44th AIAA Aerospace Sciences Meeting and Exhibit
9 - 12 January 2006, Reno, Nevada

AIAA 2006-340

Multiobjective Evolutionary Optimization of a Compressor Stage using a Grid-Enabled Environment

Daisuke Sasaki* and Andy J. Keane†
University of Southampton, Southampton, SO17 1BJ, United Kingdom

and

Shahrokh Shahpar‡
Rolls-Royce plc, Derby, DE24 8BJ, United Kingdom

Multiobjective and multidisciplinary optimization with high-fidelity analysis is becoming an essential factor in the design of turbomachinery blades. Grid-computing environments enable the solution of optimization problem requiring large computational resources. Here, the Geodise computing system is used as a Grid-enabled tool, which realizes the client functionalities for a Globus Grid service in the Matlab environment. It allows users to handle their computing jobs on Grid-enabled machines as Matlab functions. As Matlab includes various useful functions to analyze and visualize data, and to integrate several components via its scripting language, Matlab is used as the main framework of the work presented. In this research, single stage rotor/stator blades for a multistage compressor are optimized to improve aerodynamic performance in terms of efficiency, blockage and loss, while satisfying four aerodynamic constraints to maintain the flow similar to a baseline geometry. To identify the trade-off between three objectives with a reasonable number of function evaluations, the Adaptive Range Multi-Objective Genetic Algorithm is adopted as the optimizer. The benefits of constrained multi-objective optimization of single-stage blades by Evolutionary Algorithms using Grid-enabled environment are discussed.

Nomenclature

A	=	cross-sectional area
B	=	blockage
C_p	=	specific heat at constant pressure
F	=	objective function
G	=	constraint function
L	=	stage loading
$\dot{m}$	=	mass flow rate
P	=	pressure
PR	=	pressure ratio
T	=	temperature
U	=	blade speed
u	=	velocity in the x direction (axial direction)
V	=	velocity
v	=	velocity in the y direction (pitchwise direction)
x	=	axial coordinate
y	=	pitchwise coordinate
β	=	stage exit swirl angle
γ	=	ratio of specific heat capacities

* Research Fellow, School of Engineering Sciences, AIAA Member.
† Professor, School of Engineering Sciences.
‡ Aerothermal Design Specialist, Aerothermal Methods Group, AIAA Associate Fellow, FRAeS.

η_{is} = isentropic efficiency
ρ = density

Subscripts
base = baseline geometry
design = designed geometry
exit = exit condition
inlet = inlet condition
max = maximum
min = minimum

I. Introduction

OPTIMIZATION has become a key technology for developing products in many industries. In turbomachinery design, there are many components to be improved in terms of aerodynamics, structure, vibration, etc, thus multiobjective and/or multidisciplinary optimisation is essential. However, these optimizations tend to require large computational power, especially when high-fidelity analysis such as Computational Fluid Dynamics (CFD) is used in the design evaluation. Therefore, Grid-computing methods are of interest because they enable designers to conduct large numbers of computations on widely distributed computing nodes. Many approaches have therefore been considered for developing Grid-computing techniques by providing middleware and assisted tools.[1-7]

One such is the Grid-Enabled Optimisation and Design Search for Engineering (GEODISE) project which has been conducted at the University of Southampton. This aims to assist engineers to improve the characteristics of their designs by providing a Grid-enabled Problem Solving Environment.[3,5-7] As part of this work, the Geodise computational tool box has been developed for the Matlab® environment. Matlab is widely-used in the engineering community, including in industry.[8] It provides a large number of tool boxes including ones for visualization and data analysis. The Geodise computational tool box works in a similar function. Therefore, designers who are familiar with Matlab can easily conduct Grid computing using the Geodise tool kit. In addition to ease of use for Grid computing, there are other advantages to using Matlab for multiobjective/multidisciplinary optimization. Matlab has many functions and algorithms which help designers to analyze and visualize data after optimization. Moreover, several optimization methods can be easily integrated with the Matlab environment. The Geodise project provides a large collection of such methods.

The design optimization of turbomachinery blade rows is still a challenging problem, because this problem tends to have many components, a large design space, and a large computational domain.[9-13] In the case of compressor design, empirical models and through-flow analysis have been widely used as design tools to predict performance.[14] However, the actual flow in a compressor is very complicated because of viscous effects. When more precise prediction of flow in the compressor or of performance is required, high-fidelity CFD such as Reynolds-Averaged Navier-Stokes computation has to be used. Although progress in computational resources and CFD techniques enables three-dimensional Navier-Stokes computation to be conducted for complicated geometry, there are still difficulties, especially the computational effort required. To conduct optimization using three-dimensional Navier-Stokes calculations, large computational resources are required. Therefore, a Grid-enabled environment is highly desirable in such studies.

In this research, the aerodynamic design target is a blade row in a multi-stage low speed compressor. The compressor rig consists of four stages, each with identical blades, parallel annulus and cantilever stators. It is currently unrealistic to optimize the four-stage compressor directly in terms of computational time. In this study, a single stage is considered for optimization. In the end, the full-stage configuration of the optimized design is computed and compared with the datum design for the validation. The aim of the optimization is to improve efficiency, blockage and total pressure loss by changing the shapes of the rotor and stator blades. In addition, it is also important to identify the trade-offs between the three objectives in comparison to baseline flow conditions. For the purpose of finding designs that satisfy the baseline flow requirements, four equality constraints are imposed. The aerodynamic performances of the compressor stage is analyzed using the HYDRA suite that has been developed by Rolls-Royce jointly with its University Technology Centres (UTCs) to study turbomachinery flows.

Although Grid-computing is effective for aerodynamic optimization, computational time or number of evaluations still has to be kept to a practical level. A simple utility function with gradient-based method is often used to conduct multiobjective optimization efficiently to identify trade-offs. However, it is not always effective to use such a formulation, and it can cause misunderstandings of the trade-off surface. As long as the aim of optimization is to collect multiple non-dominated solutions that form trade-offs, Multiobjective Evolutionary Algorithms (MOEAs)

can be reasonable alternatives. As MOEAs generally require a large number of function calls, methods to control the number of expensive evaluations have to be used. One of the ways to conduct these problems efficiently is to use MOEAs coupled with Response Surface Methodology such as Kriging or Neural Networks.[15-18] An alternative method is to use efficient MOEAs without any approximation model. The former is quite powerful, but we need to consider that errors of approximation vary locally. In addition, this method also requires a significant number of Design of Experiment (DOE) runs to build precise approximation models if the numbers of design variable, design objective and constraint are large. Therefore, a direct apporach is considered in the present optimization. The Adaptive Range Multi-Objective Genetic Algorithm (ARMOGA) is adopted as an optimizer.[19-21] This method has been developed to efficiently identify trade-offs of aerodynamic design problems with high-fidelity codes.

This paper is laid out as follows. In Section 2, the Geodise computing tool box for Grid computing and the optimization framework based on Matlab are described. The details of the design problem are defined in Section 3. Optimization results are presented in Section 4, with concluding remarks in Section 5.

II. Grid Computing and Optimization Framework

The aim of Grid computing is to provide users with access to widely distributed computational resources. The technology has been developed to provide secure access, file transfer, data management, scheduling of jobs, etc. These functions are accomplished by middleware, which enables users to conduct grid computing. The Globus tool kit provides one such set of middleware, including secure authentication and communication (Grid Security Infrastructure, GSI), job submission and control (Grid Resource Allocation and Management, GRAM), file transfer (GridFTP) and resource information (Monitoring and Discovery System). Client software making use of Globus Grid services exists on a number of platforms, and a number of Commodity Grid (CoG) kits that expose Grid services to 'commodity technologies' are provided.[22] By using the interfaces to commodity technologies provided by Java, Python, Perl, etc, the client software can be independent of platform and operating system.

The Java CoG kit provides a number of java packages, which give Application Program Interfaces (APIs) corresponding to the Globus Grid service clients. The Geodise tool kit has been developed to expose such Globus functionalities in the Matlab environment. For this purpose, all functions are programmed in the interpretive Matlab language, and these functions make use of Java classes to access the Java CoG API. As the Geodise function behavior and syntax are consistent with the Matlab environment, Matlab users can use the tool very easily.

The Geodise tool kit provides users with a collection of functions for Grid computing, and typical commands are detailed in Table 1.[3] The authentication of valid certificate is conducted by the *gd_createproxy* command. Users are able to submit compute jobs using the *gd_jobsubmit* command, which makes use of GRAM job managers via the Resource Specification Language (RSL).[23] The command returns a unique job handle to identify the job, which may be used to query or terminate the specified job. Another job submission command *gd_submitunique* provides an extension of *gd_jobsubmit*: it creates a unique directory on the remote machine and the job is then performed in the directory. Two file transfer commands *gd_putfile* and *gd_getfile* are provided to transfer files to and from Grid-enabled resources, respectively.

In this research, Matlab is chosen as the central framework to control the optimization, which is based on time-consuming computing jobs using Grid-enabled facilities, as well as to execute the stand-alone in-house optimizer, the Adaptive Range Multi-Objective Genetic Algorithm (ARMOGA).[19] The optimization framework is illustrated in Fig.1. Matlab produces all the input files which ARMOGA requires to run, and then invokes ARMOGA to generate population data to be evaluated on the grid. Matlab loads the data, and creates design parameter files that are submitted to remote machines so that the corresponding designs are generated and evaluated on the machine (by means of the command *gd_submiunique*). Matlab polls the computing jobs until they finish using *gd_jobpoll*, and then retrieves the data from the remote computing node using *gd_getfile*. When all the results are archived, Matlab provides the required input files and invokes ARMOGA again to produce the next generation data. The optimization continues until the given generation count is reached.

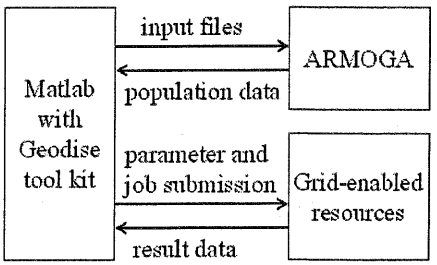


Figure 1. Optimization framework.

Table 1. Geodise toolkit commands.

Function Name	Description
gd_createproxy	Creates a Globus proxy certificate from the user's credentials
gd_jobsubmit	Submits a GRAM job specified by a RSL string to a Globus server. Returns a job handle to the user
gd_submitunique	Creates a unique directory on a remote host and submits a GRAM job that is computed at the directory.
gd_jobstatus	Returns the status of the GRAM job specified by a job handle.
gd_jobpoll	Polls the status of a GRAM job until the job is completed.

Function Name	Description
gd_jobkill	Terminates the GRAM job specified by a job handle.
gd_listjobs	Returnsjob handles for all GRAM jobs associated with the users credentials registered on a MDS server.
gd_getfile	Retrieves a file from a remote host using GridFTP.
gd_putfile	Transfers a file to a remote host using GridFTP.

III. Problem Definition

The design target considered here is the third stage blade row in a four-stage compressor, working at a low speed condition. The datum rotor blades in each stage are identical, and so are those in the stator. The target compressor incorporates a parallel annuls, in other words, the annulus area is constant between stages.

The aim of the optimization is to re-design the three-dimensional blades of both rotor and stator simultaneously to improve three aerodynamic performance measures while keeping similar flow conditions and capacity to the baseline geometry that has already been designed. ARMOGA is used to identify the trade-offs near the baseline condition. The geometry and corresponding mesh of each design candidate is automatically generated using the PArametric Design and RApid Meshing System (PADRAM).[24] The aerodynamic performance of the designs are evaluated using the three-dimensional Reynolds-Averaged Navier-Stokes solver, HYDRA,[25] which is described briefly below.

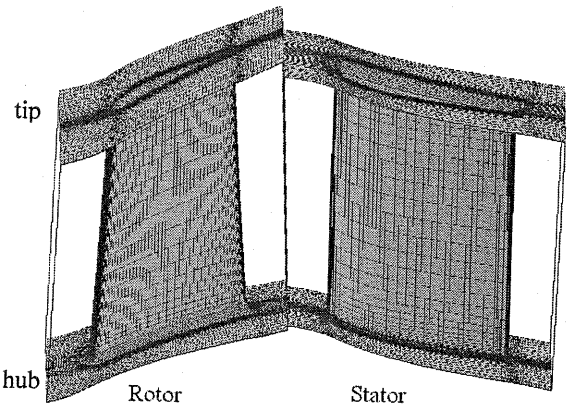


Figure 2. Base geometry and corresponding mesh.

A. Evaluation tools

The simulation codes used to evaluate the compressor stage are a collection of programs developed at Rolls-Royce jointly with its UTCs. The evaluation sequence is shown in Fig. 3. Based on the input design parameters, a three-dimensional blade geometry is defined by PADRAM, and then PADRAM creates a good quality viscous mesh. The boundary conditions and initial flow fields for the flow solver are set in a pre-CFD tool. In addition, multiple coarse meshes are generated to make use of multigrid approaches to improve the convergence performance in steady flow. An in-house, Reynolds-Averaged Navier-Stokes solver, HYDRA, is then used to predict the flow fields. Finally, objective and constraint functions are calculated from the converged flow field.

PADRAM has been developed with the aim of generating blade geometry from well-understood

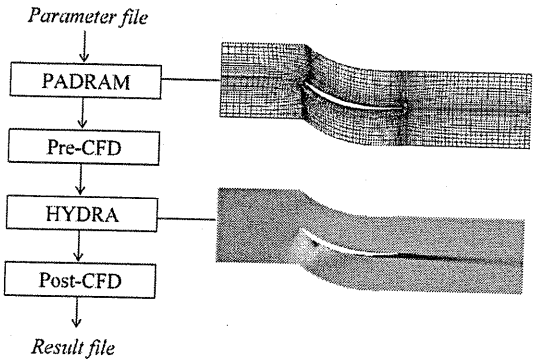


Figure 3. Evaluation sequence.

turbomachinery blade design parameters, and rapidly meshing the resultant geometry. By specifying a parameter name and value, the corresponding shape based on a change to a given baseline geometry is produced. In the present optimization, the three-dimensional blade shape is re-designed by varying stagger angle, lean and sweep at different spanwise locations. PADRAM is capable of producing good quality meshes for viscous computation in such multistage compressors. The mesh for the stage is based on O-H mesh topology, and the tip and hub gap meshes are automatically generated. The mesh density and quality are controlled by the number of nodes in the O-mesh or H-mesh, and their stretching factor.

After the geometry and mesh of the rotor and stator are created, the pre-processing tool is used to integrate separate meshes, to compute edge weights, to provide connectivity information, and to set up boundary conditions. In addition, the tool creates a number of multirgrid levels in order to improve convergence performance. In this approach, the multigrid scheme is achieved by edge collapsing, removing certain edges to selectively coarsen the mesh on which the flow is solved, and damping out high frequency error nodes.[26] In the present evaluation, a four-level multigrid is adopted.

The flow solver, HYDRA, is an unstructured hybrid code using an edge-based data structure to give the flexibility to run on different mesh types.[27] The non-linear solver computes the steady-state Navier-Stokes equations with the Spalart-Allmaras turbulence model.[28] The solver works in an explicit time marching fashion, based on a five-step Runge-Kutta stepping procedure.[29] Typical boundary conditions in compressible aerodynamics are used for the rotor inlet and the stator exit boundary. Radial profiles of total temperature, total pressure, and absolute flow angle are specified at the rotor inlet, and radial profiles of static pressure are specified at the stator exit. A mixing plane boundary is applied for the contact surface between the outflow of the rotor and the inflow of the downstream stator. In the formulation of the mixing plane boundary condition in HYDRA, circumferentially flux averaged values of density, axial velocity, circumferential velocity, radial velocity and static pressure are transformed from the inflow boundary of the mixing plane to the nodes on the outflow boundary and vice versa.

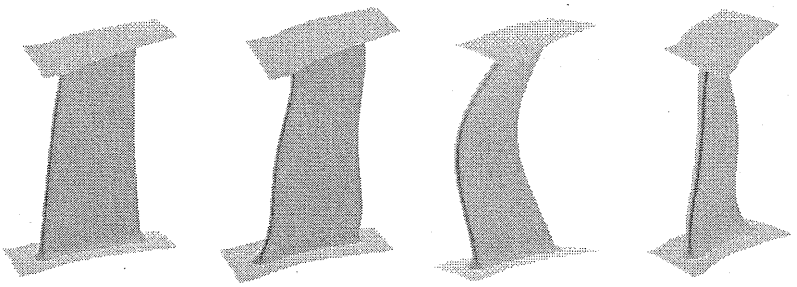
In this research, a part-converged solution is used to reduce the computational time of each CFD evaluation. The flowfield at the fully converged solution of the baseline geometry is set as the initial condition for a design candidate to be computed. This assists in providing rapid convergence compared to the use of an assumed uniform initial condition.

B. Design Parameters

The three-dimensional shape of the blades is re-designed from the baseline geometry by parameters specified in PADRAM. The following four parameters are chosen here to control the shape.

1) XCEN: axial movement of sections along the engine axis (or sweep)
2) DELT: circumferential movement of sections (or lean)
3) SKEW: solid body rotation of sections based on trailing edge position
4) PITCH: pitch angle (controls the number of blades)

Some re-designed shapes of rotor blades produced using the above parameters are shown in Fig. 4. These parameters are applied for rotor and stator blade simultaneously. The variation of the shape is defined at various radial locations. The radial profile can be represented by polynomial functions or NURBS curves.



a) Baseline b) XCEN c) DELT d) SKEW

Figure 4. Geometries defined by parameters in PADRAM.

C. Design Criteria

1. Objective Functions

The following objective functions are considered here in the design of compressor stage 3 (which consists of rotor and stator). The aerodynamic performance of the compressor is measured by the isentropic efficiency η_{is} (because isentropic and polytropic efficiencies are almost identical for single stage analysis):

$$\text{Maximize } \eta_{is} = \frac{\left(\frac{P_{exit}}{P_{inlet}}\right)^{\frac{\gamma-1}{\gamma}} - 1}{\frac{T_{exit}}{T_{inlet}} - 1}. \tag{1}$$

Blockage is perhaps one of the most critical quantities in stage design, as it causes reduction in the passage area because of viscous effects. In the definition shown in Fig. 5, the blockage B is computed purely using the axial velocity u instead of V[30]:

$$\text{Minimize } B = \frac{\int_A \left(V_{high} - \min\left[V_{high}, \max\left[V_{low}, V\right]\right]\right) dA_x}{\int_A dA_x}, \tag{2}$$

$$\text{where } V_{high} = \frac{9}{10} V_{\max} + \frac{1}{10} V_{\min}, \tag{3a}$$

$$\text{and } V_{low} = \frac{8}{10} V_{\min} + \frac{2}{10} V_{\max}. \tag{3b}$$

The blockage is an estimation of the amount of low momentum mass flow in the vicinity of the hub and casing endwalls, or the stage inefficiency. The blockage function is computed at the exit measurement station of each blade row, and an arithmetic mean blockage is adopted as the objective function.

In addition to the above two objective functions, flow loss is also considered. In the optimization, the stage total pressure loss is adopted as the third design objective:

$$\text{Minimize } Loss = \frac{P_{inlet} - P_{exit}}{P_{inlet}} \times 100. \tag{4}$$

The loss is computed for the rotor using relative total pressure and the stator, respectively, and an arithmetic mean loss is to be minimized.

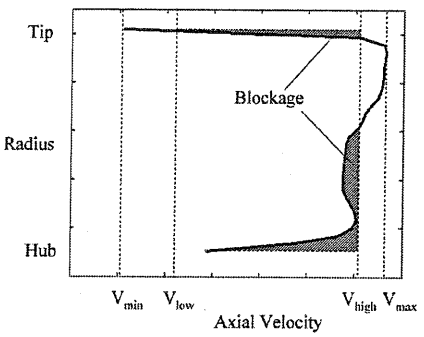


Figure 5. Blockage definition.

2. Constraint Functions

The following constraint functions are considered to maintain the flow and operating conditions similar to those of the baseline geometry; stage loading L, mass flow rate $\dot{m}$, stage exit whirl angle β, and pressure ratio PR:

$$L = \frac{c_p \Delta T}{U^2}, \tag{5}$$

$$\dot{m} = \rho V A, \tag{6}$$

$$\beta = \tan^{-1}\left(\frac{v}{u}\right), \tag{7}$$

$$PR = \frac{P_{exit}}{P_{inlet}}. \tag{8}$$

The mass flow rate and pressure ratio are used to ensure that the new design is operating at a similar location in the compressor characteristic to the baseline design. The stage exit whirl angle is controlled to prevent large flow incidence in the blade rows downstream and to maintain sensible matching between stages.

D. Optimization

ARMOGA is used to solve the present multiobjective constrained optimization problem with the aim of identifying trade-offs between objectives efficiently. The characteristic of ARMOGA is the introduction of range adaptation as described in Fig. 6. Within the range adaptation routine held every M generations, a new search region is determined according to average and standard deviation of selected better solutions. New individuals are then randomly generated in the new design space. Optimization continues and genetic operators are applied to the improved design range. In ARMOGA, each new parent set is selected from the present population and the previous parent set by CHC[31] (so-called best-N selection) and Stochastic Universal Sampling[32] according to fitness values determined by a constraint-handling Pareto ranking method[33] with fitness sharing[34]. In this optimization, Simulated Binary Crossover with a probability of 100% and Polynomial Mutation with a probability of 10% are adopted. To prevent searching similar regions, the crossover probability is set to high.

The three objective-functions (maximize isentropic efficiency and minimize mean blockage and mean total pressure loss) are normalized with the values of baseline geometry as follows:

$$F = \frac{F_{design} - F_{base}}{F_{base}} \times 100. \tag{9}$$

Ideally, the four constraint functions should be treated as equality constraints to maintain the baseline condition. However, as it is difficult to treat the constraints as equality function during optimization, a threshold for each constraint is defined and then feasible designs are those that satisfy all the constraints within these thresholds:

$$G = \frac{G_{design} - G_{base}}{G_{base}} \times 100, \tag{10}$$

$$G' = \mathrm{abs}(G) \le threshold. \tag{11}$$

To find good designs which have the required constraint values taken from the baseline geometry, the threshold value decreases as the optimization proceeds. This process is implemented in ARMOGA by making use of the feature that ARMOGA collects all the designs computed previously during the range adaptation routine. In order to search for feasible solutions effectively, constrained-Pareto ranking is adopted. In addition, several infeasible designs which violate each constraint slightly are preserved in an archive for generating new design spaces during range adaptation. This enables the process to create more new designs near the border between feasible and infeasible space.

Here, the initial population is generated based on three different strategies: uniform random generation, normal-distributed random generation, and perturbation generation. Uniform random generation can produce design candidates randomly across the whole design space. As the baseline geometry is already well-designed and the constraints based on the baseline geometry have to be satisfied, designs having similar geometry to the baseline are assumed to be more practical. Normal-distributed random number sampling produces design variables nearer the baseline with higher probabilities. In addition, perturbation generation is used: all design variables are set to values of the baseline initially, and perturbation is added to each design variable with a probability of 50%. In this optimization, 16 individuals are used at each generation, and the initial population is composed of four uniformly random generated designs, six normal-distributed random designs and six perturbation generated designs. This relatively small population is adopted because of the expense of design evaluation.

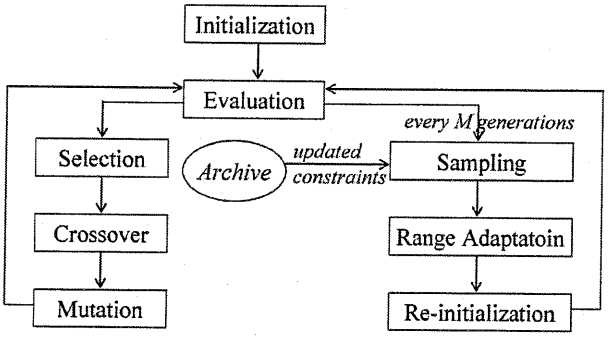


Figure 6. Flowchart of ARMOGA with constraint updating process.

E. Validation of Mesh Density and Part-Converged Computation

As each evaluation is time-consuming, it is reasonable to examine the effects of using a coarse mesh instead of a fine mesh. For the validation of mesh density, the baseline geometry is computed using the original fine mesh (967K nodes) and a coarse mesh (608K nodes). The values are normalized by the original mesh values and summarized in Table 2. The coarse mesh values are within 2%, and thus the coarse mesh reasonably matches the original one.

A rotated geometry having all SKEW values set at -3.0 degrees for the rotor and 3.0 degrees for the stator was also chosen for comparison between fine and coarse meshes. The aerodynamic performance of the geometry obtained from these meshes has been normalized based on the corresponding results for the baseline geometry. The results show a similar trend to the original mesh as indicated in Table 2.

In addition to the use of a coarse mesh, part-converged solutions may be used for rapid convergence studies. The initial flowfield is set to the converged flowfield of the baseline geometry, and 200 multigrid iterations used. The resulting data is also given in the table. As the values of the part-converged solution are almost the same as the fully-converged solution, the part-converged solution with 200 iterations is used here for evaluations during optimization.

Table 2. Comparison of normalized aerodynamic performance.

	Baseline		Rotated geometry		
Mesh	original	coarse	original	coarse	coarse
Convergence	full	full	full	full	part
Isentropic Efficiency	0	-0.26	0.055	0.093	0.081
Blockage	0	1.71	5.63	5.81	5.90
Total Pressure Loss	0	1.96	3.17	2.84	2.93
Loading	0	-0.26	71.71	71.72	71.72
Mass Flow Rate	0	-0.53	5.88	5.92	5.91
Flow Angle	0	-0.38	-16.70	-16.61	-16.62
Pressure Ratio	0	-0.0088	0.062	0.063	0.062

IV. Optimization

A. Fixed numbers of rotor and stator blades

First of all, an optimization is conducted with the same number of rotor and stator blades as the datum geometry (75 rotor blades and 96 stator blades). The three parameters (DELT, XCEN, SKEW) are chosen here to control the shape. To represent the radial profiles efficiently, the parameter radial distributions are defined by the following cubic function:

$$y = a + bx + cx^2 + dx^3 \tag{12}$$

Five control points are set along the radial direction (0, 25, 50, 75, and 100%) to represent the radial variation by this cubic function with least-square fitting to the chosen control points, as shown in Fig. 7. All values at the five control points can be specified for XCEN and SKEW. As the value at the tip (hub for stator) is fixed to 0 for parameter DELT, for this case only four values out of the five control points can be changed. Therefore, 14 design variables are used to define each blade shape, *i.e.*, a single stage shape is defined by 28 design variables. The ranges of the design variables are detailed in Table 3. The perturbation size of XCEN is normalized based on the chord length of the baseline blade at mid-spanwise location.

Constrained three-objective optimization has been conducted using ARMOGA for 20 generations. It took nearly five hours to evaluate each design using a node, and 16 different geometries at one generation were computed in parallel using the available Grid resources. In all, 320 different geometries were evaluated during the optimization. As already noted, the equality-constraint functions are treated by decreasing the thresholds on the inequality constraints as the optimization proceeds. The distributions of constraint according to the design number that has been successfully evaluated are plotted in Fig. 8. The final thresholds are also shown in the figure. The figure shows that the constraint values are widely spread and this illustrates the difficulty in obtaining similar values for all constraints at the same time when using ARMOGA. Actually, by analyzing all computed designs, only eight designs satisfy all four constraints.

The three objective-function values for all the solutions are projected onto the two-dimensional plot in Fig. 9. The direction of the optimization is drawn as an arrow. As the values are normalized by the baseline geometry, the dotted lines that cross the point (0,0) in the figure indicate the baseline value. From the figure, it may be seen that the improvement in efficiency lies within 1 % even when unfeasible solutions are included. Simple trade-offs are observed between the objectives. To investigate the relationship between constraint functions, the relationships of flow angle and pressure ratio against mass flow rate are presented in Fig. 10. When a design satisfies the required mass flow rate, the design can easily satisfy the pressure ratio constraint, because their trends are quite similar. On the other hand, it seems quite hard to obtain a design that satisfies the flow angle constraint even though the design satisfies the mass flow rate because the flow angle distribution is widely spread.

Based on the final constraint thresholds, seven non-dominated solutions have been obtained. The solutions are also plotted in Fig. 9. The trade-offs in the non-dominated solutions are clearly observed, except for the relationship between efficiency and loss. The relationship between efficiency and loss in the non-dominated solutions are roughly linear, and it seems that both performances can be improved at the same time. Three non-dominated solutions are selected for comparison and their performances are summarized in Table 4. S1 has maximum efficiency as well as minimum loss, and S2 has minimum blockage. The non-dominated solution S3 that is superior

to the baseline geometry in all three objective functions is also chosen for comparison and the shapes of these designs with their pressure surfaces are presented in Fig. 11.

As flow angle and mass flow rate are not in a simple relationship, either the mass flow rate or the flow angle of the selected three solutions has more than 1% difference compared to the baseline geometry, as described in Table 4. In the table, in addition to the mean values of blockage and loss, rotor and stator values are also noted. Regarding total pressure loss, the loss caused at the stator increases. This indicates that the stator can be improved or that the rotor's contribution to performance is higher than that of the stator. This is also true for blockage.

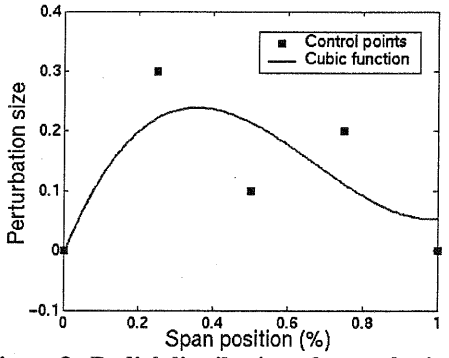


Figure 8. Radial distribution of perturbation.

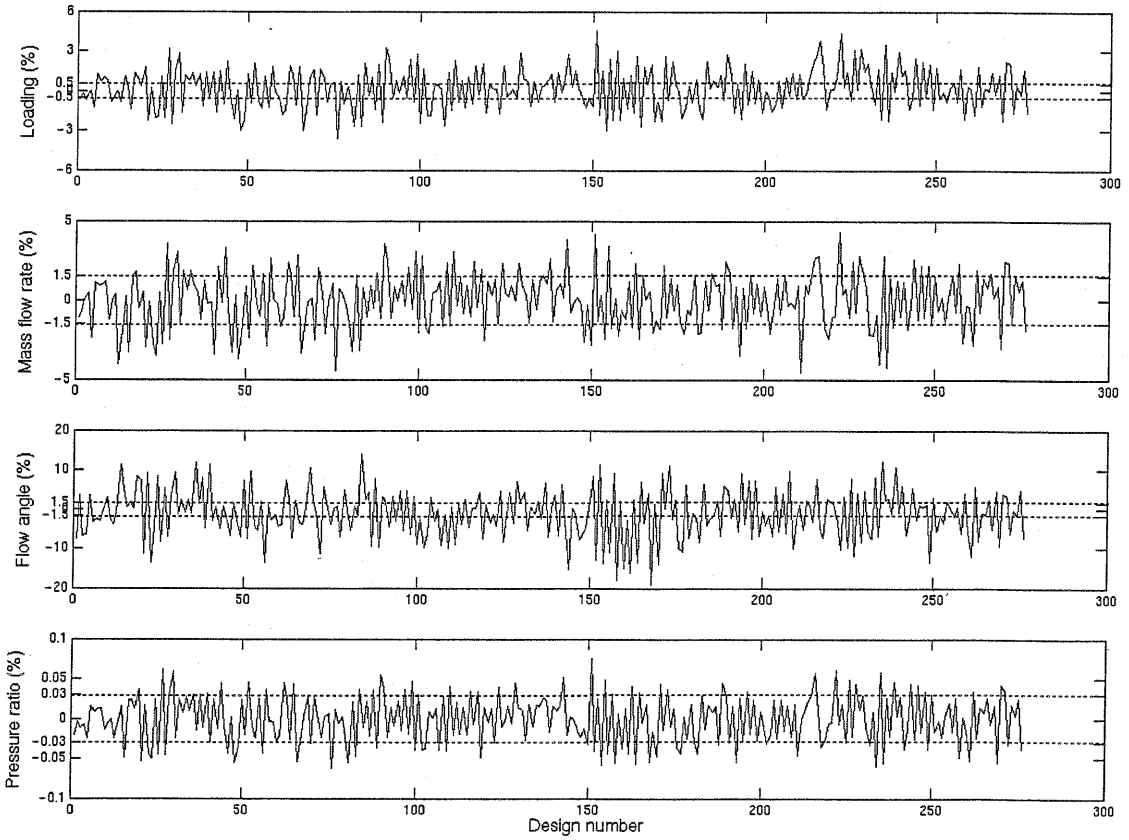


Figure 7. Constraint distributions according to design number.

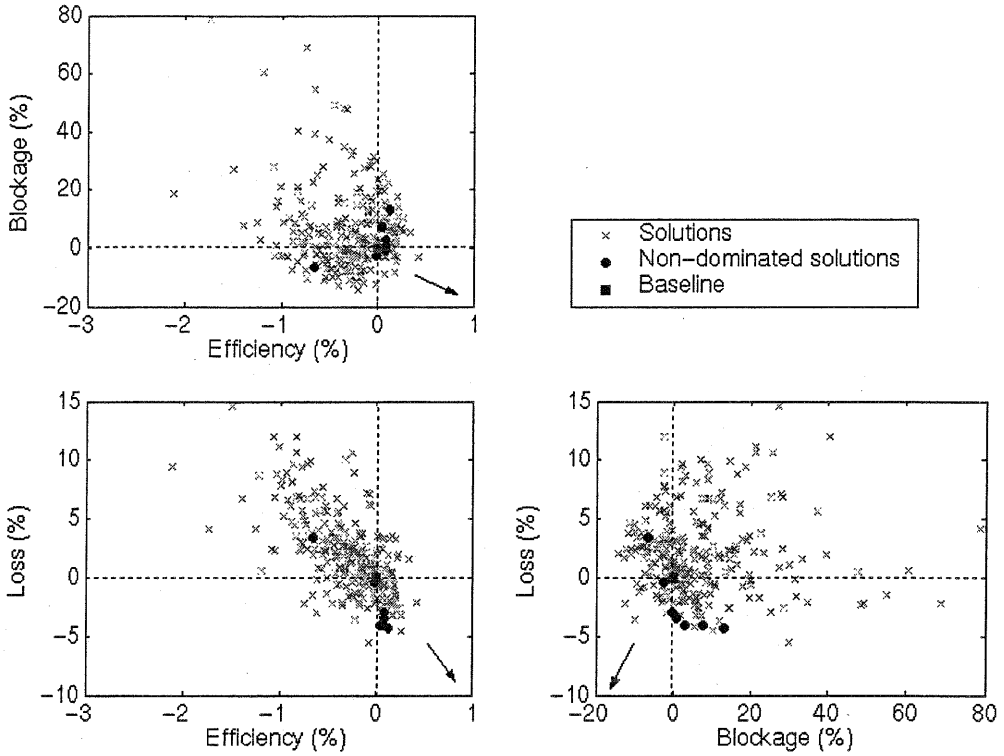
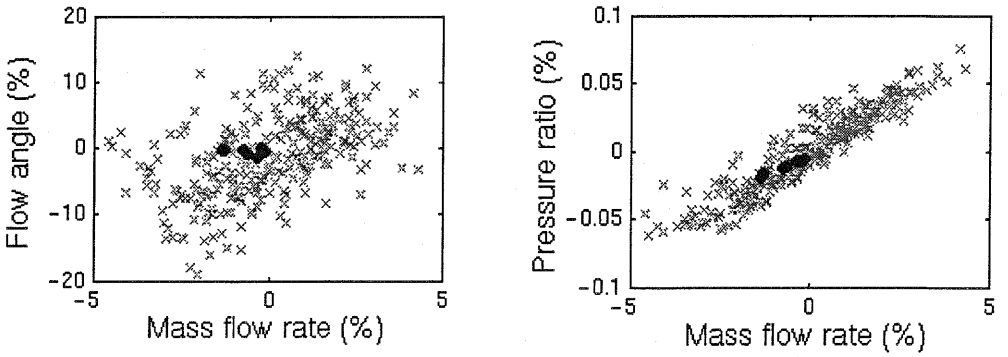


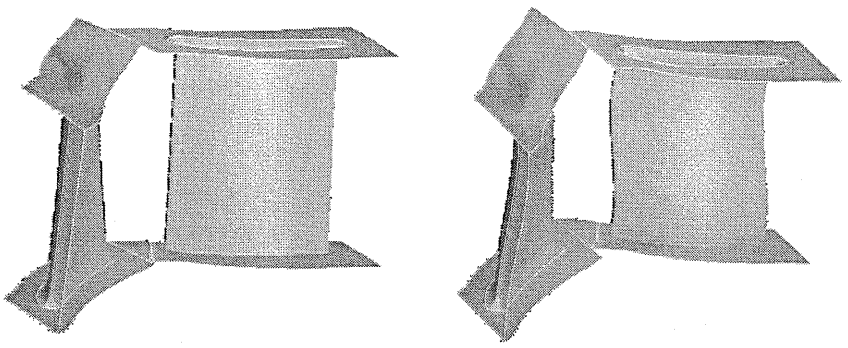
Figure 9. Objective-function space. All solutions are plotted.



a) Mass flow rate and flow angle

b) Mass flow rate and pressure ratio

Figure 10. Selected constraint-function space. All solutions are plotted.



(a) Baseline (b) SC3

Figure 11. Baseline geometry and SC3 geometry with pressure surface.

Table 3. Upper and lower boundaries of design parameters.

	Rotor		Stator	
	Lower boundary	Upper boundary	Lower boundary	Upper boundary
XCEN (%)	-5.35	5.35	-6.20	6.20
DELT (degree)	-2.0	2.0	-2.0	2.0
SKEW (degree)	-5.0	5.0	-5.0	5.0

Table 4. Comparison of normalized aerodynamic performance of selected non-dominated solutions.

	SC1	SC2	SC3
Isentropic Efficiency	0.12	-0.66	0.080
Blockage (Rotor/Stator)	13.16 (-5.1/35.3)	-9.42 (-11.9/0.08)	-0.52 (-10.6/11.7)
Total Pressure Loss (Rotor/Stator)	-4.25 (-8.2/3.0)	3.44 (3.0/4.2)	-3.01 (-7.0/4.3)
Loading	0.13	0.21	-0.073
Mass Flow Rate	1.17	0.15	0.77
Flow Angle	-0.10	-1.21	-1.30
Pressure Ratio	0.012	-0.0010	0.0069

B. Variable numbers of rotor and stator blades

In the previous optimization, the improvement in the efficiency was quite small. It is assumed that the following two factors prevent increase in the efficiency: fixed number of blades, and the low freedom of the three-dimensional shape definition using a cubic-spline function. Therefore, the PITCH parameter is introduced to change the numbers of rotor and stator blades. In addition, instead of the cubic function, B-spline curvature is used to control the radial distribution of the other three parameters (DELT, XCEN, SKEW). In the end, the total number of design variables is then 30. The boundaries of the design variables are specified in Table. 5.

Table 5. Upper and lower boundaries of design parameters (blanket represents number of blades).

	Rotor		Stator	
	Lower boundary	Upper boundary	Lower boundary	Upper boundary
XCEN (%)	-5.35	5.35	-6.20	6.20
DELT (degree)	-2.0	2.0	-2.0	2.0
SKEW (degree)	-5.0	5.0	-5.0	5.0
PITCH (radian)	0.069 (90)	0.1047 (60)	0.0537 (116)	0.0827 (76)

Optimization using these variables has been conducted following exactly the same procedure and settings as before using ARMOGA. The three objective-function values for all the solutions are projected onto the two-dimensional plot in Fig. 12. As the figure shows, nearly 1.5% increase of the efficiency is accomplished in this optimization. Figure 13 shows the relationship between the number of blades and the efficiency, and the mass flow rate as well. From the figure, reduction in the number of blades tends to increase the efficiency. On the other hand, it seems that reducing the number of blades does not affect the mass flow rate. Therefore, it is assumed that the number of blades does not have a large effect on constraint values used in this study.

Based on the same final constraint thresholds as the former optimization, 14 designs satisfy all four constraints. Within the feasible designs, there are four non-dominated solutions shown in Fig. 12. Three of these non-dominated solutions are selected for comparison and their performances are summarized in Table 6. SV1 has maximum efficiency, SV2 has minimum loss, and SV3 has minimum blockage. The non-dominated solutions SV1 and SV2 are superior to the baseline geometry in all three objective functions. Here, the shapes of SV1 and datum designs with pressure surface are presented in Fig. 14. The design SV1 will be a good candidate to increase the performance of the current compressor in terms of aerodynamics at this flow condition.

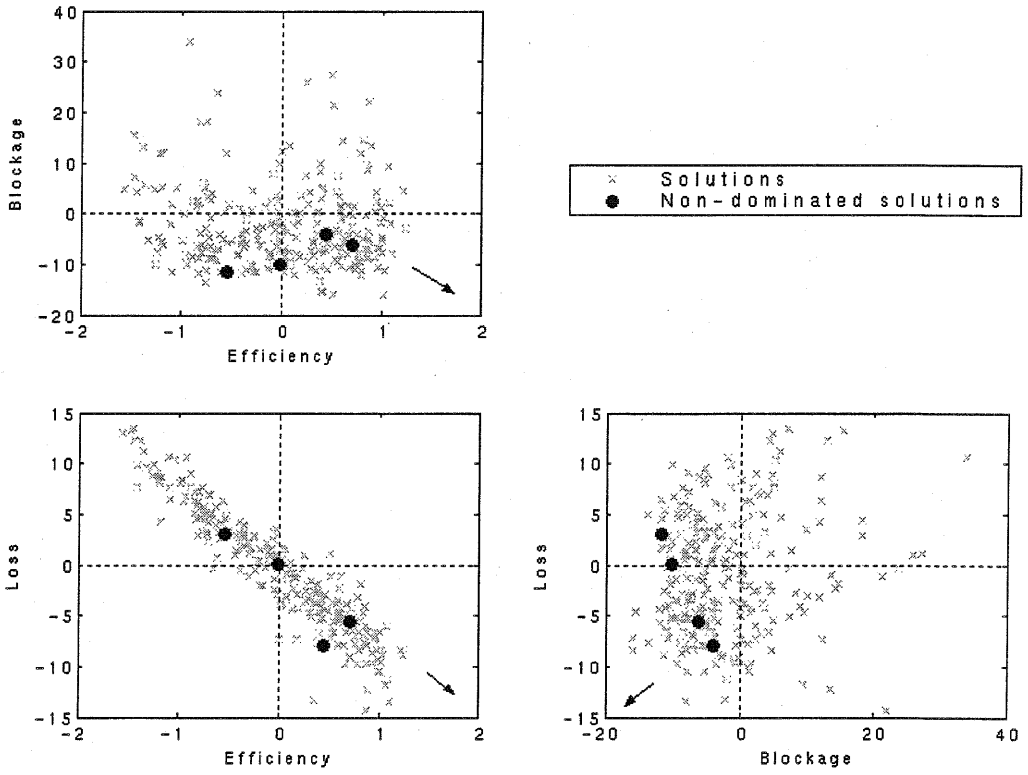
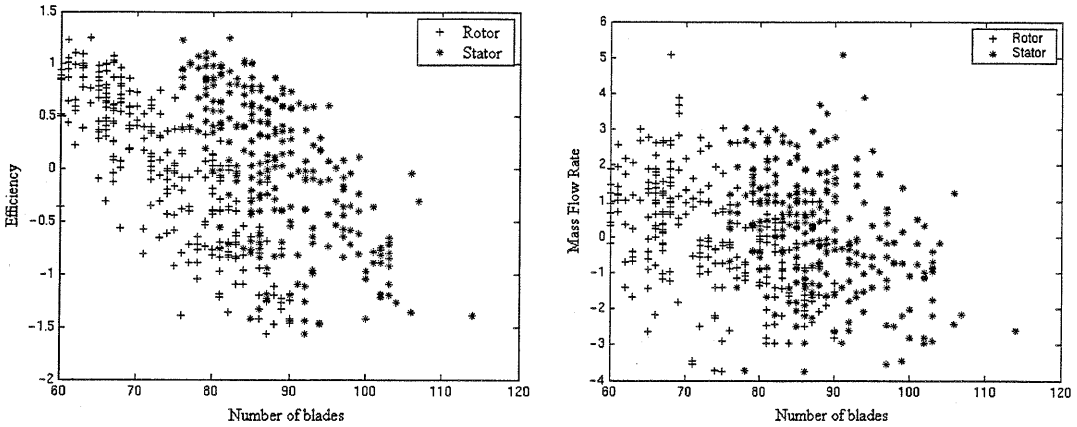
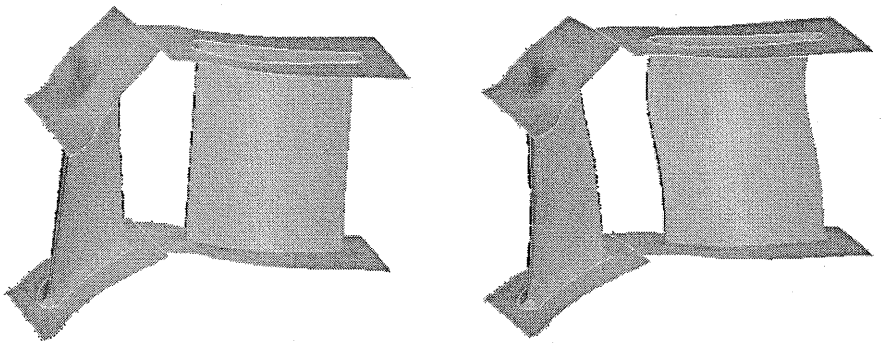


Figure 12. Objective-function space. All solutions are plotted.



(a) Efficiency (b) Mass flow rate

Figure 13. Relationship between functions with number of blades of rotors and stators.



(a) Baseline (b) SV1

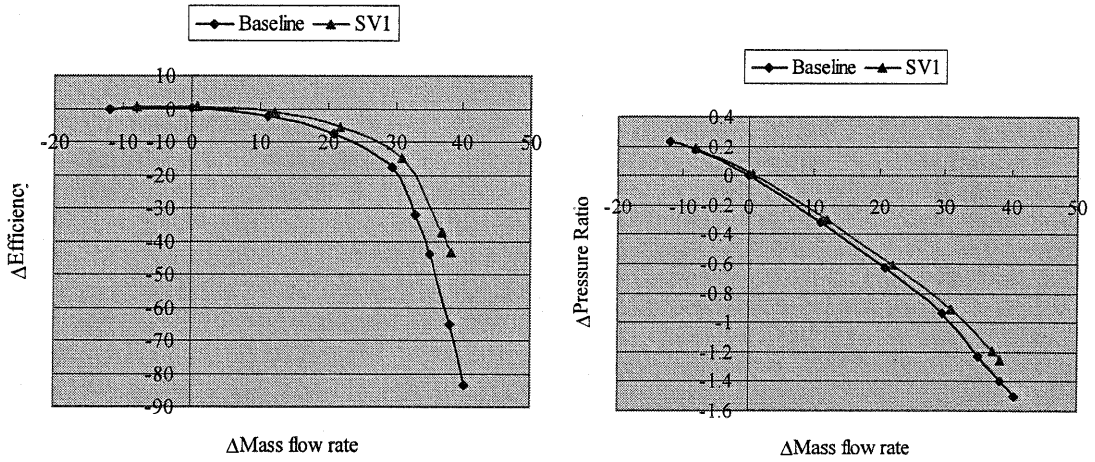
Figure 14. Baseline geometry and SV1 geometry with pressure surface.

Table 6. Comparison of normalized aerodynamic performance of selected non-dominated solutions.

	SV1	SV2	SV3
Isentropic Efficiency	0.71	0.44	-0.54
Blockage (Rotor/Stator)	-6.10	-3.99	-11.64
Total Pressure Loss (Rotor/Stator)	-5.59	-7.93	3.01
Stage Loading	-0.15	-0.095	0.038
Mass Flow Rate	0.64	1.40	-0.36
Flow Angle	1.40	-0.93	0.86
Pressure Ratio	0.0090	0.016	-0.0039
Number of Blades (Rotor/Stator)	68/84	61/81	84/90

C. Flow characteristics of the optimal and datum designs

As a compressor has to work at different conditions, it is desirable that the entire performance is reasonable. Therefore, it is important to evaluate the profile of the performance at various mass flow rates. The optimal design SV1 is analyzed here by changing the pressure profile at the stator exit. In Fig. 15, the efficiency and pressure ratio of SV1 and datum designs are shown. The values are normalized based on the values at the design point of the baseline geometry. Compared to the datum profile, SV1 shows reasonable profiles in terms of efficiency and pressure ratio over different working conditions. Therefore, SV1 is a good candidate to replace the datum configuration.



(a) Efficiency **(b) Pressure ratio**

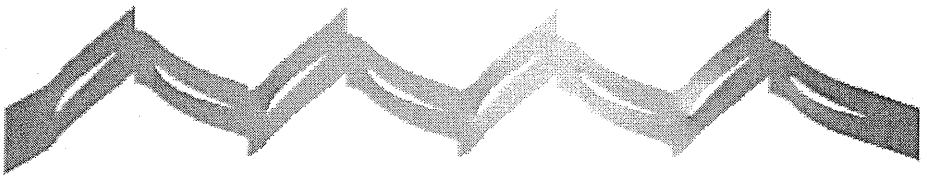
Figure 15. Profiles of flow characteristics according to different mass flow rates.

D. Full-stage calculations of the optimal and datum designs

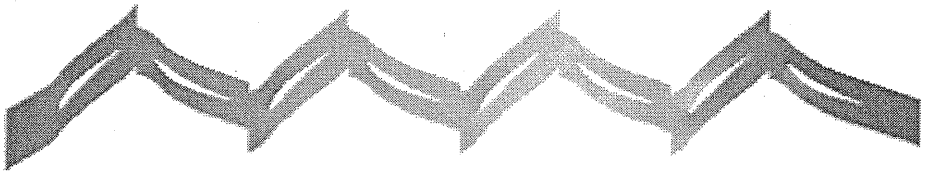
The optimal design SV1 has been obtained by single-stage compressor optimization. However, the compressor is originally composed of four stages with identical rotors and stators. It is therefore necessary to validate the performance of the optimal design in the full-stage configuration. Here, in addition to the isentropic efficiency, the following polytropic efficiency is calculated for comparison because the polytropic efficiency is more reliable for a multi-stage case:

$$\eta_{pol} = \frac{\gamma - 1}{\gamma} \frac{\ln\left(\frac{P_{exit}}{P_{inlet}}\right)}{\ln\left(\frac{T_{exit}}{T_{inlet}}\right)}. \tag{13}$$

The boundary conditions of the current full-stage calculations are specified by throughflow analysis. The pressure contours at mid-location of both SV1 and the baseline geometries are shown in Fig. 16. It is observed that both pressure fields are quite similar. The isentropic and polytropic efficiencies of SV1 normalized by the baseline geometry are described in Table 7. Although 0.7% increase of efficiency is obtained in the single-stage calculation, a slightly reduces 0.53% increase is confirmed in the multi-stage calculation. It is concluded that the improvement in the efficiency of the single-stage optimization contributes to the increase in the efficiency of entire full-stage design.



(a) Baseline



(b) SV1

Figure 16. Baseline geometry and SV1 geometry with pressure surface.

Table 7. Comparison of normalized isentropic and polytropic efficiencies of SV1 for single-stage and four-stage computations.

	Single stage	Four stages
ΔIsentropic Efficiency	+0.709	+0.535
ΔPolytropic Efficiency	+0.699	+0.529

V. Conclusion

Two constrained three-objective optimizations of a compressor stage have been conducted using ARMOGA and a Grid-enabled environment. Equality constraints to satisfy the aerodynamic performance and flow conditions of the baseline geometry were treated as tight pairs of inequality constraints by reducing the constraint thresholds as the optimization advanced. The first optimization has been conducted assuming a constant number of blades, however, the resulting improvement in efficiency was quite low. The second optimization has been conducted with the aim of increasing efficiency by after changing the number of blades. This enabled the search to find better designs that satisfy all four constraints considered. The optimal design's flow profile was evaluated over different working conditions, and it showed a reasonable profile comparable to the datum geometry. Finally, the optimal design was analyzed using a four-stage calculation. Although the size of the increase in efficiency in the multi-stage computation was slightly reduced compared to that in the single-stage computation, the improvement in efficiency was confirmed. The following observations are obtained throughout this study. The number of blades has an effect on the efficiency, but its effect on the aerodynamic constraints is not large. Single-stage optimization is useful to improve the performance of a full-stage compressor, but it cannot evaluate the precise improvement of the full-stage computation. ARMOGA can efficiently explore a large design space having tight constraints.

Acknowledgments

The authors would like to acknowledge the use of the UK National Grid Service in carrying out this work. The authors also gratefully acknowledge the permission of Rolls-Royce plc to publish this paper. The first author would like to thank the members of GEODISE project for their advice. The first author's research is supported by the Japan Society for the Promotion of Science.

References

[1]The Globus Project, URL: http://www.globus.org/ [cited 1 May 2005].

[2]The Condor Project, URL: http://www.cs.wisc.edu/condor/ [cited 1 May 2005].

[3]The Geodise Project, URL: http://www.geodise.org/ [cited 1 May 2005].

[4]Allan, R. et al, "Building the e-Science Grid in the UK: Middleware, Applications and Tools deployed at Level 2," *Proceedings of UK e-Science All Hands Meeting*, Nottingham, 2003, pp. 488-493.

[5]Pound, G. E., Eres, M. H., Fairman, M. J., Xue, G., Keane, A. J., and Cox, S. J., "Grid Middleware for Engineering Design Search and Optimisation," *Proceedings of UK e-Science All Hands Meeting*, Nottingham, 2003, pp. 736-743.

[6]Eres, M. H., Pound, G. E., Jiao, Z., Xu, F., G., Keane, A. J., and Cox, S. J., "Implementation of a Grid-enabled Problem Solving Environment in Matlab," *Proceedings of ICCS 2003*, Melbourne, 2003, pp. 420-429.

[7]Song, W., Keane, A. J., and Cox, S. J., "Two Dimensional Airfoil Optimisation Using CFD in a Grid Computing Environment," *Proceedings of Euro-Par 2003*, Klagenfurt, 2003, pp. 50-57.

[8]The MathWorks, Matlab, URL: http://www.mathworks.com/ [cited 1 May 2005].

[9]Shahpar, S., "Automatic Aerodynamic Design Optimisation of Turbomachinery Components – An industrial Perspective," *Optimization Methods & Tools for Multicriteria/Multidisciplinary Design Applications to Aeronautics and Turbomachinery*, edited by Deconinck, H., Periaux, J., and Giannakoglou, K., VKI Lecture Series 2004-07, 2004.

[10]Shahpar, S., Giacche, D., and Lapworth, L., "Multi-Objective Design and Optimisation of Bypass Outlet-Guide Vanes," ASME Paper GT-2003-38700, June 2003.

[11]Koller, U., Monig, R., Kusters, B., and Schreiber, H.-A., "Development of Advanced Compressor Airfoils for Heavy-Duty Gas Turbines Part I: Design and Optimization," *Journal of Turbomachinery*, Vol. 122, 2000, pp. 397-405.

[12]Oyama, A., Liou, M.-S., and Obayashi, S. "Transonic Axial-Flow Blade Optimization Using Evolutionary Algorithms and a Three-Dimensional Navier-Stokes Solver," *AIAA Journal of Propulsion and Power*, Vol. 20, No. 4, July-August, 2004, pp. 612-619.

[13]Sasaki, D., Shahpar, S., and Obayashi, S., "Multi-Objective Optimization of Low Pressure Compression System," *CD-Rom Proceedings of ICAS2004 (The Twenty-Fourth Congress of International Council of the Aeronautical Sciences)*, Yokohama, Japan, August 2004.

[14]Aungier, R. H., *Axial-Flow Compressors*, ASME Press, New York, 2003.

[15]Keane, J. A., "Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods," Journal of Aircraft, Vol. 40, No. 4, 2003, pp. 741-750.

[16]Poloni, C., Padovan, L., Parussini, L., Pieri, S., and Pediroda, V., "Robust Design of Aircraft Components: A Multi-Objective Optimization Problem," *Optimization Methods & Tools for Multicriteria/Multidisciplinary Design Applications to Aeronautics and Turbomachinery*, edited by Deconinck, H., Periaux, J., and Giannakoglou, K., VKI Lecture Series 2004-07, 2004.

[17]Giannakoglou, C. K., "Neural Network Assisted Evolutionary Algorithms, in Aeronautics and Turbomachinery," *Optimization Methods & Tools for Multicriteria/Multidisciplinary Design Applications to Aeronautics and Turbomachinery*, edited by Deconinck, H., Periaux, J., and Giannakoglou, K., VKI Lecture Series 2004-07, 2004.

[18]Shahpar, S., "Design of Experiment, Screening and Response Surface Modelling to Minimise the Design Cycle Time," *Optimization Methods & Tools for Multicriteria/Multidisciplinary Design Applications to Aeronautics and Turbomachinery*, edited by Deconinck, H., Periaux, J., and Giannakoglou, K., VKI Lecture Series 2004-07, 2004.

[19]Sasaki, D., and Obayashi, S., "Efficient Search for Trade-offs by Adaptive Range Multi-Objective Genetic Algorithms," *Journal of Aerospace Computing, Information, and Communication*, Vol. 2, January 2005, pp.44-64.

[20]Sasaki, D., Obayashi S., and Nakahashi, K., "Navier-Stokes Optimization of Supersonic Wings with Four Objectives using Evolutionary Algorithms," *Journal of Aircraft*, Vol. 39, No. 4, July 2002, pp.621-629.

[21]Sasaki, D., and Obayashi, S., "Visualization of Global Trade-Offs in Aerodynamic Problems by ARMOGAs," *CD-Rom Proceedings of WCCM VI in conjunction with APCOM'04*, Beijing, China, September 2004.

[22]Commodity Grid Kits, URL: http://www.globus.org/cog/ [cited 1 May 2005].

[23]The Globus Resource Specification Language RSL v1.0 URL: http://www-fp.globus.org/gram/rsl_spec1.html [cited 1 May 2005].

[24]Shahpar, S., and Lapworth, L., "PADRAM: Parametric Design and Rapid Meshing System for Turbomachinery Optimisation," ASME Paper GT-2003-38698, June 2003.

[25]Lapworth, L., "Hydra-CFD: A Framework for Collaborative CFD Development," *International Conference on Scientific and Engineering Computation (IC-SEC)*, Singapore, 2004.

[26]Muller, J. D., and Giles, M. B., "Edge-Based Multigrid Schemes for Hybrid Grids," *Proceedings of the Sixth ICFD Conference on Numerical Methods for Fluid Dynamics*, Oxford, 1998.

[27]Moinier, P., Muller, J. D., and Giles, M. B., "Edge-based Multigrid and Preconditioning for Hybrid Grids," *AIAA Journal*, Vol. 40, No. 10, 2001.

[28]Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flow," AIAA 92-0439, 1992.

[29]Martinelli, L., "Calculations of Viscous Flows with a Multi-grid Method," Ph.D Thesis, Dept. of Mech. and Aerospace Eng., Princeton University, USA, 1987.

[30]Harvey, S.A. et. al., "An Automatic Design Optimisation System For Axial Compressors. Part I: Software Development", June 2003, GT2003-38115.

[31]Eshelman, L., "The CHC Adaptive Search Algorithm: How to Have Safe Search When Engaging in Nontraditional Genetic Recombination," *Foundations of Genetic Algorithms*, edited by Rawlings, G. J. E., Morgan Kaufmann, 1991, pp. 265-283.
[32]Baker J. E., "Reducing Bias and Inefficiency in the Selection Algorithm," *Proceedings of the 2nd International Conference on Genetic Algorithms*, 1987, pp. 14-21.
[33]Deb, K., *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, Ltd., Chichester, 2001.
[34]Fonseca, C. M., and Fleming, P. J., "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generallization," *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1993, pp. 416-423.