Proceedings of the International Gas Turbine Congress 2003 Tokyo
November 2-7, 2003

IGTC2003Tokyo TS-036

The Development of a Genetic Algorithm Code for Secondary Flow Injection Optimization in Axial Turbines

Carlos Felipe Ferreira FAVARETTO[1], Ken-ichi FUNAZAKI[1] and Tadashi TANUMA[2]

[1] Department of Mechanical Engineering
Iwate University
3-5-4 Ueda, Morioka, Iwate 020-8551, JAPAN
Phone: +81-19-621-6424, E-mail: funazaki@iwate-u.ac.jp
[2]Turbine Design and Assembling Department, Toshiba Corporation, Yokohama, JAPAN

ABSTRACT

Flow injection is being used in steam turbines for some combined or geothermal plants in order to increase its total thermal efficiency. It can be easily imagined, however, that such flow injection is accompanied by additional aerodynamic loss due to the flow mixing or the change in local flow angle. The present study is devoted to finding the optimum flow injection angle and blade lean angle for minimum energy loss and minimum outlet yaw angle deviation. A multi-objective non-dominated sorting genetic algorithm coupled with a commercial *CFD* code is developed. The system is run in a fully automated fashion requiring no user interaction during the whole optimization process. The results revealed that low flow injection angles to the suction side of the nozzle combined with high positive lean angle are more suitable for minimizing the energy loss. Lower yaw angle values were obtained for no lean or small lean angle configurations, regardless of the flow injection angle.

NOMENCLATURE

Abbreviations

PS	pressure side
SS	suction side
SUS	stochastic universal sampling

Symbols

c	[m]	chord
L_ε	[m]	eddy length dissipation scale
P		pressure
Re		Reynolds number = $u_1 c_x / \nu$
Tu		turbulence intensity
α		yaw angle
β	^{0}	flow injection angle
$\Delta\alpha$	^{0}	flow angle deviation
γ	^{0}	blade lean angle
ν	[m^2/s]	kinematic viscosity
σ_{share}		sharing parameter
ξ		energy loss coefficient

Subscripts

1, 2	inlet, outlet
0	stagnation condition
x	axial
s	slot (injection)

Superscripts

$\bar{}$	area average

Manuscript Received on April 4, 2003

INTRODUCTION

Secondary flow injection into a flow field can be found in many mechanical engineering applications, ranging from micro air injection in axial compressors (Nie et al., 2002) to steam injection in high efficiency combined cycle steam turbines. In some kinds of steam turbines for combined cycle or geothermal power plants, injection flows (from 5 to 15% of main mass flow rate) from steam generators are introduced between turbine stages, resulting in a substantial improvement of the thermodynamic efficiency. As a counterpart of the gain due to the injection of steam, the loss due to the mixing of the secondary and the main streams may incur in additional pressure loss. The upstream steam injected from the outer casing is usually at a different temperature and velocity than the main flow. Since the injected steam and the main flow are not completely mixed in the vicinity of the stator leading edge, pitchwise and spanwise variations of the mass flow and the unsteadiness of the inlet flow angle occur (Fig. 1). Such variation will change the turbine stage operating conditions, affecting the blading flow pattern and efficiency. In the present paper, such disturbances upstream the nozzle vanes are denominated “inlet distortion”. The study of the effects of the flow distortion, which is caused by the flow injection or due to upstream flow disturbances has been investigated by many researchers (Moser, 1989, Hirai et al., 1987).

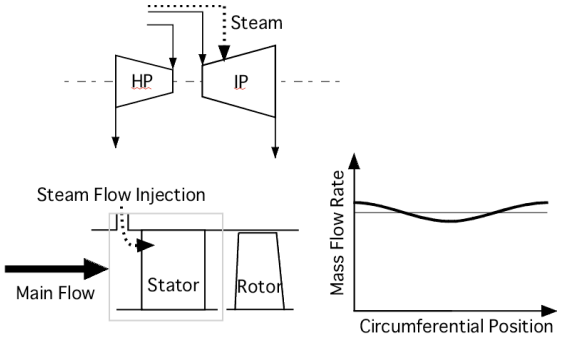

Figure 1 - Flow injection schematic diagram.

In some turbine stage configurations the flow is tangentially injected upstream of the stator. In such case, the injection flow may be directed to the suction side of the stator at some circumferential positions and to the pressure side of the blade for other locations. This circumferential variation of the injection angle may cause a significant impact on the flow field, i.e., the variation in the outlet flow angle and increase in loss. In order to quantify these effects, Funazaki et al. (2003a), performed numerical calculations in a

turbine stage and compared it with the experimental data (Kamata et al. 2002) measured in the single stage turbine rig. Figure 2 shows the circumferential variation of the velocity vectors at the vicinity of the injection slot. It was observed that the loss due to the flow injection could be reduced if the flow angle is orientated to the suction side of the stator, following the direction of the passage vortex.

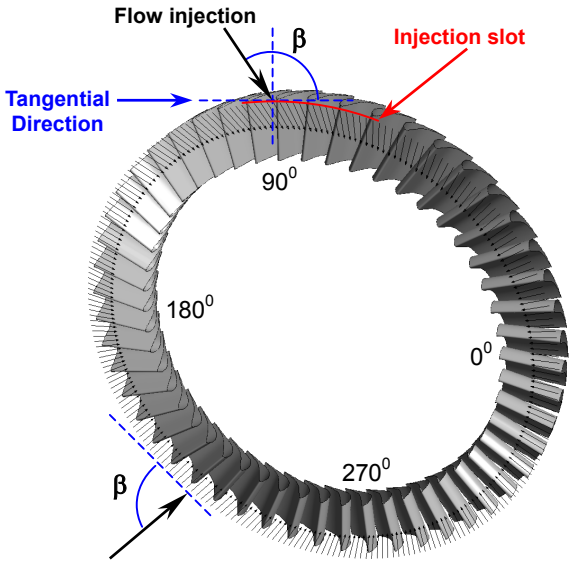

Figure 2 - Velocity vector distribution in the vicinity of the flow injection slot (Funazaki et al., 2003).

Apart from controlling the injection angle, the optimization of the blade geometry was also found to be effective in reducing the loss. Funazaki et al. (2002a) analyzed the effect of injection flow rate on leaned nozzle vanes. It was found that the blade which was leaned to the positive (pressure) side provided a reduction in the overall energy loss but increased the outlet flow angle deviation.

The combined effects of both optimum flow injection angle and blade lean angle, however, have not yet been documented. Based on the previous studies the authors propose a multi-objective optimization adopting as design variables the blade lean and flow injection angle and as target functions minimum energy loss and minimum flow angle deviation.

The use of *GA* for multi-parameter optimization has become a popular technique in a wide range of engineering fields, such as building thermal design (Wright et al., 2002), aerodynamic design of airfoils (Vicini and Quagliarella, 1997) and gas turbine related applications (Trig et al., 1999, Akmandor et al., 2002, Funazaki and Favaretto, 2003b). The reason for adopting such technique is found by the fact that it is robust, simple to implement and provides a multi-directional search. Conventional optimization methods such as "hill climbing" algorithms search in one direction of the domain only and are strongly limited to well behaved target functions. The *GA*, on the other hand, can handle complex non-linear target functions and searches for the optimum point in many directions, avoiding premature convergence at local peaks, which may not represent the global maximum of the search domain.

The simplicity in translating the *GA* method into a computational code is also one of its great advantages. A *GA* code consists of basic mathematical operations which can be parallelized in a very straightforward manner. Another motivation for the increasing number of researchers on the *GA* field may be the fact that it can be directly applied to multi-objective problems. This class of problems is often found in many kinds of applications where the objectives are conflicting with each other, such as minimizing the deflection of a cantilever beam while minimizing its weight.

Among many authors, Goldberg (1989) described the basics about *GA*, including sample program lists. For more detailed information on multi-objective *GA*, interested readers are encouraged to refer to the book written by Deb et al. (2002).

OPTIMIZATION PROBLEM DESCRIPTION

The model for the current study consists of a turbine nozzle vane with an injection slot located at approximately 25% of the axial chord upstream of the leading edge (Fig.3).

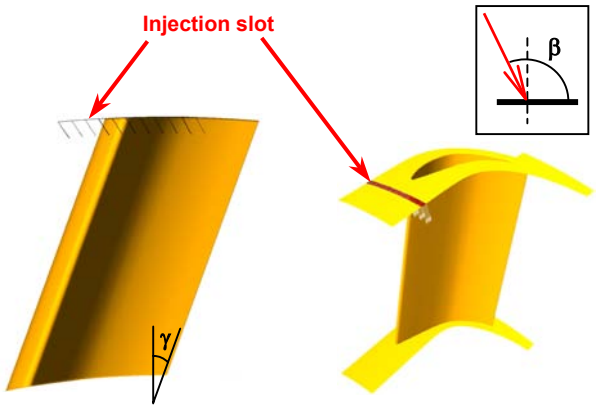

Figure 3 - Flow injection and blade lean definition.

The design variables for the optimization task are the blade lean angle ($0^0 \le \gamma \le 20^0$) and the flow injection angle ($60^0 \le \beta \le 120^0$). The limits for γ were established in order to avoid excessive outlet flow angle deviation and the β was constrained to the calculated range shown in Fig.2. The negatively leaned blade case was discarded from the analysis since it is not as efficient as the positively leaned one for reducing the loss (Funazaki et al., 2002a). The objectives of the present optimization task are to minimize the energy loss coefficient $\bar{\xi}$ defined by Eq.1 and minimize the flow angle deviation $\Delta\alpha$ given by Eq.3.

$$\bar{\xi} = \frac{\left(\bar{P}'_{01}/\bar{P}_{02}\right)^{\frac{\kappa-1}{\kappa}} - 1}{\left(\bar{P}'_{01}/\bar{P}_{2}\right)^{\frac{\kappa-1}{\kappa}} - 1} \tag{1}$$

$$\bar{P}'_{01} = \frac{\bar{P}_{01}\cdot\dot{\bar{m}}_1 + \bar{P}_{0s}\cdot\dot{\bar{m}}_s}{\dot{\bar{m}}_1 + \dot{\bar{m}}_s} \tag{2}$$

$$\Delta\alpha_2(\gamma,\beta) = \left|\,\alpha_2(\gamma,\beta) - \alpha_2(\gamma{=}0,\beta{=}0)\,\right| \tag{3}$$

where: $\bar{\xi}$ is the area averaged energy loss coefficient, $\bar{P}'_{01}$ the weighted averaged inlet total pressure, $\bar{P}_{02}$ the area averaged outlet total pressure, $\bar{P}_2$ the area averaged outlet static pressure, $\bar{P}_{01}$ the area averaged inlet total pressure, $\bar{P}_{0s}$ the area averaged total pressure at the slot, $\dot{\bar{m}}_1$ the area averaged inlet mass flow, $\dot{\bar{m}}_s$ the area averaged mass flow at the slot, κ the specific heat ratio, $\Delta\alpha_2$ the outlet flow angle deviation, α_2 the yaw angle, γ the blade lean angle and β the flow injection angle.

The geometry of the unleaned blade (γ=0) with no injection (datum) was initially generated based on the original design data. Total pressure measurements on a traversing plane located 17% of c_x downstream of the same nozzle were performed by Kamata et al. (2002). The results in Fig.4 show the agreement between the CFD and experiments for almost whole spanwise direction. However, from 92% span to the casing wall the CFD results produced higher total pressure ratio values. The authors believe that the reason for such trend stems from some modeling simplifications employed in the numerical simulation. The actual geometry where the experiments were taken consists of a single turbine stage. Thus, the

results from Kamata et al. (2002) are taking into account additional effects, which may have produced the difference observed near the casing. In the actual case, the runner is rotating close to the traversing plane and there is also an interaction between the rotor shroud and casing recess. In the numerical simulation, however, the rotor is not modeled neither the casing recess. Instead, no-slip condition is applied at the casing and wall function is used. Some wiggles in the experimental data, possibly caused by non-uniform inlet condition upstream of the nozzle, were found from 10% to 70% span. These effects are not believed to have affected the overall energy loss evaluation since the area averaged results showed a difference lower than 0.1%.

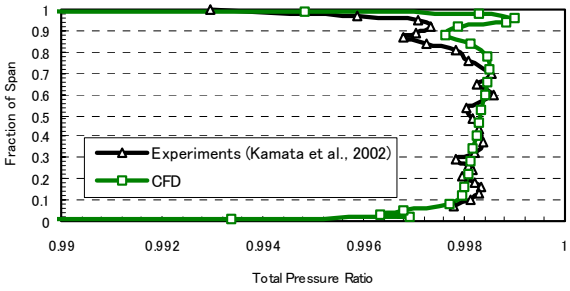

Figure 4 - Total pressure ratio downstream of the nozzle (no blade lean, no flow injection).

OPTIMIZATION TOOL

Genetic Algorithms

The *GA* is a powerful optimization tool based on the theory of evolution, which means that the "best fit" individuals in one generation survive. The "fitness" in *GA* is the function to be optimized (target function) and the parameter set or problem variables are denominated a "chromosome". In the present study, the real value of each design parameter is encoded as a string of binary digits. For instance, string *10111* refers to $\gamma = 14.83^0$ and string *00100* to $\beta = 112.25^0$. The strings for each one of the parameters are blended into a large string, forming the chromosome *1011100100*. The *GA* works with a number of chromosomes for each iteration or generation, providing a search in multiple directions of the domain simultaneously. Preliminary tests with the *GA* code here described showed that a constant population size of *40* chromosomes was suitable for the optimization task.

The starting point for an optimization using *GA* is a process called *initialization*. The initial population of chromosomes can be generated automatically by invoking a pseudo-random number subroutine, usually available in computer language compilers. The *GA* converges to the same result independent of the starting population. According to Trigg et al. (1999), a given start with specified values for the chromosomes does not seem to be an important feature since initial convergence is rapid.

After the initial population has been created, the *GA* needs to evaluate the fitness for each one of the chromosomes belonging to the initial generation. This means judging how well each chromosome is performing according to their *phenotype* (design parameters). The tool used to evaluate the fitness is independent on the *GA* code and can be externally called by the program. This is a remarkable feature which allows *GA* to be applied in almost any research field. For the present optimization task, the commercial *CFD* code *CFX-5.5.1* (*Ansys Inc.*) was used for calculating the energy loss and yaw flow angle deviation downstream of the nozzle vane.

With the fitness values for the chromosomes of the initial generation calculated, the *GA* code must select the candidates for mating. In other words, the individuals will be rated according to their fitness and the ones with the highest fitness are likely to be elected for reproduction. This part is the most important one in a *GA* code since it will directly influence the search direction. In the case of single objective *GA* the criteria for selecting the best fit individual is based simply on the result obtained from the CFD analysis. For multi-objective *GA*, however, there will be a family of optimum solutions for each generation. Therefore, it is not possible to simply base the selection criteria on the "raw" fitness values. A fitness scaling is necessary. The procedure adopted in the present code is based on the *non-dominated sorting* technique. First, all *40* chromosomes in the population are evaluated according to their rank. The definition of rank states that an individual with rank one is the one who is not dominated by anyone else, i.e., no other chromosome is better than him in every objective. For instance, a chromosome that provides the lowest flow angle deviation and the highest loss is not better nor worse than a chromosome that produces the lowest loss and the highest flow angle deviation. Both chromosomes are considered to produce an optimum solution, therefore, they are *non-dominated* and belong to the *optimal front* or *Pareto optimal front*. All chromosomes with rank higher than one are *dominated*. The rank therefore indicates the number of solutions that dominate (are better in all aspects) each one of the chromosomes. The higher the rank, the farther the distance from the Pareto optimal front. The fitness is recalculated by Eq.4 so that chromosomes in the same rank will have the same fitness.

$$F_i = N - \sum_{k=1}^{r_i - 1} \mu(k) - \sigma_{share}\left(\mu(r_i) - 1\right) \qquad (4)$$

where F_i is the rank based fitness, N the population size, σ_{share} the sharing parameter and $\mu(r_i)$ the number of solutions in rank r_i. In the present paper σ_{share} was assumed as constant through all generations and equal to *0.158*. This value was chosen based on preliminary *GA* calculations for a two-objective problem validated with the analytical solution. Figure 5 presents an example of fitness recalculation based on the rank and the sharing parameter (in this case $\sigma_{share} = 0.5$) for a minimization problem of both objectives.

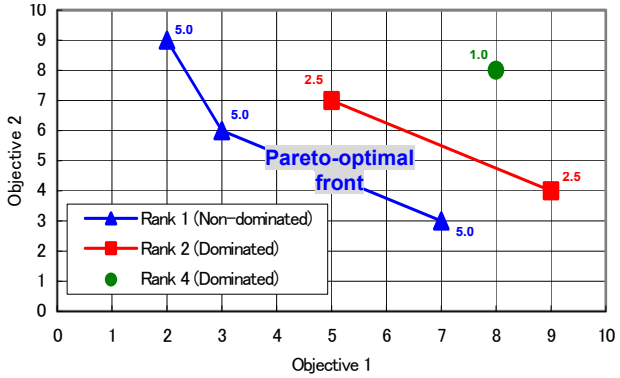

Figure 5 - Ranking of solutions for a minimization problem.

After assigning the rank-based fitness, a niching operator (Fonseca and Fleming, 1993) among solutions of each rank is invoked. This technique is implemented in order to provide a suitable distribution of the optimal solutions along the front, avoiding sections of the curve clustered with points while others poorly represented. In terms of *GA*, it means that chromosomes in crowded regions (clustered points) should share more among themselves than chromosomes that are far from each other. From such definition stems the concept of sharing function, which is calculated based on the inter-chromosomal distance and σ_{share}. The niche count is then evaluated based on the sharing function for each one of the chromosomes. The rank-based fitness is scaled by the niche count.

Based on the scaled fitness the selection process is performed in the same manner as for single objective *GA*. Several methods for accomplishing this task are described in the literature. The most

common one is called *roulette wheel*, in which all chromosomes of a population share a certain sector of a wheel, proportional to their fitness. The wheel is then spun and the chromosome selected. This method, however, causes premature convergence of the results because the best fit chromosome tends to dominate the others and cause their early extinction. The *GA* will follow one direction, leaving other possible maximum locations behind. Considering these limitations, the *stochastic universal sampling (SUS) selection* method was chosen instead. This method is quite similar to the previous one except that, in addition to the proportionally divided wheel, a uniform spaced scale is also defined around the wheel. The number of divisions of this scale equals the population size. Thus, only one spin is necessary for selecting all *40* chromosomes for mating. This method not only provides high probability of selecting the best fit chromosome but also offers a higher chance for chromosomes with poor performance to be selected, maintaining the diversity of the population. The set of chromosomes selected for reproduction is defined as *mating pool*.

After selecting the eligible chromosomes for reproduction, the *recombination* process is performed. The chromosomes mate (parents), generating two new chromosomes (children) in order to keep the population size constant. The *genes* or the string bits are exchanged between the children at a probability of *0.9*. This value was increased from the one adopted for the single objective *GA* code previously developed by the authors. In this way, the mutation operator, which is necessary for diversity of the population, could be eliminated. Like the selection process, there are many ways to perform recombination. The one-point crossover was chosen for the present paper. In this recombination method a bit location is randomly chosen. The information from the bit location just after the selected bit to the end of the string is exchanged between *child1* and *child2*. For instance, suppose the selected bit location is *7*, *child1 = 1001000111* and *child2 = 1010111100*. After crossover, *child1* would become *1001000100* and *child2 1010111111*.

After the four main operations (selection, recombination, elitism and mutation) have been completed the children replace the current population of chromosomes. The process is repeated until the convergence criteria has been satisfied. Table 1 shows a summary of all *GA* parameters adopted in the present paper.

Table 1 - Genetic algorithm parameters.

Chromosome length	10
Type of coding	binary
Population size	40 (constant)
Selection method	SUS
Recombination method	one-point crossover
Crossover probability	0.9
Number of objectives	2
σ_{share}	0.158 (fixed)

Optimizer Implementation

The *GA* code here described was developed in a hybrid *FORTRAN/Perl/UNIX* shell script language. This technique was used so that the *CFX-5.5.1* code could be combined with the optimizer. Figure 5 presents a flowchart describing the mechanism of the code.

One of the critical problems when using a *GA* code is the *CPU* time. The fitness has to be evaluated for all chromosomes belonging to every generation. Thus, for a population of *30* chromosomes, after *30* generations, the flow solver would have been called *900* times. If each one of the solver executions are performed sequentially, the *CPU* time would be *900* multiplied by the time required for each run. In order to provide faster results fully automated parallel processing and database subroutines were implemented in the code. In the beginning of each generation, the code reads the database and discards all chromosomes that have already been solved. The grid system is then generated sequentially for the remaining chromosomes and the solution loop starts. The code submits to the queue the jobs to be solved until the number of available *CFX-5.5.1* solver licenses has been reached. The pending jobs wait until another license is granted. The loop ends when all jobs have been submitted to the queue. After that, the post-processing tool is run in batch and the two objectives ξ and $\Delta\alpha$ are evaluated.

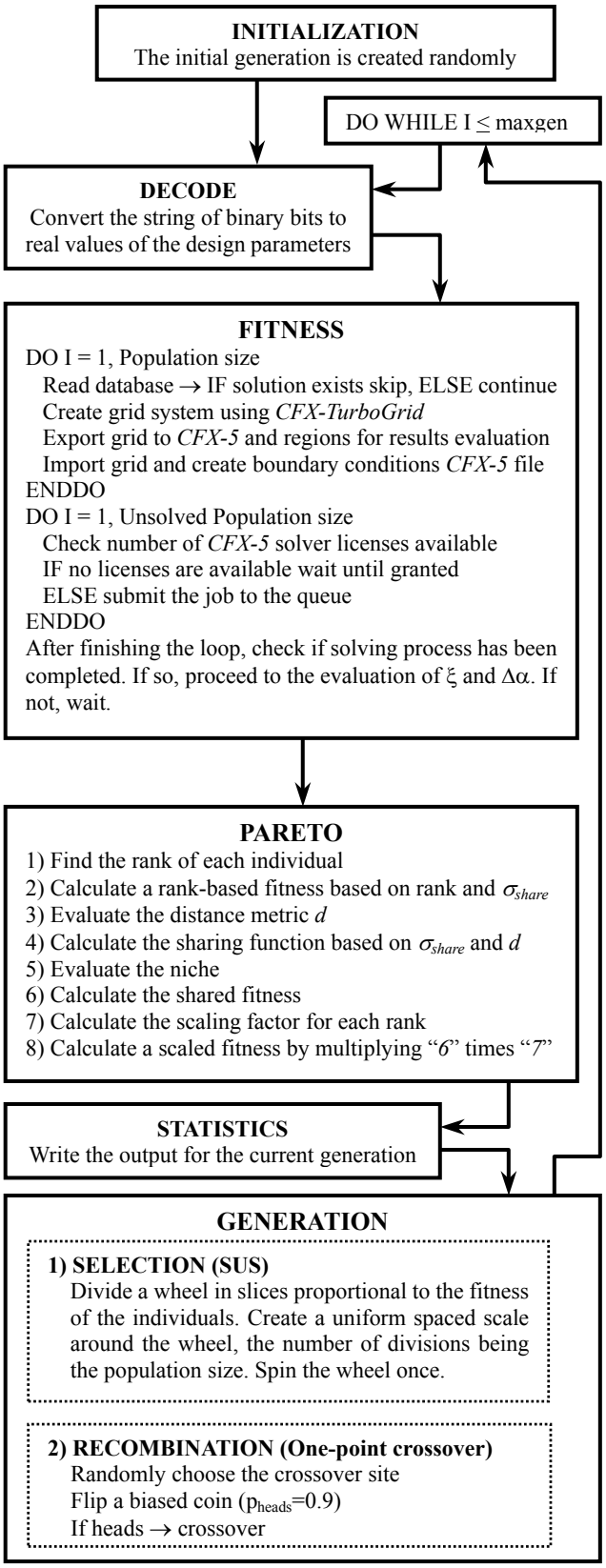

Figure 6 - Genetic algorithm optimizer flowchart.

NUMERICAL SIMULATION

Grid System

A UNIX shell script code was developed by the authors in order to automate the computational grid generation. For each selected chromosome (which have not been solved before), the code reads the decoded real values for the design parameters, generates a geometry file and executes the *CFX-TurboGrid* (*Ansys Inc.*) in batch mode. A *CFX-5.5.1* session file is then played in order to import the grid system, apply the boundary conditions, define solver parameters and export all data to a binary file that will later be used as input for the flow solver. Figure 7 shows a mesh for $\gamma=20.0^0$.

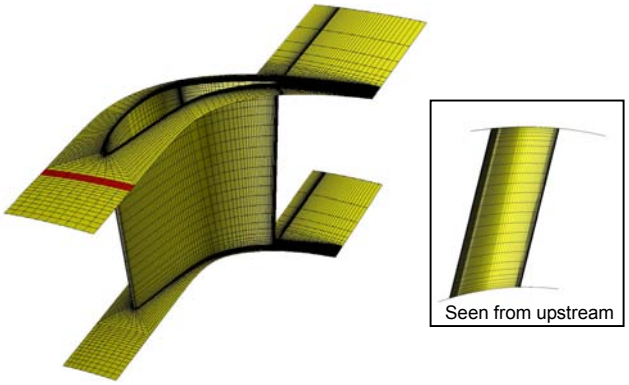

Figure 7 - Grid system.

The grid system shown in Fig.7 has the same pitchwise and streamwise node distribution as in the previous complex stage analyses performed by the authors (Funazaki et al., 2002a, 2002b, 2003a). The spanwise distribution, however, was reduced from *50* points to *26* in order to minimize the CPU time. The total number of grid points was 270,270.

Computational Code

The three-dimensional, incompressible, Reynolds-averaged, steady Navier-Stokes equations were solved with the finite volume *CFX-5.5.1* computational code. Considering the velocity-pressure coupling, the code uses a single cell, non-staggered, collocated grid to overcome the decoupling of pressure and/or velocity. In the equation of continuity, a second-order central difference is used to approximate the first-order derivative in velocity, modified by a forth-order derivative in pressure, which acts to redistribute the influence of the pressure. This overcomes the problem of checkerboard oscillations that are found when the variables are collocated (CFX-5 User's Manual). The method is similar to that used by Rhie and Chow (1982). A second order accurate scheme was selected for the discretization of the advection term. This scheme adds an extra term to the conventional upwind differencing scheme, known as *numerical advection correction*. The parameters for the extra term are calculated based on the report of Barth and Jespersen (1989). The flow solver uses the *algebraic multigrid* method (Raw, 1996) with *additive correction* (Hutchinson and Raithby, 1986), which forms a system of discrete equations for a coarse mesh by summing the fine mesh equations. This results in a virtual coarsening of the mesh spacing during the course of the iterations, and then re-refining the mesh to obtain a more accurate solution.

An automatic time step calculation based on the boundary conditions, initial guess and geometry of the domain is performed by the code every 6 iterations. This feature was found to be effective in improving the convergence rate. The numerical simulations were assumed to be converged when all residuals became smaller than 10^{-4} or after 100 iterations had been completed. In most cases convergence was obtained 10 or 20 iterations before reaching the iteration limit.

Boundary Conditions

The boundary conditions employed were based on the experimental data and on the previous calculations similar to the present analysis (Funazaki et al., 2002a). For the inlet region, total pressure, inlet turbulence intensity (*Tu=3.7%*) and the dissipation length scale ($L_\varepsilon=0.01$) were prescribed. For the outlet region, static pressure was prescribed. Since this quantity had not been measured, back-calculations were performed for the datum by changing the outlet static pressure values until the resulting outlet total pressure best matched the experimental data. The same value of the static pressure obtained for the datum was specified for the injection cases. The blade and casing wall regions were assumed as adiabatic and the non-slip condition was applied. For the pitchwise boundaries, periodic boundary conditions were applied. The working fluid used in the calculations was air at room temperature. Concerning the flow injection cases, the velocity was calculated as 10% of the main stream flow rate and prescribed according to the injection angle, maintaining the magnitude of the injection velocity vector constant. The Reynolds number based on the stator axial chord length and inlet velocity for all cases was 38,000.

Turbulence Model

The k-ε model with scalable wall function was chosen for its robustness and low computational time requirement. Unlike other conventional wall function models, the scalable one avoids inconsistencies in the wall function in case of excessively fine grids. The principle of this model is to simply neglect all grid points in the viscous sublayer, i.e., $y^+ \leq 11$. Therefore, consistency for arbitrary grid refinements is guaranteed regardless of the Reynolds number of the problem.

RESULTS

Pareto-Optimal Front

The *GA* code solved *90* generations in *12* days using the Origin 3800 machine at the Supercomputing Center of Iwate University. The CPU time for the present *GA* run was substantially reduced in comparison with the previous single objective *GA* code developed by the first two authors (Funazaki and Favaretto, 2003b). The reason for such improvement can be attributed to the automated parallel processing routine and the implementation of a database, which stores all unique design configuration fitness and recalls such values when similar case appears in subsequent generations. Another aspect for the less time consuming *GA* is due to a better compromise between CPU time for each run (4 to 5 hours) and numerical accuracy.

Figure 8 shows the population fitness for generations *1* (diamonds) and *90* (circles). The first generation was bred from a random population, thus presenting a scattered distribution in the objective space. The last generation completely lies on the Pareto-optimal front, with exception of five or six individuals. One of the reasons for such behavior could be justified by the selection process adopted in the algorithm. Even though the *SUS* tends to favor the best fit individuals there is no guarantee that a chromosome with lower performance will be completely excluded from the mating pool. This is because there is also some randomness in such selection process. In addition to that, the selected individuals have their fitness rescaled by the Pareto subroutine shown in Fig.5. The shared fitness computation procedure does not make sure that a solution in a poorer rank will always have a worse scaled fitness than every solution in a better rank. There are crowded solutions in some parts of the Pareto front which produce lower niche count values. On the other hand, some individuals with a lower rank and located far from clustered solution regions have a higher niche count. In the present code, the parameter which is directed related to the niche count (σ_{share}) was set as constant through all generations. This value may not be suitable for all generations and may also depend on the shape of the Pareto-optimal front (Deb, 2002). Recently, however, some authors have suggested a dynamic update of σ_{share}, which would account for

changes in the Pareto-optimal front. The validity of such approach has not yet been verified by the present authors, remaining as an important issue for future improvements of the described code.

Figure 8 also shows two Pareto-optimal fronts. The green line refers to the converged Pareto front obtained at the generation *90*. It should be noted that even if all chromosomes belonged to the Pareto front the maximum number of points describing the curve would be restricted to the population size. In order to obtain a more representative curve, all solutions in the database were ranked after generation *90*. The red curve in Fig.8 shows a more detailed Pareto-optimal front with 37 points widely spread. This global front shows similar trends as the local one, which could be an indication that convergence was indeed achieved at generation *90* and further runs would only change the relative location of the chromosomes on the local Pareto-optimal front.

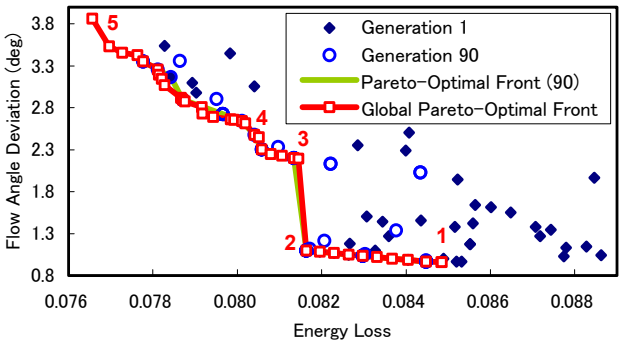

Figure 8 - Pareto-optimal fronts and population.

The numbers shown in Fig.8 indicate chromosomes located in key positions of the Pareto front and the respective design and fitness values are described in Table 2. The Pareto front can be described by a linear curve from chromosome *5* to *4*. This part of the curve shows the optimum configurations for a low loss model. The values indicate that the injection angle should be held constant ($\beta=120^0$) while a linear increase of the lean angle would increase the flow angle deviation and decrease the loss. In the region between chromosomes *4* and *3* a "plateau" can be found. No significant change in loss or flow angle deviation is detected for such region. Between chromosomes *3* and *2*, however, the results show a substantial reduction in the flow angle deviation with almost no cost in terms of loss. This "cliff " topology suggests that chromosome *2* is definitely more fit than chromosome *3* and the abrupt drop in $\Delta\alpha$ is caused mostly by the lean angle. Naturally, the unleaned blade will produce a lower flow angle deviation regardless of the injection angle. Following this trend, from chromosome *2* to *1* the flow angle deviation is hardly affected and the loss is practically dependent only on the injection angle.

Table 2 - Selected Optimal Solutions.

Chromosome	γ	β	ξ	Δα
1	0.0^0	102^0	0.0848	0.95^0
2	0.0^0	120^0	0.0816	1.09^0
3	10.9^0	110^0	0.0814	2.19^0
4	11.6^0	120^0	0.0804	2.47^0
5	20.0^0	120^0	0.0765	3.86^0

The results from Fig.8 and Table 2 provide a general view of several ideal configurations. The step further, which is selecting from the family of solutions the one that best suits a particular interest, cannot be performed by the *GA* code. Therefore, after obtaining the Pareto-optimal front the turbomachinery designer has to make the final judgment. For instance, if the objective of the optimization is more intended to minimizing the loss then a 20^0 leaned blade with 120^0 flow injection angle (chromosome *5*) should be adopted. If the goal is also to reduce the loss but without compromising the flow angle deviation, as it would be in a nozzle redesign task for a constrained turbine rotor geometry, the unleaned blade with $\beta=120^0$ would be the best choice.

Physical Interpretation of the Optimum Solutions

It must be stated that the conclusions drawn from the *GA* results are limited to the constraints established beforehand. Extrapolating the Pareto-optimal front to the direction of interest would not necessarily give the desired fitness. It is therefore important to also have a thorough comprehension of the flow events occurring in the proposed model before attempting to extend the range of applicability of the results.

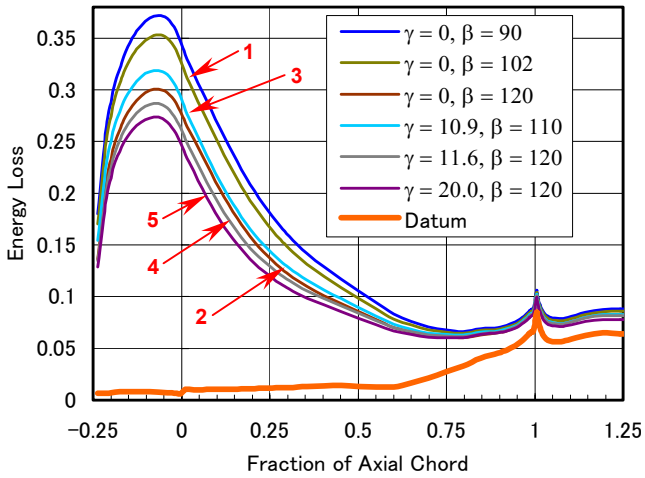

Figure 9 - Energy loss distribution along the axial direction.

Figure 9 presents the energy loss distribution along the axial direction calculated locally by using Eq.1. The injection slot is located at *–25%* c_x, the leading edge at *0%* c_x and the trailing edge at *100%* c_x. The numbers in the figure indicate the same chromosome numbers used in Fig.8 and Table 2. The energy loss for the datum is increased proportionally to the axial direction from leading to trailing edge, indicating the natural behavior of a flow through a nozzle vane. For such case the energy loss would be the summation of inlet boundary layer, profile and endwall losses. A substantial increase in loss is observed from approximately 60% axial chord. This phenomena can be attributed to the increased dissipation due to the suction surface velocity increase, which is related to the blade solidity, and the loss core formed in the surface suction endwall corner as the passage vortex lifts-off from the suction surface.

The curves for the injection cases presented a completely different behavior from -25% c_x to 79% c_x in comparison with the datum. These curves include not only the losses previously described but extra loss due to the mixing of the injection flow and the main stream. In the vicinity of the injection slot the loss increases substantially, reaching its maximum at –6.4% c_x for all analyzed cases. In this part of the domain a huge vortical structure is formed, creating a blockage for the upstream flow, which extends from the casing wall up to 30% of span fraction depending on the injection angle. The choice of a suitable injection angle was found to be the most important factor for the loss reduction since it is directly related to the size of the vortical region. A smaller tangential angle will necessarily reduce the penetration of the secondary injection flow and as a consequence reduce the high loss region, which is propagated to the blade passage (Funazaki et al., 2002b, 2003a). By observing the curves for chromosomes *2* and *5* one may find that blade lean also plays a role in reducing the loss, though with a lower intensity. Figure 9 also shows that both curves begin diverging from each other at 19% c_x upstream of the leading edge even though the injection angle is the same for both cases.

This suggests that the effects of blade geometry change (lean angle) propagate upstream of the leading edge, affecting the pattern of the vortical structure. In terms of the energy loss, the positively leaned blade produces a similar effect as the injection flow angle but with a smaller magnitude.

Figure 10 presents streakline plots coloured by fraction of span for the datum (a) and chromosome *5* (b) configurations. In Fig.10a it can be observed that the small spheres representing the streakline path do not present any variation in their color, i.e., there is no significant radial force acting on the flow. However, Fig.10b shows a noticable change in the spanwise location of the streakline, which departs from upstream at 29% span and passes through mid chord at 34% span. This result shows the existence of a radial inwards blade force acting on the flow, causing the streaklines to move contrary to the intuitive expectation. Denton and Xu (1999) deduced simple equations for qualitative understanding of the *streamline shift* (the difference between the two spanwise locations) and the radial force. Their analysis was performed for the no injection case and therefore cannot be applied for chromosome *5*. In the flow injection case, an additional term is added to radial force. Figure 10b shows a large recirculation region confined between the injection slot and the leading edge. The streaklines emitted from upstream, when encountering such region, are forced inward to the direction of the blade root. Apparently, flow injection seems to be beneficial in reducing the energy loss but this effect is counteracted by another loss core formed upstream of the leading edge. The energy loss is reduced in the axial direction from –6.4% c_x to 79% c_x probably due to such effect. From 79% c_x to downstream the secondary flow has been completely mixed with the main stream and the trend of the energy loss follows the datum situation. The shift in the curves for the injection case in comparison with the datum shows clearly the cost in terms of energy loss caused by the size of the vortical strutures upstream of the leading edge.

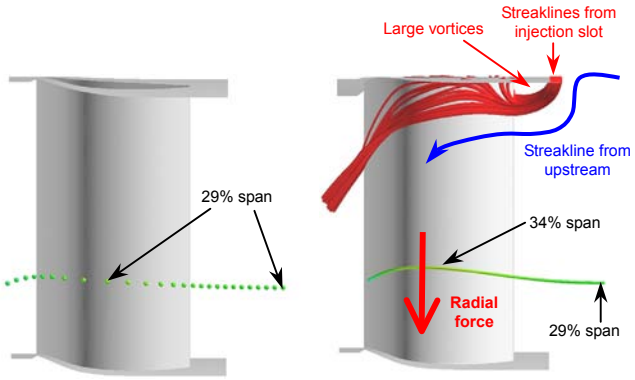

Figure 10 - Streakline plots for datum and chromosome 5.

The static pressure contours on the first two columns of Fig.11 show the effect of blade lean and flow injection on the blade loading. Taking into account solely the geometrical effects, different lean angles may be considered as moving the blade within an almost frozen pressure field. In the case of a positively leaned blade the root section is moved into a region of low velocity thus its loading is reduced whereas the tip section is moved into a region of high velocity and its loading increased (Denton and Xu, 1999). These trends were confirmed by the numerical simulations of Funazaki et al. (2002) for smaller lean angle. Figure 11b shows the contours for the unleaned case with flow injection. This figure shows no significant change in the static pressure distribution from root to approximately 70% span comparing to the datum case. At 25% c_x the secondary flow is still mixing with the main stream. The contours for Fig.11a at this axial plane seem to be scaled in Fig.11b from root to 70% span. From 70% span to root Fig.11b shows the complex flow field caused by the large vortices. At 75% c_x the flow is almost completely mixed and the flow pattern resembles the datum. However, the loading has been increased, as shown by the inclined isobars. The third column of Fig.11b presents the total pressure ratio contours, which were calculated by dividing the local total pressure by the weighted averaged inlet total pressure (Eq.2). As previously shown in the calculations of Funazaki et al. (2002b), for $\beta > 90^0$ the injection flow follows the direction of the passage vortex. The low energy fluid region formed due to the injection mixes with the boundary layer fluid and moves along the span away from the casing. The smaller the injection angle (relative to the casing) the lower the loss. Cases for $\beta < 90^0$ provide a higher loss because the flow injection stream is directed to the pressure (concave) side of the blade causing a relative diffusion, hence raising the static pressure.

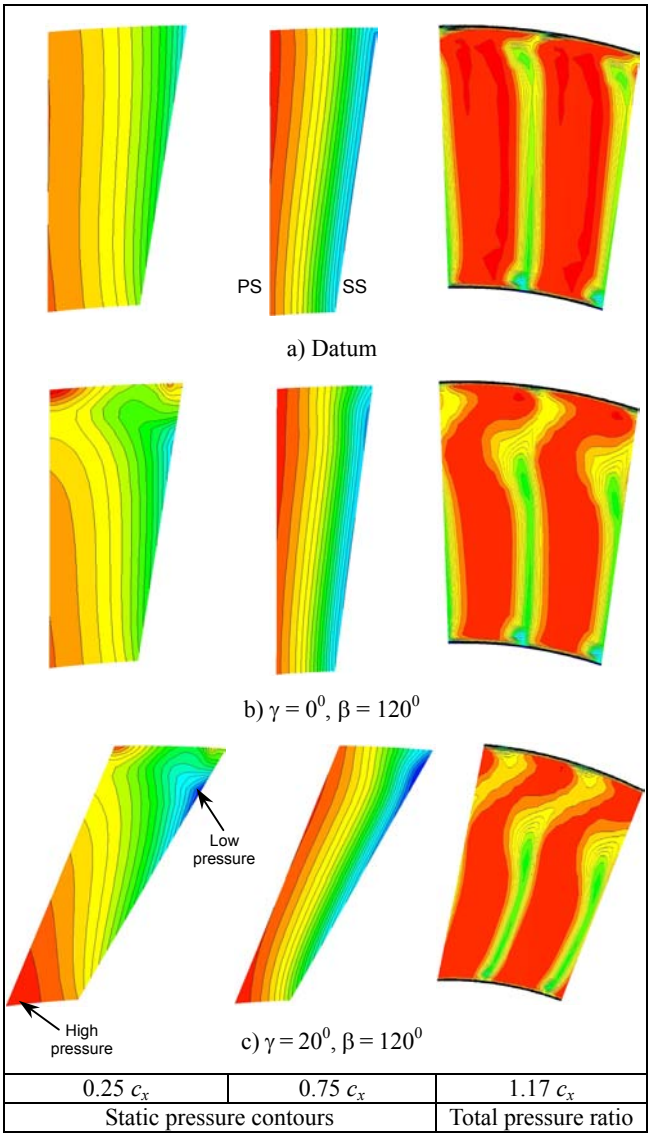

Figure 11 - Static pressure and total pressure ratio contours through the blade passage seen from upstream.

As previously shown in Fig.10b, blade lean can also be effective in reducing the energy loss for the flow injection cases. The static pressure contours show more clearly the increased pressure near the root, which is directed related to the radial blade force mechanism. The benefits of such trend are reflected on the reduced passage vortex near the root end wall, shown by the pressure ratio contours in Fig.11c. This effect is caused by the pressure gradient acting from the wall, which causes low energy fluid to move along the span towards the more highly loaded end wall on the tip.

The combination of γ and β will affect both the radial blade force and the size of the vortical structures upstream of the leading edge. It seems there is a trade-off between these two phenomena. A large recirculation region will increase the loss by extracting energy from the fluid coming from upstream. On the other hand, the streamline shift might be increased, which will also increase the radial blade force. Therefore, the combination of γ and β do not follow a linear pattern for all objective space. Regions such as the cliff between chromosomes *3* and *2* in the Pareto-optimal front of Fig.8 may occur when such non-linearity is encountered.

It must be outlined that the current optimization task is also aimed at reducing the outlet flow angle deviation $\Delta\alpha$. The results in Fig.8 and Table 2 suggest that chromosome *5* produces 10% lower loss than chromosome *2* with "only" 2.91^0 increase in the flow angle deviation. This increase in $\Delta\alpha$ does not seem to be relevant considering the substantial decrease in loss. A correct interpretation, however, requires a more detailed analysis of the flow angle distribution, i.e., pitchwise distribution. Figure 12 shows that $\Delta\alpha$ for chromosome *5* is not extremely high in average but the deviation from it can be as high as 6^0 near the root and 8^0 near the tip. If one attempts to extrapolate the results for chromosome *5* for reducing the loss the highly leaned blade could lead to extremely distorted profile near the end walls. This local change would not necessarily be reflected in the magnitude of the area average, thus requiring drastic modification of the downstream rotor blading.

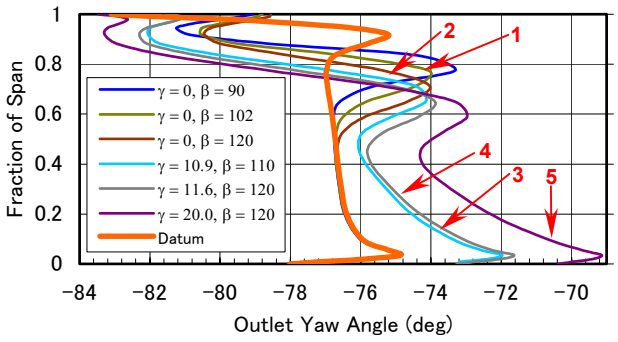

Figure 12 - Pitchwisely averaged outlet yaw angle.

CONCLUDING REMARKS

The present study described the development of a multi-objective *GA* code and its application to the optimization of secondary flow injection upstream a turbine nozzle vane. The genetic algorithms have proven to be a useful tool for tackling the challenging design tasks of modern gas turbines. The development of a hybrid *FORTRAN*/*Perl*/*UNIX* shell script program enabled the authors to combine the *GA* code with a commercial software.

The results revealed that low flow injection angles to the suction side of the nozzle combined with high positive lean angle are more suitable for minimizing the energy loss. Lower yaw angle values were obtained for no lean or small lean angle configurations, regardless of the flow injection angle.

The methodology adopted in this study may be applied to many other practical engineering problems. However, the authors acknowledge the necessity of improvements in the *GA* code. In particular, further investigation on non-dominated sorting and the implementation of an elitist model in the selection process.

ACKNOWLEDGEMENTS

The second author acknowledges the financial support of the Ministry of Education, Culture, Sports, Science and Technology of Japan.

REFERENCES

Akmandor, I.S., Öksüz, O., Gökaltun, S., Bilgin, M.H., 2002, "Genetic Optimization of Steam Injected Gas Turbine Power Plants", ASME Paper GT-2002-30416.

Barth, T.J., Jesperson, D.C., 1989, "The Design and Application of Upwind Schemes and Unstructured Meshes", AIAA Paper 89-0366.

CFX-5 User's Manual, AEA Technology.

Deb, K., 2002, "Multi-Objective Optimization using Evolutionary Algorithms", Wiley-Interscience series in systems and optimization.

Denton, J.D., Xu, L., 1999, "The Exploitation of 3D Flow in Turbomachinery Design", in *Turbomachinery Blade Design Systems*, Lecture series 1999-02, von Karman Institute for Fluid Dynamics.

Fonseca, C.M., Fleming, P.J., 1993, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion, and Generalization", *Proceedings*, *The Fifth International Conference on Genetic Algorithms*, pp. 416-423.

Funazaki, K., Favaretto, C.F.F., Tanuma, T., 2002a, "Effects of Flow Injection from Outer Casing upon Turbine Nozzle Vane Flow Field", ASME Paper GT-2002-30598.

Funazaki, K., Favaretto, C.F.F., Yamada, K., Kamata, M., 2002b, "Numerical Simulation of the Flow Field in a Turbine Stage with Upstream Tangential Flow Injection from the Outer Casing (Effects of the Injection Angle)", *Proceedings*, *Fifth JSME-KSME Fluids Engineering Conference*, Nagoya, Japan.

Funazaki, K., Favaretto, C.F.F., Kamata, M., Tanuma, T., 2003a, "Numerical Simulation on the Flow Field in a Turbine Stage with Upstream Flow Injection from the Outer Casing: Effects of the Injection Angle", *JSME International Journal*, Series B, Vol. 46, No. 1, pp.173-183.

Funazaki, K., Favaretto, C.F.F., 2003b, "Application of Genetic Algorithms to Design of Internal Cooling System for Turbine Nozzle Vanes", ASME Paper GT-2003-38408.

Goldberg, D.E., 1989, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, Reading, Massachusetts, USA.

Hirai, K., Kodama, H., Nozaki, O., Kikuchi, K., Tamura, A., Matsuo, Y., 1997, "Unsteady Three-Dimensional Analysis of Inlet Distortion in Turbomachinery", AIAA Paper 97-2735.

Hutchinson, B.R., Raithby, G.D., 1986, "A Multigrid Method Based on the Additive Correction Strategy", *Numerical Heat Transfer*, Vol. 9, pp. 511-537.

Kamata, M., Funazaki, K., Favaretto, C.F.F., Tanuma, T., 2002, "Studies on Aerodynamics of a Turbine Stage with Flow Injection from the Upstream Casing" (in Japanese), *Proceedings*, *Memorial Conference for 30th Anniversary of GTSJ*, Tokyo, Japan, pp.25-30.

Moser, W., 1989, "An Analysis of Outlet Distortion in LP Steam Turbines", ASME Paper 89-GT-237.

Nie, C., Xu, G., Cheng, X., Chen, J., 2002, "Micro Air Injection and Its Unsteady Response in a Low-speed Axial Compressor", ASME Paper GT-2002-30361.

Raw, M., 1996, "Robustness of Coupled Algebraic Multigrid for the Navier-Stokes Equations", *34th Aerospace and Sciences Meeting & Exhibit*, AIAA Paper 96-0297.

Rhie, C.M., Chow, W.L., 1982, "A Numerical Study of the Turbulent Flow Past an Isolated Airfoil with Trailing Edge Separation", AIAA Paper 82-0998.

Trigg., M.A., Tubby, G.R., Sheard, A.G., 1999, "Automatic Genetic Optimization Approach to Two-Dimensional Blade Profile Design for Steam Turbines", *ASME Journal of Turbomachinery*, Vol. 121, pp. 11-17.

Vicini, A., Quagliarella, D., 1997, "Inverse and Direct Airfoil Design Using a Multiobjective Genetic Algorithm", AIAA Paper, Vol. 35, No. 9, pp. 1499-1505.

Wright, J.A., Loosemore, H.A., Farmani, R., "Optimization of Building Thermal Design and Control by Multi-Criterion Genetic Algorithm", Energy and Buildings, Elsevier Science, Vol. 34, pp. 959-972.