

BladeCAD: An Interactive Geometric Design Tool for Turbomachinery Blades

Perry L. Miller IV and James H. Oliver
Iowa State University
Ames, Iowa

David P. Miller and Daniel L. Tweedt
Lewis Research Center
Cleveland, Ohio

Prepared for the
41st Gas Turbine and Aeroengine Congress
sponsored by the International Gas Turbine Institute of
the American Society of Mechanical Engineers
Birmingham, United Kingdom, June 10–13, 1996



National Aeronautics and
Space Administration

BladeCAD: An Interactive Geometric Design Tool for Turbomachinery Blades

Perry L. Miller IV

James H. Oliver

Iowa Center for Emerging Manufacturing Technology
Department of Mechanical Engineering
Iowa State University
Ames, IA 50011-2160

David P. Miller

Daniel L. Tweedt

NASA Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135

ABSTRACT

A new methodology for interactive design of turbomachinery blades is presented. Software implementation of the methods provides a user interface that is intuitive to aero-designers while operating with standardized geometric forms. The primary contribution is that blade sections may be defined with respect to general surfaces of revolution which may be defined to represent the path of fluid flow through the turbomachine. The completed blade design is represented as a non-uniform rational B-spline (*NURBS*) surface and is written to a standard IGES file which is portable to most design, analysis, and manufacturing applications.

INTRODUCTION

A wide variety of machines with rotating components incorporate blades for imparting energy to, or extracting it from, various fluid streams. Examples include turbines, pumps, compressors, fans, propellers, etc. In all of these applications, the blade design is critical for achieving optimal overall performance. Since the underlying function of a blade is to smoothly change the velocity of fluid flow, they are generally comprised of parametric sculptured surface models. The complex interaction between the fluid mechanics (i.e., machine performance) and the blade geometry is of fundamental importance in blade design.

Motivation

Experienced blade designers use specialized skills to determine optimal blade geometries. Blade designers represent the blade geometry as a grid of points on the surface of the blade (Figure 1) or with non-standard algebraic equations (Crouse, 1981). In order to manufacture the blade using contemporary computer-aided manufacturing technology it must be represented in a portable format. For this reason the blade is typically regenerated using a CAD package based on point data or other primitive input. However, this procedure can lead to misinterpretation of the data by the CAD operator, who essentially remodels the

blade. Ultimately, errors may be introduced and the original intended performance is sacrificed.

Axial blade design typically begins with two-dimensional stream line curvature through-flow algorithms that aid in determining the blade cross-section geometry from aerodynamic requirements (Law and Puterbaugh, 1982; Crouse, 1981). Input to these programs includes overall aerodynamic performance and basic blade section information. Output is generally a grid of points on the blade surface. The restrictions of these programs are that they will only generate certain types of blade geometry (i.e., axial compressor blades). Casey (1982) describes typical design methodology for centrifugal (radial-flow) blade designs.

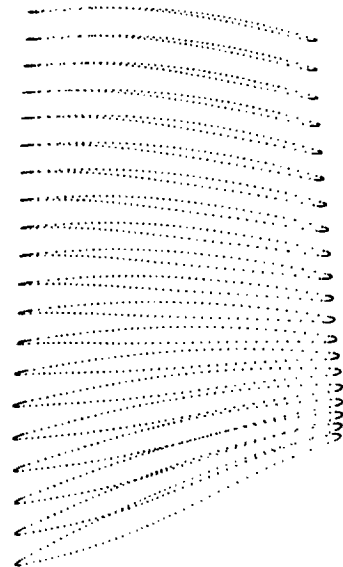


Figure 1. Blade as three-dimensional grid of points

Blades are typically defined by a series of cross-sectional profiles stacked at several radial locations. The profiles (or blade sections) are generally developed on planes, or on concentric cylinders or cones, which are intended to approximate the path of the fluid flow through the turbomachine. The blade sections are related to these surfaces and to one another by a stacking axis, which is generally limited to a line segment.

However, the stream curves generated from stream-line curvature through-flow algorithms are a more precise representation of the actual axial path of the fluid flow. By revolving these curves about the machine axis, general surfaces of revolution can be generated that exactly represent the radial boundaries of the fluid flow (i.e., hub and casing) as well as providing a more accurate representation of the internal paths than cylinders or cones. Throughout this paper these surfaces of revolution are referred to as *stream surfaces* because they are generated from two-dimensional stream curves. Thus, the stream surfaces better approximate the actual three-dimensional fluid behavior.

This paper introduces a method of designing blades that is interactive, allows for blade section development using general stream surfaces and stacking axes, and provides an interface and design methodology familiar to the aero-designer. All the available inputs are geometric by nature but are represented as familiar aerodynamic parameters. Any preprocessing, separate analysis, or designer experience is easily communicated to the blade model through a robust interface. General stream surfaces are used to allow for more accurate and advanced designs. A simple interactive method allows for stream surface generation and editing. Also the stream curves can be read in from a file to allow for compatibility with existing flow analysis codes.

A transformation is developed which allows construction of, and interaction with, two-dimensional blade sections which are simultaneously mapped onto corresponding stream surfaces. Thus, changes of these basic planar curves results in immediate changes to the three-dimensional blade model. Allowing the stacking axis to be a general space curve facilitates very flexible twist and lean capabilities. Ultimately, a robust and transportable surface model is produced. The blade is represented as a non-uniform rational B-spline surface (NURBS) (Piegl, 1991) and the final output is a standard IGES (Initial Graphics Exchange Specification) file (NIST, 1990). The blade surface geometry can then be analyzed by existing computational fluid dynamics packages.

Calculation of the interaction between the fluid flow and the blade is a complex and time consuming problem. When a blade geometry is decided upon, it must go through extensive computer simulations to determine the flow characteristics produced as a result of the interaction of blade and fluid (Chima and Yokota, 1990; Mokhtar and Oliver 1994). Almost always, the desired flow, and hence machine performance, is not what the designer intended. The blade geometry must be changed and the whole process repeated, perhaps several times. Most CAD systems are designed for general purpose surface modeling and allow shape change via direct geometric manipulation such as control point modification. Since shape change based on blade parameters such as inlet and exit angles, chord, thickness distribution, etc., are not generally supported in commercial CAD systems, changing blade geometry can be a difficult and time consuming process. This paper introduces an efficient methodology for creating a blade from scratch and subsequently modifying its design. A simple tug with the mouse on a stream curve or an alpha-numerical entry of a section profile's thick-

ness results in changes to the three-dimensional model. The modified blade is ready for analysis in a matter of minutes.

Terminology

A clear understanding of some basic concepts of blade design is paramount to understanding the need for, and the usefulness of this work. A blade is represented in a cylindrical coordinate system comprised of r -, z -, and θ -coordinates (Figure 2). The blade is oriented such that it revolves about the z -axis in the θ -direction. For axial turbomachinery components, the r -axis is referred to as the blade span direction, and the z -axis is referred to as the chord direction. In an axial flow turbomachine, fluid flows past the blade in the positive z -direction. The axial path of fluid flow can be represented as a stream curve in the r - z plane.

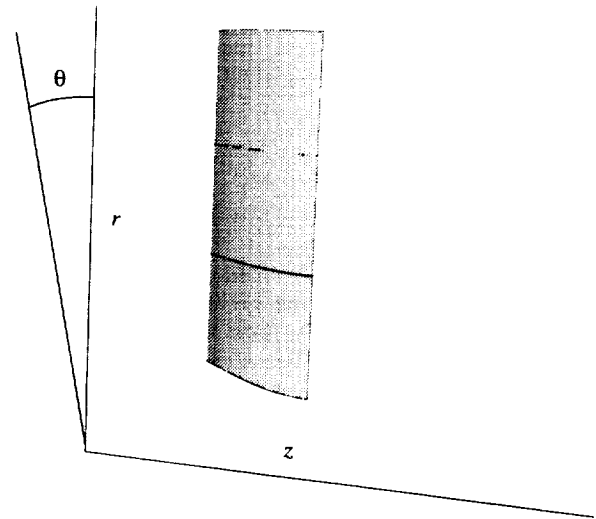


Figure 2. Blade in r - z - θ space

Typically, several stream curves are required to define a blade. Figure 3 shows two such stream curves and one stream surface for both axial and radial blade design applications. The picture on the left-hand side of Figure 3 shows two stream curves and the picture on the right shows the lower stream curve revolved about the z -axis to define a stream surface. The stream surface is a three-dimensional representation of the flow field.

A blade model is generated from several blade section curves which are each related to a specific stream surface by the stacking axis (Figure 4). A stream surface characterizes the desired axial fluid flow through the machine and is defined by a stream curve in the r - z plane.

The blade itself consists of "stacked" cross-sections or blade sections. As shown in Figure 5, the mean camber curve is the "skeleton" of the blade section, and points on the section are generated by offsetting mean camber curve points along corresponding normals by $t/2$ and $-t/2$, where t is the thickness of the blade.

The blade sections are fixed in the r - z - θ space with reference to a stacking axis. The surface definition of the blade is obtained by lofting the blade sections as shown in Figure 2 (Piegl, 1991).

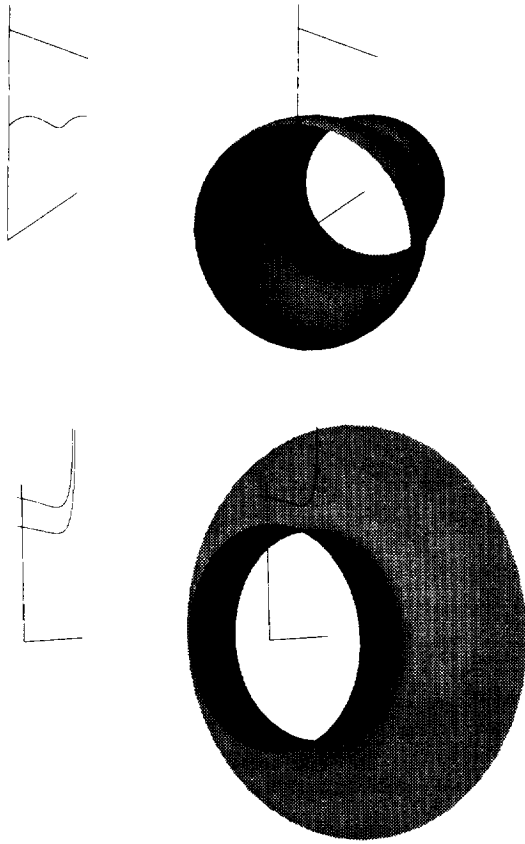


Figure 3. Top: axial stream curves, axial stream surface;
Bottom: radial stream curves, radial stream surface

BLADE DESIGN METHODOLOGY

BladeCAD introduces a new design methodology which is motivated by the need to define and modify blade sections within general surfaces of revolution, while providing intuitive interaction techniques. Direct definition and modification of a three-dimensional space curve, which characterizes both the shape of the blade section and the stream surface, would be extremely difficult. Furthermore, blade designers are accustomed to interacting with blade sections using normalized two-dimensional coordinates. The goal of this work is to provide familiar two-

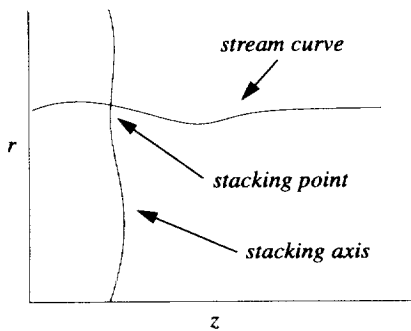


Figure 4. Stacking axis and stream curve

dimensional shape construction tools that allow definition and modification of blade sections precisely within general surfaces of revolution.

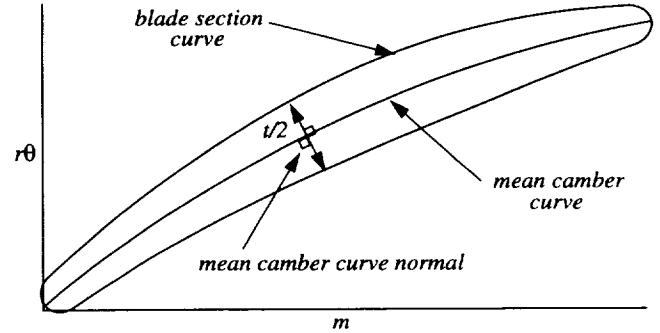


Figure 5. Blade section on mean camber normals

Coordinate Transformation

The stream curve, being a general plane curve, can be characterized by its arc length. The differential arc length dm is defined by,

$$dm = \sqrt{(dr)^2 + (dz)^2} \quad (1)$$

Figure 6 shows the m -direction on the stream surface. Since the stream curve is defined parametrically as a *NURBS* curve (i.e., $r(u)$, $z(u)$) integrating both sides of Eq. (1) results in,

$$m = \int_0^u \sqrt{(r_u(u))^2 + (z_u(u))^2} du \quad (2)$$

where $r_u(u)$ and $z_u(u)$ are the derivatives of $r(u)$ and $z(u)$ with respect to u , respectively.

The $r\theta$ -coordinate is difficult to interpret intuitively, so the m and $r\theta$ -coordinates are normalized with respect to radius. The normalized differential arc length dm' is defined by,

$$dm' = \frac{dm}{r} \quad (3)$$

Again using parametric representation and integrating yields,

$$m' = \int_0^u \frac{m_u(u)}{r(u)} du \quad (4)$$

These relationships provide a transformation between a surface of revolution defined in cylindrical coordinates (r, z, θ) and equivalent two-dimensional spaces $(m, r\theta)$ and (m', θ) . The relationship between the (r, z, θ) and (m', θ) spaces is angle preserving and is useful for definition of critical blade angle properties. The relationship between (r, z, θ) and $(m, r\theta)$ is length preserving and is necessary for construction of blade section profiles.

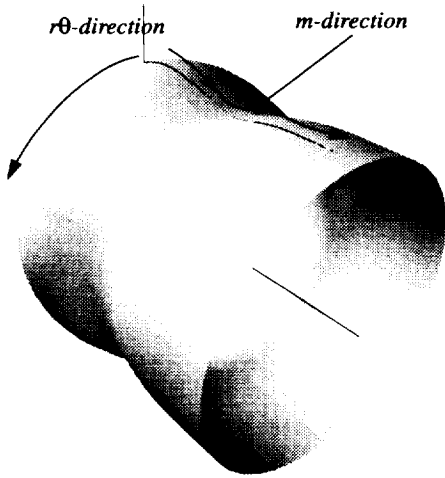


Figure 6. m - and $r\theta$ -directions on stream surface

The above equations are computationally expensive and therefore do not allow for real-time coordinate transformations. When the stream curve is defined, Eqs. (2) and (4) are used to find the corresponding m and m' -coordinates. The m and m' -coordinates are interpolated with the same parameterization as the corresponding stream curve, and the result is stored as a five-dimensional cubic B-spline (i.e., r, z, θ, m, m'). Once the B-spline is created, converting from one coordinate to another is a simple inverse problem with a Newton-Raphson search for the parameter u . Evaluating the B-spline at the parameter u yields the $r, z, \theta, m,$ and m' -coordinates.

Stream Surface and Stacking Axis

BladeCAD accepts any general *NURBS* curve for stream curve and stacking axis definition. The only requirement is that the stacking axis intersect all stream surfaces. This intersection is called the stacking point (Figure 4). In practice stream curves may be read in from *IGES* files generated from previous flow analyses. Alternatively they may be created interactively by selecting points in the r - z plane which are interpolated with a cubic *NURBS* curve. Similarly, the stacking axis may be read from an *IGES* file or created interactively.

Mean Camber Construction

In a two-dimensional space defined by m' and θ , a mean camber curve is defined by the inlet angle (α_i), exit angle (α_e), stagger angle (γ), and the chord (C') as shown in Figure 7. This information is used to represent the mean camber curve as a cubic Bezier curve (i.e., a single segment cubic B-spline). The mean camber curve can be considered the "skeleton" of the blade section. The shape of the mean camber curve (i.e., the incremental distribution of its tangents) is specified in the two-dimensional m' - θ space because this space is angle preserving with respect to r - z - θ space. The transformation preserves angles because both length components are normalized with respect to radius, i.e.,

$$\tan \alpha = \frac{rd\theta}{dm} = \frac{d\theta}{dm/r} = \frac{d\theta}{dm'} \quad (5)$$

The three angles described above are one method traditionally used to fully describe the mean camber curve (Crouse, 1981) Another method involves specifying the desired angle distribution (metal angle) and then integrating it to obtain points on the mean camber which may then be interpolated with a cubic *NURBS* curve. Of course, this method introduces some physical constraints on the metal angle curve, i.e., it can not have vertical segments. Once the mean camber is constructed in m' - θ space it is converted to m - $r\theta$ space so that thickness values may be applied in a length preserving space.

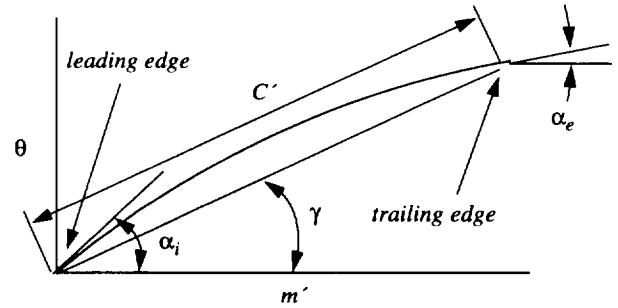


Figure 7. Mean camber definition

Blade Section Construction

The blade section is comprised of a suction side curve, and a pressure side curve (Figure 9). The suction and pressure side curves are computed by interpolating points offset along mean camber curve normals by $\pm t/2$, where t is the thickness of the blade.

The thickness (t) is defined by the following input data: chord-normalized leading edge (t_{le}/C), trailing edge (t_{te}/C), and maximum (t_{max}/C) thickness, and their corresponding locations (s_{le}, s_{te}, s_{max}) relative to mean camber curve arc length. Figure 8 and Figure 9 show how these parameters relate to the thickness function and how they apply to the associated mean camber curve. The thickness function curve is a combination of three splines: a cubic natural spline (US Navy, 1993) that interpolates the three data points, and two ellipses at either end (Figure 8). The thickness function has vertical tangents at both ends. This produces suction and pressure side curves with tangent continuity at their endpoints (Farin, 1988). A joining procedure redefines the suction and pressure side curves as one *NURBS* curve (Piegl and Tiller, 1995)

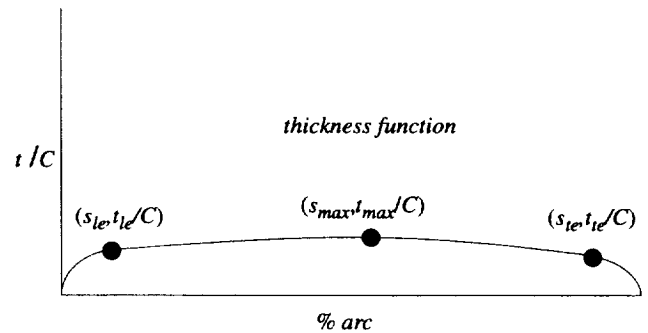


Figure 8. Thickness function

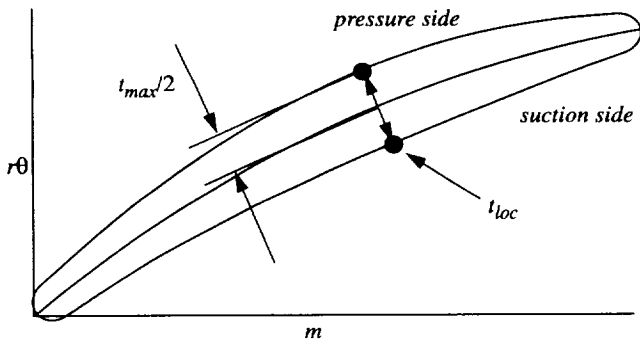


Figure 9. Thickness definitions on mean camber

Mapping Blade Section to r - z - θ Space

After a blade section is constructed in m - $r\theta$ space it is converted to its corresponding stream surface in r - z - θ space via the inverse mapping procedure mentioned above. Thus when the blade section is defined in the two-dimensional m - $r\theta$ space, there is an exact r - z - θ equivalent blade section that lies directly on the stream surface (Figure 10).

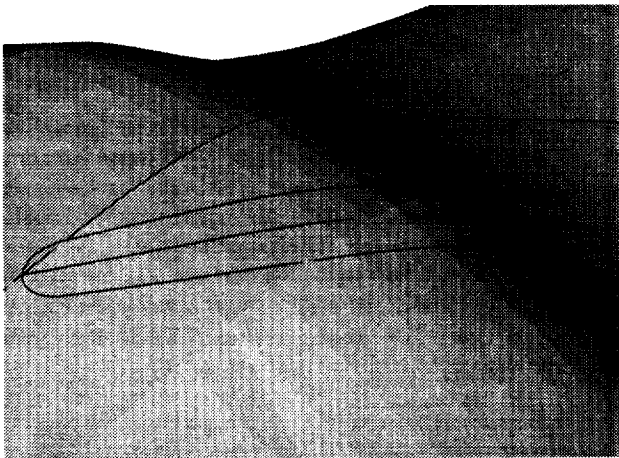


Figure 10. Blade section conforming to stream surface

IMPLEMENTATION

BladeCAD is implemented in an interactive, user-friendly software application. User interface functionality is supported by *OSF/Motif* (Young, 1992) and Silicon Graphics *Open Inventor™* (Wernecke, 1994) is used for graphical display and interaction. The following sections describe the interface components and their functionality.

Main View

The Main View window (Figure 11) has a menu bar with the standard *File*, *Edit*, and *View* menus. The *File* menu contains buttons labeled *New Section*, *Open*, *Save*, and *Quit*. This window displays the assembled blade in a three dimensional space defined by an r - z - θ coordinate system. The stream curves and the stacking axis are also displayed.

Under the *File* menu is a button labeled *New Section*. When pressed, this will prompt the user with an input box asking for the radius of the new stream curve. The new default stream curve will be a cubic B-spline with co-linear control points. The user has the option under the *Edit* menu to drag the control points of any one of the stream curves or the stacking axis in the r - z plane. This will result in a real-time update of the entire blade geometry. It is also possible to drag the stacking axis in the r - θ plane. The *View* menu has various viewing options such as displaying the stream surfaces or resetting to the home orientation. The home orientation is the initial position of the geometry in the window. The user may use the mouse to zoom, translate and rotate. Selecting the home orientation option labeled *Reset* will return the window to the initial view.

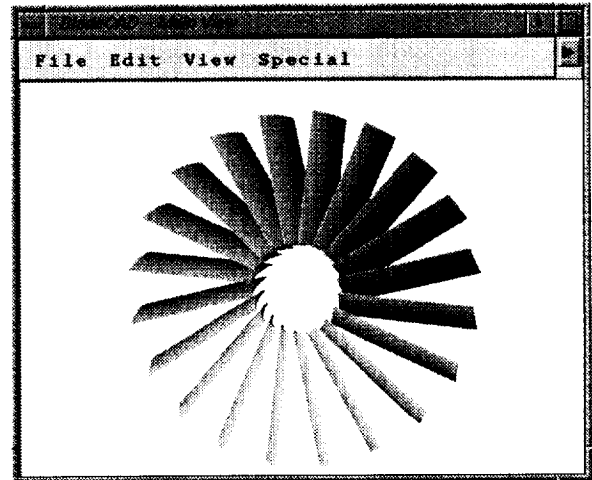


Figure 11. Main View

Input Panel

The *Input Panel* (Figure 12) has a *Section* menu at the top (upper left of Figure 12). This menu has a button to select each section used for the design of the blade. The user can add sections by picking *New Section* under the *File* menu in the *Main View*. The *Input Panel's Section* menu will add a new button accordingly.

While some editing procedures affect the entire blade, most features of *BladeCAD* allow changes to only one section at a time. The *Section* menu allows the user to choose the current blade section. Below the menu bar is an area for entering the thickness function parameters, the mean camber angles, i.e., inlet, exit, and stagger angles (α_i , α_e , and γ), and the solidity (σ), chord (C), and number of blades (N_B).

The bottom of the *Input Panel* has an *Quit*, *Undo*, and *Cancel* button. The *Undo* button will revert the blade parameters back to the previous ones in memory. The *Cancel* button will ignore any new parameters typed in and return the values displayed to the current parameter values in memory.

3.3 Thickness Function Editor

The *Thickness Function Editor* (Figure 13) permits the user to change the thickness function parameters via interaction with the mouse. This window displays the same curve shown in Figure 8. The *Thickness Function Editor* has a menu bar with an *Edit* and *View* menu.

The *Edit* menu allows the user to toggle the editing capabilities. There are nine control points (shown as white spheres), as well as the thickness function parameters represented as interpolated data points (shown as black spheres). The control points can be dragged with the mouse. When a control point is moved the thickness function is re-evaluated with the resulting new set of control points.

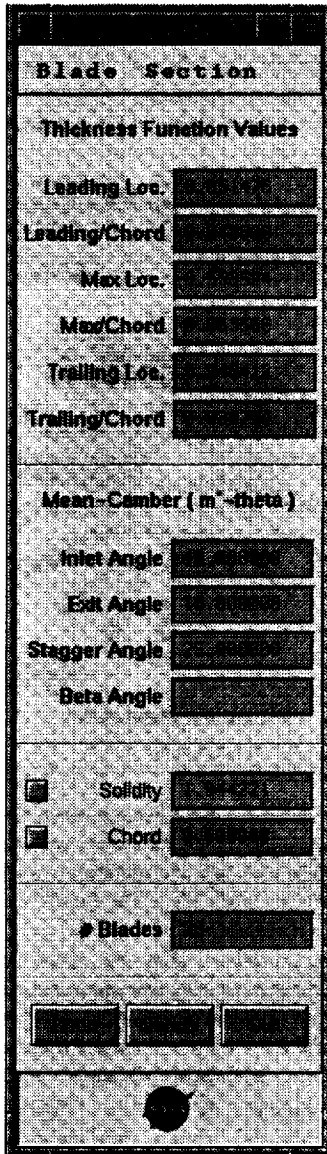


Figure 12. The Input Panel

Alternatively, when one of the three thickness function parameters is moved all three are re-interpolated to make a new thickness function and the corresponding text input fields in the *Input Panel* change to display the new values. Alternatively, the thickness function parameters may be changed alpha-numerically on the *Input Panel* (Figure 12) and the view is updated. The blade geometry in the *Main View* changes accordingly, depending on which blade section is being edited (recall

the *Section* menu in the *Input Panel*). This is one example of the flexibility of *BladeCAD*, the information can be entered in several ways and all related displays are updated automatically. The *View* menu allows the user to resize the control points and reset the display to the home orientation.

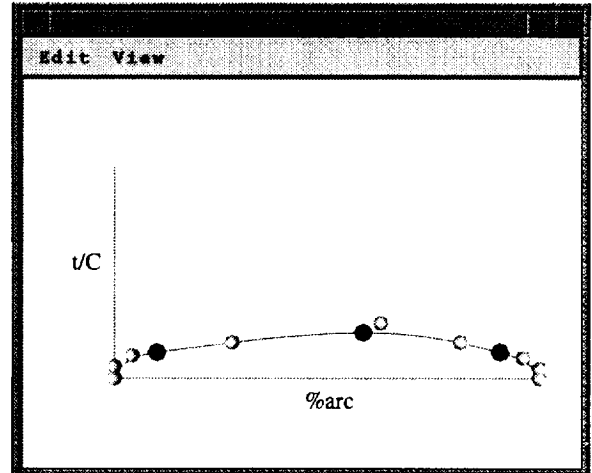


Figure 13. Thickness Function Editor

Section Editor

The *Section Editor* (Figure 14) has a menu bar with an *Edit* and *View* menu. The *Edit* menu allows the user to toggle between editing the mean camber as a cubic *NURBS* curve or as a circular arc. Using a circular arc for the mean camber geometry is a standard method employed by blade designers. Three points are required to define a circular arc. The first point is fixed at the origin. The user can drag the second or third point with the mouse. The inlet angle (α_i), exit angle (α_e), and stagger angle (γ) (Figure 7), as well as the chord (C) and solidity (σ) are automatically updated.

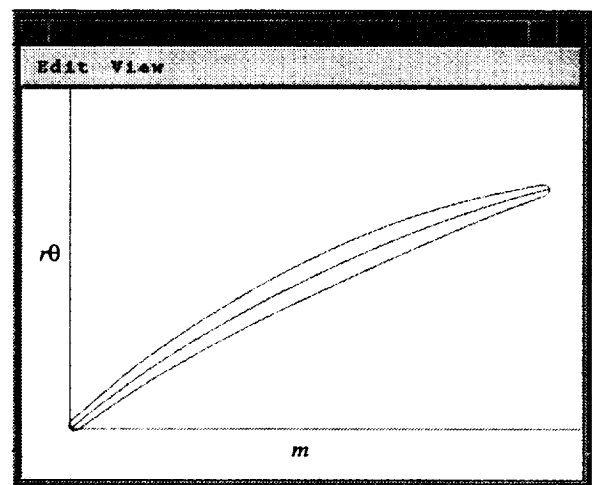


Figure 14. Section Editor

The cubic *NURBS* mean camber curve is defined by two end points and two end tangent vectors. This information comes from the three angles: inlet (α_i), exit (α_e), and stagger (γ). When editing as a *NURBS* curve, the user can drag the mean camber curve control points directly. When a control point is moved the mean camber curve is re-evaluated with the resulting new set of control points. The user can undo any changes to the control points and start again with the mean camber defined from the angles in memory. The *View* menu allows the user to resize the control points, reset to the home orientation, and hide or show the control points.

Blade Metal Angle Editor

The *Blade Metal Angle Editor* (Figure 15) displays the blade metal angle curve and has a menu bar with an *Edit* and *View* menu. The blade metal angle curve is the arc tangent of the derivative of the mean camber curve's θ -coordinate with respect to the m -coordinate ($\text{atan}(d\theta/dm)$ vs. m), essentially a plot of the angle distribution along the mean camber. The *Edit* menu has a toggle for two edit modes, editing the blade metal angle curve as a *NURBS* curve or as a straight line.

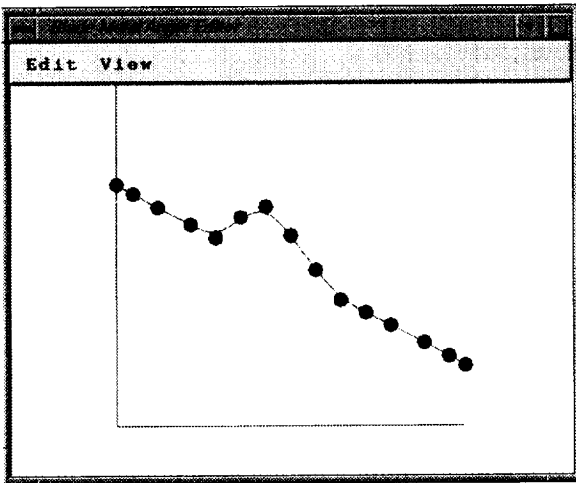


Figure 15. Blade Metal Angle Editor

The first editing option is to edit the blade metal angle curve as a *NURBS* curve. The user drags one of the control points and the curve changes accordingly. Then the blade metal angle curve is integrated to generate a new mean camber. The corresponding modified mean camber curve is then drawn in the *Section Editor*. This happens in real time.

The second editing option is to edit the blade metal angle curve as a line. When this option is chosen the end points of the previous *NURBS* curve are connected with a straight line. The line is modified by dragging one of the end points and any modifications automatically update the shape of the mean camber. The integral of a straight line is a circular arc. This is a second method of editing the mean camber as a circular arc. As before, the *View* menu allows the user to resize the control points and to reset to the home orientation just as in the other windows.

EXAMPLE DESIGN PROBLEM

The following examples are presented to demonstrate the robust capabilities of *BladeCAD*. The examples include a new AST fan blade design, a centrifugal compressor and an axial inducer pump.

Figure 16 shows an AST fan blade row. Thus far in this paper only axial blades (like Figure 16) have been shown. However, the application is not limited to this type of blade design.

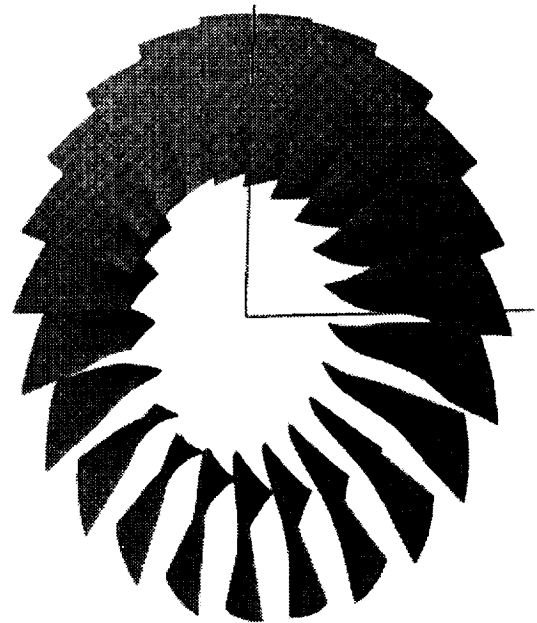


Figure 16. AST fan blade

The radial blades shown in Figure 17 are a good representation of the generality of *BladeCAD*. The same overall design methodology used to produce axial fan blade in Figure 16 was used to create the radial blade shown in Figure 17.

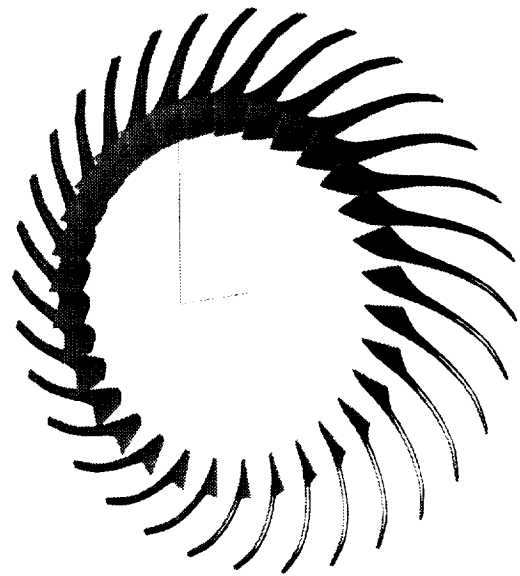


Figure 17. Centrifugal compressor

Figure 18 shows an axial inducer pump blade row. The inducer pump blade is an example of another class of blades that can be easily modeled with *BladeCAD*.

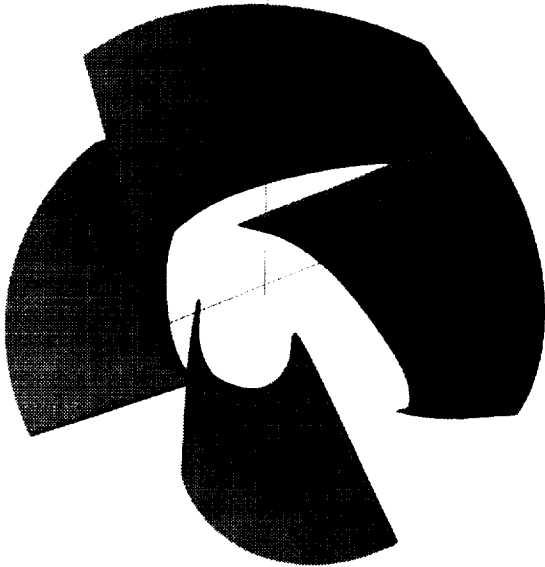


Figure 18. Axial inducer pump

CONCLUSIONS

BladeCAD offers easy creation and modification of blade geometric models using familiar blade design parameters and methodologies. Giving the blade designer the capability of fully representing the blade geometry in the portable IGES format from *BladeCAD* eliminates the previous necessity of having a CAD operator interpret the designer's point data to model the blade in a general CAD package, and also eliminates any related errors. The manual iteration process of designing a blade, testing it for flow characteristics, and making modifications accordingly is still basically the same. However, the modifications can be made quickly and easily in an environment that offers parameters familiar to the blade designer while producing a precise and portable *NURBS* surface model. The general stream surface capabilities enhance existing blade design technology by allowing the designer to better approximate the desired flow field.

BladeCAD is not limited to axial turbomachinery components. Radial flow turbomachinery components such as impellers are designed with the exact procedures mentioned for axial flow turbomachinery components. The application is general enough to design all types of turbomachinery blades.

REFERENCES

Casey, M.V., 1982, "A Computational Geometry for Blades and Internal Flow Channels of Centrifugal Compressors," ASME Paper No. 82-GT-155

Chima, R.V., and Yokota, J.W., 1990, "Numerical Analysis of Three-Dimensional Viscous Internal Flows," *AIAA Journal*, Vol. 28, No. 5, pp. 789-806

Crouse, J.E., and Gorrell, W.T., 1981, "Computer Program for Aerodynamic and Blading Design of Multistage Axial-Flow Compressors," NASA Technical Paper No. 1946

Farin, G., 1988, *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, Academic Press, San Diego

Law, C.H., and Puterbaugh, S.L., 1982, "A Computer Program for Axial Compressor Design (UD0300M)," Report AFWAL-TR-82-2074, Wright-Patterson AFB, OH

Mokhtar, J., and Oliver, J.H., 1994, "Parametric Volume Models for Interactive Three-Dimensional Grid Generation," *Advances in Design Automation*, Vol. 1, pp. 435-442

Mortenson, M.E., 1985, *Geometric Modeling*, Wiley, New York

NIST (National Institute of Standards and Technology), 1990, *Initial Graphics Exchange Specification*, Version 5.0

Piegl, L., 1991, "On NURBS: A Survey," *IEEE Computer Graphics and Applications*, Vol. 11, No. 1, pp. 55-71

Piegl, L. and Tiller, W., *The NURBS Book*, Springer Verlag, Berlin

US Navy, 1993, "DT_NURBS Spline Geometry Subprogram Library Users' Manual," Naval Surface Warfare Center, David Taylor Model Basin, Bethesda, MD

Wernecke, J., 1994, *The Inventor Mentor*, Addison-Wesley, Massachusetts

Young, D.A., 1992, *Object-Oriented Programming with C++ and OSF/Motif*, Prentice Hall, Englewood Cliffs, NJ

REPORT DOCUMENTATION PAGEForm Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE July 1996	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE BladeCAD: An Interactive Geometric Design Tool for Turbomachinery Blades			5. FUNDING NUMBERS WU-505-62-10	
6. AUTHOR(S) Perry L. Miller IV, James H. Oliver, David P. Miller, and Daniel L. Tweedt				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Lewis Research Center Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-10327	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, D.C. 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA TM-107262	
11. SUPPLEMENTARY NOTES Prepared for the 41st Gas Turbine and Aeroengine Congress sponsored by the International Gas Turbine Institute of the American Society of Mechanical Engineers, Birmingham, United Kingdom, June 10-13, 1996. Perry L. Miller IV and James H. Oliver, Iowa Center for Emerging Manufacturing Technology, Department of Mechanical Engineering, Iowa State University, Ames, Iowa 50011-2160 (work funded by NASA Grant NAG3-1318); David P. Miller and Daniel L. Tweedt, NASA Lewis Research Center. Responsible person, David P. Miller, organization code 2760, (216) 433-8352.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 07 This publication is available from the NASA Center for AeroSpace Information, (301) 621-0390.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A new methodology for interactive design of turbomachinery blades is presented. Software implementation of the methods provides a user interface that is intuitive to aero-designers while operating with standardized geometric forms. The primary contribution is that blade sections may be defined with respect to general surfaces of revolution which may be defined to represent the path of fluid flow through the turbomachine. The completed blade design is represented as a non-uniform rational B-spline (NURBS) surface and is written to a standard IGES file which is portable to most design, analysis, and manufacturing applications.				
14. SUBJECT TERMS B-spline; NURBS; BladeCAD			15. NUMBER OF PAGES 10	
			16. PRICE CODE A02	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	

**National Aeronautics and
Space Administration**

**Lewis Research Center
21000 Brookpark Rd.
Cleveland, OH 44135-3191**

**Official Business
Penalty for Private Use \$300**

POSTMASTER: If Undeliverable — Do Not Return