Computational Fluid Dynamics JOURNAL vol.17 no.4:32 2009 (pp.256–266)
Special Issue dedicated to Professor Helmut Sobieczky on the occasion of his retirement and for acknowledging his original contributions to CFD

THREE-DIMENSIONAL PARAMETRIC SHAPE OPTIMIZATION USING PARALLEL COMPUTERS

Brian H. DENNIS [1] George S. DULIKRAVICH [2] Igor N. EGOROV [3]
Shinobu YOSHIMURA [4] Djordje HERCEG [5]

1 Introduction

The design software system described here is based on a combination of several software components. These components include optimization, physics-based analysis such as computational fluid dynamics (CFD) and finite element analysis (FEA) and supporting tools, and shape design parameterization. The optimization and CFD/FEA related codes, such as mesh generation and partitioning, can be considered that can be applied directly with no modifications to any passage design situation. In the current system the shape design parameterization on the other hand is considered problem-specific and different codes are needed for different design problems. In the near future it will be routine to employ a standard computer aided drafting (CAD) package as a **black box** parameterization code. In that case the user would simply need to write a driving script for different design situations.

A key issue in the use of optimization with detailed 3-D analysis is that global multiobjective optimization methods will typically require hundreds or thousands of design. The design software system described here is based on a combination of several software components. These components include optimization, physics-based analysis such as computational fluid dynamics (CFD) and finite element analysis (FEA) and supporting tools, and shape design parameterization. The optimization and CFD/FEA related codes, such as mesh generation and partitioning, can be considered **black boxes** that can be applied directly with no modifications to any passage design situation. In the current system the shape design parameterization on the other hand is considered problem-specific and different codes are needed for different design problems. In the near future it will be routine to employ a standard computer aided drafting (CAD) package as a **black box** parameterization code. In that case the user would simply need to write a driving script for different design situations. A key issue in the use of optimization with detailed 3-D analysis is that global multiobjective optimization methods will typically require hundreds or thousands of design analyses in attempt to find the global solution. In order to complete the design process in a reasonable amount of time, a parallel computer should be employed. When the 3-D models become large, it is necessary to divide a single analysis among several processors when the core memory on a single processor is not large enough. In this case the software system should be capable of managing a collection of simultaneous parallel finite element analyses. Both the CFD/FEA analysis codes and the optimization codes used in this work were written to deal with this issue and hence make full use of parallel computing resources. The global optimization methods used here are naturally parallel algorithms and can make full use of a large number of processors.

Here, we apply global parallel optimization to two specific shape optimization applications, the shaping of turbine blade coolant passages and the shaping of a transonic transport wing.

2 Optimization Methods

The core of the design system is the optimization code. The optimizer directs the design process by generating new designs based on the performance of previously generated designs, in an iterative manner. In

Received on October 12, 2008.
1 Mechanical and Aerospace Engineering, University of Texas at Arlington, Arlington, TX 76019, U.S.A.
2 Mechanical and Materials Engineering, Florida International University, 10555 W. Flagler St., Miami FL 33174, U.S.A.
3 IOSO Technology Center Milashenkova 10-201 Moscow 127322, RUSSIA
4 Quantum Engineering and Systems Science. School of Engineering, University of Tokyo 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, JAPAN
5 Faculty of Sciences, University of Novi Sad, Trg Dositeija Obradovvica 3, 21000 Novi Sad, SERBIA

general, we wish to use optimization methods that are robust and computationally efficient. For optimization on a parallel computer, the optimizer should find a good design in the fewest possible number of iterations. Such algorithms should also be capable of making full use of large-scale parallel computers.

We also desire a robust optimization algorithm. The optimizer should not terminate in a local minimum due to a noise generated by the analysis code. It should also not terminate if the analysis cannot be completed due to, for example, failure to generate a proper grid for a candidate design.

We have found parallel genetic algorithm (PGA) based methods [1] and response surface methods based on Indirect Optimization based on Self Organization (IOSO) [2] to have the properties discussed above and work well for 3-D shape design.

2.1 IOSO Method

The Indirect Optimization based on Self Organization (IOSO) method is a constrained optimization algorithm based on response surface methods and evolutionary simulation principles [2]. Each iteration of IOSO consists of two steps. The first step is creation of an approximation of the objective function(s). Each iteration in this step represents a decomposition of an initial approximation function into a set of simple approximation functions. The final response function is a multilevel graph. The second step is the optimization of this approximation function. This approach allows for self-corrections of the morphology and the parameters of the response surface approximation. The distinctive feature of this approach is an extremely low number of trial points to initialize the algorithm (30-50 points for the optimization problems with nearly 100 design variables). During each iteration of the IOSO, the optimization of the response function is carried out within the current search area. This step is followed by the direct call to the mathematical analysis model for the obtained point. In the current research, the mathematical model is a finite element model of a candidate design. The information concerning the behavior of the objective function in the neighborhood of the extremum is stored, and the response function is made more accurate just for this subdomain of the search area. For a basic parallel IOSO algorithm, the following steps are carried out:

1. Generate a group of designs based on a design of experiments (DOE) method;
2. Evaluate the designs in parallel with the analysis code;
3. Build initial approximation based on the group of evaluated designs;
4. Use stochastic optimization method to .nd the minimum of the approximation;
5. Do adaptive selection of current extremum search area;
6. Generate a new set of designs in current extremum search area using DOE;
7. Evaluate the new set of designs in parallel with the analysis code;
8. Update the approximation with newly obtained result;
9. Goto 4. unless termination criteria is met.

The procedure is illustrated below in Figure 1.1.

Thus, during each iteration, a series of approximation functions is built for a particular optimization criterion. These functions differ from each other according to both structure and definition range. The subsequent optimization of the given approximation functions allows us to determine a set of vectors of optimized variables, which are used to develop further optimization criteria on a parallel computer.

3 Design of Turbine Blade Cooling Passages

With a perpetual goal of increasing thermodynamic efficiency of turbines, various blade-cooling schemes have been used. However, with the extremely high temperatures of the combustion gases it became apparent that film cooling causes increased production of NOx. As a remedy, a high-pressure closed-circuit internal cooling concept [3] became attractive again decades after its inception. However, the problem that has not been answered yet is where precisely to locate coolant passages and what should be the size of each individual passage [4].

There is a strong interaction among a number of engineering disciplines when studying the internally cooled gas turbine blades [5]. We will consider the temperature and the associating stresses within the blade material in detail. However, the effects of the hot gas flow and coolant flow will be treated in a very approximate way. In the design process explained in this paper, these individual disciplines will not be solved simultaneously in detail for 3-D designs, because this approach would take an unacceptably long time, even on a cluster of workstations running in parallel. For these pragmatic reasons we opted for a more approximate yet computationally affordable design approach.

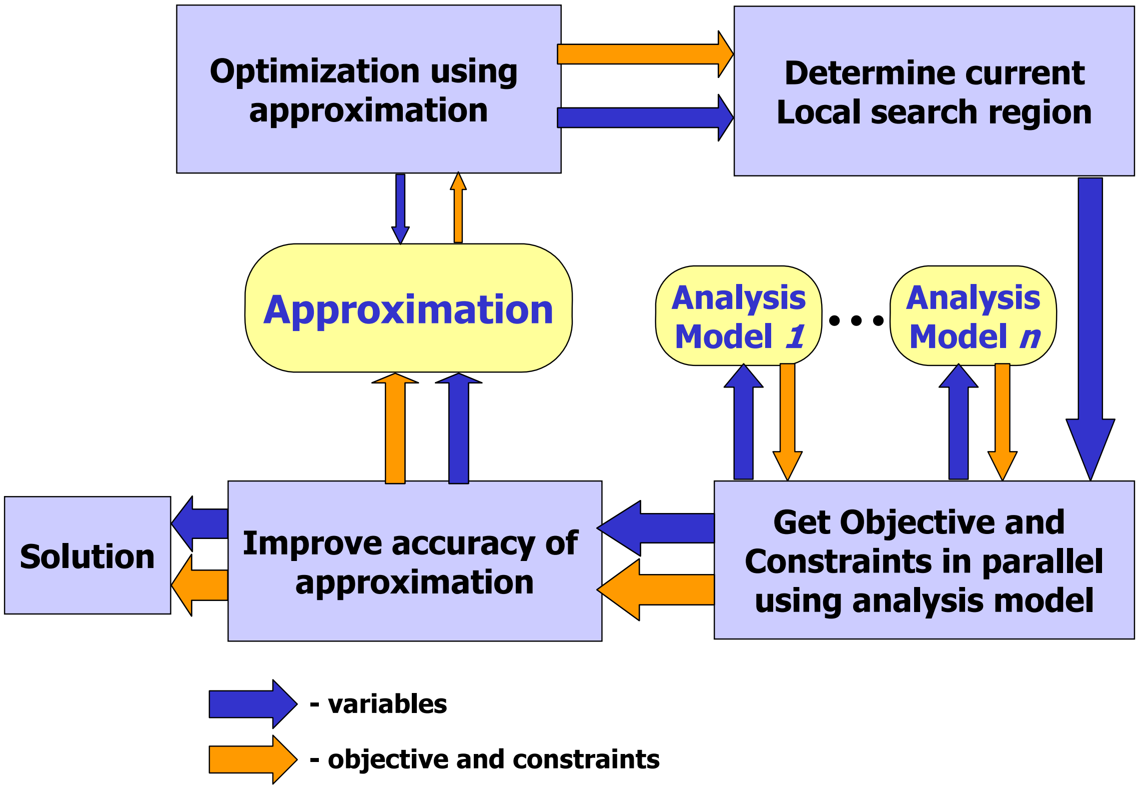

Fig.1: Parallel optimization based on approximation functions

3.1 Objective and Constraints

In this section the design objective and constraint functions are detailed. The objective of the design optimization is to minimize the variation in stress distribution within the blade material. This improves the durability of the turbine blade. The objective function is computed using the maximum principal stress at each node within the blade. Mathematically, the normalized objective function is expressed as

$$F = \sum_{i=1}^{n} \frac{\sigma_i^2}{n\sigma_{yield}} \tag{1}$$

where σ_i is the maximum principal stress at node i, n is the number of nodes within the blade, and σ_{yield} is the yield stress of the blade material. Only nodes within the blade itself are considered for the objective and constraint functions.

By minimizing this objective function, a smoothing effect on the principal stress field is achieved. In addition, this objective also drives the stresses to lower values, which is also desirable for the durability of the blade.

In addition to minimizing the objective function, the optimizer must find a design that simultaneously satisfies the design constraints. For the design of a turbine rotor blade, the maximum temperature should be less than an allowable temperature, T_{allow}. Similarly, the maximum principal stress should be less than the yield stress, σ_{yield}. These two inequality constraints are expressed mathematically as

$$G_1 = \sum_{i=1}^{n} \frac{1}{n} \left[100.0 \frac{T_i^- - T_{allow}}{T_{allow}} \right]^2 \tag{2}$$

$$G_2 = \sum_{i=1}^{n} \frac{1}{n} \left[100.0 \frac{T_i^- - T_{allow}}{T_{allow}} \right]^2 \tag{3}$$

where the constraints are satisfied if $G_1 \leq 0.0$ and $G_2 \leq 0.0$, while

$$\overline{T_1} = \begin{cases} T_i & \text{if} \quad T_i > T_{allow} \\ T_{allow} & \text{if} \quad T_i \leq T_{allow} \end{cases} \tag{4}$$

and

$$\overline{T_i} = \begin{cases} \sigma_i & \text{if} \quad \sigma_i > \sigma_{yield} \\ \sigma_{yield} & \text{if} \quad \sigma_i \leq \sigma_{yield} \end{cases} \tag{5}$$

The above constraints on maximum temperature and maximum stress could have been written more simply as

$$G_1 = T_{max} - T_{allow} \tag{6}$$

$$G_2 = \sigma_{max} - \sigma_{yield} \quad (7)$$

where T_{max} and σ_{max} are the maximum nodal temperature and principal stress, respectively. However, constraints (2)-(3) have the effect of penalizing designs with many nodes with infeasible temperature or stress, where as constraints (6)-(7) only consider the worst values at a single node. In our experience we found that the constraints (6)-(7) worked well only when an initial feasible design was given at the start of the optimization. In cases where no initial feasible design was known, the constraints (2)-(3) produced superior results in fewer iterations for both GA and IOSO algorithms.

3.2 Design Parameterization

The outer blade shape was considered to be fixed and to be provided by the user at the beginning of the design optimization. The shapes of the 3-D internal serpentine coolant passages were parameterized using analytical surface patches algorithm developed by Helmut Sobieczky [6, 7] that maintains C2 smoothness of the surface. The turbine blades considered in this research had a total of four straight passages connected by U-turn passages. The result is a single serpentine passage with a single inlet and outlet. The spanwise cross-sectional shape of each straight passage is described by four parameters as shown in Figure 2. These parameters include the degree of filleting in the passage, r, the minimum blade wall thickness, d, and the passage chordwise starting and finishing point, x_1 and x_2 respectively. The passage cross-section shapes are determined at the root and the tip by user provided parameter values. The parameters for the middle sections are found by linear interpolation along the blade span.

Three U-turn shapes are used to connect the ends of the coolant passages. The wall shape of the U-turn passage is determined by using analytic functions. For wall n, the half shape can be found by using the following equations

$$x_n = (x_{max_n} - x_c)|\cos(\theta)^{R_{fn}}| + x_c \quad (8)$$

$$z_n = Z_n|\sin(\theta)^{R_{fn}}| + z_c \quad (9)$$

where x_{max} is the x position of the end of the straight passage wall and x_c, z_c are the x and z coordinates of the strut center. Four parameters are needed to define each U-turn shape in the $x - z$ plane as shown in Figure 3. The parameters Z_1 and Z_2 control the position of the passage walls in the z-direction. The parameters R_{f1} and R_{f2} control the roundedness of the u-turn shape. U-turn shape change with the variation of the control parameters is shown in Figure 4. More details on construction of the turbine blade passages and outer shape are discussed in the references [6, 7].

The straight passage parameterization is somewhat limited, because it cannot create designs with angled struts in the $x-y$ plane. Also, in the current approach, the number of straight passages cannot be changed easily and is fixed at four. These limitations should be addressed to further increase the usefulness of this approach for creating passage shapes.

The following additional design parameters were also used: the coolant passage bulk temperature, T_c, and blade angle with the disk, θ_b. All together a total of 42 continuous design variables were used to uniquely describe a design.

Sobieczkyfs shape parameterization code generates a block-structured grid that describes the shape of the blade. Example blade and passage geometries are shown in Figures 2-6. An inner shroud and blade root geometry are generated separately and added to the base of the blade section. The block-structured grids for blade, shroud, and root are then used as the base geometry for generating a triangular surface mesh. The triangular surface mesh is then used as input to a tetrahedral mesh generation program [8].

3.3 Design Analysis

The analysis process may need to be performed thousands of times for a single optimization run so it is critical that each module be automatic, robust, and computationally efficient. The thermal and thermoelastic analysis is performed by parallel finite element analysis. The finite element analysis codes and tools for mesh generation, mesh partitioning, and others are freely available as a part of the ADVENTURE project [9] lead by the University of Tokyo. The software modules are geared towards large-scale parallel analysis and are well suited to the efficient analysis of complicated geometries.

The ADVENTURE system employs a module-based architecture. Each module is an independent application program that can be operated individually or together with other modules via standard binary file interface. Example modules include triangular surface patch generation, tetrahedral volume mesh generator, boundary condition attachment, parallel mesh partitioning, and parallel thermal/elastic finite element analysis (FEA). The parallel FEA algorithms are based on various iterative solvers including the Hierarchical Domain Decomposition Method (HDDM), which is able to achieve a high parallel efficiency. These modules have been successfully used

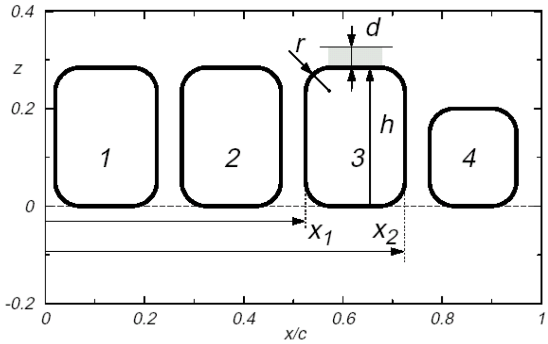

Fig.2: Parameters for passage cross-section shape in $x-y$ plane

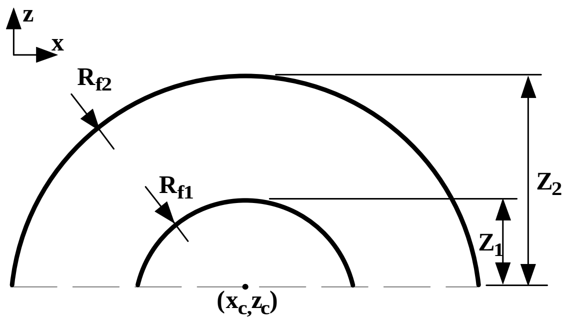

Fig.3: Parameters for U-turn shape in the $x-z$plane

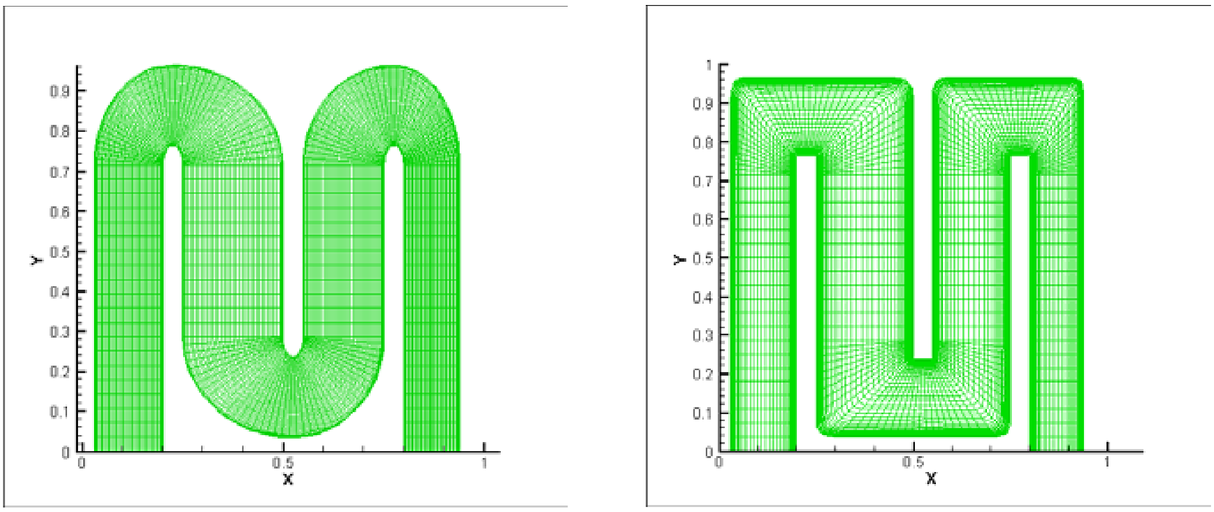

Fig.4: Example passage shape for variation parameters R_{f1} and R_{f2}

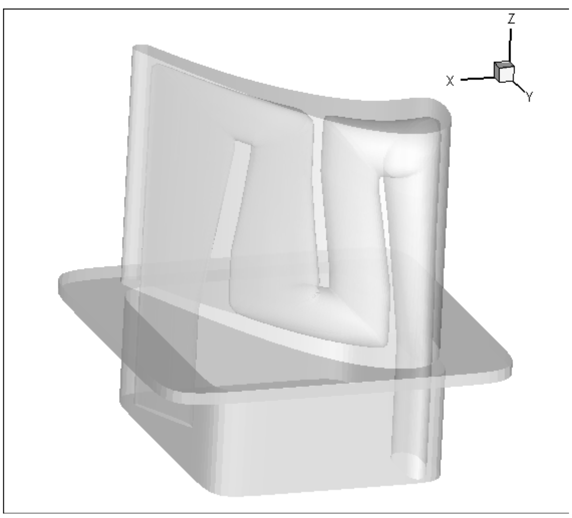

Fig.5: Internally cooled blade example

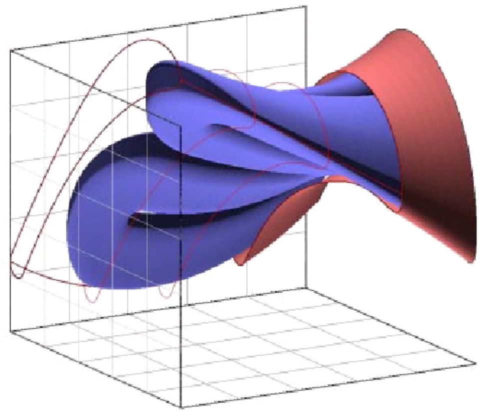

Fig.6: View of 3-D serpentine coolant passage surfaces

on analysis models ranging from one million degrees of freedom (DOF) to hundreds of millions of DOF.

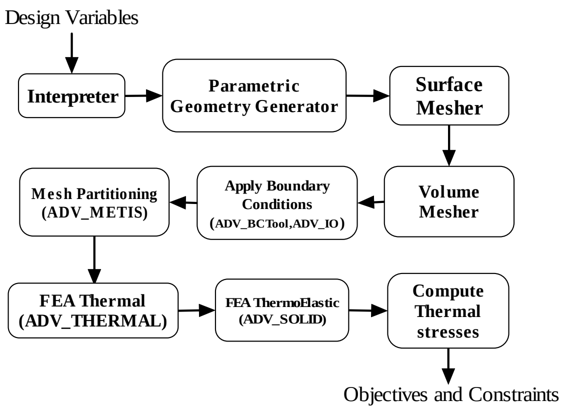

Fig.7: Modules used for automatic parallel thermoelastic FEA

For each design, a series of modules is used to automatically turn a given set of design variables into objective and constraint function values. In the current system, each module represents an individual software executable. The data is passed from module to module by file. Within some modules, such as the FEA solver and the mesh partitioner, parallel processing is used via domain decomposition algorithms. In those modules internal data is passed by MPI. Upon termination, the final results are written to file and passed to the next module. The flow of data between these modules is depicted graphically in Figure 7.

3.4 Design Optimization Example

The design system described here was used to perform an example of 3-D shape design optimization of an internally cooled turbine blade. We created the outer blade geometry by generating a series of 2-D turbine airfoils and stacking the sections along the z-axis.

In this example, the blade material was assumed to be a titanium-aluminum alloy. For each design mesh, the boundary conditions were applied automatically. The root section of the geometry was set to zero displacement while the blade and inner shroud were left free to deform. As for thermal boundary conditions, the outer surface of the blade and top surface of inner shroud were set to convection boundary conditions which require the specification of the convection coefficient, h_B, and the hot gas bulk temperature, T_B. Convection boundary conditions were also applied to the coolant passage surface inside the blade using h_C and T_C. All other surfaces were assumed thermally insulated. Both centrifugal and thermal body forces were applied automatically to each design mesh. Actual values used for this design example are shown in Table 1.

The optimization run was performed on a commodity component based PC cluster with 54 Pentium II 400 MHz processors. Both PGA and IOSO optimization methods were tested with this problem. A total of 12 analyses were performed concurrently for IOSO method while for each PGA, 36 designs were evaluated at each iteration. For both cases, each parallel thermoelastic FEM analysis used 4 processors. A typical analysis mesh contained over 150,000 degrees of freedom and required 4 minutes to complete a full thermoelastic analysis.

A converged result was found by the IOSO optimizer in 70 iterations after consuming approximately 12 hours of total computer time. For PGA, the total computer time was more than 30 hours. The PGA run was terminated before a converged result was found. The convergence history for the objective function for both PGA and IOSO is shown in Figure 8.

For all designs, the stress constraint was satisfied. However, the initial design violated the temperature constraint so the optimizer had to first determine a feasible design. The convergence history for the temperature constraint function is shown in Figure 9. This figure shows that feasible region was found at iteration 12 for IOSO and iteration 62 for PGA. These results clearly demonstrate the computational efficiency of the IOSO approach over the PGA method for this design problem.

The passage shape and resulting outer surface temperature distribution for the initial design are shown in Figures 10-11. The temperature patterns on the surface of the blade follow the shape of the cooling passage inside the blade. This shows that the passage shape will have a strong impact on the temperature distribution and hence the thermally induced stresses. Stress in the root of the blade is high due to the centrifugal loading and temperature gradients. The IOSO optimized cooling passage shape is shown in Figure 12.

The wall near the tip corners has become much thinner obviously in an effort to keep the temperature in those regions below the maximum allowable value. The temperature distribution on the outer surface of the optimized blade is shown in Figure 13. It is considerably smoother compared with that of the initial design. In addition, the principal stress distribution for the optimized design is smoother than in the initial design, as shown in Figures 14-15.

4 Aerodynamic Design of a Transonic Wing

The competitiveness of the aircraft industry along with the increasing price of fuel has forced design-

Table 1: Parameters for rotor design problem

Coolant convection coefficient, hC	$500.0 W/m^2$ °C
Coolant bulk temperature, T_C	150.0-600.0 °C
Hot gas convection coefficient, h_B	$150.0 W/m^2$ °C
Hot gas bulk temperature, T_B	1500.0°C
Maximum allowable temperature, T_{allow}	900.0 °C
Angular velocity about x-axis	5000 RPM
Inner shroud distance from x-axis	0.25 m
Blade span	0.10 m
Blade chord	0.10 m

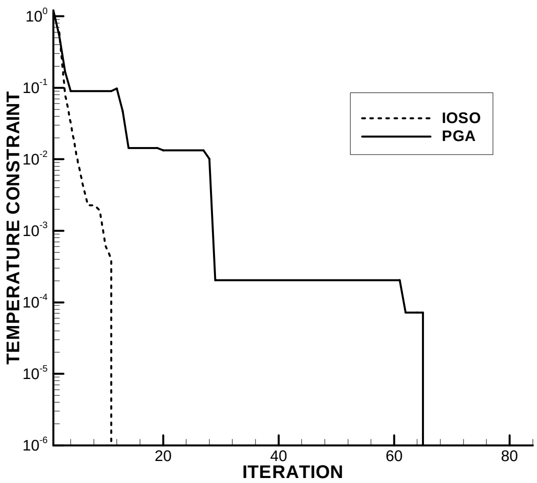

Fig.8: Objective function convergence history

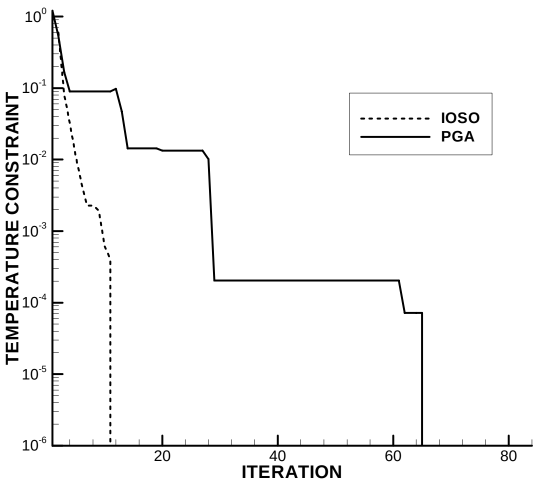

Fig.9: Temperature constraint function convergence history

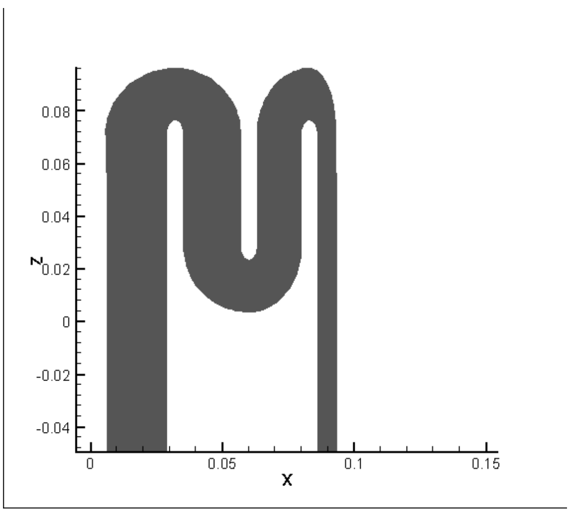

Fig.10: Cooling passage shape in $x-z$plane for initial design

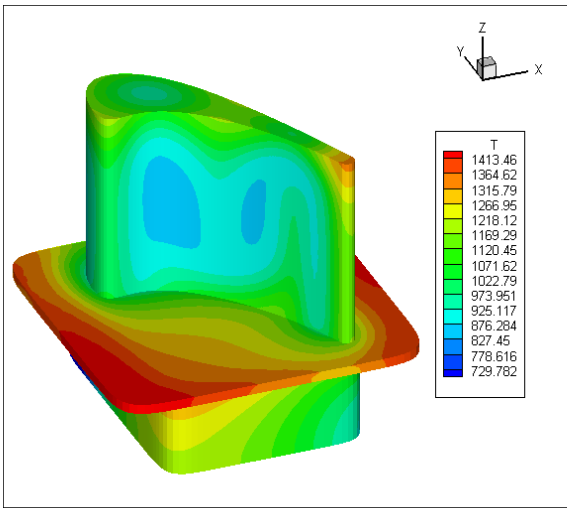

Fig.11: Blade outer surface temperature map for initial design

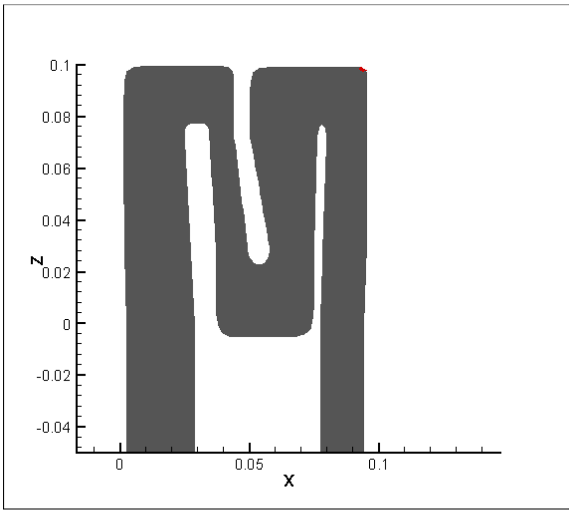

Fig.12: Cooling passage shape in $x-z$ plane for IOSO optimized design

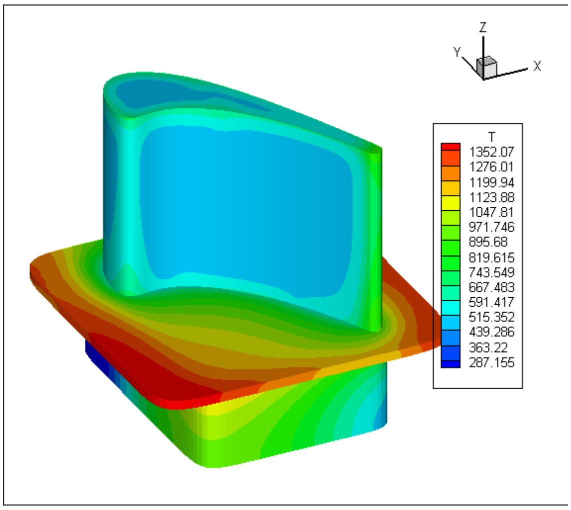

Fig.13: Blade outer surface temperature map for IOSO optimized design

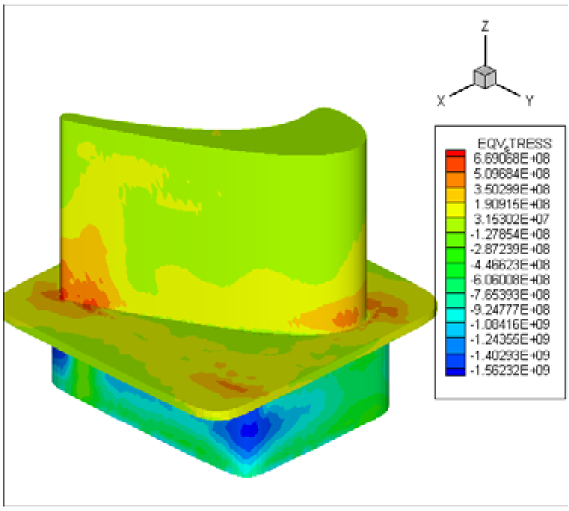

Fig.14: Principal stress map for initial design

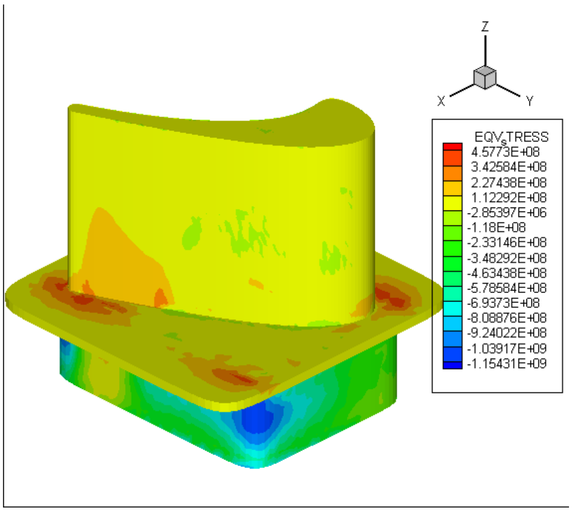

Fig.15: Principal stress map for IOSO optimized design

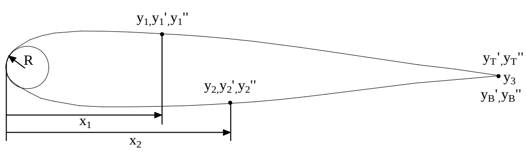

Fig.16: Airfoil action parameters

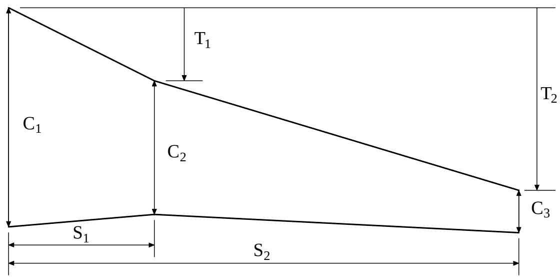

Fig.17: Wing planform parameters

ers to consider new approaches to improving aerodynamic efficiency. Simulation-based design optimization using computational fluid dynamics analysis is one approach that has the potential to improve product performance while simultaneously reducing design time. Researchers have explored a variety of different optimization approaches with varying degrees of success [10]-[12]. In this section, we demonstrate the application of IOSO parallel optimization methodology to the aerodynamic design of an isolated commercial transport transonic wing. In this design problem, the objective is to minimize the drag of a wing under cruise flight conditions, while satisfying two nonlinear constraints that are applied to make the result more practical. First, the lift of the optimized wing must be equal to a user specified value. If we consider the aerodynamic redesign of an existing wing, then the optimized wing should maintain the same lift as the initial design. We also use a constraint that forces the optimized wing to maintain the same volume as the initial wing. This is needed to insure sufficient volume for internal structure, undercarriage, and fuel tanks.

4.1 Design Parameterization

The wing model is constructed by stacking airfoil sections along the span. Each airfoil section is described entirely by analytic functions in an attempt to create a flexible description with a minimum number of parameters. The nose of the airfoil is represented with a circular arc of radius R. The upper and lower surfaces are defined by four quintic polynomials. The upper and lower surface each have a free control point. The position, slope, and curvature at each control point are required as input to the geometry routine. A third control point defines the trailing edge. The slope and curvature for the upper and lower surfaces are also specified at the trailing edge. The upper and lower surfaces join the nose circle such that the continuity of airfoil curvature is guaranteed. A graphical depiction of the parameterization is shown in Figure 16. In total, 13 parameters are required to completely define an airfoil shape with this approach. The wing planform is defined with three airfoil sections, one at the root, one at the midspan, and one at the tip. With this approach, taper and sweep can easily be introduced through proper choice of the design parameters. The 7 parameters needed to define the planform are shown in Figure 17. In total, 47 design variables are required to completely describe the wing shape. This includes the planform shape, the three airfoil sections, and the wing angle of attack, α.

4.2 Design Analysis

A 3-D CFD code for structured grids is used to obtain the flow field around the wing configuration. The CFD code solves the time-dependent conservation law form of the Reynolds-averaged Navier-Stokes equations with a two-equation turbulence model. The spatial discretization involves a semi-discrete finite-volume approach with upwind-biasing used for the convective and pressure terms [13]. Time advancement is implicit using an alternating-direction-implicit (ADI) method with the ability to solve steady or unsteady flows. Multigrid and mesh sequencing are used for convergence acceleration.

An algebraic grid generation method is used to generate a C-O type mesh around the wing configuration. The mesh is clustered towards the wing surface to improve the accuracy of the viscous drag calculation.

4.3 Design Optimization Example

In this section we present an example of transonic wing optimization using the parallel IOSO approach. The approximate planform shape and the airfoil sections of a Boeing 737-100 was chosen as the initial wing design to be optimized. The airfoil sections were obtained from the Airfoil Coordinates Database at University of Illinois at Urbana-Champaign. The wing planform was taken from 737 diagrams available on Boeing's website. The cruise Mach number was given as 0.73 at an altitude of 35,000 ft. The weight was specified as 81,900 lbs. The initial design had a non-dimensional volume of 5.87x10.3,a CL of 0.1275, and a CD of 0.01466.

Each design was analyzed with the compressible turbulent Navier-Stokes solver described in section 4.2. A C-O type grid with the size of 201x49x49 was used to analyze each design. Each analysis was run for 1000 time steps beginning from a restart file obtained from 4000 iterations on the initial wing design.

The parallel IOSO optimizer was used on 26 processors. The objective was to minimize CD while maintaining the initial wing CL and volume. The optimization convergence history is shown in Figure 18. A converged result was obtained by iteration 20. A total of 442 analyses were performed over approximately 2.5 days, including the DOE effort required to build the IOSO approximation function.

The planform shape was changed noticeably from the initial design, as shown in Figure 19. There is a slight decrease in span with an evident increase in sweep. Both the CL and volume remained unchanged, while the CD was reduced by 47.5%. This dramatic drop in aerodynamic drag can be explained by observ-

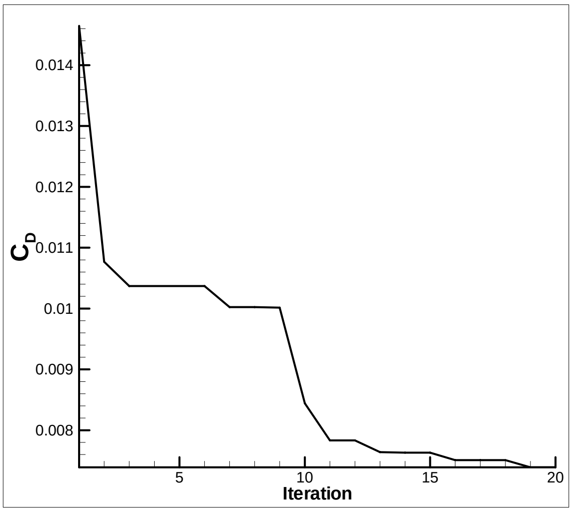

Fig.18: Transonic wing optimization convergence history

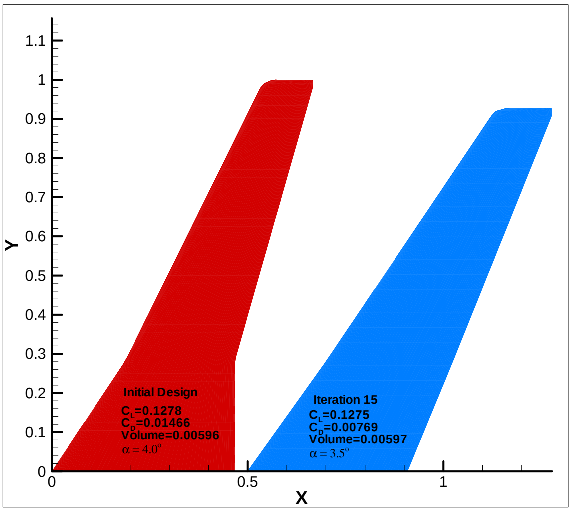

Fig.19: Planform shape for initial and optimized transonic wings

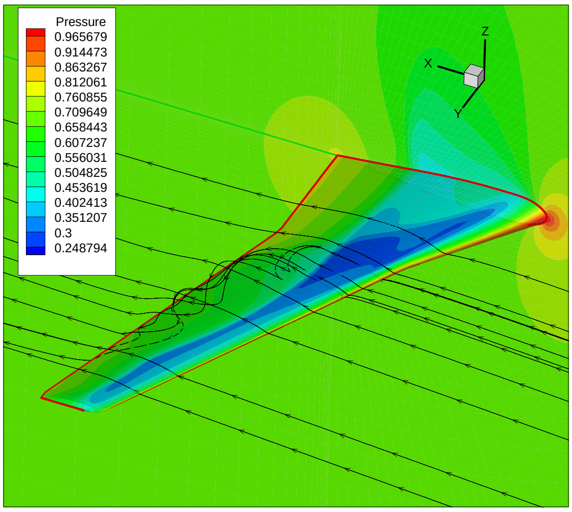

Fig.20: Streamlines and pressure contours for initial transonic wing

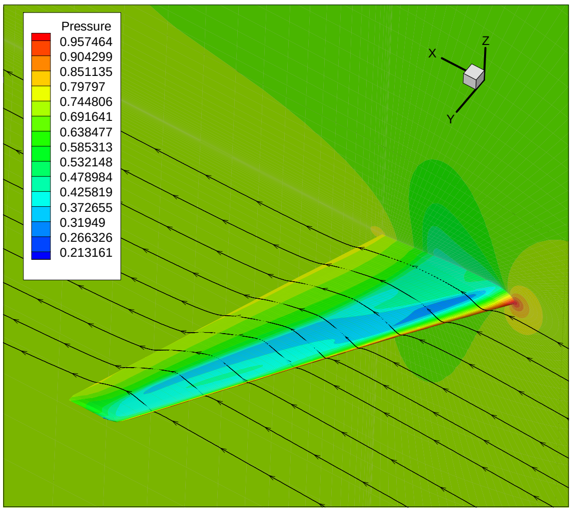

Fig.21: Streamlines and pressure contours for optimized transonic wing

ing the flow patterns on the surface of the optimized wing. In Figure 20, the strong shock-boundary layer interaction on the initial wing has resulted in a large flow separation region on the upper surface. The optimized wing in Figure 21 shows a very smooth flow pattern free of the large separation present on the initial design. Removal of this shock-induced separation has clearly allowed the optimized wing to produce the required lift at a much lower drag.

5 Concluding Remarks

A robust software system for the shape design of 3-D components has been developed using powerful optimization algorithms and efficient parallel analysis codes. The automatic parametric shape design of turbine blade internal coolant passages and a transonic wing was demonstrated. The IOSO method was found to be robust and efficient, often requiring fewer analyses than the PGA. With the recent availability of low cost parallel supercomputing based on commodity components, a complete multidisciplinary 3-D design system is expected to become computationally and financially feasible in the very near future.

Acknowledgments

First two authors are grateful for the partial financial support provided by the AFOSR grant FA9550-06-1-0170 monitored by Dr. Todd E. Combs and Dr. Fariba Fahroo and by the ARO grant 50486-MS-H monitored by Dr. William M. Mullins. The first author acknowledges support from the ADVENTURE project of the Research for the Future (RFTF) program supported by the Japan Society for Promotion of Science (JSPS).

REFERENCES

[1] Dennis, B. H., Han, Z.-X. and Dulikravich, G. S., Optimization of Turbomachinery Airfoils with a Genetic/Sequential Quadratic Programming Algorithm, AIAA Journal of Propulsion and Power, Vol. 17, No. 5, Sept-Oct 2001, pp. 1123.1128.

[2] Egorov, I. N., Indirect Optimization Method on the Basis of Self-Organization, Curtin University of Technology, Perth, Australia., Optimization Techniques and Applications (ICOTAf98), Vol.2, 1998, pp. 683.691.

[3] Nomoto, H., Konga, A., Ito, S., Fukuyama, Y., Otomo, F., Shibuya, S., Sato, M., Kobayashi, Y. and Matsuzaki, H., The Advanced Cooling Technology for the 1500 C Class Gas Turbine: Steam-Cooled Vanes and Air-Cooled Blades, ASME Journal of Engineering for Gas Turbines and Power, Vol. 119, 1997, pp. 624.632.

[4] Martin, T. J. and Dulikravich, G. S., Analysis and Multi-disciplinary Optimization of Internal Coolant Networks in Turbine Blades, AIAA Journal of Propulsion and Power, Vol. 18, No. 4, 2002, pp. 896.906.

[5] Han, Z.-X., Dennis, B. H. and Dulikravich, G. S., Simultaneous Prediction of External Flow-Field and Temperature in Internally Cooled 3-D Turbine Blade Material, International Journal of Turbo & Jet-Engines, Vol. 18, No. 1, 2001, pp. 47.58.

[6] Sobieczky, H. and Dulikravich, G. S., Parameterized Aerospace Vehicles for Aerothermodynamic Optimization, DLR-Technical Note H95F-12.93, Goettingen, Germany, 1993.

[7] Sobieczky, H., Dulikravich, G. S. and Dennis, B. H., Parameterized Geometry Formulation for Inverse Design and Optimization, 4th International Conference on Inverse Problems in Engineering, ed: H.R.B. Orlande, Rio de Janeiro, Brazil, May 2002.

[8] Marcum, D. L. and Weatherhill, N. P., Unstructured Grid Generation Using Iterative Point Insertion and Local Reconnection, AIAA Journal, Vol. 33, No. 9, 1995, pp. 1619.1625.

[9] Yoshimura, S., Shioya, R., Noguchi, H., and Miyamura, T., Advanced General-Purpose Computational Mechanics System For Large Scale Analysis And Design, Journal of Computational and Applied Mathematics, Vol. 149, 2002, pp. 279.296.

[10] Jameson, A., Aerodynamic Design via Control Theory, Journal of Scientific Computing, Vol. 3, No. 2, 1988, pp. 233.260.

[11] Obayashi, S., Yamaguchi, Y., and Nakamura, T., Multiobjective Genetic Algorithm for Multidisciplinary Design of Transonic Wing Platform, Journal of Aircraft, Vol. 34, No. 5, 1997, pp. 690.693.

[12] Epstein, B., and Peigin, S., Constrained Aerodynamic Optimization of 3D Wings Driven by Navier-Stokes Computations, AIAA Journal, Vol.43, No.9, 2005, pp. 1946.1957.

[13] Anderson, W. K., Thomas, J. L., and Whitfield, D. L., Three-Dimensional Multigrid Algorithms for the Flux-Split Euler Equations, NASA TP 2829, November 1988.