An Automated Approach to Feature-Based Design for Reusable Parameter-Rich Surface Models

Jason H. Elliott[1], Courtney L Berglund[2] and C. Greg Jensen[3]

[1]Pratt & Whitney, jason.elliott@pw.utc.com
[2]Brigham Young University, cberglund@byu.edu
[3]Brigham Young University, cjensen@byu.edu

ABSTRACT

In striving to reduce the cycle time required to deliver quality products, many engineering groups use CAD/CAM/CAE applications to design and analyze their products' models. Feature and constraint-based design concepts for solid modeling applications have been embedded in third-generation CAD systems to reduce the time required to produce virtual computer models. While very advantageous in solid modeling, complex surface models do not directly benefit from these parametric design methods. This paper presents a methodology that applies feature-based parametric design to a class of products known as parameter-rich surface models (PRSMs) by coupling knowledge of parametrics and feature-based design with high level programming and automation techniques. We describe efficient implementations of this approach to show feature and constraint-based design can be applied to PRSMs. Results show that following this automated approach will serve to reduce design time and integrate data from separate domains into a seamless design process to prepare high-fidelity CAD models for CAM/CAE applications.

Keywords: CAD, parametrics, feature-based design.

1. INTRODUCTION

Engineering research and development teams aim to reduce the time required to design and analyze parts. Many of these groups use advanced technology in the form of commercial CAD/CAM/CAE systems, parallel processors, and supercomputers. This advanced technology has helped to reduce design time as well as improve overall part quality. Since the 1980's, feature-based design and parametric methodologies have opened the door for the use and reuse of design and analysis models. In a CAD model, feature-based design and parametric technology specifies that features be created from primitive sets of lower order features and edited by manipulating their assigned parameters. When parameters are varied within a parametric CAD model, a new design or instantiation is generated without recreating the part model. This methodology allows for added flexibility to the design and configuration of CAD models, thus reducing overall design time.

Designing reusable models with flexible parameter schemes, however, is not the only advantage of using parametrics. In 1995, parametric design methodologies were extended to include complexity as well as flexibility in the modeling of solid objects [1]. Nearly all approaches to address model complexity have been implemented by reducing the number of driving parameters necessary to describe a part model. While designing CAD models with fewer driving parameters offers many advantages in solving engineering problems, feature-based design for surface models with multiple driving parameters is a complex problem that has not been greatly explored [4]. Hoffman and Kim [6] provided some insight in this area and stated that when parametric design is used for highly complex models, the instantiation of a new part with new parameter values often fails. They attribute part of the failure to the parametric schema used to describe the shape and introduced a method that formulates valid parameter ranges for solid parametric models. However, for the case of complex surface geometry with multiple parameters, they offer no solution.

Along with creating geometric models interactively, models can also be created programmatically using a high-level programming language of the CAD API [12]. In concept, CAD APIs were seen as early as 1980 [28]. Since then, abundant research has been conducted in parametric programming (see [2], [4], [19], [10], [11] and [14]). Of the works cited on programmatic CAD in industry, only two articles show work relating to PRSM features inside a parametric CAD-based environment. Tang and Chang [14] automatically generated and manipulated spline points, curves, and surfaces. However, their contribution was not in applying feature-based design to surface models, but rather in simplifying the models by reducing parameters—a conventional technique commonly employed today. Haimes and Crawford [4] looked at an increasing number of parameters within a programmatic environment; here they worked with models having roughly 130 driving parameters. Although this number of design variables is impressive, it still falls short of those found in typical complex surface geometry models.

While interactive and parametric methodologies derived from feature-based design have provided advantages to solid modeling (see [1], [6], [13]), there are still circumstances that do not directly benefit, such as when:

- Large amounts of data are required as inputs.
- CAD models require free-form surface construction.
- Features are difficult to parameterize.
- Physics-based analysis models require high-fidelity geometry.
- Design and analysis iterations take place on the analysis model, causing difficulties incorporating nodal perturbations into the CAD model.

Part families exhibiting these problematic characteristics reside in a class of products that set them apart from classical low order parametrics. Here we focus on this class of products, called parameter-rich surface models (PRSMs). PRSMs take months to iterate from design to design and often, an optimum is not found in time for production. This article summarizes an introduction to a feature-based and parametric methodology to automate the geometric and topologic creation of models within the PRSM domain.

The methodology of applying feature-based and parametric methods to solve problems in the PRSM class includes three steps: (i) planning, (ii) development, and (iii) evaluation. The planning stage involves the product-specific identification of inputs and feature parameterization schemes. Step two is the development of a design application called the Automated Parameterization Tool (APT). The product specific APT, programmatically generates the surface features and objects in the CAD model according to the parameterizations developed in the planning stage. Finally, the created surface geometry is evaluated for connectivity between adjacent surfaces with simple routines to prove the geometry is mesh-worthy and satisfactory for downstream applications, such as finite element analysis (FEA) or computational fluid dynamics (CFD). Also, the evaluation portion also verifies that the overall process meets the required standards as set out in the planning stage.

This paper will outline the steps necessary to implement this methodology on any product from the PRSM class. As a point of reference, a case study will be given of the methodology applied on an example product from the PRSM class, a jet engine shroudless hollow fan blade. Implementation of this methodology for other products within the PRSM class would follow a similar approach, with a new product-specific Automated Parameterization Tool (APT) being developed for the respective product in question.

2. PROGRAMMATIC SURFACE MODELING

Research in the area of feature-based design and parametric technology has reduced the time required to design, analyze, and manufacture a wide variety part families. Work relating to the application of these principles to parameter-rich free-form surfaces, however, is limited in describing its benefits to any given community. Here, methodology is introduced that applies feature-based and parametric design concepts to the automated parameterization and geometry creation of PRSMs.

2.1 Planning

In order to apply feature-based concepts to a new class of products, planning is required to map out the precise amount of work to be done in the development stages. These stages include: (1) identifying the necessary inputs to the product specific APT; (2) designing a modeling strategy for the specific product in question, and (3) creating a parameterization scheme for the respective surface model features to be created.

2.2 Development

Once the planning stage is complete, the Automated Parameterization Tool (APT) must be developed. The APT is the design application tool comprised of multiple algorithms developed to automatically parameterize and create geometry for the PRSM in question. Developing an APT for a product class such as PRSMs helps to transform the difficult characteristics of the class into non-cumbersome attributes describing its design requirements. It is important to note that by using the APT to generate a complete CAD model for a given set of input parameters, the PRSM changes into a reusable product model.

This methodology prescribes that the APT consist of algorithms written using a high level programming language and a third-generation CAD system API. In developing the APT, algorithms will be designed to implement each task of the surface model generation. The necessary code fragments will be organized according to their individual function, and necessary mathematical formulas and equations designed for the features which require them.

2.3 Evaluation

To verify an APT is effective and the product created by the APT meets design criteria, various new algorithms must be created depending on the product in question. The type and amount of evaluation procedures that need to be developed to test the end models for mesh worthiness depends largely on the actual product chosen from the PRSM class. In the case of the hollow fan blade APT, a variety of algorithms were created to evaluate the resulting fan blade models for various capabilities including mesh worthiness. The evaluation criteria, along with actual results are discussed in section 3.3.

3. IMPLEMENTATION

The PRSM test case used to validate this methodology was a jet engine shroudless hollow fan blade (Figure 1). The hollow fan blade fully embodies the PRSM characteristics in that it requires free-form surface construction and large amounts of data as inputs, contains non-reusable features that are difficult to parameterize, requires high-fidelity geometry for mesh generation, and is part of a complex and lengthy multidisciplinary process. As previously stated, application of the following methodology can be applied to any member of the PRSM class by following a similar approach. Implementation on a different PRSM however, would require a new product-specific Automated Parameterization Tool (APT) to be developed in conjunction with the new PRSM.

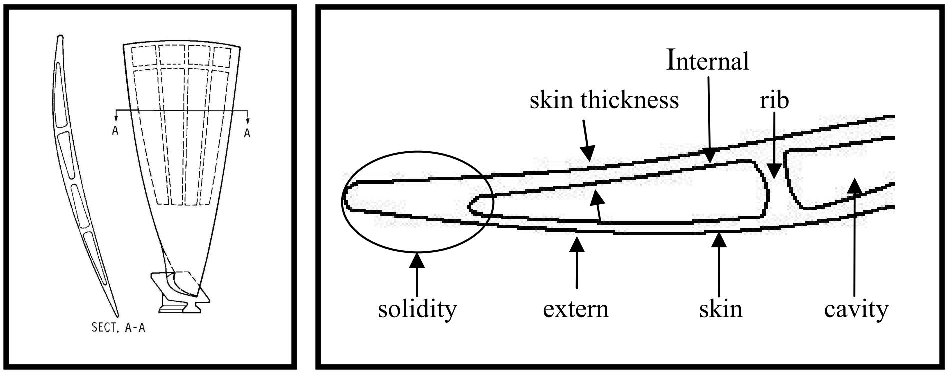

Fig. 1: A jet engine shroudless hollow fan blade with a swept external shape and simple internal configuration.

3.1 Planning

As discussed in section 2.1, the first step of implementation is the planning stage which consists of identifying necessary inputs, designing a modeling strategy and developing a parametric scheme.

3.1.1 Inputs

Identifying inputs to the APT helps determine the necessary upfront time to prepare the input data by quantifying exactly how much work is to be done in the development stage of the method. While specific inputs for the APT vary due to the PRSM in question, possible inputs could include, product definition test files, excel spreadsheets, in house program output files or base CAD models. Depending on the type and number of required inputs for the specific PRSM, specific algorithms must be developed to import and store the data within the APT database.

In the hollow fan blade test case, the required inputs are airfoil definitions in bulk point aero source (AS) files and a Unigraphics (UG) root attachment part file. In general, an AS file is a text file generated by aerodynamicists prescribing an airfoil shape by hundreds of Cartesian coordinate points. While the AS files used to validate this methodology do not represent actual proprietary industry designs, the B-Spline curves and surfaces generated are of the same mathematical form as those typically used in industry. To import the data, algorithms for the APT were developed to scan in the driving parameters from the AS files and store coordinate point information within the APT database. The second input to the APT is a UG root attachment part file. In industry, root attachment geometry does not vary significantly from engine-to-engine. The APT requires 2D sketch features used to create the root body, and datum planes used to define the trimmed leading and trailing edge faces. Inside the APT, the pointers to these objects are obtained by using UG/Open algorithms to query the object names or labels.

3.1.2 Strategy

Outlining the design strategy not only includes defining the creation sequences of CAD model geometry, but also developing the strategy involved in the design and using application user function algorithms. Also, the different disciplines involved in the design loop should be involved and rules and constraints guiding the holistic view of the model should be defined. When using a programmatic modeling approach, the created CAD model must use pre-selected features, configurations and joints resulting in a model exactly similar to one created interactively.

As with defining the inputs, strategy definition also depends on the specific PRSM in question. The hollow fan blade test case is particularly complex and the design is created in two separate domains. The first domain is a UG part file with a flat hollow fan blade representation. The flat domain provides a 2-dimensional layout where the internal cavity definition is defined. The internal layout is defined using Bezier curves and B-splines to represent the flat rib configuration. The second domain is a UG part file containing a twisted fan blade configuration, or the actual performance surface design. Once the internals are designed in the flat domain, the splines are translated to the twisted domain where the hollow fan blade surfaces are generated. For this study, the internal rib configurations generated by the hollow fan blade APT are non-proprietary configurations taken from designs used by Kielb [7], i.e. simple waffle designs with near rectangular shape. Figure 2 shows the strategy schematic for the automated free-form surface generation within the two separate domains.

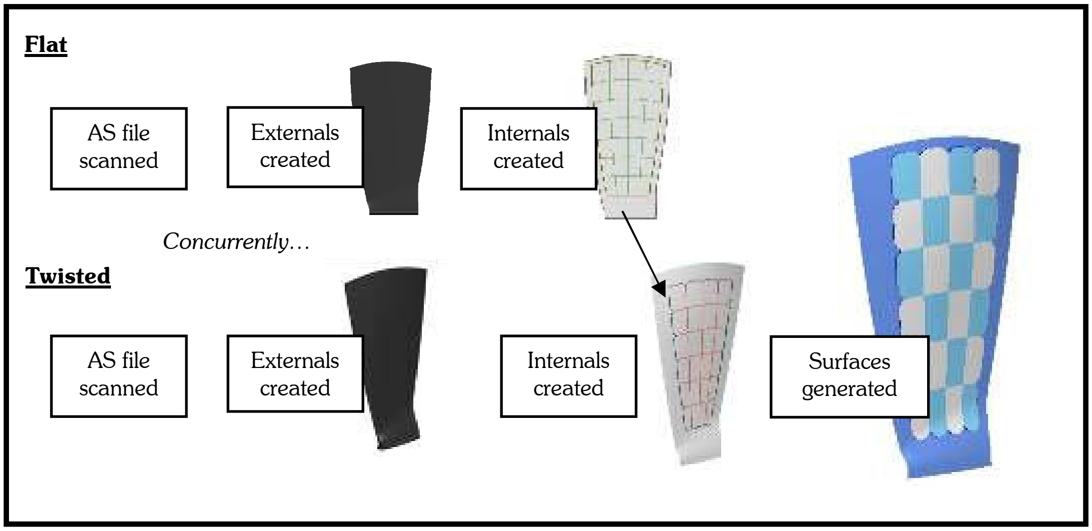

Fig. 2: Design strategy schematic for the hollow fan blade APT.

3.1.3 Parameterization

As were the previous steps within the planning stage, setting up a parameterization scheme is entirely dependent on the specific PRSM chosen. Of particular importance here is verifying the CAD model created by the APT is exactly similar to one created interactively. Understanding how the PRSM is created interactively with the product-specific inputs is paramount in defining the parameterization scheme for the specific PRSM.

To parameterize the hollow fan blade, the main hollow fan blade features were described and organized in a feature modeling tree showing their proper creation order. With exception of the input root body, the hollow fan blade APT creates all the geometry and parameterizations for each feature of the hollow fan blade. For a more detailed description of the parameterization of the hollow fan blade features, see [3].

3.2 Development

The hollow fan blade test case APT was developed using C/C++ coupled with native toolkit functions from the UG/Open API library. The design tool is an internally executed program that runs as a dynamically linked library inside the UG interface. The design tool automates the generation of hollow fan blade product models and applies parameters and constraints to detailed surface features. There are seven main algorithms within the hollow fan blade APT architecture that perform various functions and sub routines (for detailed descriptions of each algorithm, see [3]). While the algorithms created for the test case are specific to the hollow fan blade, a similar organization scheme could be used for any PRSM in question with specific algorithms developed to read in and store the input data and other algorithms developed to create CAD geometry from the input data based on the design strategy and parameterization scheme developed during the planning stage.

SThe hollow fan blade APT's controlling program, MAIN, calls each of the algorithms and controls the feature creation of both the flat and twisted domains. MAIN provides organization to the sequence of algorithms called and surface models generated. MAIN first opens the twisted root body part and then, after the flat part model completes the necessary algorithms, passes the uv-data to the twisted domain. Also developed in this research was the technology allowing near simultaneous operations to be performed in separate domains. Even when UG has changed display to the twisted domain, the data associated to the flat hollow fan blade features are within scope and data could still be passed upstream to the flat model. For a different PRSM, the APT's controlling program would be very similar to MAIN in the test case, serving to provide organization to the sequence of PRSM product-specific algorithms called and surface models generated.

3.3 Evaluation

The APT is a design tool that creates PRSM geometry and prepares it for mesh generation. For the hollow fan blade test case, validation criteria are applied in the next section and the required capabilities and qualities of the APT and resulting product models discussed. While evaluation procedures to prove whether an APT is effective will differ depending on the specific PRSM, the following questions apply to any APT created for a product within the PRSM class and help define the evaluation procedures for the APT, namely, does it:

- Allow for a large data input?
- Automatically parameterize free-form surface geometry?
- Generate reusable surface models with high-fidelity geometry?
- Build and improve upon current design processes?

4. TEST CASE RESULTS

In introducing a feature-based and parametric design methodology to the PRSMs, namely a jet engine shroudless hollow fan blade, an Automated Parameterization Tool (APT, ~7000 lines of code) was developed to automatically parameterize and generate hollow fan blade geometry. Executing the APT within the UG interface causes the creation of hollow fan blade PRSMs with a high number of independent parameters controlling the main surface features. To evaluate the effectiveness of the APT in generating hollow fan blade part models, seven case studies were completed. Each study was executed with different input parameters. Table 1 shows the input parameters for each case study, with variability from study to study in blue.

Study No.	**Airfoil shape**	**AS file**	**Root geometry**	**Rib grid**	**EWB**	**MOV**	**Offsets**
1	Simple	100 x 10	Normal	3 x 6	Normal	10 x 40	Normal
2	Complex	150 x 30	Normal	3 x 6	Normal	10 x 40	Normal
3	Complex	150 x 30	Different	7 x 15	Normal	10 x 40	Normal

4	Simple	100 x 10	Normal	5 x 9	Different	10 x 40	Normal
5	Simple	100 x 10	Normal	5 x 9	Normal	10 x 40	Divided
6	Unconventional	150 x 30	Normal	4 x 7	Different	10 x 40	Divided
7	Unconventional	150 x 30	Normal	4 x 7	Different	20 x 80	Divided

Tab. 1: Input parameter matrix for APT case studies, comparisons in blue.

The AS files generated for the case studies consisted of *simple*, *complex*, or *unconventional* with coordinate sets of 100 x 10 or 150 x 30 sections by points per section. The root part models used were characterized as either *normal* or *different* geometry. In addition to the two main inputs, the different scenarios also varied the internal designs. The informal variables used to vary the internal geometry included:

1. Grid dimensions of rib mid splines designed in flat configuration.
2. Discrete description of end wall boundary, normal or different.
3. Grid dimensions of the matrix of values (MOV) for the parameterization of the skin mids.
4. Offset values that fill the cells of the MOV, normal or different.

The variations on the end wall boundary descriptions were set by values controlling the leading, tip, trailing, and bottom offset edges of the mean line surface (MLS). The MOV was set and filled by an external program with a MOV grid of either 10 x 40, or 20 x 80 columns by rows. The MOV program also filled the cells in the matrix according to span height or external parameter value. A *normal* set of offset values was considered to taper linearly from the root of the HFB to the tip. A *different* set of offset values also tapered linearly from root to tip, but for every other matrix column, the values were divided in half, to show variability in the offset skin mids.

The evaluation procedures outlined in section 3.3 were the questions used to determine whether the APT contained the necessary attributes to fully accomplish the task of transforming the problematic characteristics of the PRSM class into simple non-impeding attributes. Here, those questions are answered.

4.1 Data Input

To effectively handle large amounts of input, the hollow fan blade APT must build hollow fan blade surface models that allow for a variety of AS files and root body part files. As shown in Table 1, test scenarios were conducted varying both size of the input AS files and topology and location of input root bodies. A comparison between case studies 1 & 2 verifies the hollow fan blade APT was successful in allowing large numbers of coordinate points to describe the airfoil definition. The difference between the two studies was the airfoil configuration and number of AS coordinate points. Typically, no greater than 4000 points are used to describe a commercial fan blade airfoil in industry. In the various case studies, the APT exhibited the capability of handling 1000 to 4500 coordinate points. These results prove the APT was successful in generating surface features from a large data input.

4.2 Reusable Models

The reusable model paradigm allows for the manipulation of CAD models to create new designs or instantiations, rather than creating them from scratch. In the programmatic approach, to reuse a model is to execute a user function that automatically creates a new design with new model design variables. In the hollow fan blade test case, the APT was developed to automatically create reusable models by assigning parameters, constraints, and relationships to newly created surface geometry. The automated parameterization of the APT was evaluated by using different MOV's defining the internal geometry of rib mid splines and the skin mids offset values, to test whether the APT was capable of creating reusable models with different feature parameterizations.

The results and reusable models created are best described by analyzing comparisons between case studies 4 & 5, 2 & 3, and 6 & 7. Studies 4 & 5 both had simple airfoil definitions with an equal number of rib mids and varied end wall boundary offset dimensions. The hollow fan blade APT easily accomplished the dimensional variation in the end wall boundary features by following simple design variables in an internal dimensions parameter file. The rib mids were generated throughout the domain of the cavity region regardless of the location of the end wall. Another difference was the MOV offset values for the skin mid parameterization. The skin mids of study 4 were created with tapered values offset from the externals, while study 5 contained offset values that tapered span-wise with additional offsets that varied with grid location. The distances used in the variation were twice those of the normal tapered values used in the specific grid location.

The parameters changed for studies 2 & 3 were location and geometry of the root body, and the number of internal ribs. Figure 3 (a) shows the resulting part models of case studies 2 & 3, highlighting the different parameters between the two cases. The root body of case 3 was located slightly more to the left, rotated counter-clockwise around the z-axis, and contained slightly more curvature than case study 2. Notice the continuity of the externals as they extend perpendicular from the top face of each root. The starting knot points of each span-wise B-spline of the externals are located at the top of the root. This continuity example shows that for different root part models, the APT set parameters and constraints governing free-form surface features. Case studies 2 & 3 also provide a difference in the internal layout of the rib mids. Figure 3 (b) shows CAD models of the two cases. By varying the number of rows and columns of the rib mid grid in an input file, the APT automatically generated variations of the rib mid splines thus showing the reusability of the parameterization schemes embedded in the code.

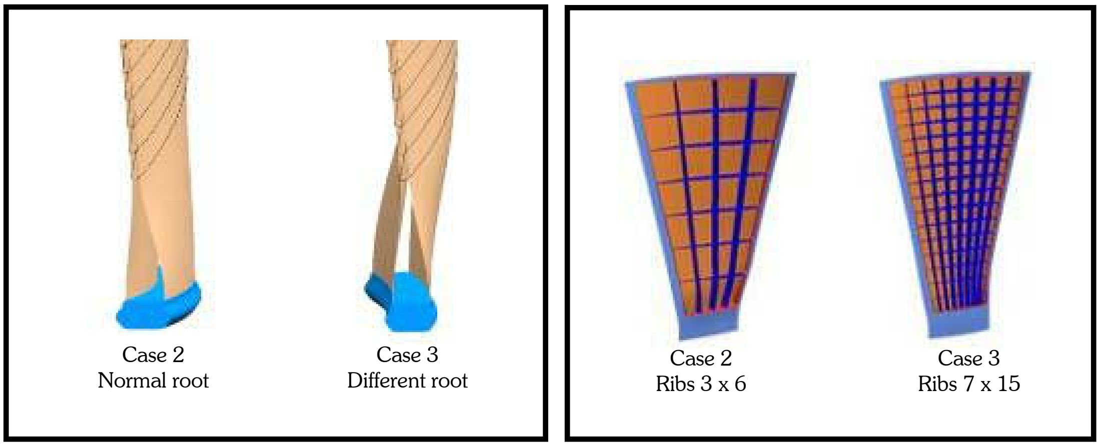
Case 2
Normal root
Case 3
Different root
Case 2
Ribs 3 x 6
Case 3
Ribs 7 x 15

Fig. 3: Different root bodies show (a) continuity constraint of externals and (b) rib grid variations.

In the final studies, 6 & 7, different MOV node grids were used. Of particular interest here were the AS files used. These AS files used a more complex definition to reiterate the robustness of the algorithms developed for the hollow fan blade APT. Figure 4 (a) shows a CAD model representation for both studies, with suction side skin mid removed for clarity and a span-wise slice of each test case shown in Figure 4 (b).

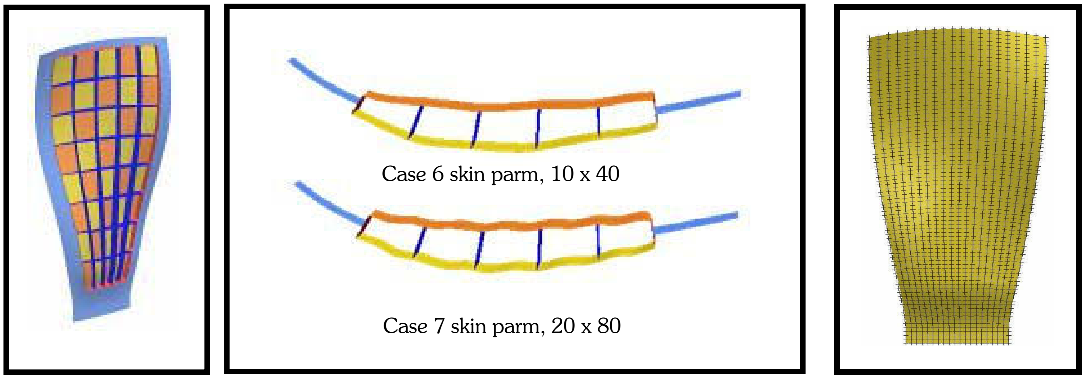
Case 6 skin parm, 10 x 40
Case 7 skin parm, 20 x 80

Fig. 4: Studies 6 & 7 (a) CAD model, (b) different skin mid parameterizations and (c) the MOV grid on the pressure side external for case 7.

The 7th study resulted in skin mid features with many more driving parameters than case 6. Figure 4 (c) shows the MOV grid overlaying the pressure-side external for study 7. The grid dimensions of the MOV were automatically changed based on the skin mid parameterization file. This varied parameterization shows not only that multiple parameterizations are possible within the same product design application, but also that a CAD model can be created with more than 3200 driving parameters. Test case 7 had 6400+ independent parameters controlling the skin mid locations and shape.

These results show the hollow fan blade APT was capable of automatically assigning constraints and relationships to PRSM geometry with a variable number of controlling parameters with upper limits exceeding all comparable models in related research. These results also show an APT has the ability to reuse parameterization schemes for different data sets and models can be instantiated from abstract data structures residing within APT algorithms.

4.3 High-fidelity Geometry

To verify the hollow fan blade APT created consistent, high-fidelity geometry, an evaluation algorithm was developed to check surface geometry for slivers, self-intersections and gaps. Another routine was created to calculate the deviations between all adjacent edges for each twisted surface model created. To further test the created geometry, deviations between the end wall and pressure-side skin mid coincident edges were also calculated and checked for accuracy. To rate the fidelity of the generated PRSMs, deviations for each of the case studies were compared to .001 inches. The APT was considered to be successful if 99% of total maximum deviations within each model generated were less than the stated metrics. For other PRSM applications these evaluation metrics would be specified by the element size of a finite element model used in analysis. As long as the CAD model is water-tight within a certain tolerance, the analysis software can discretize the domain without loosening the tolerance to a size that defeatures the model.

The results for the test cases revealed that all maximum deviations fell well below the metric of 0.001 inches. The maximum deviations, shown in chart 1, occur at the end wall vs. rib mids intersections and are 0.076, 0.104, 0.154, and 0.098, thousandths of an inch respectively. In the chart, the metric was truncated and the deviations visually amplified to provide a better view of any correlations.

Each of the four deviations mentioned is more than 80% less than the metric value, and actually follows a trend from the input parameters. Case studies 2, 3, 6 and 7 were generated with higher definition AS files, at 150 x 30 points, versus the other cases which were generated with 100 x 10 point AS files. As all the deviations measured for the case studies resulted in values substantially less than the 0.001 inches metric, it follows that the hollow fan blade APT was able to generate high-fidelity geometry sufficient for mesh generation.

4.4 Current Process

In order to measure whether the hollow fan blade APT builds and improves upon current design practices, it was compared to existing design practices currently used by hollow fan blade engineers at Pratt & Whitney. As formal inputs, the APT uses a root part model and AS file for the flat and twisted domains. The APT builds on current design practices due to the fact that the engineers at Pratt & Whitney use master model root parts with similar parameterization schemes, as well as similar formatting of AS files. Improvements made by the implementation of the APT for hollow fan blade surface models over current design practices include:

- More complete and higher definition of surface geometry.
- Seamless integration of engineering models from separate domains.
- Automatic evaluation of model quality.

The hollow fan blade APT transfered data in text files by seamlessly connecting them to the design process. The files integrated were the AS file and the MOV (matrix of values). In industry, a file similar to the MOV is normally set by perturbations from an analysis model. The time consuming work of creating transitions by hand was also addressed through algorithms which automatically create transition features preserving aero definition and maintain associativity to root geometry. In industry the process of designing and analyzing hollow fan blade designs takes months. The improvements listed above add advantages not previously enjoyed by current practices. Table 2 provides a quantitative comparison of the current hollow fan blade practice versus the new approach offered by executing the APT.

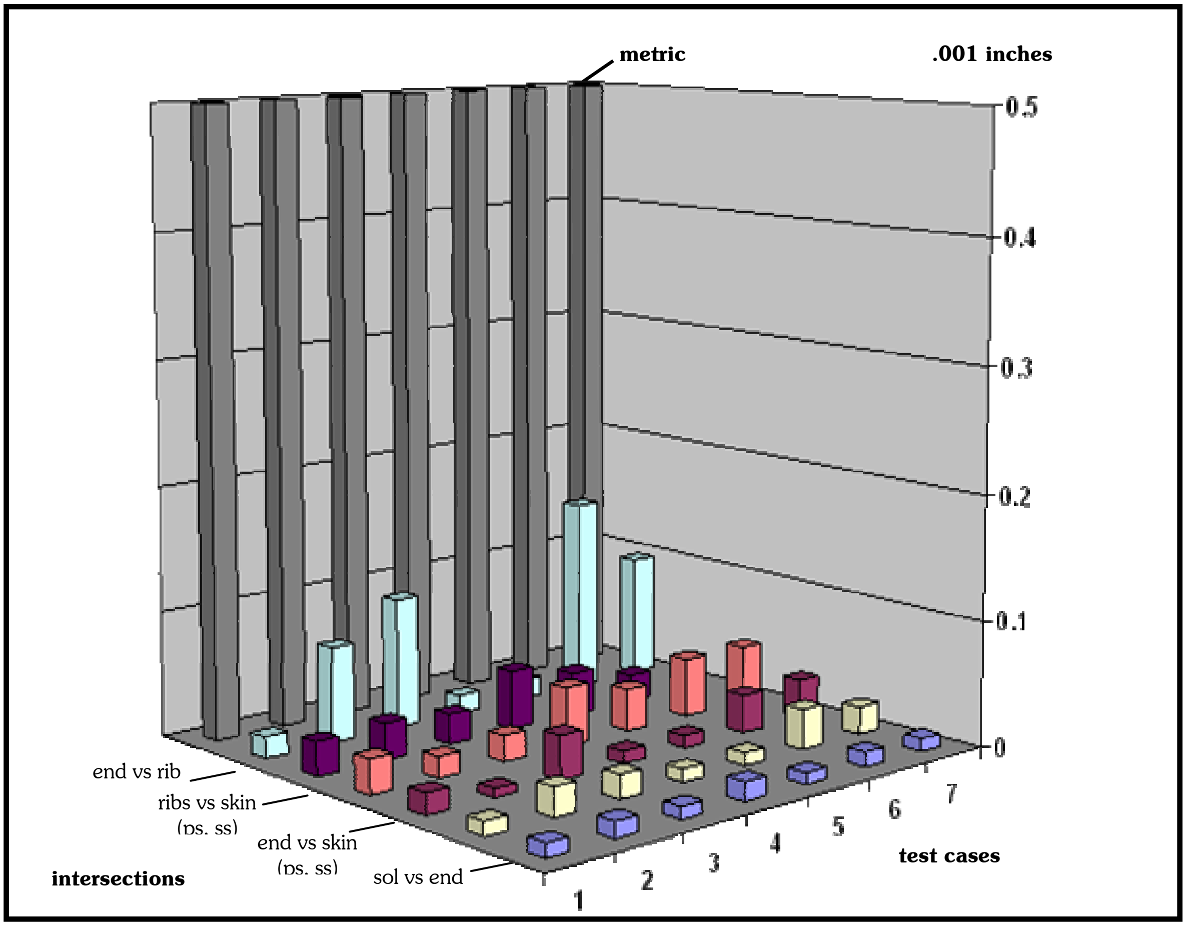

Chart 1: Maximum deviation versus intersection for the test cases 1-6.

Tasks	**Current Process**	**New Approach**
Setup/Development		3-4 weeks
Flat & Twisted model creation		
Read AS file/Create Externals	2 days	< 1 min
Create MLS	2 days	< 1 min
Flat model creation		
Create EWB/RMS	2 weeks	< 1 min
Twisted model creation		
Create EWB/RMS	2 days	< 1 min
Create end wall/rib mids	1 day	< 1 min
Create solidity/skin mids	2 days	< 1 min
Evaluation	1 day	~5 min
TOTALS	**~4 weeks**	**~4 weeks**
Succeeding Iterations	**~4 weeks**	**~30 min**

Tab. 2: Comparison of time spent in current design process at Pratt & Whitney versus the new approach.

To an experienced engineer who understands high-level programming and CAD APIs, the time spent to develop an automated parameterization tool for a complex surface model would take approximately 3-4 weeks. Naturally, for projects with much larger scales and assembly sizes, like an entire jet engine, development time would take several years and tens of thousands of man-hours. For an inexperienced designer who does not know the workings of programmatic parametrics, developing an APT would be a daunting task. Table 2 however, shows that even with an initial investment time of 3-4 weeks to develop the APT, the 1st design model iteration would take roughly the same amount of time as the current process. However, in the time it takes Pratt & and Whitney engineers to generate 10 designs and prepare them for simple analysis, the APT could generate thousands of individual designs, including time for development, debugging, and evaluation of each design.

5. CONCLUSIONS

The implementation of parametric methodologies using the programmatic CAD paradigm applied to complex surface models yielded the following accomplishments:

- CAD models from large data sets were created in a seamless integration of text file data and part models.
- Programmatic control of the variation of input parameters was implemented to successfully create feasible models within a pre-described design envelope.
- Complex free-form surface geometry with different parameterization schemes and multiple constraints were generated.
- Reusable surface models were created with high-fidelity, mesh-worthy geometry.
- Current practices were improved upon by overcoming obstacles present in current design processes.
- Two part files were controlled simultaneously to programmatically transfer data.
- Feature and constraint-based concepts were successfully implemented on a class of products that were previously unable to characterize in a class of reusable or parametric models.

It is strongly recommended that groups struggling with cycle time of complex models invest in high level design tools to automate processes and transform problematic characteristics into simple defining attributes. The application of programmatic parametric environments to complex surface models yields a higher return and in the end delivers, a faster cycle time. This research shows ground-breaking technology that exploits possibilities of CAD-based design, analysis and manufacturing to the fullest extent possible. This research shows how implementing an Automated Parameterization Tool in creating PRSM products can save months of time during succeeding product design iterations. By applying this methodology to different products within the PRSM class, huge potential is gained for great savings in design, analysis, development, process planning and production of complex products. This research study leads to a competitive edge for advancing technology and retaining market share of products.

6. REFERENCES

[1] Anderl, R.; Mendgen, R.: Parametric Design and its Impact on Solid Modeling Applications, Proceedings of the 3rd ACM Symposium on Solid Modeling and Applications, Salt Lake City, Utah, 1995, 1-12.

[2] Ardalan, S.: DrawCraft: A Spacecraft Design Tool for Integrated Concurrent Engineering, Aerospace Conference Proceedings, IEEE, 11, 2000, 501-510.

[3] Elliot, Jason H.: An automated approach to feature-based design for reusable parameter-rich surface models, Department of Mechanical Engineering, Brigham Young University, 2004.

[4] Haimes, R.; Crawford, C.: Unified Geometry Access for Analysis and Design, Proceedings, 12th International Meshing Roundtable, Sandia National Laboratories, 2003, 21-31.

[5] Hoffman, C. M.; Joan-Arinyo, R.: CAD and the Product Master Model, Computer-Aided Design, 30(11), 1998, 905-918.

[6] Hoffman, C. M.; Kim, K. J.: Towards Valid Parametric CAD Models, Computer-Aided Design, 33(1), 2001, 81-90.

[7] Kielb, R., E.: Mass Balancing of Hollow Fan Blades, Journal of Engineering for Gas Turbines and Power, 108(4), 1986, 577-582.

[8] Kong, L.; Fuh, J. Y. H.; Lee, K. S.; Liu, X. L.; Ling, L. S.; Zhang, Y. F.; Nee, A. Y. C.: A windows-native 3D plastic injection mold design system, Journal of Materials Processing Technology, 139(1), 2003, 81-89.

[9] Ramaswamy, R.; Ulrich, K.; Kishi, N.; Tomikashi, M.: Solving Parametric Design Problems Requiring Configuration Choices, ASME Journal of Mechanical Deign, 115, 1993, 20-28.

[10] Rohm III, T.; Tucker, S. S.; Jones, C. L.; Jensen, C. G.: Parametric Engineering Design Tools and Applications, ASME Design Automation Conference, Baltimore, Maryland, 2000.

[11] Rohm III, T.: Graphical Creation of CAD Parametric Application Programs, Department of Mechanical Engineering, Brigham Young University, 2001.

[12] Roller, D.: Foundation of Parametric Modeling, Parametric and Variational Design, B.G. Teubner Stuttgart, Germany, 1994.

[13] Shapiro, V.; Vossler, D. L.: What is a Parametric Family of Solids? Proceedings of the 3rd Symposium on Solid Modeling and Applications, Salt Lake City, Utah, 1995, 43-54.

[14] Tang, P. S.; Chang, K. H.: Integration of Topology and Shape Optimization for Design of Structural Components, Structural and Multidisciplinary Optimization, 22(1), 2001, 65-82.

[15] Xu, X.; Wang, Y. Y.: Multi-model technology and its application in the integration of CAD/CAM/CAE, Journal of Materials Processing Technology, 129(1) 2002, 563-567.