterally 2^+ *in* toward the right/left side of the block, then 12^+ *in* across that side towards the back, then 2^+ *in* to the trapped fly. Here, the length of the path is definitely in excess of 16 *in*, and therefore it cannot be the global optimum. The remaining path type allows the spider to move upward 1^+ *in* to the top of the block, continue diagonally towards the right/left side, then diagonally across and downward towards the back face, and finally 2^+ *in* to the fly. See Figure 2. It is not immediately obvious that the local optimum of this path type cannot also be the global optimum. Hence, the spider must investigate further.

To set up the design space for this problem, the spider adopts a cartesian coordinate system aligned with the wooden block such that the origin coincides with the front-lower-left corner of the block. The x coordinate measures positive to the right, the y coordinate measures positive along the long side of the block away from the front face, and the z coordinate measures positive upward in the vertical direction. In this coordinate system, the spider's initial position on the front face is $(XS, YS, ZS) = (2, 0, 3)$, and the fly is trapped on the back face at $(XF, YF, ZF) = (2, 12, 1)$.

The path type to optimize can be partitioned into four segments, corresponding to the four block faces to be traversed. The first segment is described by end points $(2, 0, 3)$ and $(X, 0, 4)$. The second segment's end points are $(X, 0, 4)$ and $(4, Y, 4)$. The third, $(4, Y, 4)$ and $(4, 12, Z)$. The fourth, $(4, 12, Z)$ and $(2, 12, 1)$. Hence, the complete path can be described as the piecewise linear curve that connects $(2, 0, 3)$, $(X, 0, 4)$, $(4, Y, 4)$, $(4, 12, Z)$, and $(2, 12, 1)$. In this design space, there are precisely three design variables (X, Y, Z). Further, the design space is constrained by the inequalities:

$$\begin{array}{ccccc} 0 & \leq & X & \leq & 4, \\ 0 & \leq & Y & \leq & 12, \\ 0 & \leq & Z & \leq & 4. \end{array} \tag{1}$$

Hence, the design space as defined in this problem is constrained to the interior of the wooden block.

The length of each segment is given as:

$$\begin{array}{rcl} S_1 & = & \left[1 + (X-2)^2\right]^{\frac{1}{2}}, \\ S_2 & = & \left[(X-4)^2 + Y^2\right]^{\frac{1}{2}}, \\ S_3 & = & \left[(Y-12)^2 + (Z-4)^2\right]^{\frac{1}{2}}, \\ S_4 & = & \left[(Z-1)^2 + 4\right]^{\frac{1}{2}}. \end{array} \tag{2}$$

The total path length (or objective/cost function) is defined as:

$$I \equiv S = S_1 + S_2 + S_3 + S_4. \tag{3}$$

The statement of optimization is to minimize I subject to the constraints of Eqn (1).

There are many ways in which one can proceed to solve this problem of optimization. For example, one approach is to use evolution theory, where a population of random guesses of $(X, Y, Z)_i$ is evaluated for their associated set of cost-function values, I_i. This establishes a generation of information which can be used to coerse subsequent generations towards the optimum location within the design space. An improvement to basic evolution theory is the Genetic Algorithm (GA). GAs attempt to speed the evolution process by combining the *genes* of promising pairs from one generation to procreate the next. GAs also allow some fraction of mutations to occur in order to improve the chance of finding a global optimum. However in general, this is *not* guaranteed. These methods are relatively easy to set up and program as they do not require any gradient information, and in fact may be the best choice if the cost function does not smoothly vary throughout the design space. Unfortunately, they can be computationally very expensive, even for a problem with a modest number of design variables. Nonetheless, solving the spider-fly problem with evolution