10th World Congress on Structural and Multidisciplinary Optimization
May 19 -24, 2013, Orlando, Florida, USA

Design Optimization of a High-Pressure Turbine Blade using Generalized Polynomial Chaos (gPC)

Girish Modgil[1], William A. Crossley[2], Dongbin Xiu[3]

[1] Rolls-Royce Corporation, Indianapolis, Indiana, USA, girish.modgil@rolls-royce.com
[2] Purdue University, West Lafayette, Indiana, USA, crossley@purdue.edu
[3] The University of Utah, Salt Lake City, Utah, USA, dongbin.xiu@utah.edu

1. Abstract
Gas turbine engines for aerospace applications have evolved dramatically over the last 50 years through the constant pursuit for better specific fuel consumption, higher thrust-to-weight ratio, lower noise and emissions all while maintaining reliability and affordability. This paper addresses one facet of this inter-disciplinary optimization problem – optimal design of a turbine blade. An existing Rolls-Royce High Work Single Stage (HWSS) turbine blisk provides a baseline to demonstrate the optimal aerodynamic design of a turbine blade. The optimization problem maximizes stage efficiency, for the high-pressure stage turbine, using turbine aerodynamic rules as constraints. The function evaluations for this optimization are surrogate models built from detailed 3D steady Computational Fluid Dynamics (CFD) analyses. To perform the optimization, this paper presents the generalized polynomial chaos (gPC) method – which is commonly associated with uncertainty quantification - as a viable option for sampling and constructing polynomial approximations. To reduce computational costs, the optimization uses response surfaces generated by fitting the results from the engineering toolset at a prescribed set of trial points. Instead of using traditional DoE techniques, like fractional factorial designs, the gPC method provides the trial points through sparse grid sampling. Also the gPC toolbox developed as part of this research effort facilitates construction of the response surfaces for the blade optimization via the stochastic collocation technique. The paper describes the design optimization concept, introduces basic gPC theory, provides detailed CFD results and concludes with an interpretation of the design optimization effort that results in a new aerodynamic shape for the turbine blade. The efforts are aimed at advancing the usability and impact of high-fidelity tools in the design process using automation and optimization with the focus of improving stage efficiency of the high work single stage turbine. To the best of the authors' knowledge, this paper is a first in applying gPC methods to generate surrogate models for design optimization in an industry level turbomachinery problem.

2. Keywords: Blisk, stochastic collocation, design optimization, sparse grid quadrature, High-fidelity CFD

3. Introduction
Gas turbine engines for commercial and military applications have evolved dramatically over the last 50 years through the constant pursuit for better specific fuel consumption (SFC), higher thrust-to-weight ratio, lower noise and emissions, to name a few - all while meeting requirements to maintain reliability and affordability. To meet these demands, rapid innovation has led to designs with higher stage loading, higher pressures, smaller axial gaps, lower tip clearances, higher temperature profiles, improved cooling, better nacelle designs, and newer lighter advanced capability materials. Design practices have matured significantly as well – 3D Finite Element Analysis (FEA) and Computational Fluid Dynamics (CFD), damper design, detailed material characterization programs, improved life assessment models are now standard for a typical blade design[1]. Reliable operation of a gas turbine engine depends on the structural integrity of its rotating parts. Design, analysis and testing are intensely rigorous phases of the product development cycle that help ensure that the final product on-wing meets the required specification by the manufacturer.
Gas turbine engines can be classified into four broad types: turbojet, turbofan, turboprop and turboshaft. Due to high fuel consumption and noise issues turbojets have been largely replaced by turbofans, which drive additional air outside the core of the engine through a bypass duct by the addition of a low-pressure (LP) turbine. An engine cut-out of a high bypass Rolls-Royce turbofan appears in Figure 1 (left). The different turbine stages are labeled; these remain consistent between all classes of gas turbine engines [2]. Turboshafts can power helicopters, ship propellers, generators in power stations, oil pipeline pumps, and natural gas compressors. An example of one such turboshaft engine is the Rolls-Royce Model 250, which is a 400 to 750 shaft horsepower (shp) turboshaft engine. Figure 1 (right) shows the engine cutaway for a Model 250-C20R turboshaft engine used widely in Bell Helicopter Textron 206B III and the Agusta A109C MAX. Overall, the design of the high-pressure (HP) turbine is one of the most challenging engineering tasks in the gas turbine engine. The design

is a compromise that has to satisfy a series of constraints, i.e., efficiency, SFC, cost, weight, etc., that in some cases compete against one another [3].

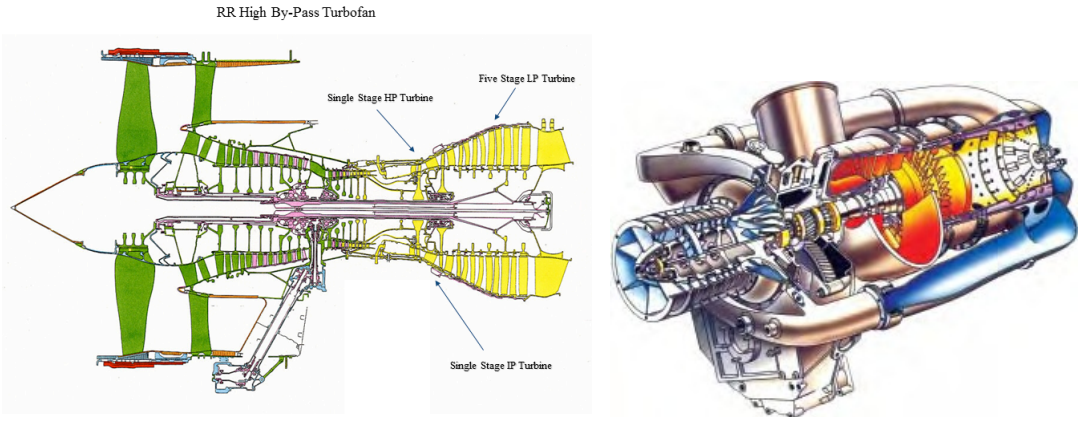

Figure 1: A High By-Pass RR Turbofan Engine [2] (left), Model 250-C20R turboshaft engine cutaway [3] (right)

4. Geometry Description and CFD Toolset

The general arrangement for the high-pressure (HP) turbine stage used for model development, here appears in Figure 2. This turboshaft engine, developed in the 1970s, was a prototype built as part of Rolls-Royce's (then Allison Engine Company) research and development efforts. In 1983, an instrumented engine core rig test (HP turbine and compressor stages) provided experimental data for comparison with the analytical predictions. Test results are discussed in detail in reference [4]. This turboshaft engine consists of a single HP turbine stage with a cooled vane. The vane uses outer end wall contouring to accelerate the flow after some initial turning to reduce the secondary flow losses. The cooled vane ejects the cooling flow into the annulus through the gill slot near the trailing edge of the vane. The design consisted of 20 upstream vanes (stator assembly) and 44 blades (rotor assembly) in the HP turbine stage. The overall turbine itself consists of a single-stage HP and a two-stage free Power Turbine (PT). The maximum engine operating speed is 51847 rpm (5429 rad/s) and the stage pressure ratio, defined as inlet total pressure divided by exit static pressure, at this operating condition is 3.52. A pressure ratio of 3.52 implies that the flow is fully choked at maximum operating condition.

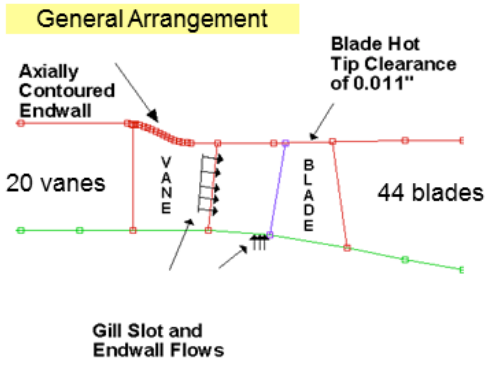

Figure 2: HP Turbine Flowpath Geometry Used for Model Development

Recent developments in CFD modeling and the availability of High Performance Computing (HPC) have made robust design studies for flow predictions possible. High-fidelity analysis was not the norm at the time of the engine development, so this baseline design provided an opportunity to investigate high-fidelity 3D CFD analysis along with state-of-the-art optimization techniques as a means to achieve better turbine stage efficiency.

For the flow analysis, two Rolls-Royce in-house codes facilitate the design optimization study. The first one is a 3D CFD meshing tool PADRAM, which stands for Parametric Design and Rapid Meshing system [5]. This meshing tool integrates geometry manipulation and parametric meshing, which avoids problems associated with import and export of geometry for meshing. For the purposes of design optimization, PADRAM runs in batch mode; the batch mode allows control of the various design parameters and mesh controls through simple ASCII input files. Additional files allow changes to blade design parameters in batch mode. The mesh style used in this work is of the H-O-H multi-grid type, which appears in Figure 3. PADRAM generates a structured O-grid

around the blade with four block structured H-grids for the remainder of the passage.

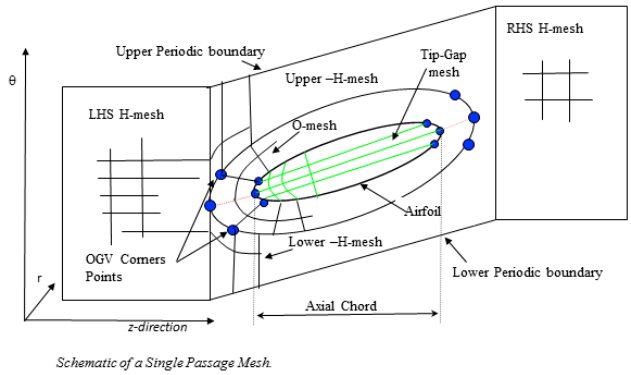

Figure 3: Schematic of a Single Passage Mesh [5]

PADRAM offers a number of ways to change the aerodynamic shape parametrically. This work uses the engineering blade parameters – stagger, lean, and sweep – as the design variables to maximize stage efficiency given a certain set of aerodynamic constraints (inlet capacity, reaction, exit swirl angle) all while paying due consideration to the mechanical limits (bearing loads etc.). Figure 4 (radial-axial view) shows how changes in these variables, affect the blade geometry. These three design variables are arguably the most conventional variables used to redesign turbine blades. In Figure 4, the original blade geometry appears in a 2D profile. For this research problem, the parameter stagger, measured in degrees, has the effect of rotating the 2D section at the mid-span of the baseline blade about the positive radial axis. A counter-clockwise rotation is considered to be positive stagger. Lean, also measured in degrees, is the tangential (or circumferential) movement of the 2D section. Sweep, measured in mm, is the forward or backward movement of the 2D section along the engine axis.

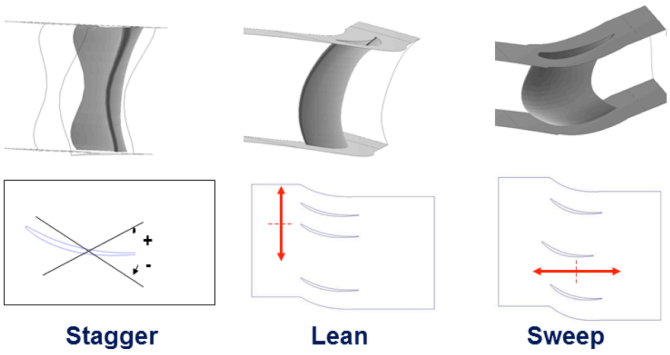

Figure 4: Engineering Blade Parameters used for Design Optimization

Using PADRAM in batch mode, ASCII files with different values of lean, stagger and sweep can be prescribed at multiple span locations. In this work, the three design variables (stagger, lean and sweep) describe the 2D blade section at the mid-span (50% radial height), and a fourth-order polynomial function describes the geometry between blade hub and tip; only the 2D section at mid-span is changed.

The CFD solver used for this work is a suite of tools of linear, non-linear and adjoint solvers built collaboratively by Rolls-Royce and its university partners and collectively known as Hydra [6]. The suite has a common core that provides a consistent framework for input, output, multi-grid acceleration, parallelization and visualization. The Hydra flow solver, based on work initially done by Moinier and Giles [7], is a general unstructured flow solver developed for turbo-machinery flow analysis and design. It is an explicit time-marching scheme that can use structured or unstructured meshes. The unstructured meshes may be composed of mixtures of tetrahedral, pyramidal, prismatic and/or hexahedral elements. These offer the geometric flexibility all while maintaining the advantages of structured meshes within boundary layers. The Hydra implementation is of the five stage Runge Kutta Monotone Upstream-centered Schemes for Conservation Laws (MUSCL) type of flux differencing algorithm. Hydra can model turbulence using one-equation Spalart-Allmaras or two-equation models such as k-ε

and k-ω[5]. The discrete equations are pre-conditioned using a block Jacobi pre-conditioner [8] and iterated toward the steady state using the five-stage Runge-Kutta scheme [9]. Various studies, have found that this process provide a speed up in convergence rate of up to three times compared to non-pre-conditioned schemes. An element collapsing multi-grid algorithm that automatically generates sequences of coarser meshes for the flow solver further accelerates convergence to steady state. This accelerated convergence typically reduces the processing time [9]. For the analyses needed in this work, developing the grid and obtaining the solution, required on the order of five to six hours, even with the accelerated convergence features in the Hydra solver and with PADRAM running in batch mode. An efficient surrogate model must be developed to reduce computational costs associated with design optimization; this is done through the gPC-based polynomial approximation. The sampling to build the gPC-based polynomial approximation invokes both PADRAM and Hydra multiple times for design space exploration and boundary condition manipulation.

5. Motivation

One of the main research goals for this study is to advance the usability and impact of high-fidelity tools in the design process using automation and optimization with a specific focus of improving stage efficiency of a high work single stage turbine. The following sections describe the process used for achieving this goal. Section 6 details generalized polynomial chaos with some basic theory for sparse grid sampling and stochastic collocation; gPC is used as a building block for DoE and response surface fitting. Section 7 details all the steps within the design optimization process. Section 8 provides a summary of all the relevant results from the design optimization. Section 9 provides the conclusions drawn from this effort.

6. Generalized Polynomial Chaos (gPC)

The generalized polynomial chaos (gPC) method is best known for providing rapid convergence with reasonable accuracy for predictions of probabilistic responses. Fundamental to gPC is the construction of an approximation to some form of measured response. The approach when applied to the problem for this paper uses the techniques associated with gPC to build surrogate models for use in design optimization. Ghanem [10, 11] first explored this through his work with spectral stochastic finite elements and named his approach Polynomial Chaos. The term chaos is arguably a misnomer, because gPC has nothing to do with chaos associated with non-linear dynamical systems but rather acknowledges the work of Wiener [12] who proposed a Homogeneous Chaos Expansion via extending Hilbert theory to infinite dimensional stochastic processes. This was done by employing Hermite polynomials in terms of Gaussian random variables to express stochastic processes with finite variance [13]. Also, Cameron and Martin [14] showed that any variable with finite variance can be represented exactly through Homogeneous Chaos expansions. For design involving complex physics, the governing system becomes one of stochastic partial differential equations (such as finite element analyses and computational fluid dynamics) where the unknowns are modeled as random variables and random fields [13]. A more accurate definition of Polynomial Chaos is as a Spectral Method. In Ghanem's case, a basis of multivariate Hermite polynomials were used to span the set of square integrable functions on the parameter space [15]. Xiu and Karniadakis [16] extended Ghanem's method to a generalized polynomial chaos (gPC) that used basis functions from the Askey family of orthogonal polynomials (Table 1). The underlying concept in gPC remains the same as proposed by Ghanem except that gPC extends the approach to a generic basis of orthogonal polynomials in addition to the Hermite polynomial basis.

Table 1: Correspondence between type of gPC and the underlying random variable

Type	Distribution of Z	gPC Basis Polynomials	Support
Continuous	Gaussian	Hermite	$(-\infty,+\infty)$
	Gamma	Laguerre	$[0,\infty)$
	Beta	Jacobi	$[a,b]$
	Uniform	Legendre	$[a,b]$
Discrete	Poisson	Charlier	$\{0,1,2...\}$
	Binomial	Krawtchouck	$\{0,1,...N\}$
	Negative Binomial	Meixner	$\{0,1,2...\}$
	Hypergeometric	Hahn	$\{0,1,...N\}$

A comprehensive theoretical treatment of spectral methods for stocahstic computations is provided in Xiu's book [15]. References [17, 18] provide a broad perspective on gPC's applicability to different classes of test problems. For problems whose response is smooth with respect to the uncertain parameters, gPC has a distinct advantage over traditional techniques such as Monte Carlo, possibly yielding exponentially fast convergence of the error as opposed to algebraically fast convergence of the error. The solution is a set of coefficients for a polynomial that approximates the response. Once a solution is obtained, statistical moments and sensitivities can be easily derived. The gPC technique can be implemented intrusively (embedded in a software) or non-intrusively as well. The latter, non-intrusive gPC implementation, uses existing software codes in a deterministic sense to calculate the coefficients of a pseudo-projection and thereby avoids the effort required for analysis code rework; this comes at the cost of conducting a few more calculations compared to the intrusive implementation.

A recent advancement in non-intrusive gPC was the introduction of spectral collocation methods as described by Xiu and Hesthaven [19]. Collocation methods are a popular choice for uncertainty quantification of the response of complex systems where well-established deterministic codes exist to predict the responses. Stochastic Collocation (SC) or Spectral Collocation is a term reserved for the type of collocation method that results in mean-square convergence to the true solution. The execution of SC methods are straightforward requiring only solutions of the corresponding deterministic problems at each interpolation point, allowing the software codes to be used as 'black-boxes', much like the Monte Carlo Sampling. Upon completing the simulations, post-processing is required to obtain the desired solution properties (statistical moments, response surfaces etc.) from the solution ensemble. This, in the classical sense, is a 'deterministic sampling technique' making the use of SC very attractive.

The core issue for efficient SC is the construction of the set of interpolation points, which is non-trivial in multidimensional random spaces. A commonly used approach for determining the sampling points is a tensor product construction of one-dimensional nodes. Because each node requires a full-scale underlying deterministic simulation, the tensor product approach is practical only for low numbers of variables (N<6). For higher-order random dimensions, sparse grids address the "curse of dimensionality" problem to a certain extent while maintaining many of the accuracy properties of the tensor grid. A sparse grid is a subset of the full tensor grid and can significantly reduce the number of nodes in higher dimensional problems while keeping high-order accuracy making it a viable choice in practical simulations. The following two sub-sections, briefly describe the two main ideas from gPC used for the design optimization – sparse grids and discrete projection. The gPC toolbox developed for the work in this paper implements the SC technique using sparse grid quadrature.

5.1. Sparse Grid Quadrature

Quadrature is a numerical scheme for estimating integrals. In its simplest form, it is in 1-D and estimated by a trapezoidal rule. To integrate in multiple-dimensions (known as cubature), one can use 1-D rules in each dimension. This number of points in the product rule is the product of the points in each 1-D grid. The most straightforward way of constructing high-order integration rules is to extend quadrature rules from the univariate case to the high-dimensional spaces via tensor construction. Figure 5 (left) below shows a 2D product rule with 289 points (17 points in each dimension). As the dimension increases beyond 2D, the solution becomes intractable because the number of points in the grid becomes too large. Sparse grids can address this problem by capturing only 'desirable' monomials; this is done by adding product grids that are dense in one direction but sparse in the other direction [20]. For a product grid in m-dimensions and degree of precision P, the cost is P^m. As the degree increases, many of the monomials can be dropped from the approximation because they contribute very little to the precision [20]. The requirement is only to capture monomials up to a degree P. So, a 2D sparse grid rule, needs P that meets the following:

$$2D: x^{P1}y^{P2} \quad P1+P2 \le P \tag{1}$$

In equation 1, x and y are the two independent variables in the approximation. With a sparse grid, the degree of precision P is greater than or equal to the sum of the degrees of each independent variable. Figure 5 contrasts the number of points required for an identical level of precision – polynomials up to degree 4 – for a two dimensional problem. Abscissae based on Clenshaw-Curtis quadrature show that a sparse grid using only 65 points can approximate a 289 point (17×17) tensor grid.

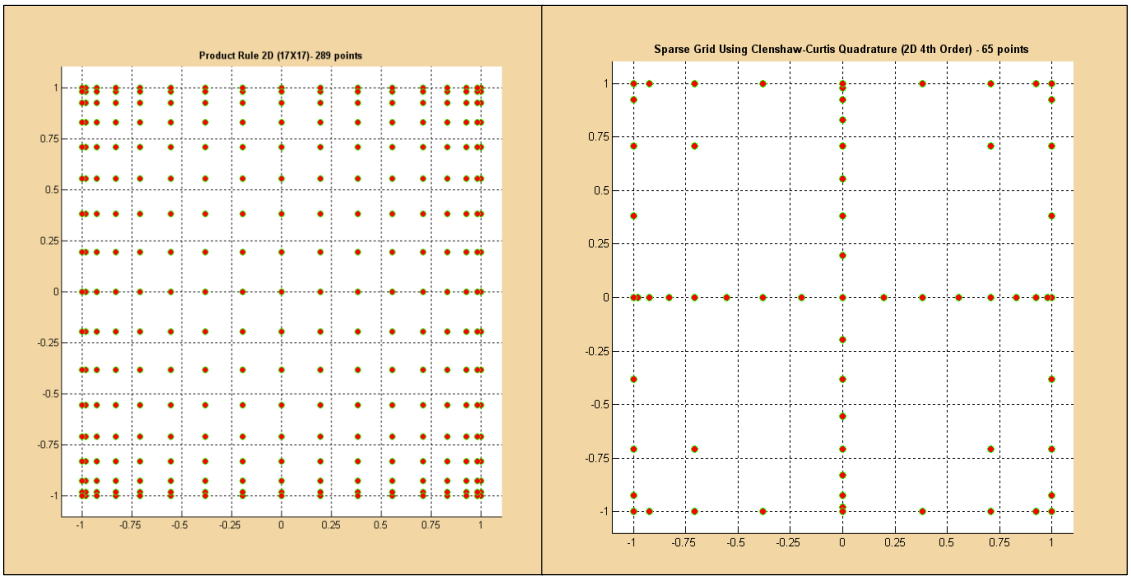

Figure 5: 2D Tensor Product Grid (left) and 2D Sparse Grid using Clenshaw-Curtis quadrature (right)

Sparse grids combine lower order product rules to achieve the same asymptotic accuracy as the product grid but with far fewer points [20, 21]. The important concept when using Smolyak's construction rule is that of '*nestedness*'. There are several choices of nested abscissae such as Clenshaw-Curtis, Gauss-Patterson, Gauss-Legendre, Gauss-Jacobi and Gauss-Laguerre to name a few [17]. Because of the Smolyak construction, these enjoy slower growth of the number of points in high dimensions (N>3) compared to other non-nested techniques. For the work presented here, sparse nested grids are created using Clenshaw-Curtis abscissae. The Clenshaw-Curtis nodes are extrema of Chebyshev polynomials defined in equation (2) with k providing an index for the point sets:

$$Z_i^{(j)} = -\cos\frac{\pi(j-1)}{m_i^k - 1}, \qquad j = 1, 2...., m_i^k \tag{2}$$

Reference [15] shows that interpolation through Clenshaw-Curtis sparse grid is exact if the function is in, $\mathbf{P}_k^d$, the polynomial space for the *d-dimension* problem. Additionally, Clenshaw-Curtis sparse grids approximate integrals in a bounded domain – a relevant feature in this optimization study where the design variables have bounds- and have become very popular due the slower growth of the number of points in higher dimensions.

5.2. Discrete Projection

Discrete projection is also known as the pseudospectral approach; this approach performs the stochastic collocation. It seeks to approximate the integral:

$$\int f(z)dF_z(z) \approx U^Q[f] \triangleq \sum_{j=1}^{Q} f\left(z^{(j)}\right)\alpha^{(j)}, \qquad Q \geq 1, d > 1, z \in R^d \tag{3}$$

The nodes and weights are $z^{(j)}$ and $\alpha^{(j)}$ respectively. The integration rule is convergent if it converges to the integral as $Q \to \infty$. Accuracy of the integration is measured by polynomial exactness. An integration rule of degree m implies that the approximation is exact for any integrand f that is a polynomial of degree up to m. For discrete gPC projection, the exact orthogonal gPC projection of $u(Z)$ is given by:

$$u_N(Z) = P_N u = \sum_{|k|=0}^{N} \hat{u}_k \phi_k(Z) \tag{4}$$

The expansion coefficients are obtained as follows:

$$\hat{u}_k = \frac{1}{\gamma_k} E[u(Z)\phi_k(Z)] = \frac{1}{\gamma_k}\int u(z)\phi_k(z)F_Z(z), \quad \forall |k| \leq N \tag{5}$$

The idea of discrete projection is to *approximate* the integrals in the expansion coefficients of equation (5) of the

continuous gPC projection by an integration rule. The *discrete* projection of the solution is:

$$w_N(Z) = \sum_{|k|=0}^{N} \hat{w}_k \phi_k(Z) \tag{6}$$

where the expansion coefficients are given by:

$$\hat{w}_k = \frac{1}{\gamma_k} U^Q\left[u(Z)\phi_k(Z)\right] = \frac{1}{\gamma_k}\sum_{j=1}^{Q} u\left(z^{(j)}\right)\cdot\phi_k\left(z^{(j)}\right)\cdot\alpha^{(j)} \tag{7}$$

The total error bound, for discrete projection, is the sum of the projection error (from using finite order polynomial), the aliasing error (difference between continuous and discrete projections) and the numerical error. The coefficients in equation (7) are substituted into equation (6), which is the polynomial approximation of the response. This approach provides the response surfaces for the objective function and constraints in the blade optimization problem. The statistical moments associated with these responses are not needed for this study, but they are easily obtained from post-processing equation (6). Reference [15] provides more details on deriving statistical moments, sensitivities and orthogonal polynomial theory.

7. Design Optimization

Optimization is the process of finding values of the independent variables of a function so that the function reaches an extreme value [22] while satisfying constraints. In formulating a design optimization problem, one has to find a set of design variables that minimize (or maximize) the objective function (turbine stage efficiency in this case) subject to different types of constraints (equality and inequality constraints). A typical numerical optimization statement appears in equation (8):

$$\text{Objective function: } \min f(x) \tag{8}$$

subject to the inequality constraints *(g)*, equality constraints *(h)* and side constraints:

$$\begin{aligned} g_j(x) &\le 0 \qquad j = 1, m \\ h_k(x) &= 0 \qquad k = 1, n \\ x_i^L \le x_i &\le x_i^U \quad i = 1, l \end{aligned} \tag{9}$$

where the design variables, x, are given by:

$$x = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_l \end{Bmatrix} \tag{10}$$

The update formula describes the iterative nature of optimization:

$$x^q = x^{q-1} + \alpha S^q, \quad q = 1, N \tag{11}$$

where x is the design variable array, S is the search direction vector, α is the scalar defining the distance to move along S and q is the iteration number. Depending on the type of optimization algorithm, the determination of the search direction may use just the function or the function and its gradient, or the function, its gradient and Hessian. A feasible design is one for which all constraints are satisfied.

For this problem, the design variables are the aforementioned blade lean, stagger and sweep, and the objective function is to maximize stage efficiency with a set of design constraints. To achieve this, the blade is defined parametrically in 3D from hub to tip either by using CAD geometry or by grid points in a text format at each span location defining the airfoil contour. The analysis toolset then reads the updated CAD or text file, meshes the geometry, submits the analysis and reports the results.

Starting from a baseline blade definition, changes to aerodynamic shape variables such as lean, stagger and sweep cause the bladed-disk system to go 'out-of-balance'. Blade balancing is routinely performed after the aerodynamic shape changes are incorporated. The blade balancing analysis involves minimizing the thrust bearing loads and shaft moments by optimally stacking the 2D airfoil sections. Blade balancing is time consuming and computationally intensive. Rather than develop an explicit constraint for blade balance, which would incur more cost and increase the complexity of the problem, the variable bounds (on lean, stagger and sweep) were used as an implicit way to address blade balance. Input and advice from experts at Rolls-Royce helped in setting these bounds. These design parameters were allowed to vary enough to enable a significant

improvement in stage efficiency. Table 2 below shows the minimum and maximum values assigned for lean, stagger and sweep design variables.

Table 2: Range of Values for Design Space Parameters

	Lean (deg)	Stagger (deg)	Sweep (mm)
Minimum	-2.0	-5.0	-3.0
Maximum	2.0	5.0	3.0

The objective function is stage efficiency; for the problem formulation shown below, $\hat{f}(x)$ represents the response surface approximation of stage efficiency. Inequality constraints, six in all because of the upper and lower bounds prescribed by design rules of thumb, supplied to the optimizer were as follows:

1. Inlet Capacity (m): does not change by more than 0.1%. $\hat{c}_1(x)$ represents the approximation of inlet capacity.
2. Rotor Reaction (n): does not change by more than 1.0%. $\hat{c}_2(x)$ represents the approximation of rotor reaction.
3. Exit Relative Swirl Angle (q): does not change by more than 0.5%. $\hat{c}_3(x)$ represents the approximation of exit swirl angle.

The optimization problem, for turbine stage efficiency, can be posed as follows:

$$\text{Objective function: } \min \hat{f}(x) \tag{12}$$

subject to the following six inequality constraints; the subscripts 1, 3, 5 are the lower limit constraints and subscripts 2, 4, 6 are the upper limit constraints for inlet capacity, rotor reaction and exit relative swirl angle respectively:

$$\begin{aligned}
\hat{g}_1(x) &= \left(\hat{c}_1(x) - m\right)/0.001 - 1 \le 0 \\
\hat{g}_2(x) &= 1 - \left(\hat{c}_1(x) - m\right)/0.001 \le 0 \\
\hat{g}_3(x) &= \left(\hat{c}_2(x) - n\right)/0.01 - 1 \le 0 \\
\hat{g}_4(x) &= 1 - \left(\hat{c}_2(x) - n\right)/0.01 \le 0 \\
\hat{g}_5(x) &= \left(\hat{c}_3(x) - q\right)/0.005 - 1 \le 0 \\
\hat{g}_6(x) &= 1 - \left(\hat{c}_3(x) - q\right)/0.005 \le 0
\end{aligned} \tag{13}$$

Matlab's 'fmincon' performs the optimization using surrogate models of the objective and constraint functions. The 'fmincon' function is an implementation of Sequential Quadratic Programming (SQP), a popular and efficient gradient based optimization routine. Figure 6 depicts the optimization process.

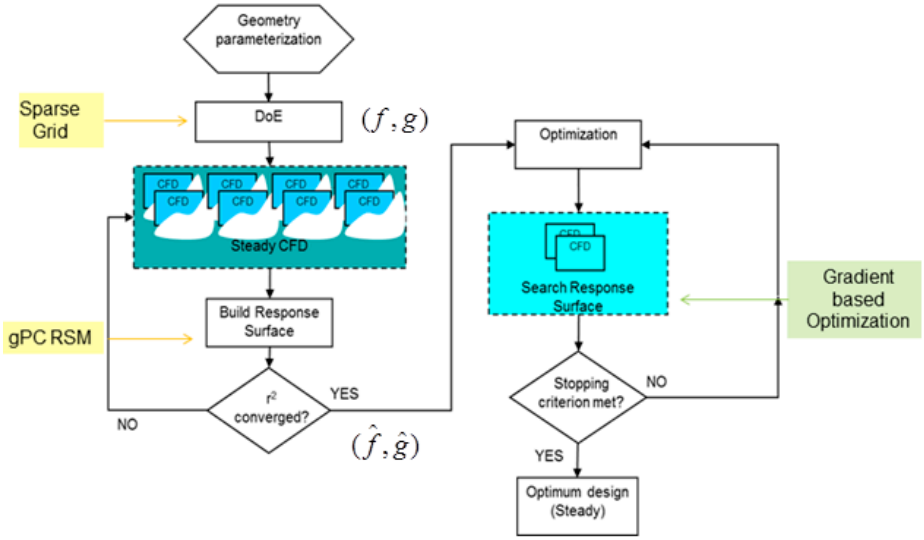

Figure 6: Process for Stage Efficiency Optimization

The steps shown in Figure 6 are discussed below:

1. Geometry parameterization sets up the folder structure for all files associated with the baseline geometry. The turbine stage has two rows – stator and rotor. Input files are needed to set up the CFD mesh and run the steady flow analysis. In the optimization, only the aerodynamic shape of the rotor (blade) changes. The stator (vane) shape is unchanged. For this optimization study, the analysis had to be performed many times over so a Python script automates the setting up of input files, folders and results folders for each different aero shape combination considered in the design study
2. For the Design of Experiments (DoE), the generalized polynomial chaos (gPC) toolbox generates a 3D, level-4 sparse grid with 177 points based on the Clenshaw-Curtis abscissae. These points (or nodes) define the combination of blade design parameters lean, stagger and sweep (from Table 2) for each steady CFD analysis.
3. Hydra calculations to steady-state used a mixing plane boundary condition at the stator-rotor interface for each CFD model representation of the blade geometry. Design system throughflow solutions of the turbine provided the inlet and exit boundary conditions. These included specification of the radial profiles of total pressure, total temperature, and flow angles at the inlet. For the exit boundary conditions, the static pressure was specified at the inner end wall (or hub), and the exit static pressure profile used the radial equilibrium equation. The analysis used the maximum power operating condition (51847 rpm). Each CFD analysis writes out a file with pertinent stage information such as stage efficiency, inlet capacity, rotor reaction, specific work and mass flow errors on an iteration by iteration basis.
4. The gPC toolkit creates a response surface to approximate stage efficiency and similar approximations for the three constraint functions of inlet capacity, rotor reaction and exit relative whirl angle. For each desired approximation, post-processing the steady CFD results via equation (7) provides coefficients for assembling polynomials. Assessment of the response surface fit quality for stage efficiency used two approaches; discussion of this appears in section 8.
5. Using the response surfaces built in the preceding step, a numerical optimization determines the design variable values associated with the local feasible maximum for stage efficiency.
6. The final step is to run the CFD flow code at the optimal design from the approximate problem to get a fully converged steady solution with the high-fidelity analysis. Results from the flow calculation are then compared against the response surface prediction for the optimal stage efficiency point as a verification step.

8. Results

As mentioned above, the mesh created for the 3D stator and rotor passages is a hybrid H-mesh and O-mesh of the multiblock structured type consisting of an O-mesh around the blade with H-blocks covering the remaining part of the meshed volume. Three different CFD models were developed to assess the sensitivity of the design features on the predicted blade stresses and the robustness of the modeling process. All the results discussed in this study involve use of the final meshed model, which is referred to as the *baseline geometry* from hereon. The

baseline geometry used for this study included tip clearance, end wall contouring and cooling flows in the CFD model. A blade tip clearance of 0.011 inches was modeled for the turbine blade. A typical mesh distribution used in CFD model development, showing the tip clearance in green, appears in Figure 7 (left). Figure 7 (right) shows a three-dimensional schematic of the turbine stage. The contours in Figure 7 (right) on the vane and the blade hub are the air density to indicate the cooling flow locations (cooler air, higher density). The location of the gill slot on the pressure surface of the vane and the hub end wall cooling injection for the blade row are apparent in this figure. The model accounts for the injection of cooling flow using a surface boundary condition specifying mass flow and total temperature.

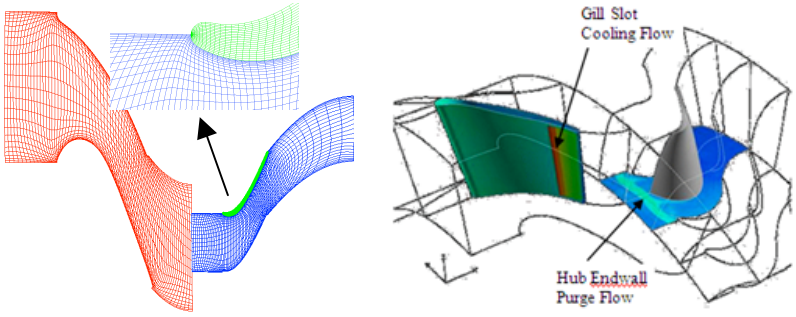

Figure 7: Typical CFD Grid showing Tip Clearance Mesh (left), 3D CFD Model with Cooling Flows (right)

Mesh for the baseline aero shape and all remaining aero shapes investigated for aerodynamic optimization were generated using the same basic mesh template file changing only the parametric design space values for lean, stagger and sweep at the mid-span location. For the single passage vane and rotor mesh, the multi-block structured mesh was around 1.53 million cells with approximately 726,000 cells on the rotor and approximately 804,000 cells on the vane. This would be considered *adequate* mesh from a CFD viewpoint for this industry level problem. A finer mesh can be implemented, but the mesh shown in Figure 8 was used for the design space exploration study for computational efficiency. Figure 8 shows a very good refinement O-mesh near the LE, TE and all around the pressure and suction surfaces of the blade. Shearing effects of the grid are apparent at the L.E., but these effects on the predicted stage efficiency, inlet capacity, stage reaction and relative exit swirl angle are minimal in the current context. For the current work, it is sufficient that mesh topology has good orthogonality on the flow surface where it is critical to the solver.

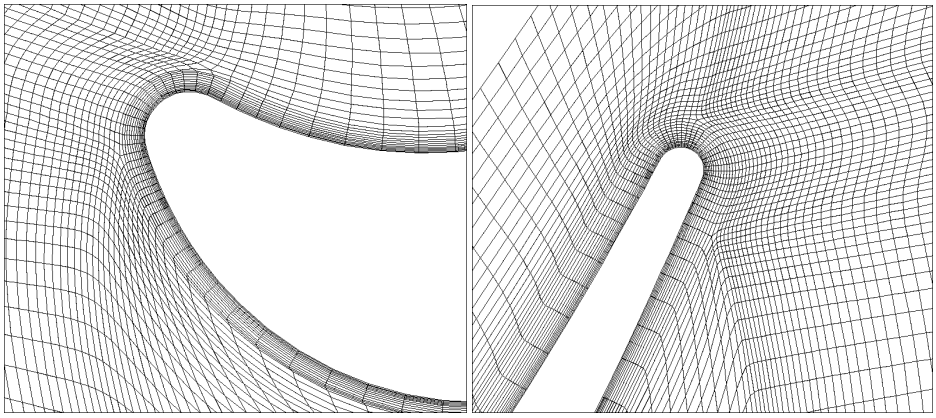

Figure 8: Rotor CFD Mesh for Baseline Aero: Leading Edge (Left), Trailing Edge (Right)

Knowing that the response surface will have to be tested for goodness of fit, a level-4 sparse grid was used. This would create a 177 point DoE; Figure 9 graphically shows the combinations of the three design variables for which the steady CFD analysis will predict the objective and constraint functions. The approach in the gPC toolbox then constructs a polynomial approximation to the objective and constraint functions using these 177 points. Running 177 DoE points for a three variable problem may seem excessive, but this cost is justified by the requirement to assess response surface fit quality. Also, the risk to an engineering project of running a very sparse DoE matrix is that, if more simulations are required, the entire DoE will have to be resubmitted to get

accurate weights, nodal locations and response surface fit. A traditional inline gradient-based search using the CFD tools directly could also have been used for the optimization search. A quick estimate yields five CFD solutions per iteration using the traditional inline gradient-based method wherein the numerical gradients require three evaluations, the function itself requires one evaluation and the line search requires one evaluation. A budget of 177 runs would allow for about 35 iterations. However, the gPC toolbox for this study can be directly used for further research involving uncertainty quantification analyses. Using the same toolset for optimization and uncertainty quantification offers a significant advantage from a software development point of view. Moreover, references [15, 16 and 19] suggest that gPC is capable of producing a very high quality response surface fit over the entire design space.

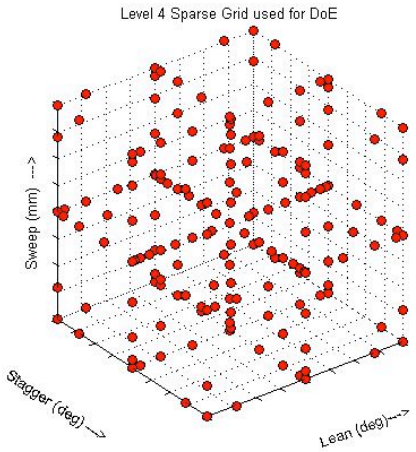

Figure 9: Level-4 Sparse Grid used to set up DoE for Constructing Surrogates with gPC toolbox

Using the prescribed nodes (design points) from the sparse grid, steady CFD analyzed blade shapes at 177 DoE points. At each DoE point, a time-averaged steady-state solution was obtained using a steady mixing-plane boundary condition between the airfoil rows. Steady flow calculations for this single stage analysis take roughly five hours to converge on a Linux cluster using a single node per mesh block. Each steady CFD analysis is post-processed to check for solution convergence and extract relevant stage information such as efficiency, inlet capacity, specific work and reaction. The intent is to be sure that the CFD results are of the quality expected in turbomachinery blade design. Figure 10 (left) shows the CFD solver convergence history for total pressure (Pa) at the vane inlet, blade inlet and blade exit and Figure 10 (right) shows the stage efficiency convergence monitor for the baseline aerodynamic shape. These indicate that the solver has obtained appropriate results.

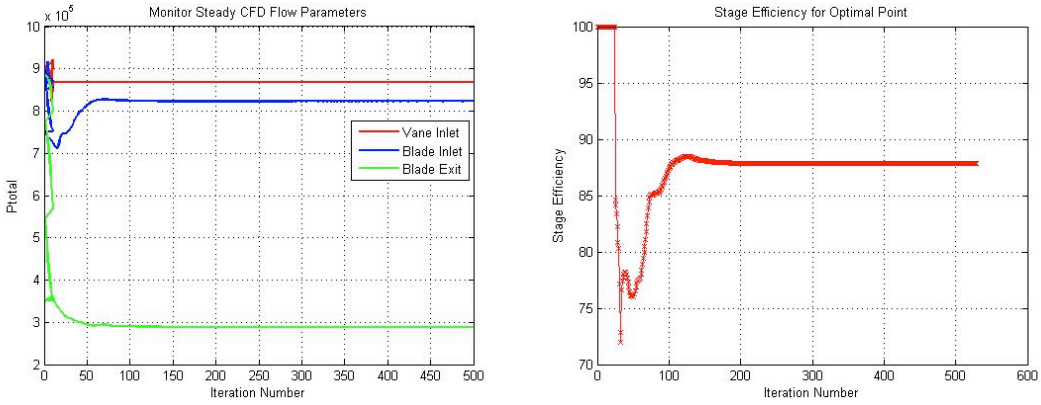

Figure 10: CFD Solver Convergence Monitor: Left - Total Pressure (Pa), Right – Stage Efficiency (%)

Table 3 shows how the quality of the CFD based predictions for the baseline blade geometry compares with the design intent of the blade; empirical data from similar styled blade designs and through-flow calculations (1D and 2D) provide the design intent for the blade . Predicted values of mass flow and stage efficiency agree quite

well with the design intent. The slight disparity in the values is understandable; the difference is caused by the under-prediction of the stage losses in the 3D CFD model along with analysis errors (numerical, discretization etc.).

Table3: Baseline Geometry Design Intent vs. Predicted Values

	Prediction	Design Intent	% Difference
Stage Efficiency (%)	87.8	86.5	1.3
Mass Flow (kg/s)	2.6135	2.5569	2.2

Figure 11 shows the relative Mach number contour for this stage calculation for the baseline geometry. In a qualitative sense, these results are representative of the solutions from all the CFD runs for the designs in the entire DoE set. The wake and shock structures appear fully developed.

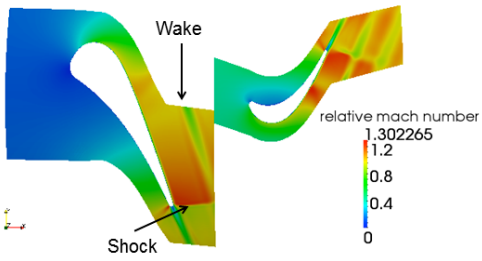

Figure 11: Relative Mach Number Contour

Mach numbers are greater than 1.0 near both the vane and the blade row exits. The shock formation is evident at the maximum engine speed operating condition due to the high pressure ratio across the stage.

Prior to running optimization to find the maximum stage efficiency, the quality of the response surface fit was evaluated using the coefficient of determination (R^2). In general for preliminary design studies in turbomachinery applications, R^2 values in the range of 0.80-1.0 are considered acceptable. This study used two approaches to assess quality of the response surfaces.

Because the level-3 sparse grid is a subset of the level-4 sparse grid, one way to provide an assessment of the response surfaces is to use only the 69 design points associated with the level-3 sparse grid and construct a polynomial approximation using only these points. The CFD predicted stage efficiency from the remaining 108 design points from the original level-4 sparse grid are then compared with the stage efficiency prediction using the polynomial approximation from the level-3 sparse grid design points. This process is commonly referred to as "leave *n* out cross validation" and allows checking results without additional simulations. Typically, increasing the number of points used to construct an approximation improves the quality of the approximation. Therefore, if the R^2 measure in the "leave *n* out" comparison is acceptable for the 69 point approximation, then the 177 point approximation is also acceptable. Figure 12 shows the results of performing this type of RSM assessment; in a perfect prediction, all the points would lie on a diagonal line. The R^2 value describing how well the 69-point, level-3 sparse grid predicts the remaining 108 CFD solutions is 0.82. Considering that more than 60% of the points were excluded from the response surface, this is a very good fit.

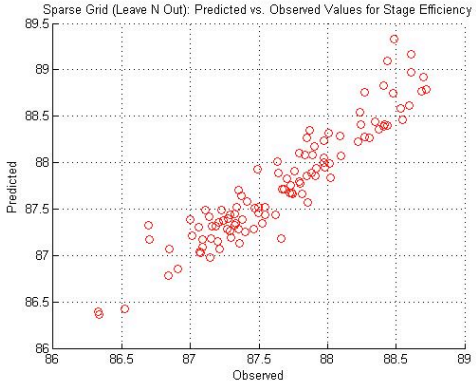

Figure 12: R^2 estimate: Leave "*n*" Out Approach

The second, more common, approach to check response surface fit is to run additional points. Because the R^2 estimate from the previous step is acceptable, the optimization found a feasible design that maximizes stage efficiency using the 177 point response surfaces. When computational resources became available, using the design from the surrogate-based optimization as a "new baseline", twenty additional blade designs near the optimal design were analyzed with the CFD solver, and these CFD-based results were compared against the response surface obtained from the level-4 sparse grid. The minimum and maximum values for lean, stagger and sweep were 50% and 150% of the optimal point. Using this method, the R^2 value, measuring how well the response surface describes these additional twenty CFD results near the optimal solution is 0.9991. This further enforces the idea that the response surface is acceptable for this optimization, and the fit is very good near the optimal design obtained from the surrogate-based optimization. Figure 13 below shows the result of the comparison of the 20 additional CFD calculations with the response surface prediction.

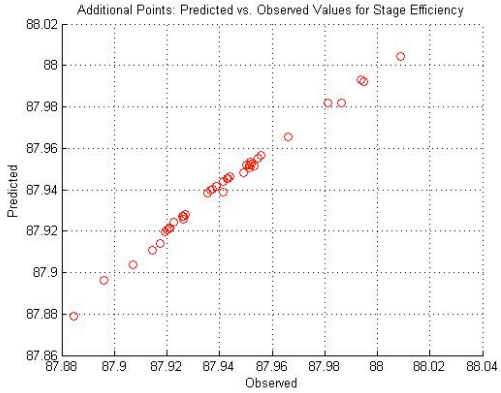

Figure 13: R^2 estimate:Twenty additional points

Matlab's optimization toolbox was used for gradient-based optimization. Gradients and Hessians were provided to the 'fmincon' function. Figure 14 shows the optimization results from Matlab. In Figure 14, design variable values appear in the top left, objective function values appear in the top right, optimization step size appears in the bottom left, and the first-order optimality metric appears in the bottom right.

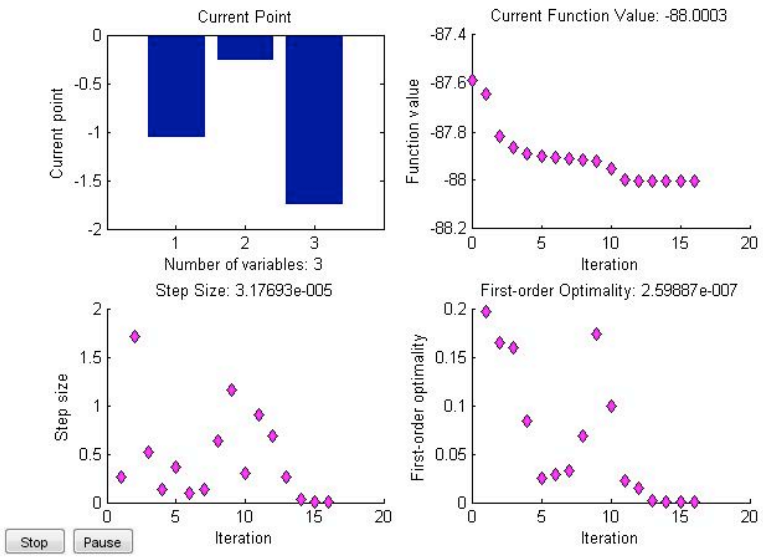

Figure 14: Optimization Results and Iteration History

With the inequality constraints included in this problem, the optimized design had a stage efficiency improvement of 0.17%. For a single stage turbine analysis, an efficiency gain of 0.17% is a very good improvement, but not surprising given the current state of the toolset compared to when the engine was designed and built. Since the optimization uses response surfaces, it is computationally inexpensive. Gradient based optimizers can be sensitive to the initial design point used for the first guess. To ensure that the true local optimum was obtained, the initial guess was changed multiple times. Another, more rigorous, check for optimum is that the Hessian matrix be positive definite. Both checks confirmed that the optimal stage efficiency, based on the response surface approximations, is obtained for the following design parameters:

i. Lean: -1.063 degrees
ii. Stagger: -0.261 degrees
iii. Sweep: -1.752 mm

Figure 15 shows the difference between the baseline aerodynamic (blue) and the optimized aerodynamic (red) shapes. Since the optimized value of stage efficiency was obtained from the response surface, the value was checked by running a steady flow calculation using the optimized aerodynamic shape and at four other design points in the vicinity of the optimized aerodynamic shape. The maximum difference of 0.05% between the RSM predicted values and the CFD-based predictions of the stage efficiency provides further confidence in the quality of the RSM fit.

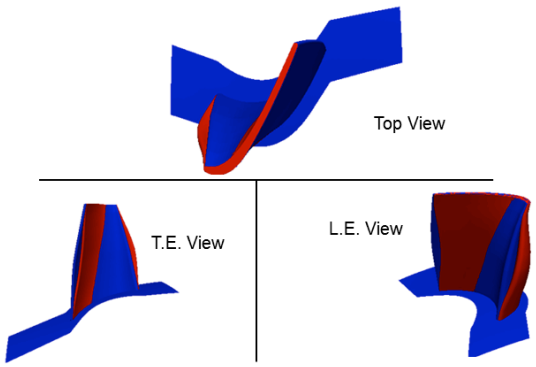

Figure 15: Baseline Aero (blue) and the Optimized Aero (Red) Shapes

The last step in the process is to make sense of the optimal blade design by using basic turbine design rules (1D and 2D). The vane and rotor are fully choked at maximum operating speed. For the optimal result point, there is an increase in stage efficiency, decrease in specific work and increase in rotor reaction all while maintaining the inlet capacity.

1. Increased rotor reaction implies a decreased expansion over the vane and higher rotor blade Mach number. Figure 16 shows the relative Mach number contour for the optimized aero shape (left) alongside the same contour for the baseline aero shape (right) confirming this rule.
2. Decrease in specific work generally agrees well with the observed increase in stage efficiency.
3. Changing the stagger in the prescribed manner has the effect of closing the throat area, which can increase the turning in the blade and increase stage loading. Increased stage loading does not necessarily lead to lower stage efficiency, because offsetting factors such as reduced shock losses across the vane may result in the observed behavior of increased stage efficiency.
4. Sweeping the blade forward along with the combined effect of stagger and lean will cause lower static pressure drop across the vane leading to an increased rotor reaction.

Figure 16: Relative Mach Number Contour: Optimized (left); Baseline (right)

The traditional way to conduct the design study would have been to fix the expansion ratio so that the amount of specific work done across the stage is the same. The current set up is slightly more flexible. For the sake of an optimization study, it is considered acceptable to allow some amount of variance in the inlet capacity (and specific work). The produced optimal point is valid from the standpoint of stress, turbine aerodynamics, optimization and practicality for manufacture which is based on the fact that the constraints are satisfied and the variable bounds enforced to avoid rebalancing.

9. Conclusions

This paper showed how generalized polynomial chaos (gPC) can be used for design optimization for an industrial level problem; in this case it was for a High Work Single Stage (HWSS) turbine blade in a Rolls-Royce turboshaft engine. The study provides a process to advance the usability and impact of high-fidelity tools in the design process using automation and optimization with the focus of improving stage efficiency of a high work single stage turbine. The research explored gPC techniques for sampling and polynomial approximation as an alternative to traditional DoE techniques and response surface fits. In particular, sparse grid quadrature provided the sampling points and stochastic collocation provided the polynomial approximations of the response (objective function and constraints). Based upon the tests of the quality fit, the gPC method produced a very high quality response surface enabling a computationally inexpensive optimization search. As an alternative to gPC, the 177 CFD runs may have allowed for 35 iterations using a traditional inline gradient-based optimization search. This may have yielded an optimal solution similar to the one obtained via gPC. However, the gPC approach, normally associated with uncertainty quantification, provides an effective highly accurate response surface for optimization that may have been difficult to obtain using traditional approaches. The design optimization led to a new aerodynamic shape for the turbine blade using blade lean, stagger and skew as design variables. The process culminated with a turbine blade design that produces a stage efficiency gain of 0.17% with the inequality constraints defined on rotor reaction, inlet capacity and exit relative swirl angle. The techniques used in this study show promise for future industrial turbomachinery related applications including the ability to conduct uncertainty analysis using the same gPC toolbox developed for this optimization study.

10. Acknowledgements

The authors would like to thank Rolls-Royce Corporation for funding this work as part of the Rolls-Royce Ph.D. Fellowship Program. The authors would also like to acknowledge Drs. Daniel Hoyniak and Kurt Weber (both from Rolls-Royce) and Dr. Nicole Key (from Purdue University) for their suggestions, and technical guidance.

11. References

[1] A.V. Srinivasan, Flutter and Resonant Vibration Characteristics of Engine Blades, *ASME Journal of Engineering for Gas Turbines and Power*, 119, 742-775, 1997.

[2] M. Taylor, Axial Flow Turbine Aerodynamic Design, *VKI Lecture Series*, 2008.

[3] Rolls-Royce, *The Jet Engine Book*, Rolls-Royce Technical Publications, 2005.(ISBN: 0902121235)
[4] K. Weber, G. Modgil, S. Neerarambam and S. Gegg, Unsteady CFD Analaysis and Stress Prediction Comparisons for a High Work Single Stage Turbine Blade, *Proceedings of ASME Turbo Expo*, GT2012 -69085, 2012.
[5] S.Shahpar, SOPHY: An Integrated CFD based automatic design optimization system, *AIAA*, ISABE-2005-1086, 2005.
[6] B.L.Lapworth, HYDRA-CFD: A Framework for Collaborative CFD Development, *International Conference on Scientific and Engineering Computation*, IC-SEC Singapore, 2004.
[7] P. Moinier and M. B. Giles, Preconditioned Euler and Navier-Stokes Calculations on Unstructured Grids, *6th ICFD Conference on Numerical Methods for Fluid Dynamics*, Oxford, UK, 1998.
[8] P. Moinier, J-D. Müller, and M.B. Giles, Edge-Based Multigrid and Preconditioning for Hybrid Grids, *AIAA Journal*, 40 (10), 2001.
[9] L. Martinelli, Calculations of Viscous Flows with a Multigrid Method, *Ph.D. Thesis, Department of Mechanical. And Aerospace Engineering, Princeton University, USA*, 1987.
[10] R. Ghanem and P. Spanos, *Stochastic Finite Elements: A spectral approach*, Springer-Verlag, New York, 1991.
[11] P.D. Spanos and R. Ghanem, Stochastic Finite Element Expansion for Random Media, *Journal of Engineering Mechanics*, 115, 1035-1053, 1989.
[12] N. Wiener, The Homogeneous Chaos, *American Journal of Mathematics*, 60, 897-936, 1938.
[13] P. Constantine, Spectral Methods for Parameterized Matrix Equations, *Ph.D. Thesis, Stanford University*, 2009.
[14] R.H. Cameron and W.T. Martin, The orthogonal development of non-linear functionals in series of Fourier-Hermite polynomials, *Annals of Mathematics*, 48, 385-392, 1947.
[15] D. Xiu, *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press, 2010.
[16] D. Xiu and G. Karniadakis., The Wiener-Askey Polynomial Chaos for Stochastic Differential Equations, *SIAM Journal for Scientific Computing*, 24, 619-644, 2003.
[17] M.S. Eldred, C.G. Webster and P. Constantine, Evaluation of Non-Intrusive Approaches for Wiener-Askey Generalized Polynomial Chaos, *Proceedings of the 10th AIAA Nondeterministic Approaches Conference*, AIAA Paper 2008-1872, 2008.
[18] M.S. Eldred and J. Burkardt, 2009, Comparison of Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Quantification, *Proceedings of the 47th AISS Aerospace Sciences Meeting and Exhibition*, AIAA Paper 2009-0976, 1-20, 2009.
[19] D. Xiu and J. Hesthaven, High-order collocation methods for differential equations with random inputs, *SIAM Journal of Scientific Computation*, 27(3), 1118-1139, 2005.
[20] J. Burkardt and C. Webster, A Low Level Introduction to High Dimensional Sparse Grids, *Lecture to Computer Science Research Institute at Sandia National Laboratory*, 2007.
[21] M. S. Eldred, Nonintrusive Method for Uncertainty Quantification, *Presentation at SIAM Conference on Computational Science and Engineering*, Miami, Florida, 2-6 March 2009
[22] A. Keane and P. Nair, *Computational Approaches for Aerospace Design*, Wiley, 2005.