AIAA JOURNAL
Vol. 46, No. 3, March 2008

Biobjective Design Optimization for Axial Compressors Using Tabu Search

Timoleon Kipouros,* Daniel M. Jaeggi,† William N. Dawes,‡ and Geoffrey T. Parks§
University of Cambridge, Cambridge, England CB2 1PZ, United Kingdom
A. Mark Savill¶
Cranfield University, Cranfield, England MK43 0AL, United Kingdom
and
P. John Clarkson**
University of Cambridge, Cambridge, England CB2 1PZ, United Kingdom

DOI: 10.2514/1.32794

At present, optimization is an enabling technology in innovation. Multi-objective and multidisciplinary optimization tools are essential in the design process for real-world applications. In turbomachinery design, these approaches give insight into the design space and identify the tradeoffs between the competing performance measures. This paper describes the application of a novel multi-objective variant of the tabu search algorithm to the aerodynamic design optimization of turbomachinery blades. The aim is to improve the performance of a specific stage and eventually of the whole engine. The integrated system developed for this purpose is described. It combines the optimizer with an existing geometry parameterization scheme and a well-established computational fluid dynamics package. Its performance is illustrated through a case study in which the flow characteristics most important to the overall performance of turbomachinery blades are optimized.

Nomenclature

B	=	blockage
C	=	blade tip clearance
C_{lim}	=	blade tip clearance limit
$\boldsymbol{F}$	=	vector of objective functions
f	=	objective function
i_{local}	=	local iteration counter (tabu search parameter)
im_{size}	=	intensification-memory size counter (tabu search parameter)
$\text{im}_{\text{thresh}}$	=	intensification-memory size threshold (tabu search parameter)
$\dot{m}$	=	mass flow rate
n	=	number of design variables
n_{regions}	=	number of regions used for each variable in long-term memory (tabu search parameter)
n_{sample}	=	number of points sampled in the local search (tabu search parameter)
n_{stm}	=	short-term memory size (tabu search parameter)
R_{LE}	=	blade leading-edge minimum radius
SS	=	initial step size as a percentage of the variable range (tabu search parameter)
SSRF	=	factor by which step sizes are reduced on restarting (tabu search parameter)
ssr	=	step-size-reduction counter used in the restart strategy (tabu search parameter)
$\text{ssr}_{\text{thresh}}$	=	step-size-reduction threshold used in the restart strategy (tabu search parameter)
$\boldsymbol{x}$	=	vector of design variables
$\Delta\theta$	=	mass-averaged flow turning

Subscript

0	=	datum blade parameter

Presented as Paper 4023 at the 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Tucson, AZ, 10–13 July 2005; received 13 June 2007; revision received 9 October 2007; accepted for publication 15 October 2007.

*Research Student, Engineering Design Centre, Department of Engineering, Trumpington Street. Member AIAA.
†Research Student, Engineering Design Centre, Department of Engineering, Trumpington Street.
‡Professor, Computational Fluid Dynamics Laboratory, Department of Engineering, Trumpington Street. Senior Member AIAA.
§Senior Lecturer, Engineering Design Centre, Department of Engineering, Trumpington Street.
¶Professor, Department of Aerospace Sciences. Senior Member AIAA.
**Professor, Engineering Design Centre, Department of Engineering, Trumpington Street.

I. Introduction

THE optimization of airfoil designs is a challenging, computationally expensive, highly constrained, nonlinear problem. As with most real-world problems, there are multiple (usually conflicting) performance metrics that an engineer might seek to improve in optimizing, for example, the design of turbomachinery blades, wings, or other aerodynamic surfaces. This suggests a multi-objective approach, a notion that is reinforced by the recognition that any consideration of robustness [the retention of performance over a range of operating conditions or in the face of geometry variation (e.g., through creep or wear)] must also inevitably entail multiple objectives [1]. The consideration of robustness is essential if a design is truly to be optimized for real-world performance.

Despite this obvious motivation, multi-objective aerodynamic optimization has, until recently, been restricted largely to variously weighted single-objective schemes. However, several true multi-objective optimization studies for two or more objectives simultaneously have now shown the benefits of this approach compared with single-objective optimization using a composite-objective function.

The multi-objective integrated design system (MOBOS3D) used in the present work has been developed and described by Kipouros et al. [2], building on the single-objective integrated design optimization system (BOS3D) developed by Harvey [3] and described by Dawes et al. [4]. The system combines an existing efficient and flexible geometry parameterization scheme, a

well-established computational fluid dynamics (CFD) package, and a novel multi-objective variant of the tabu search (TS) optimization algorithm for continuous problems [5,6]. The system can readily be run on parallel computers, substantially reducing wall-clock run times: a significant benefit when tackling computationally demanding design problems.

In previous work [2], the performance of this system has been investigated considering a compressor blade design test case. The effectiveness of the multi-objective optimization procedure was verified and the expected tradeoffs between the chosen objectives were confirmed. In the work presented in this paper, we use our system to tackle a more realistic turbomachinery design test case, taking advantage of the greater computational power offered by exploiting its parallel processing capabilities, and we test enhancements to the optimizer that improve the search efficiency.

II. Background

A. Optimization in Aerodynamics

Optimization in aerodynamics is not new: an early study was conducted by Hicks and Henne [7], mainly to assess the feasibility of performing computerized wing design by numerical optimization. Burgreen et al. [8] proposed an improvement to the parameterization stage of the design process. They suggested replacing a grid-point-based approach with a Bezier–Bernstein polynomial parameterization of the surface and achieve a 75% decrease in total CPU time, as well as a significant reduction in computational memory requirements.

Bowen and Zhide [9] performed single-objective optimization of a transonic airfoil for multiple operating conditions using a combination of full potential and boundary-layer flow analysis in their CFD solver. On the other hand, Topliss et al. [10] proposed a surrogate-model-based integrated design system and achieve a substantial reduction in CPU time at the cost of a lower quality of objective-function approximation. Similarly, Patnaik et al. [11] concluded that surrogate models, such as neural networks and linear regression approximations, can increase the applicability of optimization to real-world design tasks. However, they note some difficulties associated with the training of the approximating analyzers in terms of time and accuracy. Low-order models cannot accurately predict a variety of nonlinear phenomena, such as wave drag, which can play an important role in the search for the optimal design.

De La Garza and Darmofal [12] presented a multidisciplinary approach for the aeroelastic design optimization of a subsonic wing, and the difficulties of such systems were revealed. In contrast, Pierret [13] successfully demonstrated a fully automated design system for 3D blade shapes using artificial neural networks (ANNs), detailed flow analysis, and a genetic algorithm (GA) for the optimization of a single global performance objective function. He concluded that these design process components allow more design options to be explored in a given period of time than traditional methods.

Anders et al. [14] highlighted the importance of the integration of flexible parametric representation schemes into the design system [15] and explained how the parameterization techniques facilitate optimization in turbomachinery blade design systems; their use is essential to maintain high design-process efficiency yet still allow innovation.

Reuther et al. [16] presented a new framework for the coupled optimization of aerostructural systems using high-fidelity modeling for both the aerodynamic and structural analysis. Their results illustrate the possible advantages of this design methodology in evaluating the tradeoffs between aerodynamic performance and structural weight for complete aircraft configurations. Moreover, Chamis [17] described an optimization environment for a multidisciplinary (thermal, structural, and acoustic) engine design problem. His results demonstrate that superior designs can be achieved through multidisciplinary computational simulations with multi-objective optimization algorithms.

Gaiddon et al. [1] performed multi-objective optimization on a supersonic missile inlet. They compared a number of optimization algorithms using both composite and multiple objective functions and concluded that "performing real multi-objective optimization and finding a Pareto front is the only effective way to find a set of designs satisfying several performance criteria in an industrial context."

Nemec et al. [18] performed multi-objective optimization on both a single- and a multi-element 2D aerofoil. Their integrated approach combined a Newton–Krylov adjoint CFD code, a B-splines-based parameterization scheme, and both a gradient-based optimizer and a GA. They obtained good results on some simple test problems.

Chiba et al. [19] optimized the wing shape of a transonic regional jet aircraft from a multi-objective, multidisciplinary perspective. This is a large-scale, real-world application with aerodynamic, structural, and aeroelastic objectives using high-fidelity evaluation models. The system used exploits parallel processing and the adaptive-range multi-objective genetic algorithm (ARMOGA) developed and described by Sasaki et al. [20]. The optimizer is designed for aerodynamic and multidisciplinary design optimization problems using computationally expensive high-fidelity CFD solvers.

A different approach to enhancing the efficiency of automated design optimization systems is by improving the parameterization stage of the process: the geometry manager model. Samareh [21] suggests that these models should have the following set of characteristics: be automated, provide consistent geometry across all engineering disciplines, be parametric, provide sensitivity derivatives, and use relatively few control variables yet provide high geometrical flexibility, to be compatible with product design cycle timescales.

Demeulenaere et al. [22] studied turbomachinery blade shape optimization, supporting the view that this has the potential to improve further the performance of rotating machines. Their proposed approach incorporated an ANN during the optimization phase, facilitating the efficient use of a GA. The system's performance was demonstrated through two multipoint design problems: seeking to increase in the efficiencies of an axial compressor and an industrial pump over a wide range of operating conditions.

Harvey et al. [23] deployed a partial differential equation (PDE) parameterization technique [24] to compress the CAD file of an axial compressor blade into a design vector of 26 dimensions. They sought to improve the performance of a compressor stage through a single-objective optimization test case, thereby demonstrating the great flexibility of the PDE approach. The results were experimentally validated, and significant improvement in the stage efficiency was observed [25], corroborating the high performance of this PDE technique recorded by Samareh [26] in a high-fidelity shape-optimization real-world environment.

Another approach to improving optimization system performance is to integrate an optimizer well suited to the nature of the problem and able to adapt its behavior as the optimization landscape changes. Sugimura et al. [27] presented an autonomous design optimization system for aerodynamic applications, distinguished by its use of a hybrid simulated annealing (SA) optimization algorithm and an ANN for global and local searches, respectively.

Battiti [28] supported the idea that reactive heuristic search has the flexibility needed to cover a wide spectrum of problems by integrating a simple history-based feedback scheme for the online tuning of the optimization parameters. Through metaheuristic methods that exploit information obtained earlier in the run or by hybridization with global optimization algorithms, local search can thus be guided to go beyond local minima [29].

B. Multi-Objective Optimization

The goal of single-objective optimization is to find a vector $\boldsymbol{x}$ such that an objective function $f(\boldsymbol{x})$ is minimized. The components of $\boldsymbol{x}$ are known as design variables. A point $\boldsymbol{x}_1$ in design space can be considered "better" than another point $\boldsymbol{x}_2$ if $f(\boldsymbol{x}_1) < f(\boldsymbol{x}_2)$. Thus, a single-objective optimization problem has only one optimal solution, or perhaps multiple optima with equal objective-function values.

We are interested in problems in which multiple objective functions are each to be minimized. When true multi-objective design optimization is performed, the solution is a set of designs representing the optimal set of tradeoffs between the objectives. This set is known as the *Pareto-optimal set*: solutions in the set are said to be *Pareto-equivalent* to each other and nondominated by any other solution. If $\boldsymbol{F}_n$ is the vector of objective-function values for design point $\boldsymbol{x}_n$, $\boldsymbol{x}_n$ is said to be Pareto-equivalent to another point $\boldsymbol{x}_m$ if some components of $\boldsymbol{F}_n$ are larger than the corresponding components of $\boldsymbol{F}_m$, whereas other components of $\boldsymbol{F}_n$ are smaller than the corresponding components of $\boldsymbol{F}_m$. Similarly, $\boldsymbol{x}_n$ *dominates* $\boldsymbol{x}_m$ if at least one component of $\boldsymbol{F}_n$ is strictly less than that of $\boldsymbol{F}_m$ and the remaining components of $\boldsymbol{F}_n$ are less than or equal to the corresponding components of $\boldsymbol{F}_m$. The goal in multi-objective design optimization is to identify (a close approximation to) the Pareto-optimal set, providing the designer with a clear picture of the achievable tradeoffs between the competing objectives, and thus to make a well-informed choice of the final solution.

There are three possible approaches to solving a multi-objective optimization problem. The first reduces the multiple objectives to a single objective by generating a composite-objective function, usually from a weighted sum of the objectives. This composite-objective function can then be optimized using existing single-objective optimizers. However, the weights must be preset, and the solution to this problem will be a single vector of design variables, rather than the entire Pareto-optimal set. This can have undesirable consequences: setting the weights implicitly introduces the designer's preconceptions about the relative tradeoff between objectives. The second is to cast the problem in a form amenable to solution by a goal programming approach [30]. Again, this requires the designer to annunciate preferences or define weightings in advance and thus has similar disadvantages to the composite-objective approach. The third approach is to search directly for the entire Pareto-optimal set. This can be achieved in a number of ways and requires modification to existing single-objective algorithms.

C. Existing Multi-Objective Tabu Search Algorithms

Our choice of TS as the optimizer is informed by the work of Harvey [3], who tested a number of metaheuristic methods on a representative single-objective aerodynamic design optimization problem and found TS to be superior to the GA and SA methods. The highly constrained nature of many aerodynamic design optimization problems renders much of their search space infeasible. The local search paradigm at the heart of TS enables it to navigate such complex constrained search spaces more effectively than methods (such as GAs) that tend to make larger changes to the design vector and generate many more infeasible solutions in consequence. This observation is supported by the work of Molinari et al. [31], who identified the intrinsically local nature of the design space of compressor blades and characterized it as "too irregular" to be modeled via a global regression model, because the feasible regions are too disconnected. There are many other stochastic local search methods that could potentially also be used [32]. However, the computational expense of performing a single optimization run for the application of interest makes a comparison of the performance of different optimizers impractical here, and a continuation from Harvey's [3] work is then a natural progression.

The authors know of only two other attempts to develop a local-search-based multi-objective TS algorithm, although others (for instance, Molina et al. [33]) have combined TS concepts with other search methods. Hansen's algorithm [34] is an extension of the composite-objective approach: it performs several composite-objective Tabu searches in parallel. Each search has a different, dynamically updated, set of weights, and in this way, the optimizer can be driven to explore the entire Pareto front. Although a good implementation of TS, this algorithm has the disadvantages common to all weighted-sum approaches: for problems with concave Pareto fronts, there may be regions of the front that are not defined by a combination of weights, and conversely, certain combinations of weights represent multiple points on the front.

Baykasoglu et al. [35] developed a TS algorithm combining a downhill local search with an *intensification memory* (IM) to store nondominated points that were not selected in the search. When the search fails to find a downhill move, a point from the IM is selected instead. When the IM is empty and all search paths exhausted, the algorithm stops. The facts that the search is restricted to downhill moves and that there is no diversification strategy mean that this algorithm is, in reality, an elaborate local search algorithm, rather than a true TS algorithm.

The other TS algorithms mentioned by Jones et al. [36] in their comprehensive review of the state-of-the-art in multi-objective metaheuristics either use a composite-objective function or are little more than local search algorithms similar to the algorithm of Baykasoglu et al. [35].

Jaeggi et al. [5] developed the original version of the multi-objective TS variant presented in this paper and executed a performance comparison on a set of unconstrained test functions. Its constraint-handling approach and the performance of the algorithm on benchmark-constrained optimization problems are presented in [6].

III. Multi-Objective Tabu Search Adaptation

A. Algorithm Overview

The single-objective TS implementation of Connor and Tilley [37] is used as a starting point for our multi-objective variants. This uses a Hooke and Jeeves (H&J) [38] local search algorithm (designed for continuous optimization problems), one of the best-known direct search methods [39], coupled with short-, medium-, and long-term memories to implement search intensification and diversification, as prescribed by Glover and Laguna [40]. These TS features are essential additions to overcome the difficulties encountered when the underlying direct search becomes trapped at a local minimum.

1. *Local Search Iteration*

TS operates in a sequential iterative manner: the search starts at a given point and the algorithm selects a new point in the search space to be the next current point. The basic search pattern in our implementation is a modified version of the H&J search. Allowing for points that violate constraints or are tabu (discussed next), the H&J local search strategy requires approximately $2n_{\text{var}}$ solution evaluations, where n_{var} is the number of design variables. A real-world problem may contain a large number of design variables, and this strategy can become prohibitively expensive. One solution is to incorporate an element of random sampling in the H&J step. We generate the $2n_{\text{var}}$ new points, remove those that are tabu or infeasible, and only evaluate $n_{\text{sample}} \leq 2n_{\text{var}}$ points from those that remain, selecting randomly to avoid introducing any directional bias. If one of these points dominates the current point, it is automatically accepted as the next point. If two or more points dominate the current point, a nondominated point from among those is randomly selected. If no points dominate the current point, a further n_{sample} points are sampled and the comparison is repeated. If all the feasible, nontabu points were sampled without finding a point that dominates the current solution, a nondominated point from among all those sampled is chosen randomly to be the next point.

In addition, a *pattern move* strategy is implemented in the same way as by Connor and Tilley [37]. Before every second H&J move, the previous move is repeated. This new point is compared with the current point, and, if it dominates it, is accepted as the next point; if not, the standard H&J move is made. In this way, the search may be accelerated along known downhill directions.

2. *Memories and Search Control*

Recently visited points are stored in the *short-term memory* (STM) on a first-in/first-out basis and are *tabu*: the search is not allowed to revisit these points. Although in many TS implementations only objective information is stored in the STM, in our implementation, complete solutions (design vectors and their associated objectives) are stored, and solutions are compared on the basis of their design

vectors. This is because, for our application, the computational cost of evaluating a solution, which is avoided if the solution is known to be tabu, far exceeds the computational cost of searching the STM on a design vector basis.

The *medium-term memory* (MTM) maintains an unbounded record of the Pareto-optimal points found thus far during search, and its contents are the primary output at the end of the optimization.

In a similar strategy to that used by Baykasoglu et al. [35], an IM is employed. This is a set of Pareto-equivalent points; at each H&J step, points that dominate the current solution, but are not selected as the next point, are considered as candidates for addition to the set. At *search intensification*, a point is chosen randomly from the IM. The IM is continuously updated, and points that become dominated by the addition of a new point are removed. Thus, the IM should always contain unexplored points that are on the current Pareto-optimal front (stored in the MTM).

The *long-term memory* records the regions of the search space that were explored and is used in *search diversification*, directing the search to regions that are underexplored. This is achieved by dividing the allowed range for each design variable into a certain number of regions and tallying the number of solutions evaluated in those regions. In diversification, a new starting point for search is generated by sampling randomly within the least visited regions for each design variable. If the resulting design proves to be geometrically or aerodynamically infeasible, then another design vector is constructed in a similar fashion.

A local iteration counter i_{local} is used and reset upon a successful addition to the MTM. When i_{local} reaches user-specified values, the algorithm will intensify or diversify the search or reduce the search step size and restart the search from a randomly selected point from the MTM. Thus, TS combines a systematic local search with a stochastic element and an intelligent coverage of the entire search space. Figure 1 presents a flow diagram for our multi-objective TS algorithm, and Fig. 2 illustrates the local search pattern and the relationship between the various memories.

3. *Constraint Handling*

As can be seen in Sec. IV.A, for the application under consideration, some constraints are handled using penalty functions, whereas others are treated as hard constraints: any point that violates a hard constraint is deemed to be tabu and the search is not allowed to

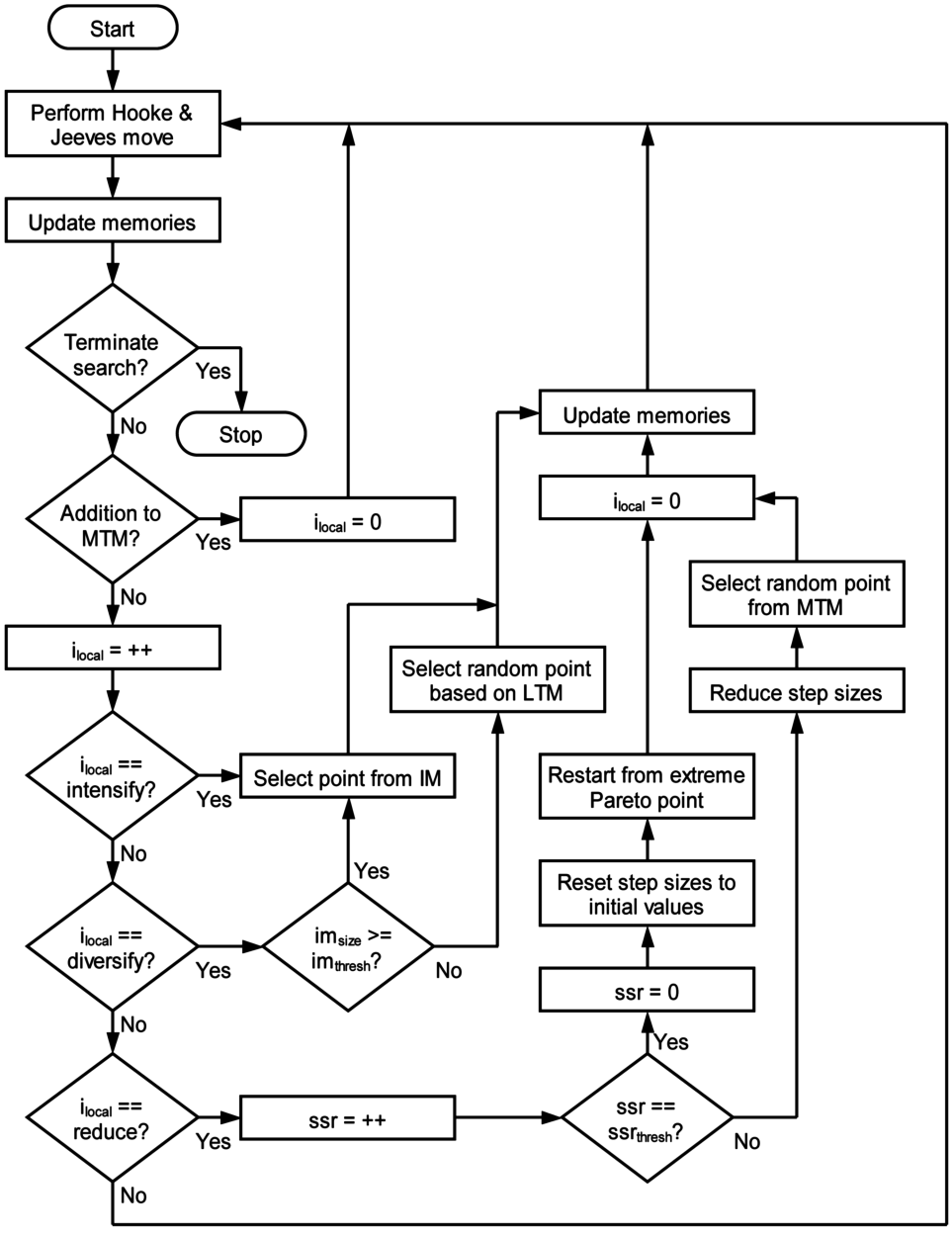

Fig. 1 Flow diagram of the multi-objective TS algorithm.

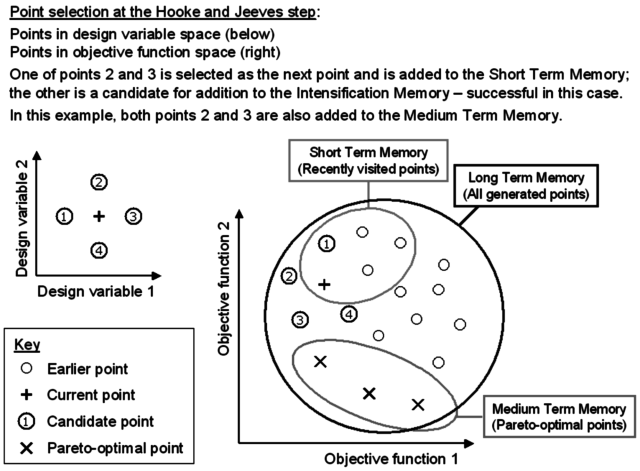

Fig. 2 Point selection for the Hooke and Jeeves move and TS memories.

visit that point. On search diversification, the algorithm loops until a point satisfying all the hard constraints is generated. The constraint-handling approach is discussed in more detail in [6].

4. Parallelization Strategy

Any optimization procedure that forms part of a real-world design cycle must be able to complete in a reasonable time frame. Parallel processing offers a large potential speedup; an optimization algorithm should ideally be designed with this in mind. Our algorithm is parallelized by means of functional decomposition: at each H&J move, the required objective function evaluations are computed in parallel.

B. Algorithm Enhancements

1. Application-Specific Intensification/Diversification Strategy

A new intensification/diversification strategy was developed and adopted in the present work. For the design optimization of gas-turbine compressor blades, intensification was found to be particularly beneficial to overall optimizer performance; the optimizer was able to make steady progress (indicated by the rate of addition to the MTM: the number of designs lying on the Pareto front) by performing frequent intensification steps. This suggests that this class of problem contains regions with many locally Pareto-optimal points, and TS is able to search these effectively through its intensification strategy. To exploit this characteristic and enhance the performance of the optimizer, the original diversification strategy has therefore been changed as follows: at the diversification stage of the optimization process, if the size of the IM is above a user-specified threshold (i.e., there are several candidate designs still to be explored through intensification), then intensification takes place instead; if the size of the IM is below this threshold, diversification takes place as before.

2. Application-Specific Restart Strategy

A further improvement made to the optimization algorithm relates to the restart strategy. Given the highly constrained nature of the aerodynamic design problems in which we are interested and the number of locally Pareto-optimal regions in the search space, it is quite likely that the optimizer will become trapped in such a region if the search step sizes become too small. To overcome this potential problem, a new restart strategy was introduced. If the search step sizes were reduced more than a user-specified number of times ssr_{thresh}, then on restart, the step sizes are reset to their initial values and an extreme Pareto point (one for which one of the objectives is the minimum found thus far) is randomly selected as the next base point. To avoid the repeated selection of the same extreme Pareto point, we introduce an additional new type of STM in which previously selected restart locations are stored. These locations are characterized as tabu, because they cannot be visited again.

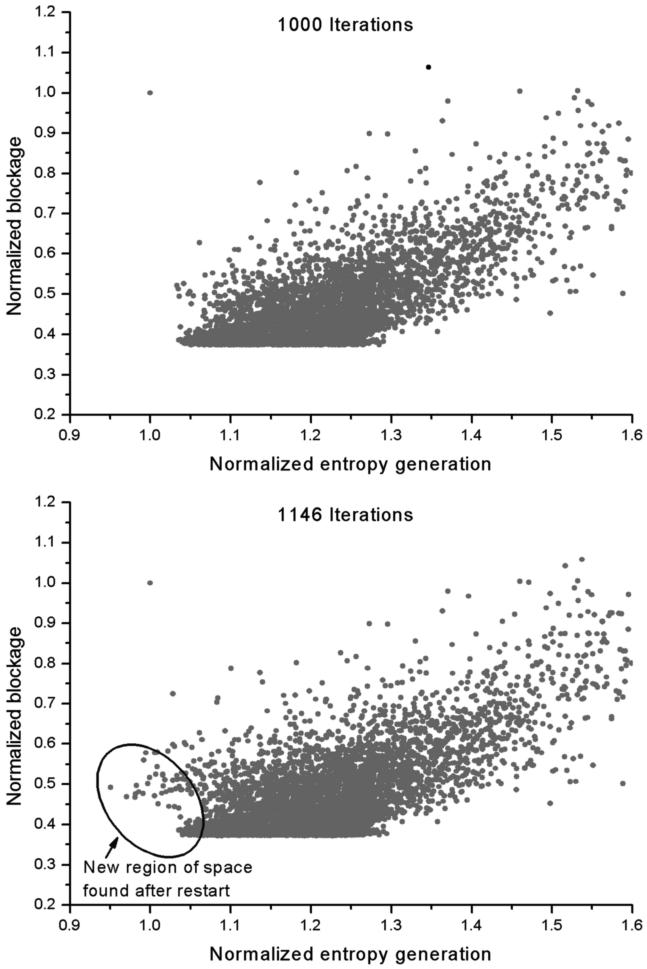

Fig. 3 Illustration of the effectiveness of the new restart strategy.

This new restart strategy has proved to be effective in directing the search to unexploited areas of the design space and locating new optimal designs, as illustrated in Fig. 3. This shows all the locations visited in the first 1000 iterations (upper figure) of an optimization on a two-objective problem in which blockage and entropy generation are to be minimized (see Sec. IV.A). It is apparent from the very high density of solutions with low values of the *y*-axis objective (normalized blockage) that the search has become trapped. The search is then restarted from the solution with the lowest value of the *x*-axis objective (normalized entropy generation) using reset step sizes, and by 1146 iterations (lower figure), a number of new Pareto-optimal designs with significantly lower values of normalized entropy generation were located in a previously unexplored part of the search space.

The effectiveness of this mechanism can be further extended with the implementation and integration of additional augmentations. The Epanechnikov function [41] can be used as a statistical quantity, measuring the density of visited designs in the extreme areas, or even the compromise area, of the Pareto front. Hence, when the restart strategy is executed, the search can be moved to the least investigated area, according to the ranking of these densities. At the same time, the step sizes of the design variables can be tuned according to the ratio of the density of visited designs in the area to which the search was moved, to the density of all feasible designs examined thus far. Large step sizes should be assigned if this ratio is small, and small step sizes should be assigned if this ratio is large. This measure can be easily extracted using the Epanechnikov function.

It would also, of course, be straightforward to implement a mechanism by which restarts were directed to other known good (existing) designs, if more were available than the one used to initiate the search.

IV. Aerodynamic Shape Optimization

A. Problem Description

The most important flow parameters that affect the performance of a turbomachinery blade are 1) flow separation (blockage), 2) losses (any flow feature that reduces the efficiency of the turbomachine), and 3) deviation in flow turning.

Here, we seek to find, starting from a gas-turbine compressor guide vane (stator) specification, a blade geometry that efficiently gives a good pressure rise at a particular flow coefficient. Thus, a global performance measure of a given blade geometry is needed for the optimization process. Efficiency is only one of a number of possible design objectives when undertaking detailed aerodynamic shape optimization. A good design must also respect mechanical and manufacturing constraints while achieving the required aerodynamic performance (with respect to flow turning, separation, and good off-design performance, etc.).

The design optimization of compressor blade geometries has previously been studied by Harvey [3] from a single-objective perspective. In this multi-objective test case, we retain Harvey's objective function, span-averaged blockage for a given mass flow rate, as an essential 1D (throughflow) measure of blade performance, based on the nonuniformity of the flow in the trailing-edge (TE) plane. This is a normalized function, including penalty-function terms for specific flow characteristic and geometry constraints:

$$f_1 = \frac{B}{B_0} + 250\left(1 - \frac{\dot{m}}{\dot{m}_0}\right)^2 + 0.4\max^2\left(0,\ 1 - \frac{R_{\mathrm{LE}}}{R_{\mathrm{LE},0}}\right) + 500\max^2\left(0, 1 - \frac{\Delta\theta}{\Delta\theta_0}\right) + 0.5\max^2\left(0, 1 - \frac{C}{C_{\mathrm{lim}}}\right) \tag{1}$$

In Eq. (1), B represents the blockage, the extent to which viscous forces restrict the effective flow area in a blade passage, which is probably the most critical quantity in high-speed compressor design. (In highly loaded compressors, the flow tends to separate from the blade under conditions of low mass flow. Flow separation acts as a blockage in the flow path, which limits pressure recovery.) Then $\dot{m}$ is the mass flow rate, R_{LE} is the minimum radius of the leading edge of the blade, $\Delta\theta$ is the mass-averaged flow turning, C measures the tip clearance of the blade, and C_{lim} is the defined lower limit on tip clearance. The zero subscripts identify the equivalent quantities for the datum blade geometry, the initial design in the optimization: a real compressor blade design, shown in Fig. 4.

The mass flow associated with the design should be equality constrained (i.e., very close to its initial value) for two reasons. First, if it was not, the inlet dynamic head from the rotor would vary, which is not modeled by the boundary conditions. Second, if the axial velocity drops, then the inlet static pressure must be higher (because inlet pressure and flow angles are prescribed), so that the static pressure rise across the stator will be lower (outlet pressure is fixed), and the blade row will not be an effective diffuser. Equally, it is important that if the mass flow is fixed, the flow turning in the stage should not be reduced during optimization, otherwise the static pressure recovery will not be sufficient. Therefore, control of the flow turning is achieved by treating it as a penalty term as well.

In addition, there are two terms in Eq. (1) describing the geometrical constraints on the blade. The first limits the sharpness of the blade's leading edge, whereas the second allows a weighted penalty factor to tradeoff aerodynamic performance against mechanical proximity. The objective-function value is penalized when the blade design has less than C_{lim} (1.5 cm) clearance. Both these penalty terms reflect a concern for robust aerodynamic performance from the design, because these geometric characteristics are closely related to the off-design performance of the blade.

Harvey [3] found that it was necessary to use a penalty-function approach with these constraints to successfully navigate the highly constrained, nonlinear search-space characteristic of aerodynamic design optimization problems. He established suitable values for the weightings for each of the penalty terms through extensive testing. Other constraints, such as those on the geometric feasibility of blade designs and on their operational feasibility (a design that produces unsteady flow patterns is not acceptable), are handled as hard constraints: designs violating them are not accepted.

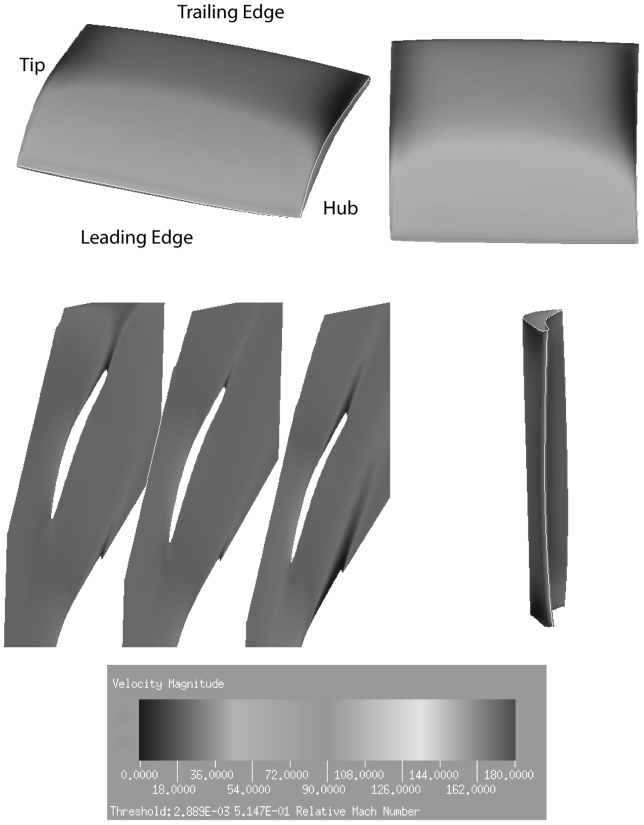

Fig. 4 Datum blade geometry showing its velocity magnitude distribution; top left: 3D view; top right: suction side normal view; bottom left: 2D profiles in hub-midspan-tip regions; and bottom right: downstream side view.

In our multi-objective case study, we introduce a second objective function that quantifies losses associated with the design. This is defined in terms of the rate of entropy generation $\dot{S}$ [42], subject again to the same set of penalty-function constraints:

$$f_2 = \frac{\dot{S}}{\dot{S}_0} + 250\left(1 - \frac{\dot{m}}{\dot{m}_0}\right)^2 + 0.4\max^2\left(0, 1 - \frac{R_{\mathrm{LE}}}{R_{\mathrm{LE},0}}\right) + 500\max^2\left(0, 1 - \frac{\Delta\theta}{\Delta\theta_0}\right) + 0.5\max^2\left(0, 1 - \frac{C}{C_{\mathrm{lim}}}\right) \tag{2}$$

In a turbomachine, the isentropic efficiency is defined as the ratio of the actual work to the isentropic work for a work-producing device (such as a turbine) and the ratio of the isentropic work to the actual work for a work-absorbing device (such as a compressor). The only factors that change this efficiency are departures from isentropic flow. These may arise due either to heat transfer or to thermodynamic irreversibility. For most turbomachines, the flow is close to adiabatic (no heat transfer) and so only entropy creation by irreversibilities contributes significantly to the loss of efficiency, which means that the entropy generation rate is the only rational measure of overall loss in an adiabatic machine. Any irreversible flow process creates entropy and thus reduces the isentropic efficiency. The sources of loss in a turbomachine can be categorized as profile loss, secondary (or endwall) loss, and tip-leakage loss. In many machines, the three are comparable in magnitude, each accounting for about one-third of the total loss.

One common approach to the treatment of constraints in multi-objective optimization is to construct additional objectives to be minimized that quantify the extent of constraint violation. A penalty-function-based approach has been preferred here because of the difficulty observed in satisfying these constraints [31] and the computational expense in evaluating candidate designs. Increasing

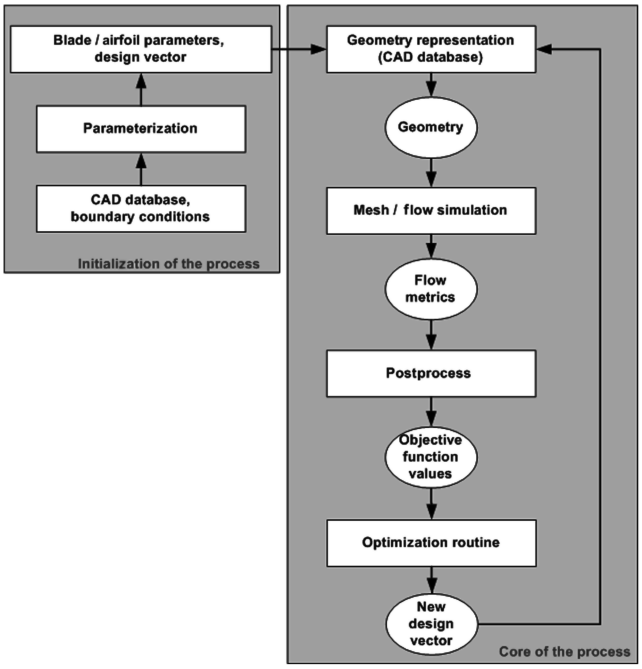

Fig. 5 Structure of MOBOS3D.

the number of objectives increases the dimensionality of the tradeoff surface to be explored, thus necessitating the generation and evaluation of more designs at considerable additional computational cost. Experience also shows that allowing the search to explore too far from feasible space results in the generation of excessive numbers of designs that cannot be evaluated by CFD because converged flow solutions cannot be obtained. Such solutions effectively mean that the optimizer is then "stumbling about in the dark," which is again undesirable and inefficient.

B. Software Implementation

Figure 5 presents a flow diagram showing the stages of the process executed by MOBOS3D. The first stage is the parameterization of the initial blade design, input through a CAD geometry file together with boundary conditions for the flow solution. This is an important stage of the process, in which the long and complicated CAD file description is transformed into a short manageable string of numbers: the design vector. The geometry is parameterized using a PDE approach [24], giving a compact but flexible representation of the design, in a design vector comprising 26 variables. See [23] for a detailed explanation and definition of these design variables. This parameterization method ensures that smooth geometries are generated. There are integrated criteria that secure the integrity of all new blade geometries, and any violation of these criteria is treated as a hard constraint, as described in Sec. III.A.3. This design vector is the input to the main loop of the design system, which consists of the flow simulation and optimization processes.

On receipt of a new design vector, a computational mesh is automatically generated from the geometry specification, and then a

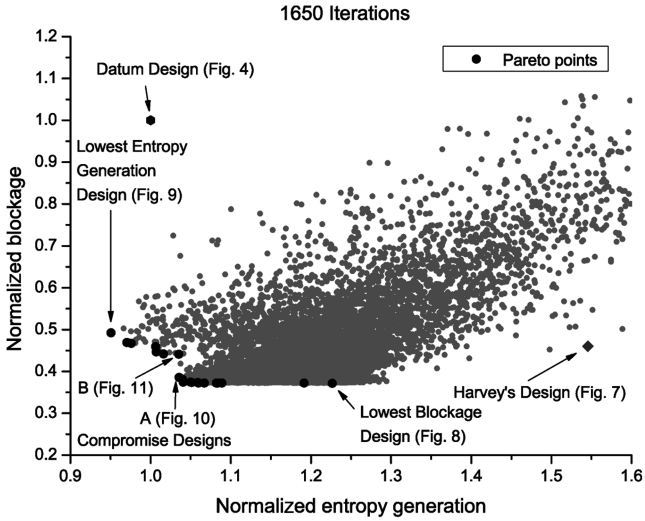

Fig. 6 Optimization search pattern and the Pareto front found.

detailed CFD analysis (blade to blade) is performed. The mesh is a 3D H-type structured grid consisting of $31 \times 97 \times 33$ nodes in each direction (tangential, axial, and radial). The flow simulation is performed by a CFD code solving the 3D Navier–Stokes equations, and this routine returns all the necessary metrics that describe the flow around the blade [43]. It is essential to use a Reynolds-averaged Navier–Stokes solver, because, to improve the blade design, accurate predictions of the complicated viscous flow in compressors are required, especially to capture the secondary flow effects responsible for the profile and endwall losses. Based on this simulation, the objective-function evaluations follow and the optimizer then generates new design vectors that are, in turn, meshed and evaluated. This process continues until a stopping criterion is met.

Initially, the flow solutions are not fully resolved to enable good regions of the design space, worthy of more detailed exploration, to be located quite fast. Then the fidelity of the flow simulation is increased and promising regions of the search space are investigated with higher accuracy. Five quality levels, in which the flow solution quality varies from 1300 to 1700 CFD iterations, are used with the switch between these being performed every 200 optimization steps. The mesh quality is constant to secure converged flow solutions.

At the end of the optimization process, the best design vectors identified and their associated flow solutions are converted into a single file, in the final stage of representation. This stage is accomplished by using nonuniform rational B-splines (NURBS) [44], a representation technique appropriate for complex 3D reconstructions, approximating curves well using a small number of control points. The optimal geometries can then be examined in detail through, for instance, contour plots.

C. Results and Discussion

The optimization was initiated from the datum geometry shown in Fig. 4. Figure 6 shows the progress in objective space of the search performed by our multi-objective TS algorithm over the 1650

Table 1 TS parameter settings

Parameter	Value	Description
Intensify	25	Intensify search after *intensify* iterations with adding to the medium-term memory (MTM)
Diversify	75	Diversify (or intensify) search after *diversify* iterations with adding to the MTM
Reduce	95	Reduce step sizes and restart after *reduce* iterations with adding to the MTM
n_{stm}	15	short-term memory size—the last n_{stm} visited points are tabu
n_{regions}	4	In the long-term memory each variable is divided into n_{regions} to determine which regions of the search space were underexplored
SS	1	Initial step size as percentage of variable range
SSRF	0.7	Factor by which step sizes are reduced on restarting
n_{sample}	9	Number of points randomly sampled at each H&J move
$\text{ssr}_{\text{thresh}}$	5	Number of step-size reductions before new restart strategy is executed
im_{thresh}	10	Intensification-memory size threshold; above intensification replaces diversification

Table 2 Datum and optimized flow metrics

Flow metric	Datum design	Single-objective optimum	Multi-objective optimum
Annulus mass flow, kg/s	15.6760	15.4646	15.5745
Entropy generation, J/K	0.367781	0.532644	0.444749
Blade loading, N	435.98	392.173	458.568
Mass-averaged inlet static pressure, Pa	34375.8	34410.4	34365.9
Mass-averaged outlet static pressure, Pa	35306.6	35330.2	35314.4
Mass-averaged inlet stagnation pressure, Pa	36403.0	36395.1	36393.8
Mass-averaged outlet stagnation pressure, Pa	36269.9	36307.6	36260.6
Mass-averaged flow turning, deg	22.5605	20.3158	22.5482
Mass-averaged velocity standard deviation	10.1124	8.07365	8.11117
Blockage	0.293881	0.141806	0.110876
f_1 (blockage)	1	0.4602	0.3719
f_2 (entropy)	1	1.5456	1.2267

iterations of the run, using the control parameters specified in Table 1. See [6] for a detailed explanation of these parameters. The values of these control parameters were chosen based on experience, but it should be noted that studies in [6] show that the algorithm's performance is relatively insensitive to the control parameter settings. Thirty-nine objective-function evaluations were executed on the average per optimization step, and 11,000 out of the total of 65,000 new designs were feasible. It should be mentioned here that the number of function evaluations required at each iteration increases linearly with the number of design variables, and there is a potentially exponential relationship between the number of iterations required and the number of objectives. The optimization was run on a nine-node parallel PC cluster of 2.8-GHz Pentium 4 machines to reduce wall-clock run times by exploiting our system's parallel capabilities.

From the search pattern, it is clear that the search space is strongly constrained in the region in which the lowest normalized blockage designs occur. Many of the feasible designs found have very similar values of this objective, and the tradeoff surface in this region is almost horizontal, indicating that small improvements in normalized blockage are obtained only at the cost of large deteriorations in the normalized entropy generation rate.

It is also apparent that significant performance improvements are achievable. For instance, the designs lying toward the right-hand end of the Pareto front reduce the blockage significantly compared with the initial datum design, at the cost of about a 20% increase in the rate of entropy generation.

The lowest-blockage design performs comparably with Harvey's blockage-optimized blade [3] (shown in Fig. 7), but has a significantly lower entropy generation rate. A number of Pareto-optimal designs also perform better than the datum design in terms of normalized entropy generation rate, these designs having been found after a restart, as discussed in Sec. III.B.2.

The Pareto front shows the achievable tradeoff between blockage and entropy generation to the designer, enabling them to make a well-informed final design choice. In this case (Fig. 6), it also exhibits an interesting feature: a clear discontinuity in the vicinity of the compromise-design area, which will be discussed later.

Table 2 presents the flow metrics for the datum design (Fig. 4), the single-objective optimal design found by Harvey (Fig. 7) and the lowest-blockage optimal design from the multi-objective test case (Fig. 8). The primary component of the second figure of merit was reduced to 48% of its initial value in the single-objective test case and to 38% in the multi-objective test case. The mass flow has reduced by just 1.3 and 0.6% for the two optimization approaches, and the static pressure rise as a fraction of inlet dynamic head has risen from 0.459 to 0.463 (+1%) and to 0.468 (+2%), based on the average pressures listed in Table 2. This increase in diffusion is possible for similar mass flow and flow turning, because the blockage region of viscous mixing was reduced in size, and therefore more of the fluid making up the mass flow was able to recover about half of the inlet head as static pressure rise. In contrast, the primary component of the first figure of merit (normalized entropy generation rate) was increased by 45 and by 21%, respectively. This is to be expected, because the two flow metrics are incommensurate and the single-objective optimal design has not been optimized for lower entropy generation, whereas the multi-objective optimal design is taken from the right-hand (highest-entropy-generation rate) end of the Pareto front in Fig. 6.

Another encouraging sign from the flow metric comparison (Table 2) is the reduced mass-averaged measurement of the outflow's standard deviation of velocity for the optimized designs. This suggests more uniform outflows that are therefore possibly less vulnerable to subsequent mixing losses.

It should be mentioned that some of the flow metrics in Table 2 are statistical nondimensional quantities and therefore appear without units.

Optimized geometries found for our test case are presented in Figs. 8–10. These show, respectively, the lowest-blockage design, the lowest-entropy-generation design, and a compromise design from the tradeoff surface (Fig. 6). All the figures showing blade designs use the same format, as defined in the caption to Fig. 4. In all of these, the magnitude of the flow velocity across the blade surface and within the compressor is shown using the same shading scale throughout, again defined in Fig. 4. Dark areas indicate low velocities, characteristic of flow separation.

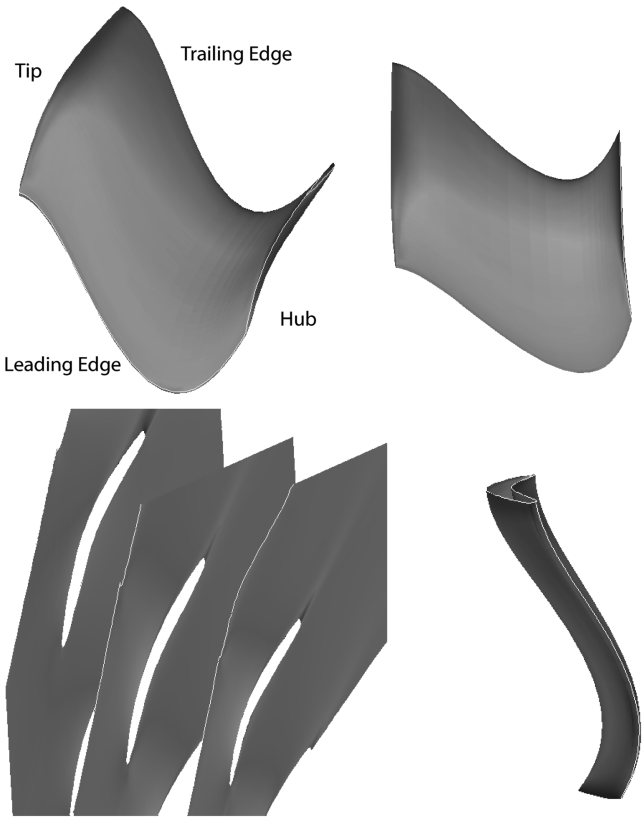

Fig. 7 Optimized geometry for the single-objective test case [4], as in Fig. 4 (identical velocity magnitude scale range).

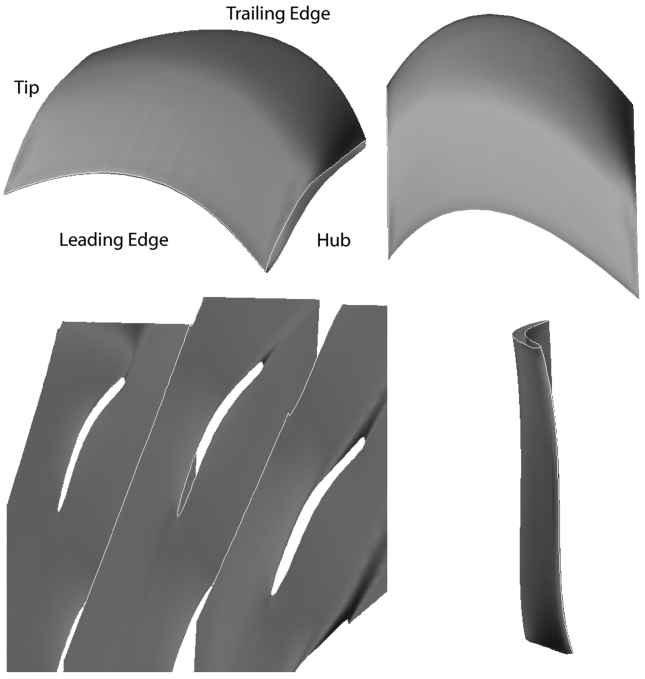

Fig. 8 Optimized geometry for lowest blockage, as in Fig. 4 (identical velocity magnitude scale range).

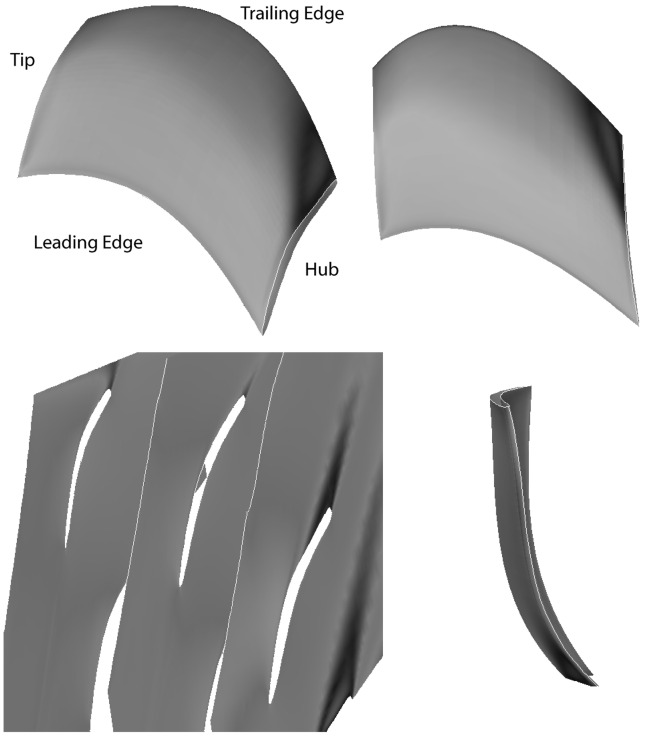

Fig. 9 Optimized geometry for lowest entropy generation, as in Fig. 4 (identical velocity magnitude scale range).

These blades are quite different from the datum geometry used to initiate the optimization; in addition, there are significant differences between them. The blade for the lowest entropy generation (Fig. 9) has a completely different leading-edge (LE) shape across the span profile, close to the tip area, compared with the blade for lowest blockage (Fig. 8) and the compromise-design blade (Fig. 10). In addition, the blade for lowest entropy generation is leaned toward the hub region. In contrast, the blade for lowest blockage and the compromise blade have similar geometrical characteristics. Their LE and TE are almost the same, as are their

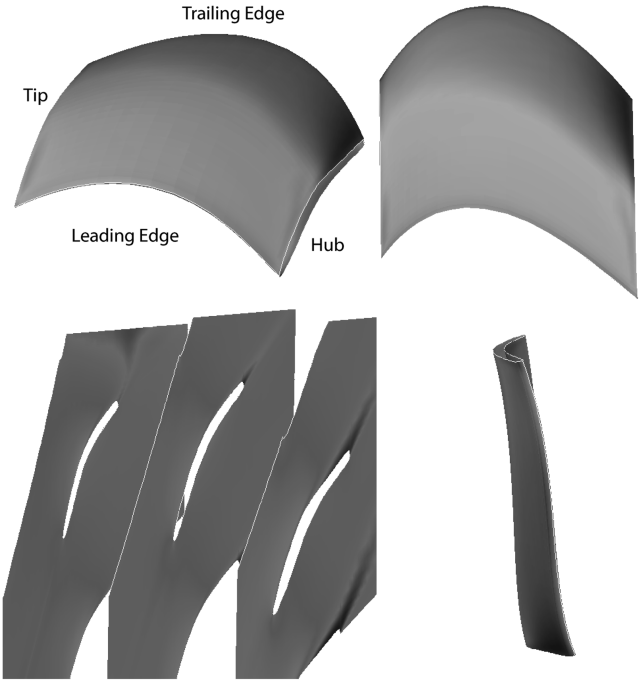

Fig. 10 Optimized geometry for compromise design A (Fig. 6), as in Fig. 4 (identical velocity magnitude scale range).

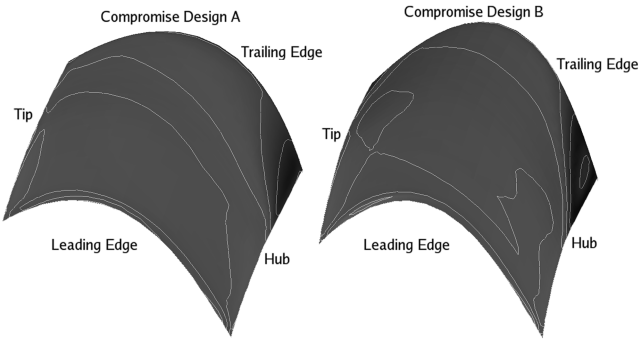

Fig. 11 Velocity magnitude contour plots for compromise designs A and B (Fig. 6).

camber and thickness. Furthermore, there are considerable differences in the geometrical characteristics of the multi-objective and single-objective designs.

Regarding aerodynamic performance, all the blades have reasonable velocity distributions. However, there is noticeably stronger flow separation along the TE of the blade optimized for lowest entropy generation. The wake for this blade is thicker than those for the other two optimal geometries. The only noticeable difference between the lowest-blockage design and compromise design A is on the rear hub corner, from which the flow of the compromise blade separates earlier.

As mentioned, interestingly, the Pareto front exhibits a clear discontinuity in blockage values in the compromise-design area. This discontinuity occurs between compromise design A (Fig. 10), already discussed, and an alternative compromise design B. Of the two, the latter has a slightly lower value for the entropy generation objective, but a noticeably higher value for the blockage objective, and its geometry is more similar to that of the lowest-entropy-generation design (Fig. 9). Detailed velocity contour plots for these two compromise designs (Fig. 11) clearly show that there is a sharp step change in the strength and extent of the corner separation at this point. This appears to be associated with a switch between essentially 2D and 3D cellular flow structure across the blade span, with the latter accounting for the higher blockage without significantly affecting the entropy generation. In terms of their geometrical

features, the two blade shapes differ in the lean on both the hub and tip ends. The lean on the hub is maintained throughout the optimal geometries for lower entropy generation, whereas the lean on the tip disappears for the extreme Pareto design for lowest entropy generation.

The large changes made to the blade geometries during the search demonstrate the flexibility of the geometry management system used, and the geometrical differences between the optimal designs found from single-objective and multi-objective approaches indicate the richness of the design space and highlight the need for an efficient search/optimization tool to support the designer in exploring this space.

V. Conclusions

The results for the foregoing test case demonstrate that our multi-objective integrated turbomachinery design optimization system can successfully tackle a real-world problem, negotiating the highly constrained, nonlinear search space and presenting the designer with a range of designs showing the tradeoffs between the objectives under consideration, giving insight into the nature of the design space and suggesting innovative designs for further consideration.

In this test case, the performance of the designs found match or exceed the performance of the optimized blade identified in an earlier single-objective study. The design resulting from that test case achieved a measured rise in efficiency of over 3% when tested experimentally [25]. Considering the comparison between the single-objective optimized blade and the blade for lowest blockage from our multi-objective test case (Sec. IV.C), we can anticipate that the new optimal designs will attain similar performance.

Although the case study presented here is a biobjective one, the method developed is applicable to problems of arbitrary dimension, and its performance on higher-dimension problems is the subject of further study.

The factors influencing the efficiency of turbomachinery blades and the tradeoffs between them are extremely complex, and therefore this area needs further investigation. The next steps will be to define additional loss objectives for the design problem examined previously. Thus, new objective functions will individually evaluate the profile losses and the secondary losses to improve understanding of the tradeoffs between them in design. These investigations will require the tackling of three- and four-objective problems and will therefore also allow us to test the effectiveness of our multi-objective TS variant on these higher-dimension problems.

In addition, because the local search enhancements of the optimizer described here proved to be beneficial for the design of turbomachinery blades, this suggests that further investigations into how better to explore and manage the design space of this problem are warranted. Another variant of our TS algorithm is under development, with an intelligent design variable selection scheme based on the idea of *path relinking*, proposed by Glover [45]. It is hoped that this will further reduce the computational time needed for a complete optimization and therefore increase the efficiency of the integrated system.

Acknowledgments

The first author gratefully acknowledges the support of the Embiricos Foundation and the Cambridge European Trust. The second author acknowledges the support of the Engineering and Physical Sciences Research Council (EPSRC) under grant number GR/R64100/01.

References

[1] Gaiddon, A., Knight, D. D., and Poloni. C., "Multicriteria Design Optimization of a Supersonic Inlet Based upon Global Missile Performance," *Journal of Propulsion and Power*, Vol. 20, No. 3, 2004, pp. 542–558.

[2] Kipouros, T., Parks, G. T., Savill, A. M., and Jaeggi, D. M., "Multi-Objective Aerodynamic Design Optimisation," *ERCOFTAC Design Optimization: Methods and Applications Conference* [CD-ROM], European Research Community on Flow, Turbulence and Combustion, Lausanne, Switzerland, 2004, Paper ERCODO2004_239.

[3] Harvey, S. A., "The Design Optimisation of Turbomachinery Blade Rows," Ph.D. Thesis, Dept. of Engineering, Univ. of Cambridge, Cambridge, England, U.K., 2002.

[4] Dawes, W. N., Kellar, W. P., Harvey, S. A., Dhanasekaran, P. C., Savill, A. M., and Cant, R. S., "Managing the Geometry is Limiting the Ability of CFD to Manage the Flow," 33rd AIAA Fluid Dynamics Conference, Orlando, FL, AIAA Paper 2003–3732, 2003.

[5] Jaeggi, D. M., Asselin-Miller, C. S., Parks, G. T., Kipouros, T., Bell, T., and Clarkson, P. J., "Multi-Objective Parallel Tabu Search," *Parallel Problem Solving from Nature*, Vol. 3242, Lecture Notes in Computer Science, Springer–Verlag, Berlin, 2004, pp. 732–741.

[6] Jaeggi, D. M., Parks, G. T., Kipouros, T., and Clarkson, P. J., "A Multi-Objective Tabu Search Algorithm for Constrained Optimisation Problems," *Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico*, Lecture Notes in Computer Science, Vol. 3410, Springer–Verlag, Berlin, 2005, pp. 490–504.

[7] Hicks, R. M., and Henne, P. A., "Wing Design by Numerical Optimization," *Journal of Aircraft*, Vol. 15, No. 7, 1978, pp. 407–412.

[8] Burgreen, G. W., Baysal, O., and Eleshaky, M. E., "Improving the Efficiency of Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 32, No. 1, 1994, pp. 69–76.

[9] Bowen, Z., and Zhide, Q., "Multiobjective Optimization Design of Transonic Airfoils," 19th ICAS Congress, Los Angeles, International Council of the Aeronautical Sciences Paper 94-2.1.1, 1994.

[10] Topliss, M. E., Toomer, C. A., and Hills, D. P., "Rapid Design Space Approximation for Two-Dimensional Transonic Aerofoil Design," *Journal of Aircraft*, Vol. 33, No. 6, 1996, pp. 1101–1108.

[11] Patnaik, S. N., Guptill, J. D., Hopkins, D. A., and Lavelle, T. M., "Neural Network and Regression Approximations in High-Speed Civil Aircraft Design Optimization," *Journal of Aircraft*, Vol. 35, No. 6, 1998, pp. 839–850.

[12] De La Garza, A., and Darmofal, D. L., "An All-at-Once Approach for Multidisciplinary Design Optimization," 16th AIAA Applied Aerodynamics Conference, Albuquerque, NM, AIAA Paper 98-2515, 1998.

[13] Pierret, S., "Three-Dimensional Blade Design by Means of an Artificial Neural Network and Navier–Stokes Solver," *Turbomachinery Blade Design Systems*, Lecture Series 1999-02, Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 1999.

[14] Anders, J. M., Haarmeyer, J., and Heukenkamp, H., "A Parametric Blade Design System (Parts 1 and 2)," *Turbomachinery Blade Design Systems*, Lecture Series 1999-02, Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium, 1999.

[15] Farin, G., *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, 3rd ed., Academic Press, San Diego, CA, 1990.

[16] Reuther, J. J., Alonso, J. J., Martins, J. R. R. A., and Smith, S. C., "A Coupled Aero-Structural Optimization Method for Complete Aircraft Configurations," 37th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, AIAA Paper 99-0187, 1999.

[17] Chamis, C. C., "Coupled Multidisciplinary Optimization of Engine Structural Performance," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 190–199.

[18] Nemec, M., Zingg, D. W., and Pulliam, T. H., "Multipoint and Multi-Objective Aerodynamic Shape Optimization," *AIAA Journal*, Vol. 42, No. 6, 2004, pp. 1057–1065.

[19] Chiba, K., Obayashi, S., Nakahashi, K., and Morino, H., "High-Fidelity Multidisciplinary Design Optimization of Wing Shape for Regional Jet Aircraft," *Evolutionary Multi-Criterion Optimization (EMO 2005)*, Vol. 3410, Lecture Notes in Computer Science, Springer–Verlag, Berlin, 2005, pp. 621–635.

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

[21] Samareh, J. A., "Status and Future of Geometry Modeling and Grid Generation for Design Optimization," *Journal of Aircraft*, Vol. 36, No. 1, 1999, pp. 97–104.

[22] Demeulenaere, A., Ligout, A., and Hirsch, C., "Turbomachinery Optimization Using FINE/Design3D^{TM}," *ERCOFTAC Bulletin*, Vol. 66, 2005, pp. 43–47.

[23] Harvey, S. A., Dawes, W. N., and Gallimore, S. J., "An Automatic Design Optimisation System for Axial Compressors, Part 1: Software Development," ASME Turbo Expo 2003, Atlanta, GA, American Society of Mechanical Engineers Paper GT2003-38115, 2003.

[24] Bloor, M. I. G., and Wilson, M. J., "Efficient Parametrization of Generic Aircraft Geometry," *Journal of Aircraft*, Vol. 32, No. 6, 1995, pp. 1269–1275.

[25] Harvey, S. A., Dawes, W. N., and Gallimore, S. J., "An Automatic Design Optimisation System for Axial Compressors Part II: Experimental Validation," ASME Turbo Expo 2003, Atlanta, GA, American Society of Mechanical Engineers Paper GT2003-38650, 2003.

[26] Samareh, J. A., "Survey of Shape Parameterization Techniques for High-Fidelity Multidisciplinary Shape Optimization," *AIAA Journal*, Vol. 39, No. 5, 2001, pp. 877–884.

[27] Sugimura, K., Savill, A. M., and Dawes, W. N., "Development of Autonomous Aerodynamic Design Optimisation System by Using Simulated Annealing Coupled with Neural Network Adaptations," Dept. of Engineering, Univ. of Cambridge, Cambridge, England, U.K., 2004.

[28] Battiti, R., "Reactive Search: Toward Self-Tuning Heuristics," *Modern Heuristic Search Methods*, edited by V. J. Rayward-Smith, I. H. Osman, C. R. Reeves, and G. D. Smith, Wiley, Chichester, England, U.K., 1996, pp. 61–83.

[29] Gandibleux, X., and Ehrgott, M., "1984-2004-20 Years of Multiobjective Metaheuristics. But What About the Solution of Combinatorial Problems with Multiple Objectives?," *Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico*, Vol. 3410, Lecture Notes in Computer Science, Springer–Verlag, Berlin, 2005, pp. 33–46.

[30] Miettinen, K., *Nonlinear Multiobjective Optimization*, Kluwer Academic, Boston, 1999.

[31] Molinari, M., Jarrett, J. P., Clarkson, P. J., and Dawes, W. N., "Characterizing the Design Space in Multiobjective Axial Compressor Blade Optimization," 2nd AIAA Multidisciplinary Design Optimization Specialist Conference, Newport, RI, AIAA Paper 2006-1908, 2006.

[32] Hoos, H. H., and Stützle, T., *Stochastic Local Search: Foundations and Application*, Morgan Kaufmann, San Francisco, 2004.

[33] Molina, J., Laguna, M., Martí, R., and Caballero, R., "SSPMO: A Scatter Tabu Search Procedure for Non-Linear Multiobjective Optimization," *INFORMS Journal on Computing*, Vol. 19, No. 1, 2007, pp. 91–100.

[34] Hansen, M. P., "Tabu Search for Multiobjective Optimization: MOTS," *Trends in Multicriteria Decision Making*, Vol. 465, Lecture Notes in Economics and Mathematical Systems, Springer–Verlag, Berlin, 1998, pp. 574–586.

[35] Baykasoglu, A., Owen, S., and Gindy, N., "A Taboo Search Based Approach to Find the Pareto Optimal Set in Multiple Objective Optimization," *Engineering Optimization*, Vol. 31, No. 6, 1999, pp. 731–748. doi:10.1080/03052159908941394

[36] Jones, D. F., Mirrazavi, S. K., and Tamiz, M., "Multi-Objective Meta-Heuristics: An Overview of the Current State-of-the-Art," *European Journal of Operational Research*, Vol. 137, No. 1, 2002, pp. 1–9.

[37] Connor, A. M., and Tilley, D. G., "A Tabu Search Method for the Optimization of Fluid Power Circuits," *Proceedings of the Institution of Mechanical Engineers, Part I (Journal of Systems and Control Engineering)*, Vol. 212, No. 5, 1998, pp. 373–381. doi:10.1243/0959651981539541

[38] Hooke, R., and Jeeves, T. A., "'Direct Search' Solution of Numerical and Statistical Problems," *Journal of the Association for Computing Machinery*, Vol. 8, No. 2, 1961, pp. 212–229. doi:10.1145/321062.321069

[39] Lewis, R. M., Torczon, V., and Trosset, M. W., "Direct Search Methods: Then and Now," *Journal of Computational and Applied Mathematics*, Vol. 124, No. 1, 2000, pp. 191–207. doi:10.1016/S0377-0427(00)00423-4

[40] Glover, F., and Laguna, M., *Tabu Search*, Kluwer Academic, Boston, 1997.

[41] Silverman, B. W., *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London, 1986.

[42] Denton, J. D., "Loss Mechanisms in Turbomachines," *Journal of Turbomachinery*, Vol. 115, No. 4, 1993, pp. 621–656.

[43] Dawes, W. N., "Development of a 3D Navier–Stokes Solver for Application to all Types of Turbomachinery," ASME Gas Turbine Conference, Amsterdam, American Society of Mechanical Engineers, Paper 88-GT-70, 1988.

[44] Rogers, D. F., *An Introduction to NURBS with Historical Perspective*, Morgan Kaufmann, San Francisco, 2000.

[45] Glover, F., "Scatter Search and Path Relinking," *New Ideas in Optimization*, edited by D. W. Corne, M. Dorigo, and F. Glover, McGraw–Hill, Maidenhead, England, U.K., 1999, pp. 297–316.

N. Alexandrov
Associate Editor