From *SIAM News*, Volume 38, Number 4, May 2005

The Extensive Use of Splines at Boeing

By Thomas A. Grandine

Splines are used extensively at Boeing and throughout much of the industrial world. There are very few remaining areas in either manufacturing or engineering in which these interesting functions have yet to play a role, and their use continues to grow at a rapid rate.

A spline, for the purposes of this article, is defined as any function made up of one or more polynomial pieces joined together to satisfy given (and possibly different) differentiability requirements. At Boeing, these functions are used not only to represent geometric designs, by modeling curves and surfaces that describe the edges and faces of geometric solids, but also to model engineering analysis and performance data. As an example of the latter, splines can be used to model airplane drag as a function of mach number, the speed of the airplane with respect to the free stream airflow.

Ideal Modeling Technology

In these contexts, Boeing uses splines in four very different kinds of applications. The most familiar is computer-aided design, where splines are used to represent the geometric entities that form the basis of product-definition data.

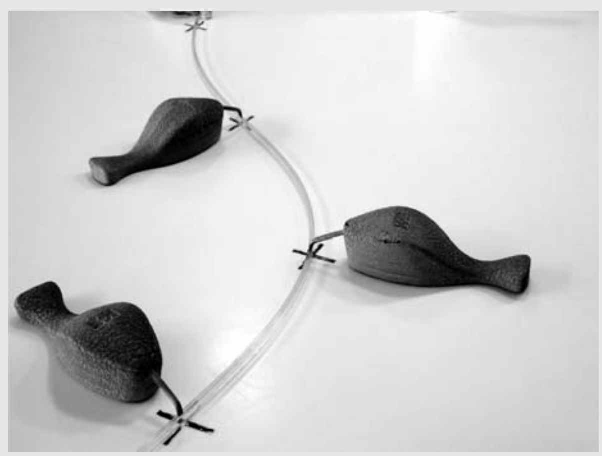
A related application, common across many industries, is computer-aided manufacturing. Here, in addition to representing geometric parts, splines can be used to represent machine tool cutter paths. They can also be used as engineering data modeling functions. Compensation tables on machine tools, for example, can be represented this way; compensation tables themselves are used to model positional corrections to a large variety of environmental or external conditions, such as temperature.

A third application class, engineering an-alysis and simulation, is especially important at a large engineering company like Boeing. Examples range from sophisticated computational fluid dynamics simulations to simple linear analysis codes involving simplified physical models. In many cases, splines are required to represent geometry or geometric boundary conditions. In others, splines are called on to model material properties and thicknesses, atmospheric chemical composition and reactivity, and experimental results, such as wind tunnel data. Some codes use them to calibrate other computations, while others use the B-splines that form a basis for any given spline function space as finite elements. In the finite element case, engineering codes create spline approximations to the solution of ordinary differential equations, partial differential equations, integral equations, and differential–algebraic equations, an approach popularized by de Boor and Swartz in 1973.

In the last class of applications, splines are components of embedded systems in the products themselves—for example, navigation and guidance systems. They are a key component of Boeing's Automated Flight Manual system, which is run by an airplane crew on every mission and in which they are used to represent airplane performance data.

Splines have not always been applied so extensively. Before 1971, when Carl de Boor's recurrence relation for splines was published, calculations involving splines were often tedious affairs that practitioners tended to avoid if possible. The recurrence relation radically altered that situation, making splines an ideal modeling technology. Today, on a daily basis, Boeing airplanes make more than 42,000 flights, and within the company, more than 10,000 design applications and 20,000 engineering applications are run. This means that Boeing uses de Boor's recurrence relation to perform something like 500 million spline evaluations every day, and the number is increasing every year.

One activity contributing directly to this increase is the relatively recent rise of virtual engineering and design space exploration. This technology makes possible the analysis of proposed designs through the construction of digital models and computational simulation rather than physical testing and prototyping. Long envisioned as a revolutionary design methodology, virtual engineering is now starting to deliver on its promise of greatly improved designs at reduced engineering cost. This achievement has been enabled by a large number of accomplishments in many areas, including engineering, computer science, and mathematics.



Draftsman's spline. When acquired by Carl de Boor, in the 1970s at the Mathematics Research Center at the University of Wisconsin, it was already serving to create "the (quite wrong) impression that the cubic spline is a good model of the draftsman's spline." Still, readers will appreciate its appeal for de Boor, whose work on spline functions, beginning with his recurrence relation for splines (1971), has been recognized by several prizes and honors; most recently, he received the National Medal of Science this year for "fundamental contributions to mathematics that strongly assisted numerical computation in science and engineering" (see SIAM News*, April 2005, page 1). Spline functions and their descendants in current work are the subject of two articles in this issue of* SIAM News*: the accompanying article by Tom Grandine on the extensive (and growing) use of splines at Boeing and Ron DeVore and Amos Ron's look at theoretical advances (page 5).*

Design Optimization with Splines

One of my longstanding interests has been finding effective means of modeling not only an individual proposed design, but simultaneously all of the admissible design variations as well. As an example, consider the airplanes shown in Figure 1. In order to choose the best airplane from the different design alternatives, the space of designs itself must be modeled. Accomplishing this feat has proved surprisingly elusive in practice, for several reasons: (1) analysis results need to vary continuously as functions of the design variations if meaningful trends in design space are to be detected and exploited, (2) design variations need to conform to requisite shape characteristics of the design (for example, the upper half of an airfoil curve should be convex), and (3) design variations should be controllable by relatively few design variables.

Figure 1. *Many different design possibilities.*

The first reason eliminates the vast majority of hybrid spline-fitting algorithms, as the internal switches between methods in those algorithms usually lead to discontinuities in the output. The second reason eliminates all linear mappings from design parameters to splines because those mappings, with the exception of the piecewise linear mapping, do not necessarily preserve the underlying shape of the input data (Ferguson's theorem, 1985). Together, these two reasons eliminate most published spline-based curve and surface construction schemes.

In 2001, when fellow Boeing mathematician Tom Hogan and I were trying to figure out how to accomplish an effective parametrization of 2D curves, we were led to consider a cubic spline-fitting scheme of de Boor, Höllig, and Sabin, published in 1987. This particular curve-fitting scheme takes a sequence of points at which curve direction and curvature are also specified and produces a curve that interpolates all of the given data. Because the scheme has the property that the sign of the curvature does not change between any of the data points, it preserves the shape of the given data. Moreover, the scheme is $O(h^6)$ in the given data, so each doubling of the number of data points improves the accuracy of the curve fit (relative to a smooth, procedurally defined given curve) by a factor of 64. In short, the scheme seems to have the three requisite properties, being both shape-preserving and controllable by relatively few design variables.

Unfortunately, a closer look at the scheme reveals that it is not necessarily continuous with respect to its input data, and that the interpolant does not necessarily even exist for some seemingly innocuous data sets. For each consecutive pair of input points, the scheme requires the solution of the quadratic system of equations

$$\begin{aligned} \rho_0 &= 1 - R_1/\rho_1^2 \\ \rho_1 &= 1 - R_0/\rho_0^2 \end{aligned}$$

in the variables ρ_0 and ρ_1, where the numbers R_0 and R_1 are positive constants that depend on the curve direction and curvatures specified at the given pair of points. Figure 2 depicts the number of solutions that this quadratic system has for different positive values of R_0 and R_1. That there are regions in which there are no solutions is a problem, as is the discontinuous jump from one solution branch to another at $R_0 = R_1 = 3/4$.

As an example, consider the airfoil data depicted in Figure 3. One would like to generate a curve that passes through these data points, satisfying the indicated directions and curvatures. However, the de Boor, Höllig, and Sabin scheme for this innocent data set has no solution; the two highlighted data points require solution of a quadratic system in one of the two zero regions depicted in Figure 3.

Faced with the dilemma of an almost-but-not-quite-perfect curve-fitting scheme, Tom Hogan and I attempted to repair it by increasing the polynomial degree of the spline from 3 to 4. After trying half a dozen or so different approaches over a period of three or four months, we were ultimately successful by solving a nonlinear system similar to the one given earlier, but that depends on the value of the curvature at the midpoint of the given curve segment. We were able to prove that that value can always be chosen in such a way that the nonlinear system has exactly one solution, thereby eliminating the only drawbacks to the scheme of de Boor, Höllig, and Sabin. (A side-benefit of the modified scheme is that its accuracy increases to $O(h^8)$ for data extracted from a conic section.) For the sample data given in Figure 3, the modified scheme was successful in producing an airfoil curve, as shown in Figure 4.

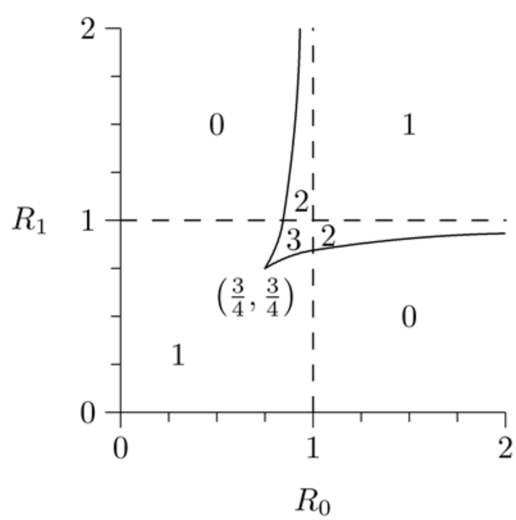


Figure 2. *Number of positive solutions.*

This is a common theme in industrial mathematics: A very promising method, seemingly ideally suited to a given application, is available in the technical literature. Preliminary deployment of the method on real, industrial problems reveals an unsuspected flaw or two. Additional refinement and augmentation of the initial idea sometimes provides the needed fixes, as in this case. Only rarely, as in the case of the recurrence relation for B-splines or de Boor–Swartz collocation, does the idea work almost perfectly right out of the box.

My colleagues and I continue to pursue the broad challenge of geometric modeling of design variation, and our efforts to date have been fairly productive, though many challenges are yet to be overcome. Among them are discovery of an effective means of parametrizing surfaces that enclose predefined collections of objects, automatic generation of design space for given fixed designs, and automatic calculation of derivatives of geometric quantities (e.g., surface area, center of mass) with respect to the defining parameters. In the more general context, we have really only scratched the surface in terms of what splines can offer a company like Boeing. The possibilities for improving consistency across engineering applications—by replacing data tables with splines, making more effective use of B-splines as finite elements, improving part quality and productivity by more extensive use of splines in machining, and improving modeling of physical tests—are almost endless. We may well discover that 500 million spline function evaluations per day is a tiny fraction of what's to come.

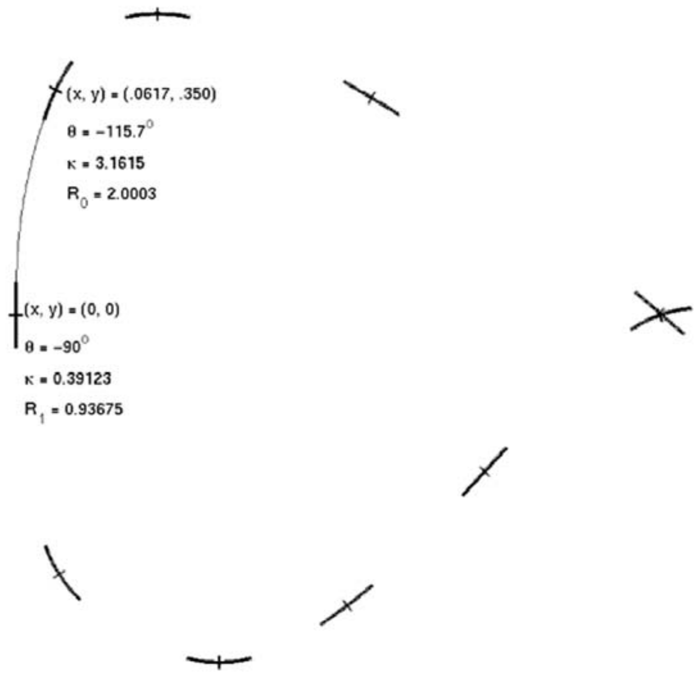


Figure 3. *Sample airfoil data.*

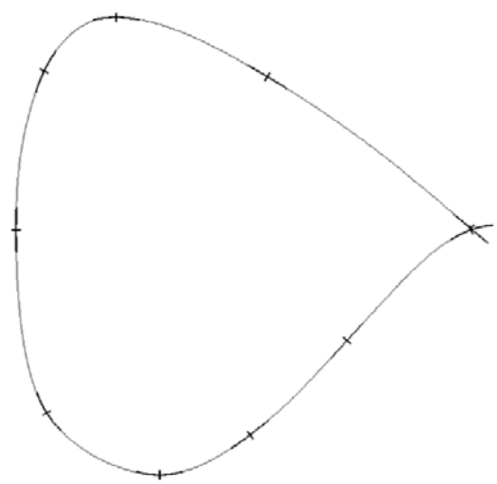


Figure 4. *Resulting airfoil curve.*

Thomas A. Grandine is a Technical Fellow at The Boeing Company.