Progress in Aerospace Sciences 93 (2017) 1–23

Contents lists available at ScienceDirect

Progress in Aerospace Sciences

journal homepage: www.elsevier.com/locate/paerosci

Review of design optimization methods for turbomachinery aerodynamics

Zhihui Li, Xinqian Zheng *

Turbomachinery Laboratory, State Key Laboratory of Automotive Safety and Energy, Tsinghua University, Beijing, 100084, China

A B S T R A C T

In today's competitive environment, new turbomachinery designs need to be not only more efficient, quieter, and "greener" but also need to be developed at on much shorter time scales and at lower costs. A number of advanced optimization strategies have been developed to achieve these requirements. This paper reviews recent progress in turbomachinery design optimization to solve real-world aerodynamic problems, especially for compressors and turbines. This review covers the following topics that are important for optimizing turbomachinery designs. (1) optimization methods, (2) stochastic optimization combined with blade parameterization methods and the design of experiment methods, (3) gradient-based optimization methods for compressors and turbines and (4) data mining techniques for Pareto Fronts. We also present our own insights regarding the current research trends and the future optimization of turbomachinery designs.

1. Introduction

The aerodynamic design of turbomachinery blades is a complex task due to the complicated flow phenomena and the interactions between various disciplines. The traditional trial-and-error design method depends strongly on the experience of the designers so the process cannot lead to globally optimized design. Moreover, these processes usually require very long design cycles.

The rapid development of computational capabilities has led to the increasing use of numerical design methods coupling numerical simulations and optimization methods to release the design process from depending on the designers' experience. Various optimization frameworks have been developed based on CFD simulations. Optimization methods can generally be divided into stochastic models and gradient-based models. The stochastic model globally searches for the optimum, while the gradient models use an optimization routine based on gradient information.

Besides the optimization algorithms themselves, the design platform also used the design of experiment (DOE) method, Surrogate Model and Data-Mining method. The design of experiment step is the design of any task that aims to describe or explain the variation of information for conditions that are hypothesized to reflect the variation. The term is generally associated with actual experiments in which the design introduces conditions that directly affect the variations, but may also refer to the design of quasi-experiments, in which natural conditions that influence the variation are selected for observation.

Optimization algorithms, especially the stochastic models, are very computationally expensive so they cannot be easily used in practical applications. Surrogate models have been used to approximate the computationally expensive functions to provide expensive reasonable predictions to the real functions. The Response Surface Method, Kriging Model and Artificial Neural Networks are popular surrogate models for turbomachinery design. The challenge of surrogate modeling is to generate a model that is as accurate as possible while using as few simulation evaluations as possible.

The difficulty for multiple objective problems is how to get useful design information from the Pareto-optimal solutions. Thus, data mining techniques can be a powerful tool to address this issue. Data mining is an interdisciplinary subfield of computer science that searches for patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

2. Optimization of turbomachinery designs

Aerodynamic design optimization methods can be distinguished into inverse and direct designs. Inverse design methods specify a pressure distribution to develop a profile shape by iterative modifications of the blade shape. The computational cost is proportional to a small number of flow analyses and is, thus, comparably inexpensive. Inverse design methods can be combined with an optimization method in an efficient design process [1–3]. However, the pressure distribution may pass through a number of iterations to obtain an acceptable profile. This approach relies strongly on the experience of the designer who needs to specify a pressure distribution which fulfills the various aerodynamic

* Corresponding author.
E-mail address: zhengxq@tsinghua.edu.cn (X. Zheng).

http://dx.doi.org/10.1016/j.paerosci.2017.05.003
Received 3 May 2017; Received in revised form 14 May 2017; Accepted 16 May 2017
Available online 12 July 2017

design aspects in terms of the flow turns, boundary layer properties and flow losses which also performs well for off-design conditions. Another shortcoming of the inverse design method is how to integrate the geometric and mechanical constraints. Unlike with inverse design process, the direct design method optimizes the shape based secondary aerodynamic properties like the aerodynamic losses with the computational cost involving many single flow calculations.

The optimization algorithms used with the direct design method are mainly the gradient based methods and the stochastic algorithms. Gradient-based methods rely on derivative information for all the objectives and all the constraints to determine the optimization search direction. These methods start with a single design point and use the local gradient of the objective function with respect to changes in the design variables to determine a search direction by using methods such as the steepest descent method, conjugate gradient method, quasi-Newton techniques, or adjoint formulations. These methods are efficient and can find a true optimum as long as the objective function is differentiable and convex. However, the optimization process can sometimes lead to a local, not necessarily a global, optimum close to the starting point. Furthermore, such computations can easily get bogged down when many constraints are considered.

Genetic Algorithms and Evolutionary algorithms are typical stochastic optimization algorithms. These methods are robust optimization algorithms that can cope with noisy, multimodal functions, but are also computationally expensive in terms of the necessary number of flow analyses required for convergence. They start with multiple points sprinkled over the entire design space and search for true optimums based on the objective function instead of the local gradient information by using selection, recombination, and mutation operations.

Fig. 1 shows a typical optimization flowchart that includes the parametric modeling, the N-S solver, the optimization algorithm and data mining. The geometry is first parameterized and the shape is modified by changing key parameters. The blade shape is inserted into a flow region with the N-S equation solver then used to solve for the flow around the blade. Then, the results are post-processed by the data mining tool to identify the flow mechanisms. The optimization algorithm then predicts the optimum combination of parameters that are transferred to the parametric model with the flow solver used to validate this design. The optimization process continues until it satisfies the design criteria.

3. Optimization using stochastic methods

3.1. Parameterization

The geometric representation of the blade profile is an important part of any shape optimization procedure which directly determines the number of design variables in the optimization process. The parameterization requirements are:

1) Have sufficient flexibility to cover the entire search space, even for "non-traditional" shapes.
2) To minimize the number of design parameters.
3) To avoid curvature discontinuities at the junctions of curves through local approximation or interpolation models.
4) To preferably include design variables linked directly to the constraints, while excluding design variables which have little effect on the aerodynamic performance of the shape.

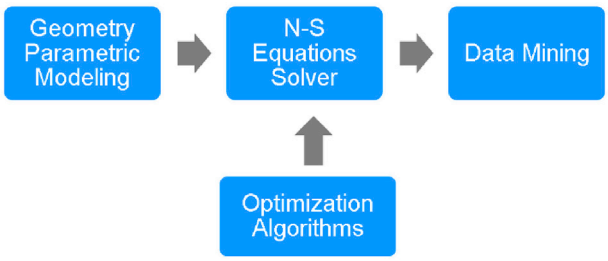

Fig. 1. Optimization framework.

Basically, the blade parameterization method is based on conventional multi-section blade parameterization [6,41,42] which defines the camber line, the stacking axis, the thickness distributions and other parameters. The design parameters are controlled by Bezier [4,11,12,20,75], B-Spine [5,76,77,78], and even Non-Uniform Rational B-Spine (NURBS) [6–10] shapes. Often, the blade is described by a number of profile sections on the same conical surfaces. Then, the spanwise profiles are connected by the stacking lines. A blade is considered acceptable if these quantities are continuous along the blade span without turning points. Each blade can be decomposed into N points (N-1 segments) using third-, fourth- or fifth- order splines, Bezier curves or circular arcs [74]. These methods can also be combined with the inverse design method. The detailed blade geometry is then calculated using the through-flow design or blade formatting design methods [1,2,3]. The advantage of this method is that the aerodynamic parameters are the design variables so the design parameters more closely correlate with the flow field to give more direct control of the aerodynamics. Fig. 2 shows an airfoil parameterization based on NURBS curves with the parameterized airfoil fitting well with the original shape. However, high turning or high curvature blades cannot be easily defined by lower order splines, so Hicks and Henne shape functions [160] have been used as an alternative. Recently, Wang et al. [161] rejected the traditional blade aerodynamic design and used a trumpet-shaped flow path which gradually turned to the desired angle.

The blade optimization methods also need to parameterize the fine blade features such as the fillet near the blade root with a limited number of design variables. Thus, advanced parameterization methods are becoming more important in the optimization process.

3.2. Design of experiment

In the simplest form, the DOE process predicts the outcome by introducing a change in the preconditions. Experimental design involves not only the selection of suitable predictors and outcomes, but also the planning of the experiment for statistically optimal conditions given the constraints on the available resources. The experimental designs seek to provide the maximum information with the minimum number of design experiments to reduce the number of computationally intensive design calculations.

The Central Composite Design method (CCD) is one of the most widely used experiment design methods [15]. CCD composite designs offer an efficient alternative to second-order response surface models [44]. However, the number of points in the CCD process increases exponentially with the number of design variables, so this is inefficient for high dimensional design problems [45]. The alphabetical optimal designs, especially the D-optimal designs, have also been widely utilized [44,45,46,47]. The D-optimal process requires smaller number of designs than the CCD, since the CCD is over-determined and needs more experiments than necessary. However, D-optimal has only a model-dependent D-efficiency and does not address the prediction variance.

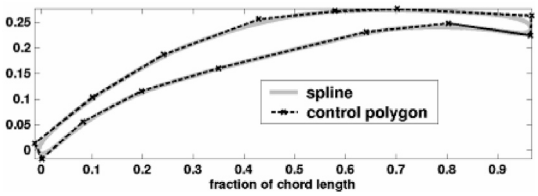

Fig. 2. Airfoil parameterization using NURBS curves [8].

Generally, DOE uses two types of simulation models, stochastic models and deterministic models. Since computer experiments involve mostly systematic errors rather than random errors as in physical experiments, a good experimental design for deterministic computer analyses tends to fill the design space rather than to concentrate on the boundary [28,29]. The space filling methods include orthogonal arrays [48,49] and various Latin Hypercube Designs (LHD) [14,50,51]. The LHD designs were found to more accurately estimate the means, variances and distribution functions of an output than random sampling and stratified sampling. Moreover, LHD ensures that each of the input variables is represented over portions of its range. Also, LHD can cope with many input variables and is less expensive computationally.

LHD was first introduced by McKay et al. [14] as a space-filling design process that provides more information within the design space and can be used with approximate computer experiments which mainly have system errors rather than random errors. The LHD process is relatively straightforward with the range of each input design variable divided into n intervals with each observation on the input variable made in each interval using random sampling. Thus, there are n observations for each of the d input variables. One of the observations on variable x_1 is randomly selected (each observation can be selected equally), matched with a randomly selected observation on x_2, and so on through x_d to build a design vector, X_1. One of the remaining observations on x_1 is then matched at random with one of the remaining observation on x_2 and so on to get X_2. The procedure is followed for $X_3, X_4, \ldots, X_n$, resulting in n LHD sampling points.

In real designs, some combinations of the variables are not feasible or can even crash the CFD code. However, LHD allows the flexibility to adjust a variable some without undermining the fundamental properties of the LHD sample [43]. Various optimal LHDs have been proposed based on minimax, minimum mean square error, maximum entropy or orthogonal algorithms. The LHD sample size can be controlled by the designer based on their time, budget or other limitations. There is no comprehensive theory about the number of design points required to construct the response surface models with LHD. Unal et al. [44] mentioned that the number of experimental samples for building response surface models with LHD should be 20–50% of the number used with D-optimal designs. Fig. 3 compares the Full Factorial method using 3^3 variable combinations with the LHD sampling method using only 3 samples. LHD methods have been frequently coupled with surrogate models for optimization of turbomachinery designs [52–59].

Wang et al. [60] mentioned that the sampling strategy should be based on the function to be approximated. Thus, the sampling strategy should generate a minimum number of samples so that the metamodel accurately reflects the "black-box" function in the area of interest. This means that the sampling process should be iterative, adaptive and progressive, as has been shown in some publications [125,126].

3.3. Surrogate model

For complex systems, the design process is a daunting optimization task involving multiple disciplines, multiple objectives and computationally intensive models. The total time consumed is always unacceptable in practice. Despite continual advances in computing power, the finite element (FE) and finite difference codes are very complex. Thus, approximation-based optimization methods have attracted much attention in the past 20 years [60]. These optimization methods approximate the objective functions by simplified analytical models. The simple models are often called surrogate models or metamodels. Surrogate models approximate computationally expensive functions with computationally orders of magnitude cheaper models while still providing reasonably accurate approximations to the real functions. Artificial Neural Network (ANN), Response Surface Model (RSM) and the Kriging Model models are also widely used in turbomachinery applications.

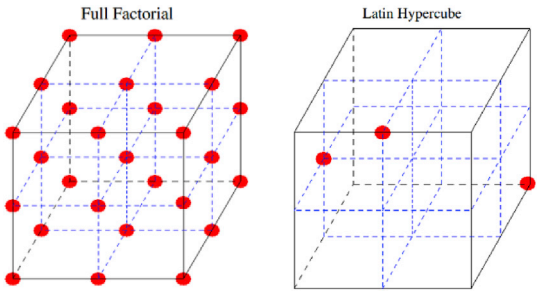

Fig. 3. Comparison of the Full Factorial and Latin Hypercube datapoints.

3.3.1. ANN

ANN algorithms use a neurological model with learning from experience, making generalization from similar situations and judging states [16]. The ANN method began in the early 1940s and became practical in the mid-1980s. Nowadays, there are many different types of ANN methods including the multilayer perceptron (MLP) (which is generally trained using the back-propagation of the error algorithm), learning vector quantization, and the radial basis function (RBF). Some ANNs are classified as feed forward while others are recurrent depending on how the data is processed through the network. ANN types can also be classified based on their learning method with some ANN using supervised training, while others are self-organizing.

The most widely used ANN is the Back Propagation Neural Network (BPNN). The first step is to initialize the weight and bias factors using small random values. Then, the input vector for the first training sample is input into the network input with the signal propagated to the output layer. Generally, the output vector provided by the network does not correspond to the desired output vector associated with this input vector, which is called the forward training phase. The error between the real and the desired output vector is back-propagated to the network input with this error used to adjust the connection weights to minimize the error. The learning process requires a set of input/output vectors that will be sequentially input to the network input/output layers. This process of presenting the input and output vectors to the network and updating the weights is repeated for each training set until the weights converge, which is called the backward process.

Fig. 4 shows the basic structure of a BPNN. In the forward phase, the sums of the input contributions are connected to the nodes in the next layer by a sigmoidal transfer function. The standard sigmoid transfer function, f_s, is:

$$f_s(x) = \frac{1}{1+e^{-x}} \tag{1}$$

In the backward propagation process, the least mean-spare error of the output is used to evaluate the convergence:

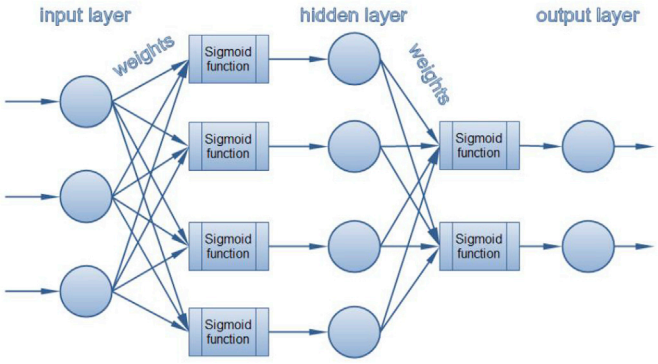

Fig. 4. Artificial neural network configuration [17].

$$E = \sum_{i=1}^{N_s} \sum_{k=1}^{N_o} \left(d_k^i - y_k^i\right)^2 \quad (2)$$

where N_s is the number of samples, N_o is the number of outputs, d_k^i is the value of the kth output of the ith sample and y_k^i is the corresponding approximated output value. The approximation accuracy of the BPNN not only depends on the number and quality of the training samples, but also on the neural network topology [17]. Although a BPNN can approximate any continuous functions in theory, the convergence of the learning process does not guarantee that the network correctly predicts the actual function. Fig. 5 shows how a network with 2 hidden layers gives better results for the non-linear function than a network with 1 hidden layer [17,61].

Radial Basis Neural Networks (RBNN) is an alternative to the more widely used BPNN that requires less computer time for network training. An RBNN consists of an input layer, a hidden layer, and an output layer. The nodes within each layer are fully connected to the previous layer. The input variables are each assigned to a node in the input layer and pass directly to the hidden layer without weights. Thus, each hidden node receives each input value unaltered. The hidden nodes contain the Radial Basis Functions (RBFs) which are also called the transfer functions. An RBF is symmetric about a mean or center point in a multidimensional space. The RBNN has a number of hidden nodes with RBF activation functions connected in a feed forward parallel architecture. The second layer of connections is weighted with the output nodes analyzed using a simple summation. Fig. 6 describes the RBNN framework.

The RBF parameters are optimized during the network training. These parameters are not necessarily the same throughout the network nor are they directly related to or constrained by the actual training vectors. When the training vectors are presumed to be accurate and a smooth interpolation is desired between them, then linear combinations of the RBFs which give no error in the training vectors. The RBNN has a faster training speed than the BPNN [19] and allows for easier optimization since the only parameter that can modify the network structure is the number of neurons in the hidden layer.

Wang et al. [17] used multi-objective optimization of turbomachinery designs based on an improved NAGA-II and BPNN. Bellman et al. [18] improved the GA efficiency with the BPNN method. Other similar applications of BPNN are published in Ref. [20,61–63]. Many other researchers have used RBNN based optimization for turbomachinery designs. The RBNN method based optimization procedure was used by Huppertz et al. [64]. The RBNN method was coupled with the response surface method for turbomachinery design by Rai and Madyavan [21].

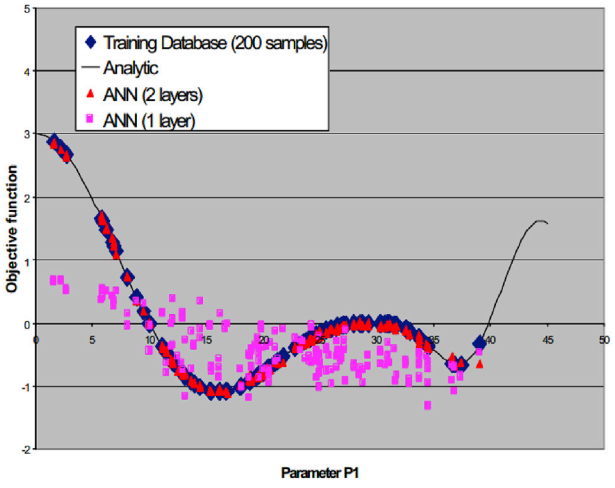

Fig. 5. Artificial neural networks with 1 and 2 hidden layers [61].

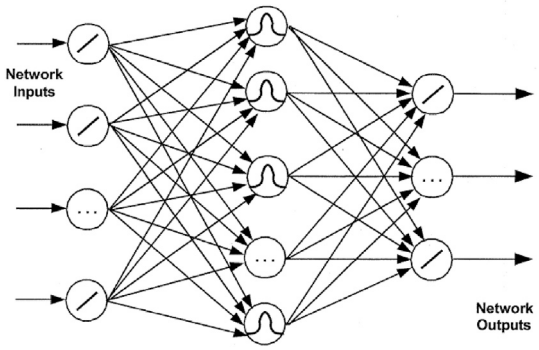

Fig. 6. Structure of the Radial Basis Neural Network [16].

Kim et al. [65] designed a centrifugal compressor impeller using RBNN. Industrial applications of the ANN methods were reviewed by Meireles et al. [16].

3.3.2. Polynomial response surface method

The basic RSM process involves selecting a number of design points at which the computationally expensive function is evaluated. Then, after these points are analyzed, response surfaces are constructed for the functional relationships between the design variables and the objective functions. A regular optimization procedure is then applied to the response surfaces to find optimal solutions.

RSM uses low-order polynomial approximations in place of computationally intensive simulations. The second order model is widely used due to its flexibility and ease of use [22,26]. A second-order response surface model with N variables can be written as:

$$y = \beta_0 + \sum_{i=1}^{N} \beta_i x_i + \sum_{i=1}^{N} \beta_{ii} x_i^2 + \sum \sum_{i \neq j}^{N} \beta_{ij} x_i x_j + \varepsilon \quad (3)$$

where x_i are the predictor variables, β are the regression coefficients, and ε denotes the total error that is the difference between the actual and the predicted response. The coefficients are typically estimated using a method of least squares regression. Assume that the prediction error at design point x_i is defined as

$$e_i = y_i - \hat{y}_i \quad (4)$$

where $\hat{y}_i$ is the predicted value. The adjusted Root Mean Square (RMS) error is then

$$\sigma_E = \sqrt{\sum_{i=1}^{N} e_i^2 / N} \quad (5)$$

Several statistical measures used to evaluate the RMS result are the coefficient of determination statistic, R^2, the adjusted statistic, R_{adj}^2, and the root mean square error, σ. Their definitions involve partitioning the total sum of the squares into a sum of the squares due to the model and a sum of the squares due to the error,

$$\sigma_T = \sigma_R + \sigma_E \quad (6)$$

where σ_T is the mean of the response and σ_R is the sum of the squares due to the model. The coefficient of determination statistic is

$$R^2 = \frac{\sigma_R}{\sigma_T} = 1 - \frac{\sigma_E}{\sigma_T} \quad (7)$$

which is the proportion of the variation in the response around the mean that can be represented by the model and this is similar to the ANOVA idea that will be discussed in section 5.1. A value of 1 means a perfect

model with no error while an R^2 of 0 means that the prediction has a prediction capability worse than the overall response mean method. The adjusted coefficient, R^2_{adj}, can be defined as

$$R^2_{adj} = 1 - \frac{N-1}{N-P}(1-R^2) \tag{8}$$

where P is the number of coefficients. This is a better measure than R^2 in which it does not increase when additional parameters are added. Thus, the modified RMS is

$$\sigma_{adj} = \sqrt{\frac{N}{N-P}}\sigma_E \tag{9}$$

When there are not enough extra data points to build the RSM, the predicted residual error sum of squares (PRESS) statistic is used to predict the RSM performance. The residual is obtained by building the RSM over the design space after eliminating one design point from the training set and then comparing the RSM estimated value with the expected one. PRESS is then given by

$$PRESS = \sqrt{\sum_{i=1}^{N}\left(y_i - \widehat{y_i}\right)^2 \Big/ N} \tag{10}$$

where $\widehat{y_i}$ is the value predicted by the RSM with the ith point excluded from the process. The modified model is regarded as good if this value is close to σ_{adj}.

The RSM have been widely used to optimize turbomachinery designs. Jang and Kim [66] used this method to optimize the shape of a stator blade in a single-stage transonic axial compressor. The blade optimization used the RSM with a 3D N-S analysis. Papila et al. [70] combined the RBNN with a polynomial-based RSM for global shape optimization of a supersonic turbine blade with similar applications in the literature [3,22,25,26,71,72]. The RSM has also been applied for data mining based on the trade-offs among objective functions [67]. Reliability-Based Design Optimization (RBDO) has based the RMS prediction interval on the Moving Least-Squares Method and Sensitivity (MLMS) [68,69].

3.3.3. Kriging model

The RSM are generally second-order polynomial models that have limited capability to accurately predict nonlinear results. Higher-order models are more accurate with nonlinear problems, but may be unstable. Kriging models show greater promise for accurate global approximations of the design space [26,27,30–33]. They are flexible due to the wide range of correlation functions which can be used to build the approximation framework. Thus, Kriging models can accurately predict both linear and nonlinear functions. An advantage of the Kriging model is its ability to reduce the number of needed parameters if the database is small. Thus, the model can be successful even if there are less training members than design variables [34]. Kriging models combine a global model with localized departures as:

$$\mathrm{y(x)} = \mathrm{f(x)} + \mathrm{Z(x)} \tag{11}$$

where y(x) is the objective function, f(x) is an approximation function which is usually a polynomial function and Z(x) is a stochastic function with zero variance and nonzero covariance. When f(x) globally approximates the design space, Z(x) creates deviations. The Z(x) is given by:

$$\mathrm{Cov}[\mathrm{Z}(x^i), \mathrm{Z}(x^j)] = \sigma^2 \mathbf{R}[\mathrm{R}(x^i, x^j)] \tag{12}$$

where $\mathbf{R}$ is the correlation matrix and R (x^i, x^j) is the correlation function of any two of the N data points x^i and x^j. The correlation function R (x^i, x^j) can be specified by the Gaussian correlation function of the form

$$\mathrm{R}(x^i, x^j) = \exp\left[-\sum_{k=1}^{N}\theta_k \left|x_k^i - x_k^j\right|^2\right] \tag{13}$$

where θ_k are the unknown parameters to fit the model, and x_k^i and x_k^j are the kth components of points x^i and x^j. Predicted values of $\widehat{y}$ are given by

$$\widehat{y} = \widehat{\beta} + r^T(x)R^{-1}\left(y - f\widehat{\beta}\right) \tag{14}$$

where $r^T(x)$ is the correlation vector between untried x and sampled data points $\{x^1, \ldots, x^N\}$:

$$r^T(x) = \left[\mathrm{R}\left(x, x^1\right), \mathrm{R}\left(x, x^2\right), \ldots, \mathrm{R}\left(x, x^N\right)\right]^T \tag{15}$$

Here $\widehat{\beta}$ is estimated using

$$\widehat{\beta} = \left(f^T R^{-1} f\right)\left(f^T R^{-1} f\right)^{-1} f^T R^{-1} y \tag{16}$$

The variance $\widehat{\sigma}$ is given by

$$\widehat{\sigma}^2 = \left[\left(y - f\widehat{\beta}\right)^T R^{-1}\left(y - f\widehat{\beta}\right)\right] \Big/ N \tag{17}$$

Finally the θ_k used to fit a Kriging model can be obtained by

$$\max\ \emptyset(\theta_k) = -\left[N ln\left(\widehat{\sigma}^2\right) + \ln|\mathbf{R}|\right]/2(\theta_k > 0) \tag{18}$$

where both $\widehat{\sigma}^2$ and $|\mathbf{R}|$ are functions of θ_k. Following the above approach, the Kriging model was built successfully. Simpson et al. [27] investigated the use of Kriging models as alternatives to second-order polynomial response surfaces for constructing global approximations in the design of an aerospike nozzle. Their error analysis showed that both the response surface and Kriging models yielded comparable results, meaning equal prediction level for the multidisciplinary design optimization. However, the Kriging models offer slightly more accurate approximations in terms of the prediction errors. Aulich and Siller [35] compared neural networks and Kriging models for optimizing a fan stage. Unlike the neural network, one benefit of Kriging models is that they are directly coupled with the database. In addition, neural networks are rather complex due to the many training weights. Their optimization process always used the Kriging model and restarted the neural network when the optimization process stagnated.

Li and Padula [36] and Queipo et al. [23] gave comprehensive reviews of various surrogate models used in the aerospace industry. Jin et al. [39] compared various surrogate models based on several defined criteria, including the prediction accuracy, efficiency, and robustness. They concluded that the polynomial response surfaces can be used to approximate lower-order nonlinear problems, while the Kriging models should be used for lower-order nonlinear problems in high dimensional design spaces and neural network should be used for higher-order nonlinear problems.

Zerpa et al. [37] assembled various surrogate models to construct a weighted average surrogate model. Goel et al. [38] then extended the utility of the ensemble of surrogates with weights associated with each surrogate model based on the individual error. They demonstrated that the best surrogate can change when the DOE method is changed while the weighted average surrogate model showed relatively low sensitivity to the choice of the DOE. Shahpar [160] used the Midrange Approximation Method (MAM) to solve large design optimization problems with the approach based on the assembly of multiple meta-models into one model using linear regression. The multiple surrogate model in general outperformed the single surrogate model in terms of the errors with much more robustness. Various multiple surrogate models have been used for turbomachinery optimization in the literature [40,160,162].

3.3.4. Gradient-enhanced kriging

It is straightforward that the computation cost can be reduced with

the gradient information. There are several ways of implementing Gradient-Enhanced Kriging (GEK). Fig. 7 shows the prediction comparison of the Rosenbrock function based on Kriging model and GEK model and it is apparent that GEK outperforms Kriging for the same number of samples. The indirect GEK uses the gradient information to provide additional data to the observation results with defining a small but finite step size [194]. However, the direct GEK modifies the prior covariance or changing the prior covariance matrix with appending the partial derivatives to observation results [195,196]. The main advantages of direct GEK over indirect GEK are: 1) direct GEK doesn't need choose the step size; 2) the observation uncertainties can be included for direct GEK; 3) direct GEK is more robust to poor conditioning of the matrix.

The GEK model based on adjoint CFD solver was used to optimize a counter rotating turbofan in Ref. [204]. The results showed that a considerable improvement of the fitness function approximation was taken into account when the sensitivity information was taken into account. A design with higher isentropic efficiency at the aerodynamic design point was created after that. The GEK has also been developed to solve the 2D airfoil drag minimization problems by Yamazaki et al. [205].

3.3.5. Support vector machine

Support vector machines (SVMs) are widely used in machine learning to analyze data used for classification and regression analysis. In the example of point classification in space, SVM model separated the categories by a clear gap that is as wide as possible. SVM can efficiently perform a non-linear classification using the kernel trick which implicitly maps the inputs into high-dimensional feature spaces. Normally the separation is easier in higher-dimensional space. The hyperplane in the higher-dimensional space is defined as the set of points whose dot product with a vector in that space is constant. With the choice of the hyperplane, the points x are mapped into the hyperplane by the relation:

$$\sum_i a_i k(x_i, x) = constant \tag{19}$$

where $k(x_i, x)$ is a kernel function, a_i represents the selected parameter and x_i is the test points. The $k(x_i, x)$ measures the relative nearness of test point x and the corresponding data base point x_i and $k(x_i, x)$ increases when x moves near to the point x_i. Thus, SVM can easily discriminate the categories.

Generally SVMs are helpful in text categorization, image classification and biological sciences. A version of SVM for regression was proposed by Drucker et al. [198] in 1996. The support vector regression (SVR) depends on a subset of the sample points due to the fact that the cost function doesn't care about the points that lie beyond the margin. Another SVM model based on least squares method has been proposed by Suykens and Vandewalle [199]. Within the research field relative to turbomachinery, SVM was mainly applied for equipment fault detection or diagnosis [200,201].

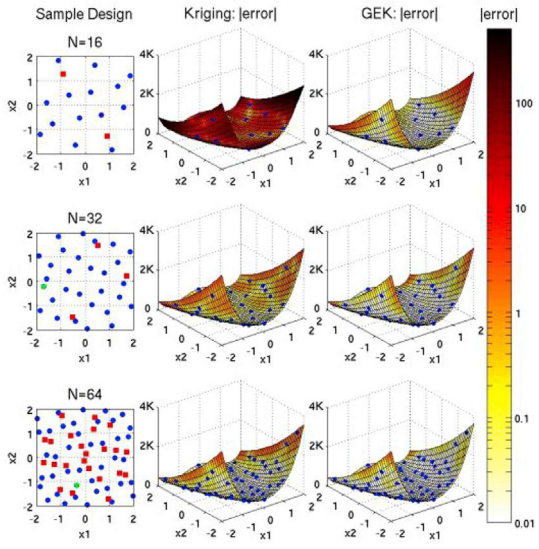

Fig. 7. Prediction comparison of the Rosenbrock function based on Kriging model and GEK model [197].

3.4. Stochastic optimization method

3.4.1. Particle swarm optimization

The urgent need for search methods capable of escaping local optima has led to the development of non-traditional search algorithms. Traditional optimization methods iteratively search for the optimum on the gradient information as will be discussed in Section 4. However, the final solutions can depend on the initial solution and these methods easily get stuck in a local optimum. More robust methods can be developed by introducing probabilistic processes into the traditional algorithms. These new methods are stochastic and iterative and are based on the individual solutions being updated simultaneously. Simulated Annealing (SA), Particle Swarm Optimization (PSO), GAs and EAs are typical stochastic optimization methods.

In computer science, PSO is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It solves a problem by having a population of particles, and moving these particles around in the search-space according to simple mathematical formula over the particle's position and velocity. Each particle's movement is influenced by its local best known positon, but is also guided toward the best known positions in the search-space, which are updated as better positions. As a result, it is expected to move the swarm toward the best solutions.

In PSO, a potential solution is called a particle. Each particle has two representative parameters, i.e., the current position x_i and the current velocity v_i. As for each particle, there exist the best position P_{best} itself and the best position for the whole swarm that has experienced S_{best}. During each generation, the velocity and position of each particle is updated using the following formulas:

$$v_i(t+1) = wv_i(t) + c_1 r_1 (P_{best}(t) - x_i(t)) + c_2 r_2 (S_{best}(t) - x_i(t)) \tag{20}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{21}$$

The PSO algorithm was firstly introduced by Kennedy and Eberhart [165]. The motivation of this achievement was to simulate the simplified animal social behaviors such as the bird flocking. Generally it is believed that PSO has high efficiency of convergence and is one of the most promising algorithms in the global optimization methods. Therefore, great effort has been employed on PSO in the optimization application [166–169].

However, the standard PSO always traps into local optimal solution or converges to precocity. Some algorithms have been devised to make the main coefficients become more suitable to the optimization issue, for example, algorithm with linear inertia weight (PSO-CIV) [170], method with dynamic inertia weight and elite velocity (PSO-DIV) [171], algorithm with convergence coefficients (PSO-C) [172] and PSO with particle generator (PSO-PG) [173]. In addition, some researchers focus on the combination of different intelligent optimization algorithms into PSO to improve results quality such as SA-PSO [174], GA-PSO [175] and Ant Colony Optimization (ACO)-PSO [176].

Recently, PSO has been gradually introduced into the optimization field of turbomachinery due to its more efficient computation cost than EA. Safari et al. [177] proposed the metamodel guided particle swarm optimization (MGPSO) algorithm in which a major modification to an original PSO was made by using all previously evaluated points aiming to increase the computational efficiency. The developed algorithm was then used to optimize the aerodynamic design of a gas turbine compressor

blade airfoil and the results illustrated the MGPSO's capability to achieve more accurate results with a considerably smaller number of function evaluations. Duan et al. [178] utilized PSO to optimize the NASA Rotor 37 and the peak efficiency was improved by 1.36% with pressure ratio enhanced by 0.26 points. In Ref. [179], Bahrani used PSO to maximize the isentropic efficiency of NASA Rotor 67, almost improved by 1.9 points while the maximum strain of the blade did not exceed the allowed strain. An efficient automated optimization of radial turbine meridional profiles using standard PSO algorithm has been presented by Tsalicoglou and Phillipsen [180]. Song et al. [181] improved the PSO algorithm based on adaptive control of particles, parameters and population diversity to optimize a large-turning tandem blade. Optimization results illustrated that the improved PSO method obviously reduced the optimization cost with the total pressure loss decreased by 40.4%.

Generally PSO is easier to implement and shows a faster convergence speed than traditional EAs. However, as mentioned by Ratnaweera [182], the major drawback of PSO is its convergence capability to local optima. Thus, PSO has been improved with the following four aspects:

1) The modification of population diversity. The main factor that makes PSO converge to local optima is the lack of the population diversity. Diversity control can be accomplished by preventing too many particles from gathering in a region with adaptively choosing neighborhood [183,184].
2) Improvement of population topology. The population topology significantly determines the way to share information among particles. Both static and dynamic topology techniques were proposed by researchers. PSO with ring neighborhood and Von Neuman neighborhood was tested in Ref. [185]. Suganthan [186] proposed PSO with dynamic topology, where the topology begins with ring neighborhood and the neighborhood number increased until the model is reached.
3) Hybrid PSO. As mentioned above, some researchers focus on the combination of different intelligent optimization algorithms into PSO to improve results quality such as PSO with differential evolution (PSO-DE) [187], SA-PSO [174], GA-PSO [175] and ACO-PSO [176].
4) PSO with parameter control. Main coefficients become more suitable to the optimization issue, for example, algorithm with linear inertia weight (PSO-CIV) [170], method with dynamic inertia weight and elite velocity (PSO-DIV) [171], algorithm with convergence coefficients (PSO-C) [172] and PSO with particle generator (PSO-PG) [173].

3.4.2. Simulated annealing

The simulated annealing (SA) search uses a probabilistic rule for accepting a new current best solution. The probability of acceptance of a worse solution is proportional to the difference in the fitness or cost between the current best solution and the new competitor normalized through a parameter called the temperature T which gradually decreases during the process. SA is able to escape from local optimums by accepting inferior solutions [72]. The term “annealing” refers to the process in which a solid material is first heated and then allowed to cool by slowly reducing the temperature. When the solid part is cooled too quickly, it will not reach the global minimum state of its potential energy function. In nature, the energy states of a system follow the so-called Boltzman probability distribution. The basic idea is that a system in thermal equilibrium at temperature T has its energy probability distribution among various energy states. The simulation of the annealing is regarded as an approach that found out a minimization of a function of large number of variables to the statistical mechanics of equilibration of the mathematically equivalent artificial multiatomic system.

SA was invented by Kirckpatrick [188] who generalized the Metropolis Monte Carlo integration algorithm in order to handle non-convex cost functions arising in a variety of problems, e.g. finding the optimal wiring for a densely wired computer chip. Then SA has been widely used in the optimization applications for turbomachinery. Tong and Feng [189] used multi-objective optimizations for the transonic turbine cascades with SA method. Tiow et al. [190] explored to seek the optimal target distribution of turbomachinery cascades by using SA with inverse design methodology. Pierret and Braembussche [191] described a knowledge method for the automatic design of more efficient turbine blades by means of SA and the results showed that both the aerodynamic and mechanical constraints were satisfied with higher efficiency. Other applications of SA algorithm for the optimization of turbomachinery were also shown in Refs. [79,91]. Some comprehensive overviews about SA were presented in Refs. [192,193].

Totally, SA is a randomly search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure and the search for a minimum in a more general system. It has been proved that by carefully controlling the rate of cooling of the temperature, SA can finally find the global optimum. The following elements must be provided:

1) a representation of possible solutions
2) a generator of random changes in solutions
3) a means of evaluating the problem functions
4) an initial temperature and rules for lowering it as the search progress

The strengths of SA are that:

1) SA is a robust and general technique due to its ability to deal with highly nonlinear models with chaotic and noisy constrains
2) It's easy to implement because of the flexibility for different nonlinear optimization issues
3) It's amenable to parallel computation

However, the weakness of SA is as following:

1) There is a clear tradeoff between the quality of the solutions and the time required to compute them
2) The tailoring work to account for different constraints and parameters makes the algorithm much more delicate
3) The precision of the coefficient chosen in the implementation shows a significant effect upon the final optimization results

3.4.3. Genetic algorithms

The GA was designed by Holland in the 70s and improved and made famous by Goldberg [85]. The GA imitates natural processes of the evolution of genes in a stochastic search for the optimal values, which makes it different from other methods, such as the gradient-based methods. The populations are encoded as binary codes, like chromosomes, in which each bit is called a gene and each population represents a set of solutions to the problem. The offspring are generated through the crossover, mutation and selection of chromosomes. The breeding process is repeated iteratively until converging to a set of solutions, which are the optimal results for the problem. The attraction of the GA is its simple algorithm.

The GA calculation has two main parts. One is the genetic operation involving chromosome crossover and mutation, while the other part is the evolution or reproduction selection using genetic operations that imitate genetic inheritance to create a new generation called the offspring. The evolution operation comes from Darwinian evolutionism where a new generation is selected based on the fitness of the offspring. In this work, the GA operations include selection, crossover and mutation with the elitist strategy always used.

Selection imitates the creating of the next generations and the fitness represents the weightings occupied by the population. The weightings calculation is based on the Darwinian principle of reproduction and survival of the fittest. An individual is probabilistically selected from the population on the basis of its fitness and then the individual is copied, without change, into the next generation of the population. The selection is done in such a way that the better fitness is more likely to be selected.

Crossover is the main procedure for chromosomal exchange in the GA because the process relies on chromosomal crossover to achieve an effective knowledge database. The crossover process combines the characteristics of two chromosomes to generate an offspring. The number of populations is determined by the crossover rate, which represents the crossover probability of the population in each generation. When the crossover rate is set too high, the possible search space becomes very broad with time-consuming searches in suboptimal solution spaces. Fig. 8 shows an example of crossover where the information in the red chromosomal is exchanged with that in the blue one.

Mutation plays a secondary role in the GA that complements the crossover and selection procedures. When the population converges to a local optimum, only mutations can give a chance to find a more optimal solution. The determination of whether a mutation will occur is decided by the mutation rate. If this rate is set too low, the potential optimal population may be missed and only a suboptimal solution will be found. However, if the rate is set too high, the historical inheritance from the parents will disappear. Fig. 9 shows a typical GA optimization flowchart.

The GA population optimization method has many advantages for multi-objective optimization problems. The multiple objective functions are initially the sums of the weighting factors that finally form a single objective function. Since this gives the optimal result, the designers have no alternative optimization options since only one "best" solution is obtained by the optimization which differs from normal engineering practice. Moreover, the weights are difficult to set for different objectives due to the lack of knowledge about the inner relations.

Multi-objective genetic algorithms based on the Pareto optimal concept have also been used for optimization. They provide a set of non-inferior solutions rather than one "best" solution. A multi-objective genetic algorithm (MOGA) is preferred and an approximation model is needed for turbomachinery aerodynamic optimization due to limited computational resources. The Non-dominated sorting genetic algorithm II (NSGA-II) has also been widely used for turbomachinery optimization. The well-known NSGA-II was proposed by Deb et al. [80] is now one of most widely used MOGAs since it provides excellent results compared with other multi-objective genetic algorithms [82,83]. The individuals in the first few generations are evaluated by CFD simulations to provide the initial approximation of the Pareto-optimal front. Then, an ANN trained by the existing points is used to predict the objective in the remaining generations of each segment. The same sequence is then implemented in the following segments. The locations of the training samples in the segments approach the Pareto-optimal front and the neural network approximation accuracy is improved. This coarse-to-fine strategy will lead to the Pareto-optimal front.

GAs have been widely used to optimize turbomachinery designs [13,63,77,84,87–89,92–94]. Demeulenaere and Hirsch [61] presented a methodology to evaluate successive designs using an ANN instead of a flow solver. The ANN accuracy was based on design examples in a database. The commercial optimization framework Fine/Design 3D also uses this idea [86]. The advantages of GAs are:

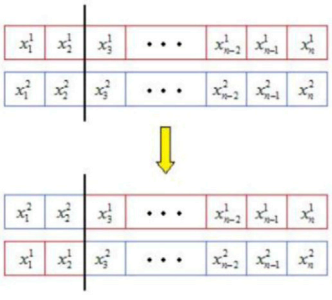

Fig. 8. Example of chromosomal exchange [18].

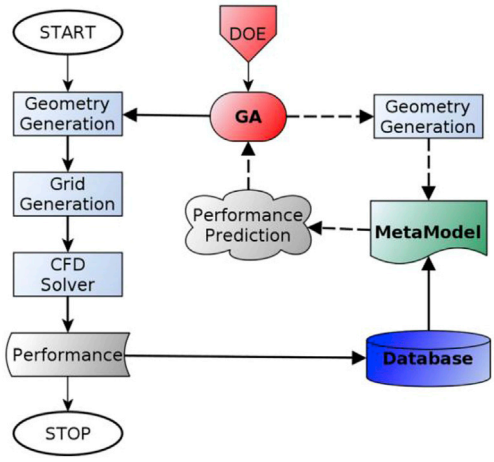

Fig. 9. GA optimization framework.

1) GAs are robust and can capture the global optimal solution without being trapped to local optima.
2) GAs can easily be used to build a multi-objective optimization framework using parallel computations.
3) GAs do not need gradient information, so they are independent of the objective function.

3.4.4. *Evolutionary algorithms*

Evolutionary Algorithms (EAs) are mainly based on a GA, Evolution Strategy (ES) and Evolutionary Programming with GA and ES s the two most widely. EAs mimic the mechanics of natural selection and natural genetics with a biological population evolving over generations to adapt to an environment. EAs start with a random population of candidates (chromosomes) with both the objective and constraint functions evaluated for all of them. A metric (fitness) is assigned to each candidate based on the objective function and constraint violations. A penalty is added to infeasible candidates so that all infeasible solutions have a worse fitness than the feasible solutions.

Typically, EAs involve the three operators for selection, crossover, and mutation, similar to the GAs. The primary purpose of the selection operator is to make duplicates of good candidates and eliminate bad candidates in a population while normally keeping the population size constant [80] through tournament selection, proportionate selection, and ranking selection. For single objective optimization problems, the ranking is based on the candidate fitness. For multi-objective optimization problems, the ranking can be based on Fonseca's non-dominated ranking method [95] in which an individual's rank is equal to the number of individuals in the present generation who are better than the corresponding individual in all the objective functions. After ranking, the N best candidates, which is the same size as the initial population, are chosen from both the current and previous generations and then placed in the mating pool. The elitist strategy [96] is often used to ensure a monotonic improvement in the EA, in which some of the best individuals are copied directly into the next generation without applying any evolutionary operators. Various EAs have been used to solve multi-objective optimization problems, for example, Fonseca and Fleming's MOGA [95], Srinivas and Deb's NSGA [97], Deb et al.'s NSGA-II [80,81] and Horn et al.'s NPGA [98]. There are several excellent review papers [99,100,101,105,158,159] about EAs for multi-objective optimization problems.

EAs have been successfully applied to aerodynamic design optimization problems for turbomachinery because of their ease of use, broad applicability, and global perspective. For example, Oyama et al. [102] used an EA for their redesign of the NASA Rotor67 transonic compressor blade. Benini [83] used an EA to improve the performance of the NASA rotor37 blade. Oyama and Liou [103] and Lian et al. [55,104] utilized

EAs to redesign rocket turbo pumps. Buche et al. [71] built an automated, multi-disciplinary optimization procedure for the design of subsonic gas turbine compressor blades based on the EA. Dennis et al. [106] optimized an airfoil cascade row to simultaneously minimize the total pressure loss, maximize the total aerodynamic loading, and minimize the number of airfoils using an EA. Another reason for these many applications is that EAs are particularly suitable for multi-objective optimization problems, which are often encountered in aerospace designs. For example, in the turbo pump design problem, the objective was to maximize the total head rise and to minimize the input power. Unlike a single objective optimization problem, a multi-objective optimization problem does not have such an optimal solution that is better than others for all the objectives. Instead, one expects a set of compromise solutions, each of which is better than the others in one objective but is worse in the other objectives.

These solutions are largely known as non-dominated solutions or Pareto-optimal solutions. When dealing with multi-objective optimization problems, classical methods, such as the gradient-based methods, usually convert the multi-objective problem into multiple single-objective problems by introducing parameters such as weight vectors. In such approaches, each optimal solution is associated with a particular vector. To find another Pareto-optimal solution, one has to choose a different weight vector and again solve the resulting single objective optimization problem. However, the EA's population approach can be used to equally emphasize all non-dominated solutions in a population and to preserve a diverse set of non-dominated solutions using a niche-preserving operator. Consequently, EAs eliminate the need for choosing different parameters and can find as many Pareto-optimal solutions as possible in one run.

Although the EAs are powerful optimization tools, they suffer from slow convergence speed due to the lack of the gradient information. As a result, the whole process usually needs a tremendous amount of computing resources. To address this issue, researchers [104,107,108] have proposed hybridized stochastic EAs that are combined with deterministic gradient-based methods. The idea is to resort to a gradient-based method whenever the EA convergence rate slows. This strategy takes advantage of the fast convergence of the gradient-based methods. Fig. 10 shows the time cost advantage of the hybrid method for high-dimensional problems.

3.5. Remarks

As an important part of the optimization process, the geometric parameterization defines the design variables linked directly to the constraints, while eliminating unimportant ones. Turbomachinery optimization is always based on a multi-section blade parameterization based on Bezier, B-spine or NURBS curves. The recent developed fine-tuned designs capture a high-fidelity model of the geometry with a limited amount of design variables, so these methods deserve further study.

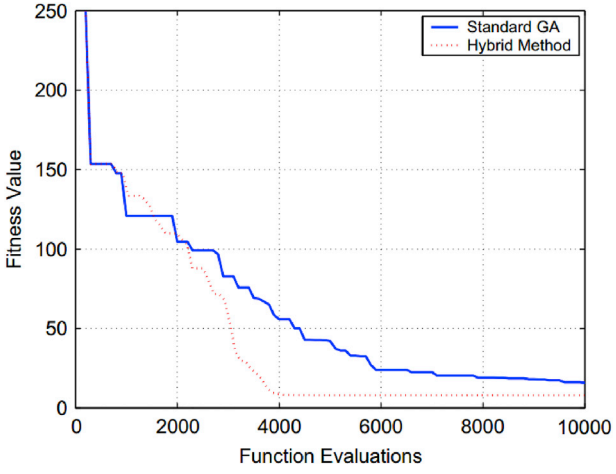

Fig. 10. Comparison of a standard GA with a hybrid method in terms of the number of function evaluations [104].

The DOE method explores the design space by sampling to provide the maximum amount of information with the minimum number of design experiments to reduce the computational costs. Various experimental design methods have been proposed, for example, the CCD, D-optimal, and LHD methods. However, for computation design experiments, in which the systematic error is larger than the random error, the CCD and D-optimality designs are inefficient or even inappropriate. LHD is a popular modern DOE method for turbomachinery optimization that uses space-filling designs based on stratified sampling rather than concentrating the designs on the boundary, thus ensuring that each input variable is represented over all of their range while still being less computationally intensive. More “intelligent” sampling schemes using iterative, adaptive and progressive sampling processes should be developed to further advance these sampling techniques.

Despite continual advances in computing power, the costs of the computationally intensive optimization processes are still significant. In the past three decades, surrogate models and approximation-based optimization methods have attracted much attention. Four types of surrogate models are widely used for the global optimization of turbomachinery flows, the polynomial RSM, artificial neural networks (BPNN and RBNN) and the Kriging method. There is no agreement on which model is superior. Some studies [20,24,28,36,37,39,40,43,127] have shown that polynomial response surfaces are better for approximations of lower-order nonlinear problems, while Kriging models are better for lower-order nonlinear problems in high dimensional design spaces and neural network are better for higher-order nonlinear problems. Among these models, the Kriging model and RBNN give the best approximations of the exact functions. In addition, unlike neural networks, Kriging models are directly coupled with the database. In addition, neural networks are rather complex due to their large number of training weights. In addition, the multiple surrogate model, which is constructed by weighting various average surrogate models, in general give smaller errors than the surrogate model with much more robustness [38,39,41]. Further research is needed to show how to build surrogate models for large problems, how to develop more flexible and generic metamodeling approaches and how to eliminate the uncertainty in surrogate models.

SA, EAs and GAs are both widely used for turbomachinery optimization. The first generation of multi-objective EAs/GAs included the Non-dominated Sorting Genetic Algorithms (NSGA), the Niches-Pareto Genetic Algorithm (NPGA), and the Multi-Objective Genetic Algorithm (MOGA). The second generation included the Strength Pareto Evolutionary Algorithm (SPEA), the Strength Pareto Evolutionary Algorithm 2 (SPEA2), the Pareto Archived Evolution Strategy (PAES) and the Non-dominated Sorting Genetic Algorithm II (NSGA-II). Further research should focus on developing hybridized optimization algorithms [101], self-adaptation parameter control [128], reducing the number of needed fitness function evaluations [129], developing methods that are independent of the platform and programming language [130], developing fast methods for a large number of objectives [131], and uncertainty quantification [132]. Recently PSO has been gradually introduced in the optimization field of turbomachinery due to its faster convergence speed and resulting relatively lower computation cost than EAs. However, the major drawback of PSO is easily convergence capability to local optima. Advanced PSOs should be improved based on adaptive control of particles, parameters, population diversity or hybrid combination with other optimization algorithms.

4. Optimization using gradient-based methods

The gradient-based methods include the finite difference method, the linearized method, and the adjoint method depending on how the

gradients are calculated. An essential part of the gradient-based optimization methods is to have a fast, accurate way to calculate the gradient information because this is the most time-consuming part of the whole design process. The traditional gradient-based method depends on the step size and the time cost is usually proportional to the number of design variables; thus, this method is not good for a large number of design variables. The adjoint method does not have this well-known disadvantage of the traditional method but can quickly calculate the derivatives of the objective function with respect to the design variables independent of the number of design variables.

4.1. Traditional gradient-based method

A typical constrained minimization or maximization problem entails a group of physical quantities that are the design variables and another group of constant quantities called problem parameters. In aerodynamic applications, the design variables and the problem parameters are related to the geometry and the flow field. In most optimization procedures, the dominant contributor to the computational cost is the calculation of the derivatives of the objective function and the constraints with respect to the design variables. These derivatives are called the sensitivity coefficients [109]. Therefore, any optimization procedure must have an efficient numerical or analytical method to determine the sensitivity coefficients and efficient computational methods to solve the resulting equations. The traditional gradient-based methods were based on the finite difference approach and the linearized method [133].

The straightforward idea is to calculate the derivative as a finite difference approximation. For example,

$$\frac{dF}{dx} = \frac{F(x+h) - F(x)}{h} \tag{22}$$

An obvious shortcoming of this idea is the uncertainty in the choice of the perturbation step size, h. Also, this approach necessitates solving for the flow field for each perturbed design variable which is particularly expensive, especially for a large number of design variables and many flow fields.

4.2. Adjoint method

Assume that the object function, I, in an aerodynamic design optimization problem is a function of the flow variable vector, U, and a design variable, α, as:

$$I = I(U, \alpha) \tag{23}$$

Then, the relationship between the flow variable and the design variable is determined through the flow equation,

$$R(U, \alpha) = 0 \tag{24}$$

The gradient of the objective function relative to the design variable is

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} \tag{25}$$

where $\frac{\partial I}{\partial \alpha}$ and $\frac{\partial I}{\partial U}$ are to be calculated. However, the calculation of the flow variable sensitivity, $\frac{\partial U}{\partial \alpha}$, involves solving the flow equations,

$$\frac{\partial R}{\partial \alpha} + \frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha} = 0 \tag{26}$$

This linearized equation also depends on the design variable, which means that each design variable requires one solution of the flow equation. The key is to find a way to decouple the influence of the design variables on the objective function by means of the flow sensitivity, i.e. eliminating the explicit dependency of the objective function sensitivity on the flow variable sensitivity, $\frac{\partial U}{\partial \alpha}$.

To achieve this goal, the right side of the flow equation is multiplied by the adjoint factor, λ, and the product is subtracted from the gradient expression as:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} + \frac{\partial I}{\partial U}\frac{\partial U}{\partial \alpha} - \lambda^T\left[\frac{\partial R}{\partial \alpha} + \frac{\partial R}{\partial U}\frac{\partial U}{\partial \alpha}\right] \tag{27}$$

This expression can be regrouped as:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} - \lambda^T\frac{\partial R}{\partial \alpha} + \left[\frac{\partial I}{\partial U} - \lambda^T\frac{\partial R}{\partial U}\right]\frac{\partial U}{\partial \alpha} \tag{28}$$

If the adjoint equation is zero:

$$\frac{\partial I}{\partial U} - \lambda^T\frac{\partial R}{\partial U} = 0 \tag{29}$$

Then the gradient is given by:

$$\frac{dI}{d\alpha} = \frac{\partial I}{\partial \alpha} - \lambda^T\frac{\partial R}{\partial \alpha} \tag{30}$$

This expression no longer depends on the flow variable sensitivity. Furthermore, the adjoint equation (26) does not depend on any design variable. This implies that the gradient of a scalar objective function relative to all the design variables can be obtained by solving only two sets of equations, the flow equation in Eq. (21) and the adjoint equation in Eq. (26). Once the flow and adjoint solutions are obtained, these can be substituted into the new gradient expression in Eq. (27) to efficiently calculate the gradients.

The adjoint algorithm for aerodynamic design optimization was pioneered by Jameson [134]. The two main types of adjoint methods are the continuous adjoint method and the discrete adjoint method, based on the way the adjoint system is formed. For the continuous method, the nonlinear flow equations are first linearized with respect to one design variable. Then, the adjoint equations are derived from the linearized flow equations. However, for the discrete adjoint method, the flow equations are first discretized, followed by the linearization and adjoint formulation. Fig. 11 shows the difference between the discrete adjoint method and the continuous method.

If all of the solutions are sufficiently smooth, all the approaches should be consistent and converge to the correct gradient of the objective function [135]. However, there are important conceptual differences between the different approaches. The advantages and disadvantages of the two methods are as follows:

1) The discrete adjoint method provides the exact gradient of the objective function. Furthermore, the programming accuracy can be easily checked.
2) The discrete adjoint program is quite straightforward.
3) The physical meaning of the continuous adjoint method is clearer.
4) The continuous adjoint method requires less memory and the program structure is simpler.

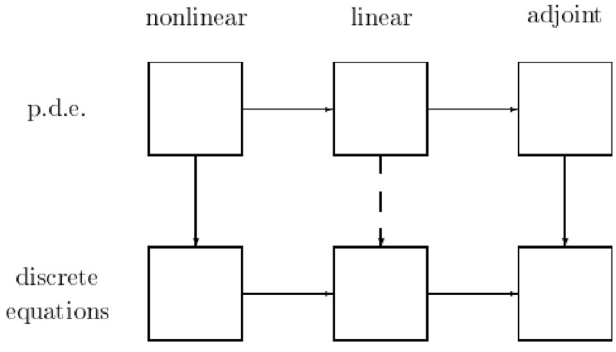

Fig. 11. Alternative approaches to forming the discrete adjoint equations [135].

There have been increasing efforts to apply the adjoint method to turbomachinery blading aerodynamics. Fig. 12 shows a flow chart of the optimization process based on the adjoint method. Arens et al. [116] optimized the turbine blades by using adjoint approach and the 2D Euler equations. Li et al. [117] combined the continuous adjoint method with the quasi-Newton method to optimize the 2D aerodynamic shape design of a turbine blade. The continuous adjoint method was also used for the aerodynamic design of cascades in a 2D inviscid, compressible flow [118]. Wu et al. [119] used the adjoint method to minimize the entropy generation of a 3D turbine stator blade. Papadimitriou and Giannakoglou [124] used the continuous adjoint method to improve the aerodynamic performance of a 3D peripheral compressor blade cascade. Flora and Hall [120] used a discrete adjoint solver to calculate the sensitivity of an unsteady inviscid flow in a turbomachine. Thomas et al. [121] presented a discrete adjoint approach for computing steady and unsteady aerodynamic design sensitivities for compressible viscous flows with the aid of the advanced automatic differentiation software tool known as Transformation of Algorithms in Fortran (TAF). Luo et al. [112] used the adjoint method to optimize a turbine blade, including the imposition of the appropriate boundary conditions for the adjoint Euler equations. Li et al. [122] used the adjoint method and an N-S solver to optimize turbine blades in a 2D viscous flow. Walther and Nadarajah [123] described a fully automatic gradient-based aerodynamic shape optimization method for a multi-row turbine including the derivation of flow-consistent adjoint boundary conditions, a discrete adjoint mixing plane formulation and an automatic grid perturbation scheme using RBFs. Luo et al. [145] used a viscous continuous adjoint method to optimize a low-aspect-ratio turbine blade row through endwall contouring. Lei and Jiang [146] applied the continuous adjoint method to optimize the aerodynamic shape of high pressure turbine blades based on the S2 surface and the Euler governing equations. Zamboni et al. [163] used an adjoint RANS solver to study the effect of geometric non-conformance on the efficiency and flow capacity of turbine blades.

Although there have been many studies using adjoint methods, they mainly focused on the optimization of a single row. Methods are also needed to apply the adjoint method to multi-stage machines with information transfer through the rotor-stator interface. Wang and He [110,[111] proposed an adjoint mixing-plane treatment that can be used for aerodynamic shape design optimization of multiple turbomachinery. The gradient results given by the mixing-plane treatment compare well with finite difference results demonstrating the validity of the adjoint mixing-plane treatment and the benefit of using it in a multi-blade row environment. This method was also used by Jing et al. [114]. To further increase the aerothermal and aero-elastic performance in modern blading designs, He and Wang [113] proposed a discrete adjoint method for concurrent blading aerodynamic and aero-elastic design optimizations in which a nonlinear harmonic phase solution method is used to solve the unsteady RANS equations. Alternatively, the unsteady adjoint-based optimization method can be used to reduce the effect of the interface for multistage configurations, [115].

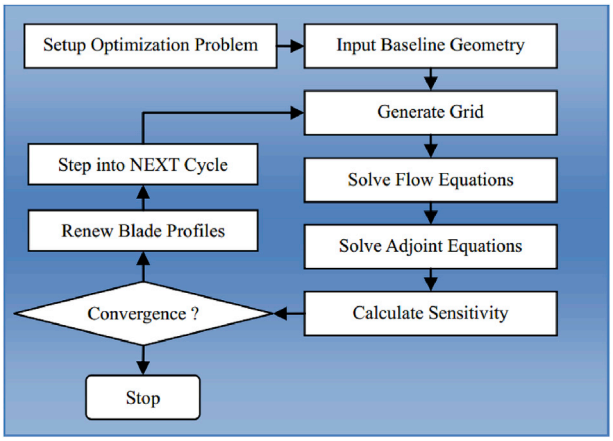

Fig. 12. Flow chart for adjoint-based optimization [136].

4.3. Remarks

Over the past two decades, the adjoint approach has been extensively used for external flow problems. However, due to the complexity of formulating the adjoint equations and the challenge of defining the adjoint boundary conditions, the adjoint method has not been often applied to internal flows. There is currently much research on applying the adjoint approach to gradient-based optimization of turbomachinery. Most of these publications described the optimization of only isolated blade row configurations. The designs produced by concurrent optimization are generally better than those based on isolated optimizations. The difficulties are the treatment of the boundaries and the rotor-stator fluxes. Conservative adjoint mixing planes have been proposed to address this issue by many researchers [110,111,123,137–139]. Further research is also needed on the effects of uncertainties on the turbomachinery performance. Some recent studies have analyzed Uncertainty Quantification (UQ) and robust optimization techniques [140,141,142].

Further research is also needed on adjoint methods for multiple objectives and unsteady optimization problems. Due to the low computational costs of multi-variable optimization problems, the adjoint method should be used for concurrent multi-disciplinary unsteady optimization problems, such as aero-thermal, aero-elastic, or aero-acoustic problems The adjoint method can also be used to refine the mesh topology to improve the accuracy of numerical simulations [143,144]. Essentially, the adjoint method is based on the gradient information and will eventually approach the local optimum. Thus, another important issue is how to couple global optimization with the adjoint method. Recently, Tang et al. [164] showed that the gradient-enhanced response surface model is especially advantageous for finding the global optimum for multi-parameter design optimization cases where the adjoint gradient information is introduced in the construction of the response surface.

5. Data mining

A multi-objective optimization problem can be converted into a single-objective problem by introducing weight factors to get one optimum solution. However, in practical applications, the weight factors are difficult to specify for different objects with variable dimensions. Therefore, real-world design processes do not obtain exactly optimal solutions. For instance, the peak efficiency of a compressor configuration can be found that eliminates the losses by changing the blade shape in the optimization algorithm. However, these changes will also modify the stable operating range and the mechanical strength which should also be taken into account.

The Pareto-optimal solutions of a multi-objective problem rather than a single result of a converted single-objective optimization problem can provide useful information such as which parameters are dependent or independent, which design parameters are more sensitive to the final result, which objective functions are independent or correlative, and so on. Aerodynamic shape optimization usually results in hundreds or even thousands of Pareto-optimal solutions. Fig. 13 shows the design candidates and the Pareto-optimal solution of an aerodynamic transonic airfoil shape design [101]. Jeong et al. [147] proposed the “multi-objective design exploration (MODE)” concept with a multi-objective evolutionary algorithm used to find the Pareto-optimal solutions and a data mining method used to extract the design information from the Pareto-optimal solutions. This information is very useful for designers because it provides meaningful guidance for the real-world design process.

Various methods have been proposed by many researchers to understand the Pareto-optimal solutions, such as the star coordinate method [148], scatter-plot matrix [149,150], value path method [151], analysis of variance (ANOVA) [147,155,157], self-organizing map

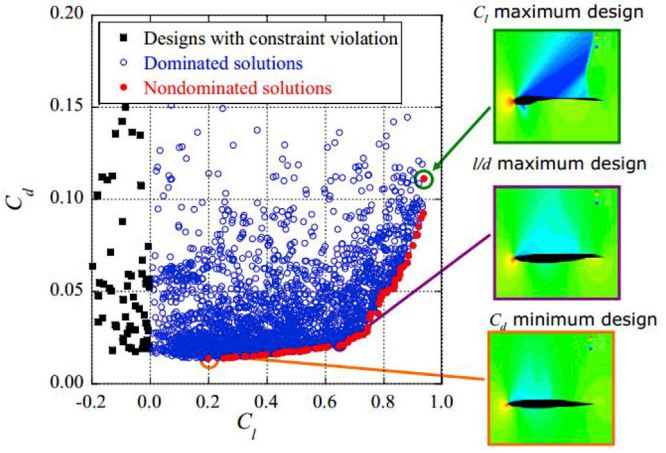

Fig. 13. Distributions of the Pareto-optimal solutions and other design candidates [101].

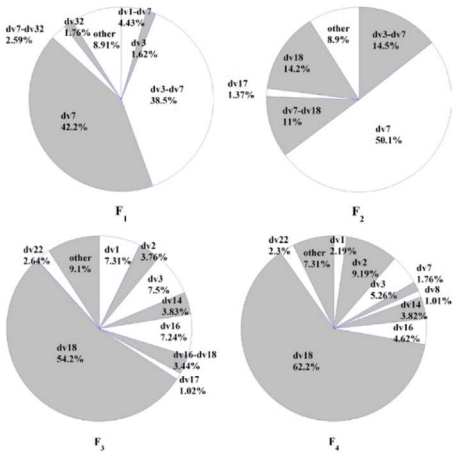

Fig. 14. ANOVA results [147].

(SOM) [147,152,155,157], fuzzy multiple discriminant analysis (FMDA) [153], and proper orthogonal decomposition (POD) [154].

5.1. ANOVA

ANOVA can identify not only the quantitative effect of the design variable but also the effect of interactions between design variables on the objective functions. The total variance of the objective results can be decomposed into the variance components due to the design variables. For example, the total mean, $\overline{u}$, and the variance of the objective functions are:

$$\overline{u} = \int \ldots \int y(x_1, \ldots\ldots, x_n) dx_1 \ldots dx_n \tag{31}$$

$$\overline{\sigma}^2 = \int \ldots \int [y(x_1, \ldots\ldots, x_n) - \overline{u}]^2 dx_1 \ldots dx_n \tag{32}$$

where $y(x_1, \ldots\ldots, x_n)$ is the objective function and x is the design variable with n samples. The main effect of a single variable, $\overline{u}(x_i)$, and the interactions variables, $\overline{u}(x_i, x_j)$, are:

$$\overline{u}(x_i) = \int \ldots \int y(x_1, \ldots\ldots, x_n) dx_1 \ldots dx_{i-1} dx_{i+1} \ldots dx_n - \overline{u} \tag{33}$$

$$\overline{u}(x_i, x_j) = \int \ldots \int y(x_1, \ldots\ldots, x_n) dx_1 \ldots dx_{i-1} dx_{i+1} \ldots dx_{j-1} dx_{j+1} \ldots dx_n - \overline{u}(x_i) - \overline{u}(x_j) - \overline{u} \tag{34}$$

Thus, the variance due to design variable, x_i, is:

$$\overline{\sigma}_i^2 = \int [\overline{u}(x_i)]^2 dx_i \tag{35}$$

The proportion of the variance, $\overline{\sigma}_i^2$, due to total variance, $\overline{\sigma}^2$, is:

$$\frac{\overline{\sigma}_i^2}{\overline{\sigma}^2} = \frac{\int [\overline{u}(x_i)]^2 dx_i}{\int \ldots \int [y(x_1, \ldots\ldots, x_n) - \overline{u}]^2 dx_1 \ldots dx_n} \tag{36}$$

Fig. 14 shows the variances of the design variable and their interactions for four objective functions as pie charts. The important parameter and which parameter is not sensitive to the objective function can be easily seen.

5.2. SOM

A Self-Organizing Map (SOM) uses a projection algorithm with clustering from high to low dimensions. As a result, one can qualitatively visualize the trade-offs and the relations between the design variables and the objective functions.

SOM uses unsupervised neural networks with the weights between the input vector and the neuron array changed to show the features of the high-dimensional data on a low-dimensional map. If two samples are close in the original space, the response of two neighboring neurons in the low-dimensional space will also be close. The SOM learning algorithm begins by finding the best-matching unit. Once the best-matching unit is determined, the weights are adjusted not only for the best-matching unit but also for its neighbors. Fig. 15 shows the original topology represented by the solid line with the weight vectors shown by the black dots. The best-matching unit is closest to the input vector. The modified topology is represented by dashed lines with white dots for its weight vectors. Finally, the weight vectors become smooth globally with repeated iterations. Thus, the sequence of closely spaced vectors in the original space results in a sequence of corresponding neighboring neurons in the 2D map.

Fig. 16 shows the SOM result for four objective functions generated with 102 non-dominated solutions. The map is classified into 10 clusters based on the similarities. Fig. 16a and b have similar color patterns, which means that these two objective functions are not in a trade-off relation. However, maximum F_3 and F_4 are not located at the same location as the maximum F_1 and F_2, thus, there are trade-offs.

5.3. Scatter plot matrix

A scatter plot is a mathematical diagram in Cartesian coordinates that displays the values for a set of data as a collection of points each having the value of one variable determining the position on the horizontal axis while the other variable determines the position on the vertical axis. Fig. 17 shows the scatter plot matrix containing all the pairwise scatter plots of the design parameters and objectives in matrix format. The figure

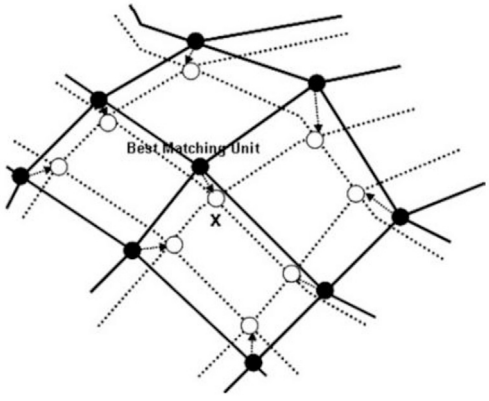

Fig. 15. Relationship between the best-matching unit and its neighbors in SOM [147].

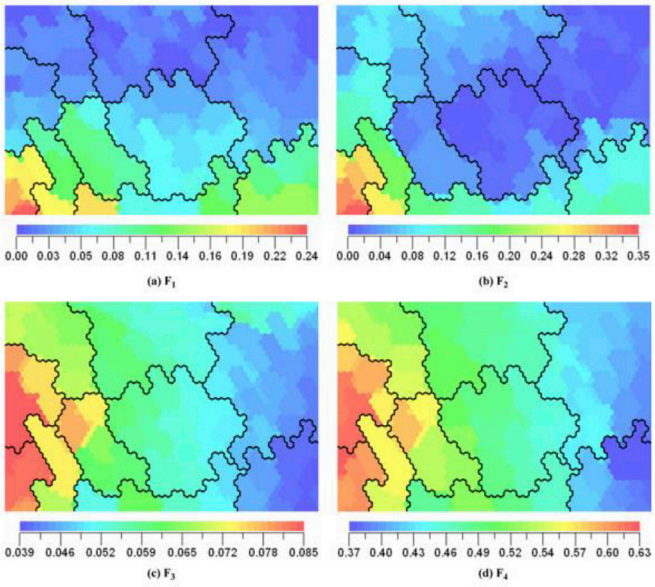

Fig. 16. SOM colored by the objective functions [147].

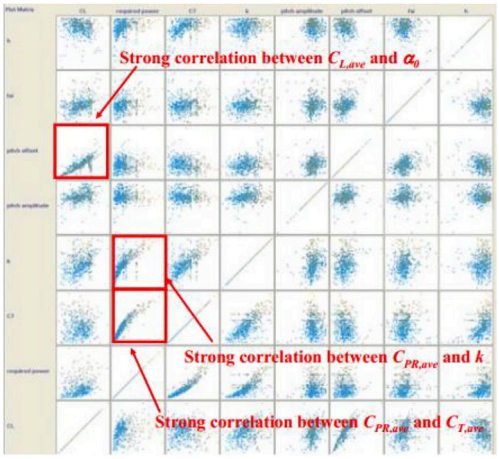

Fig. 17. Analysis using a scatter plot matrix [150].

clearly shows the strong correlation among objectives.

5.4. POD

A POD is a statistical method that can extract the dominant features out of the data by decomposing the data into a set of orthogonal base vectors with deceasing importance. This method can be used to identify the more important design parameters by solving for the eigenvectors of the principle models [154]. The data can be decomposed into a mean vector and the fluctuation vector from the mean vector to maximize the variance. The eigenvectors are determined so that the energy is maximized. Fig. 18 shows the first 10 principal orthogonal base vectors for the optimization of an airfoil shape [154]. The maximum lift-to-drag airfoil shapes for the four main orthogonal base vectors are shown in Fig. 19. The major differences occur near the leading edge, on the upper suction surfaces and on the lower pressure surface. The second base vectors show that the trailing edge thickness is reduced.

5.5. Remarks

Design exploration methods combining design optimization and data mining techniques have been used to facilitate knowledge-oriented design optimization. Many data mining methods can be utilized to identify the key information for the designers and to improve the work efficiency during the design process. The analysis of variance method can be used to determine the dominant variables and the interaction effects between the design variables, which are used in the latter data mining process. Self-organizing maps and other visualization methods can be used to find qualitative lower-order correlations, particularly trade-off relationships between the objective functions. In addition, Sugimura [156] proposed a method that uses association rules after obtaining the proper rule length by rough set theory. This method was shown to be superior to the Taguchi method for multi-objective design. Future developments in data mining techniques will generally be directed at refining the algorithms or their integration with existing methods.

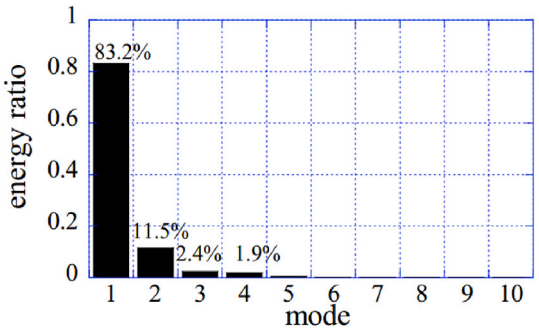

Fig. 18. Energy ratio of the top 10 principal models for an airfoil shape [154].

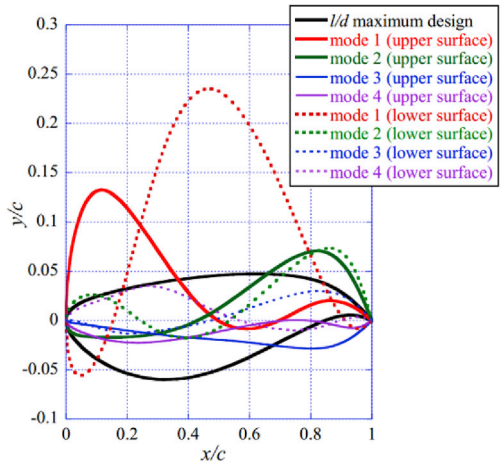

Fig. 19. Shape of the maximum lift-to-drag ratio airfoil design and the orthogonal base vectors of the first four models of the airfoil shape [154].

6. Discussion

Three actual optimization examples are shown here, the first one for the compressor design, the second one for the turbine case and the last one for the data mining technique. A comprehensive discussion about the further development of design optimization methods for the turbomachinery design is conducted in the end.

6.1. Example 1: multi-objective design of NASA Rotor 67

The NASA Rotor 67 is a transonic axial rotor which is embedded in the first stage of a two-stage fan developed by NASA. The solid picture is shown in Fig. 20. Some key parameters of this configuration are summarized in Table 1. Lian and Liou [203] presented a representative work on multi-objective and multidisciplinary design optimization for NASA

Fig. 20. Solid picture of NASA Rotor 67 [206].

Table 1
Some key design parameters of the NASA Rotor 67.

Parameter	Value
Inlet hub-tip diameter ratio	0.375
Outlet hub-tip diameter ratio	0.478
Hub solidity	3.11
Tip solidity	1.29
Tip solidity	1.3
Inlet tip diameter	514 mm
Outlet tip diameter	485 mm
Number of blade	22
Inlet Reynolds number	1.797E+06
Designed tip speed	429 m/s
Designed mass flow rate	33.25 kg/s
Designed stage pressure ratio	1.63
Designed relative inlet tip Mach number	1.38
Material	Ti-6Al-4V

Rotor 67 by using MOGA in conjunction with quadratic response surface model. The objective functions are shown as followed:

$$\text{maximize} \left(P_{outlet}/P_{inlet} \right) \tag{37}$$

$$\text{minimize} \left(W\text{blade} \right) \tag{38}$$

$$\text{subject to} \frac{|\dot{m} - \dot{m}_{baseline}|}{\dot{m}_{baseline}} \leq 0.5\% \tag{39}$$

Here the optimized blade geometry was defined by employing perturbations on baseline blade along the span (hub, 31% span, 62% span, and tip). Each 2D airfoil profile was parameterized by a mean camber line and thickness distributions with third-order B-spline curves. The camber is determined by three variables while the thickness distributions by five variables. Thus, within the optimization process, 32 design variables were selected and 1024 design points were sampled based on the LHD method. These points were evaluated by the 3D CFD solver TRAF3D and the blade weight was computed by integrating the blade volume. The response surface was then built and its accuracy was evaluated by statistical measures. The population size of the MOGA was set as 320. Finally the Pareto Fronts cases were figured out and the representative solutions were validated using the CFD solver. The convergence history was shown in Fig. 21. The convergence was improved with an increase in the generation size.

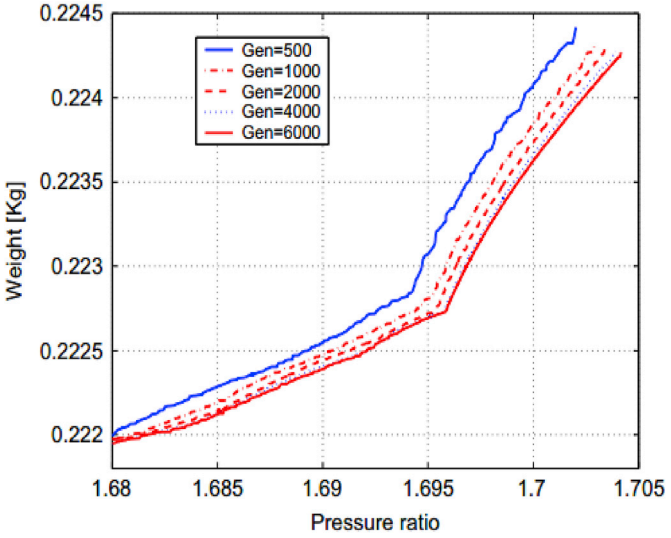

Fig. 21. Convergence history of optimization [101].

The Pareto Front results contain 473 solutions, among which 16 representative points were verified with the CFD solver. Compared to the prototype, all of the solutions increased the stage pressure ratio and reduced the blade weight. As for the optimum cases, the pressure ratio was maximally increased by 1.8% with the blade weight reduced by 5.4% to the almost maximum possible extent. The geometry comparison of the representative Pareto Front cases with the baseline blade was shown in Fig. 22. The higher pressure ratio design has a larger camber but less thickness than the prototype and the thinner airfoil leads to the blade with a lighter weight. The similar trends occurred on the 50% and 90% span. Fig. 23 shows the contours of the relative Mach number of the maximal pressure deign at the 90% span from the hub. The relative Mach number of the maximal pressure design was 1.48 and this is weaker than the Mach number of 1.52 in the baseline design, which is beneficial for decreasing the shock losses.

Meanwhile, the 3D pressure distributions on the suction surface of the rotor blade were shown in Fig. 24. The difference between the maximal pressure design and the Rotor 67 design was discussed here. A strong passage shock occurs in the upper part of the rotor. The intensity of the retrofitted passage shock was reduced in the central span part and thus moved upstream. The resulting flow separation region was reduced due to the weaker shock-boundary layer interaction which was validated in

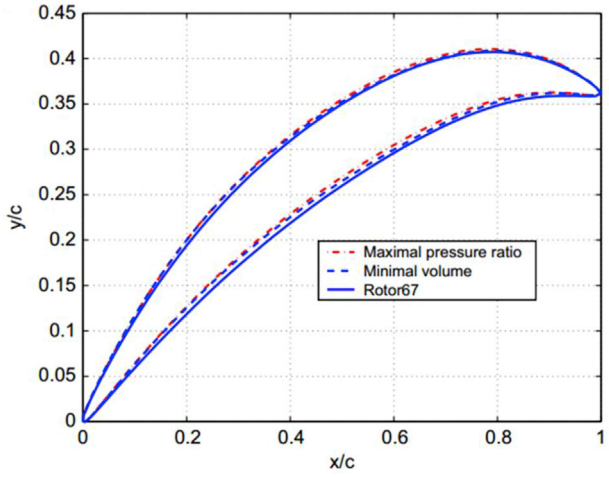

Fig. 22. Airfoil shape comparison between the Pareto Front case and the baseline at the 10% span from the hub [203].

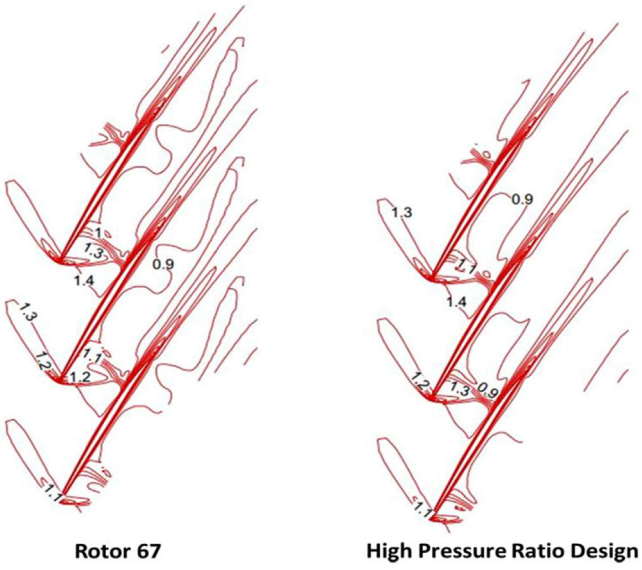

Fig. 23. Contours of the relative Mach number at the 90% span from the hub [101].

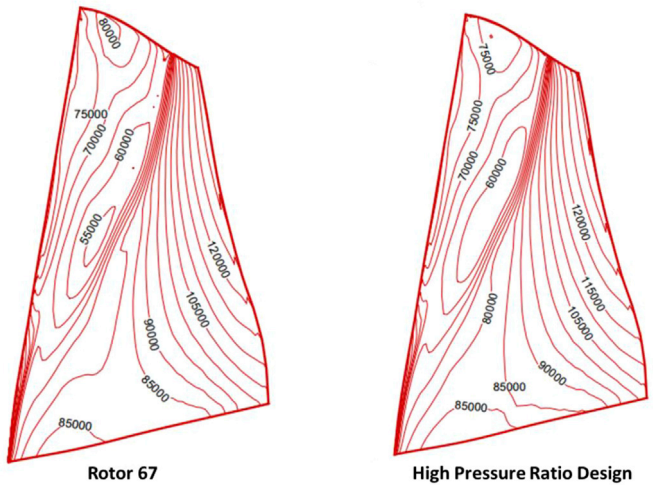

Fig. 24. 3D pressure distributions on the suction surface of the rotor blade [203].

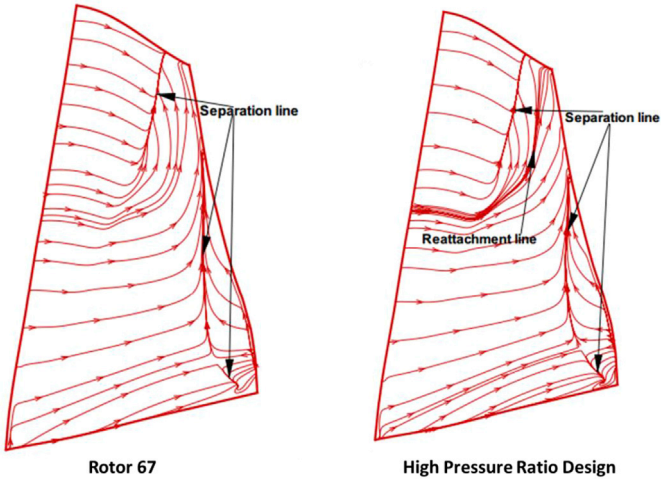

Fig. 25. 3D streamlines on the suction surface of the rotor blade [203].

Fig. 25. The reattachment flow appeared on the suction surface of the rotor with maximal pressure ratio and this was also responsible for the eliminated separation flows.

6.2. *Example 2: optimization of turbine blade with adjoint method*

Luo et al. [145] presented the application of a viscous continuous adjoint method for the optimization of a turbine blade row by means of the endwall contouring technique. Entropy production through the blade row was used as the objective function. Within each optimization cycle, the needed gradient information was obtained by only solving the flow governing equations and the corresponding adjoint equations one time regardless of the number of the input parameters. The cost function is shown as following:

$$\mathrm{I} = s_{gen} + \Lambda|\overline{\beta} - \overline{\beta}_0| \tag{40}$$

where s_{gen} means the generated entropy, $\overline{\beta}_0$ represents the reference mass-averaged flow turning of the turbine blade, $\overline{\beta}$ is the actual mass-averaged flow turning and Λ is a selected coefficient. The perturbations were added to the endwall contours in the form of a Fourier summation of four harmonics in the pitchwise direction. The parameterization of the contours is:

$$\mathrm{dz}(\mathrm{x}, s_p) = \sum_{i=1}^{8} b_i(x)\left\{\sum_{j=1}^{4}\left[V_{1ij}\sin\left(j\pi\frac{s_p}{s_o}\right) + V_{2ij}g(x)\cos\left(j\pi\frac{s_p}{s_o}\right)\right] + V_{3i}\right\} \tag{41}$$

where s_p means the pitch displacement, s_o is the local pitch, V_{1ij}, V_{2ij} and V_{3i} are the design parameters, $b_i(x)$ represents eight bump functions that are uniformly distributed from 10% axial chord upstream of the leading edge to about 50% chord downstream of the trailing edge and $g(x)$ is a defined blending function. The convergence history within 33 optimization cycles is shown in Fig. 26. In this figure, the total pressure ratio is increased by 0.2% and the adiabatic efficiency is increased by about 0.6%. Meanwhile, the produced exit flow angle is not larger than 0.04 deg when compared to the reference value.

The resulting 3D contoured endwall profile is shown in Fig. 27. A convex bump occurs near the blade pressure surface and a concave expansion occurs near the blade suction surface which leads to the accelerated flows near the pressure surface and the decelerated flows near the suction surface. Those effects are beneficial in reducing the cross-passage pressure gradient. To validate this, the detailed contours of

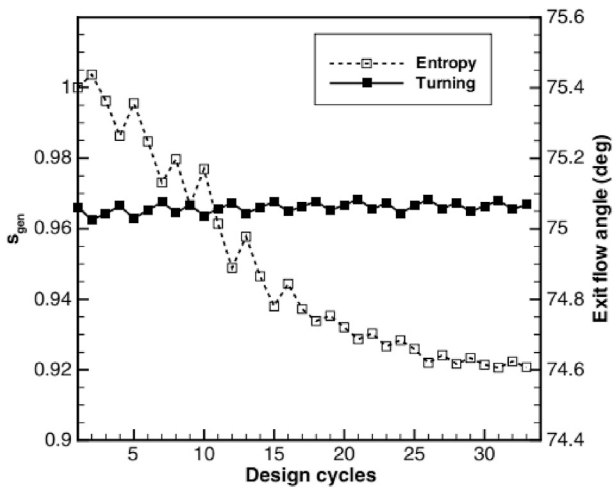

Fig. 26. Convergence history of entropy and flow turning [145].

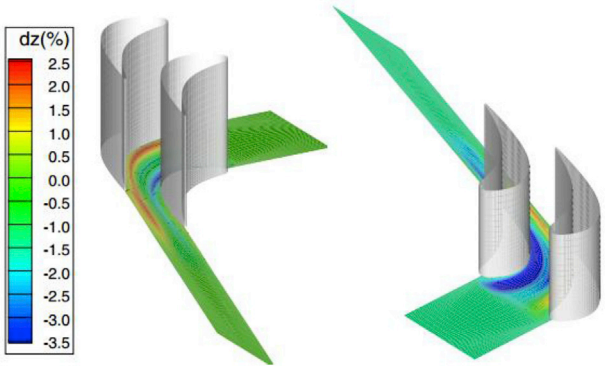

Fig. 27. Solid view of the 3D contoured endwall [145].

streamwise vorticity and secondary kinetic energy on planes normal to the axial flow direction located at the trailing edge are shown in Fig. 28. As for the optimized blade row, both the strength and size of the passage vortex and those of the wall-induced vortex are reduced. The secondary loss of the optimized blade is decreased by about 36.4% when compared to the prototype.

6.3. Example 3: date mining method

An automated 3D multi-objective optimization for a highly loaded centrifugal compressor was presented by combining the multi-objective evolutionary algorithm and data mining technique of self-organizing map in Ref. [202]. The mean camber line and the thickness distribution of section profiles near the hub and shroud region were parameterized by NURBS. The impeller was optimized for maximizing the adiabatic efficiency and total pressure ratio with constrains on the average mass flow rate as following:

$$\text{maximize}(\text{designed adiabatic efficiency}) \tag{42}$$

$$\text{maximize}(\text{designed total pressure ratio}) \tag{43}$$

$$\text{subject to}\left(0.99\dot{m}_{baseline} \le m_{design} \le 1.01\dot{m}_{baseline}\right) \tag{44}$$

As for the optimization process, the population size and the maximum generation are set to be 60 and 150 respectively. The Pareto Front cases are shown in Fig. 29. The isentropic efficiency of Design A, Design B and Design C is increased by 2.21%, 1.98% and 1.81%, respectively. Meanwhile, the total pressure ratio of Design A, Design B, and Design C is increased by 6.9%, 9.2% and 11.49%, respectively. The mass flow rate for all of the optimal Pareto Front cases meets the constraints mentioned above. SOM method is then used to explore the design information from the view of the optimization results. As shown in Fig. 30, the SOM neutrons are connected to its adjacent neutrons by neighborhood relation and are usually shown with hexagon topology. Here 1000 SOM neutrons are used for the SOM training. The correlations between the objective functions as well as the interactions among the design variables and objective functions are analyzed. Fig. 30 shows the component maps of the objective functions. It can be concluded that isentropic efficiency obtains larger values in the upper part while the total pressure ratio gets smaller values there. There exists a trade-off between the isentropic efficiency and total pressure ratio in high-loaded compressors.

The interaction among the design variables x_7 and the objective functions is visualized in Fig. 31 based on SOM-based scatterplots. As the value of x_7 increases, the isentropic efficiency increases while the total

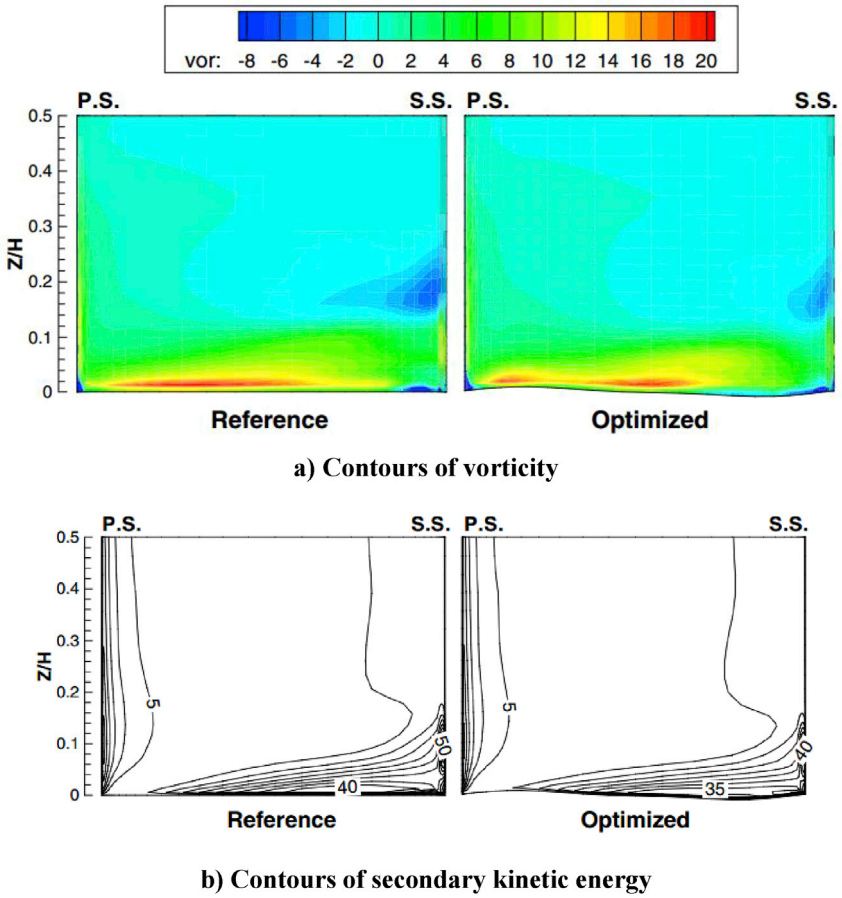

Fig. 28. Contours of streamwise vorticity and secondary kinetic energy on planes located at the trailing edge [145].

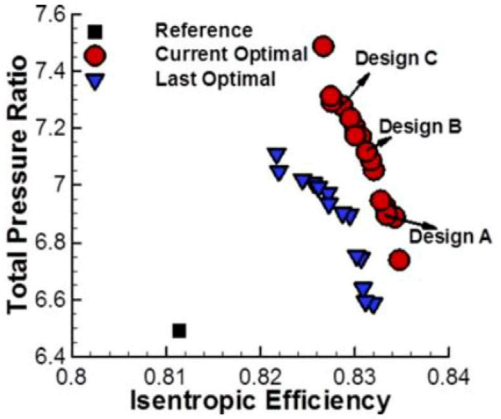

Fig. 29. Optimal Front solutions [202].

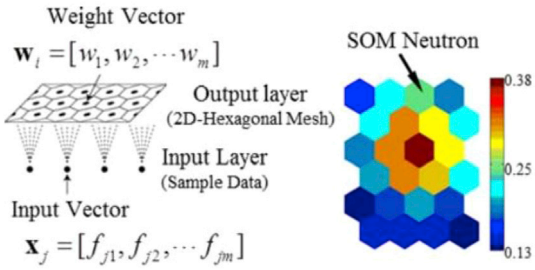

Fig. 30. Schematic map of SOM [202].

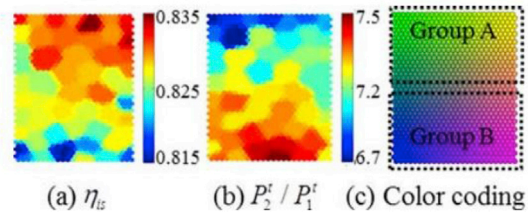

Fig. 31. SOM maps of isentropic efficiency, total pressure ratio and color coding (from left to right) [202].

pressure ratio decreases at the same time. The scatters are divided into two groups as Group A and Group B. Group A corresponds to SOM neutrons located in the upper part where the designs with higher isentropic efficiency occurs. Likewise, the scatters in Group B represent designs with lower isentropic efficiency but higher total pressure ratio.

6.4. *Further developments of design optimization methods*

All of the optimization results mentioned above validate the usefulness of the advanced optimization methods mentioned above in practical engineering applications. Although much progress has been made in turbomachinery designs by optimization methods, there are still significant gaps between the numerical optimization results and engineering applications. Information from papers published in this field and the authors' own insights suggest the following gaps:

1) Methodological issues. Only two generations of evolutionary optimization algorithms have been proposed [105] with the first generation characterized by the algorithm simplicity and lack of methods to validate the algorithms and the second generation emphasizing the efficiency at both the algorithmic level and the data structure level. Current algorithms do not behave with multiple objectives (normally more than 3 or 4). In addition, gradient-based methods can be easily trapped into local optimums due to the limitations of the methodologies themselves. The metamodel-based approaches are less attractive or even unworkable for large problems with the uncertainty in metamodels creating big challenges for engineers [60]. Adjoint methods are more effective, but they need to be able to transfer information through multi-row mixing planes and better approaches are needed for multi-objective and global optimization problems to meet practical application requirements.
2) Computational cost problems. Advanced optimization strategies require extensive calculations and costs. For example, Large Eddy Simulations (LES) or even Direct Numerical Simulations (DNS) are preferred for capturing fine flow field details. In addition, more design parameters and objective functions will lead to better optimization results. Furthermore, much more information can be recovered by data mining techniques to help designers determine the optimal search route. Finally, practical engineering problems require multidisciplinary optimization studies, concurrent multi-row optimization, uncertainty quantification based optimization and unsteady simulation based optimization methods that all require extensive computational resources.
3) Understanding of the physical mechanisms. Although the optimization methods strongly depend on the algorithms, the results also depend on a thorough understanding of the physical mechanisms. For example, compressor optimization designs always pursue higher peak efficiencies with sufficient surge margin. Unfortunately, there is still no consistent method for how to define the surge using numerical simulations with many researchers still analyzing this question. Physical mechanism-oriented optimization methods are urgently needed with properly formed objective functions. Multidisciplinary optimization studies need composite mechanisms that can be used to reduce the design parameter ranges or increase the number of ignorable factors.
4) Influence of manufacturing uncertainty. The manufacture tolerance effect on the efficiency and flow capacity of the turbomachinery has gradually attracted much more attention [132,163]. The tolerances impact the manufacturing costs with tighter tolerances leading to increased manufacturing complexity. Robust reliability-based design optimization tools should provide not only the performance ratings but also a confidence range for each variable that describes the effects of uncertainties on the design process.

These needs then lead to some promising topics for future research:

1) The design variable parameterization should only include design variables directly linked to the constraint such while excluding design variables with little effect on the aerodynamic performance of the shape. The parameterization method directly controls the number of input parameters which then impacts the computational cost. Advanced parameterization approaches are then needed to improve the optimization results.
2) DOE methods explore the design spaces to provide the maximum amount of information with the minimum number of design experiments to further reduce the computational costs. More “intelligent” sampling schemes with iterative, adaptive or progressive sampling processes should be developed to improve the sampling techniques [125,[126].
3) Surrogate models and approximation-based optimization methods have attracted much attention. The multiple surrogate model, which is constructed by weighting individual surrogate models, in general gives lower errors with better robustness than single surrogate models such as the polynomial RSM, the Kriging model and ANN [38,39]. Gradient information has recently been introduced into the surrogate models to improve their convergence [164]. In addition, the surrogate model result should be coupled with the DOE method with the preferred meta-model depending on the specific objective function.

4) The GA and EA are both widely used to optimize turbomachinery designs due to their better robustness, global optimal search abilities and flexibility for multiple objectives. Some promising paths are hybridized optimization algorithms [101], self-adaptation parameter control [128], minimum number of needed fitness function evaluations [129], algorithms that are independent of the platform and programming language [130], larger numbers of objectives [131], and uncertainty quantification [132].
5) Aerodynamic optimization involving gradient-based optimization needs fast, accurate ways to calculate the gradient information. The adjoint method is much faster than the finite difference method with calculations of the derivatives of the objective function with respect to the design variables that are fast and independent of the number of design variables. Thus, multiple objective, concurrent, and unsteady adjoint methods are needed, especially for multi-row global optimizations.
6) The stochastic and gradient-based optimization methods have their own advantages and disadvantages. Hybridization and automatic switching between optimization platforms can provide significantly improved performance at lower costs for turbomachinery designs. For example, a global optimization method can be used to explore wider parameter ranges in the preliminary turbomachinery design process, with a more locally efficient gradient-based method used for the subsequent 3D design process.
7) Pareto-optimal solutions of multi-objective design problems contain huge amounts of data, so data mining methods are needed to extract the design information. Current advances in data mining methods generally involve improved algorithms or further integration with existing methods.

7. Conclusions

This paper provides a general overview of recent developments for optimizing turbomachinery designs. Many other methods have been used with only the most significant methods mentioned here. This paper describes the basic turbomachinery optimization design framework, the stochastic and gradient-based algorithms, DOE strategies, various surrogate models, and data mining issues and their applications. The authors also provide their own insights regarding the most important current research trends and the future for this field. Some of the major conclusions are:

1) The geometric parameterization is an important part of the shape optimization procedure due to its direct relation to the number of design variables. Turbomachinery design methods use multi-section blade parameterization based on Bezier, B-spine or NURBS curves. The parameterization should preferably include design variables directly linked to the constraints, while excluding design variables having little effect on the aerodynamics. The parameterization needs to accurately model the geometry with a limited number of design variables.
2) The DOE method selects conditions that cover the entire design space to provide the maximum amount of information about the performance with the minimum number of experiments to reduce the computational costs. Various experimental design methods have been proposed including the CCD, D-optimal, and LHD methods for compressor and turbine designs. The CCD and D-optimality methods are inappropriate for computation design experiments where the systematic error is larger than the random error. LHD is a popular DOE optimization method for turbomachinery that fills the design space based on stratified sampling rather than concentrating on the boundary. More "intelligent" sampling schemes involving iterative, adaptive or progressive sampling methods should be developed to improve the sampling efficiency.
3) Computational power is increasing, but these optimization processes are very computationally intensive. Surrogate models and approximation-based optimization methods have attracted much attention as faster optimization methods. The polynomial RSM, ANN (mainly BPNN and RBNN), Kriging and GEK surrogate models have been widely used for the global optimization of turbomachinery flows. Polynomial response surfaces can be used for lower-order nonlinear problems, Kriging models are better for lower-order nonlinear problems in high dimensional design spaces and neural networks are better for higher-order nonlinear problems. The multiple surrogate model which is a weighted average of various surrogate models provides better robustness with smaller errors than the individual surrogate models. However, none of the surrogate models perform well for large objective spaces. Advanced surrogate models need to be integrated with improved smart DOE strategies for future optimization methods.
4) SA, GA and EA algorithms are widely used for optimizing turbomachinery designs since they are more robust, give globally optimal searches and can handle multiple objectives. More efficient stochastic algorithms are needed that can model high dimensionality problems, such as modified PSO, hybridized optimization algorithms, self-adaptation parameter control, less fitness function evaluations, platform and programming language independent models, models for large numbers of objectives, and uncertainty quantifications.
5) The optimization of aerodynamic problems using gradient-based optimization methods needs fast, accurate methods to calculate the gradient information. The traditional gradient-based method needs small step sizes, so the methods are expensive, especially for a large number of design variables; thus, they are not good for large problems. The adjoint method does not have this limitation but uses a much faster method to calculate the derivatives of the objective function with respect to the design variables. Adjoint methods are also needed for multi-objective, unsteady optimization problems with global optimizing capabilities.
6) Data mining methods are used to extract valuable design information from Pareto-optimal solutions for multi-objective problems. Various methods are used to understand the Pareto-optimal solutions, such as the star coordinate, scatter-plot matrix, value path, analysis of variance, self-organizing map, fuzzy multiple discriminant analysis, proper orthogonal decomposition, decision tree, and rough set methods. New developments in data mining methods focus on improving the algorithms and integrating them with existing methods.

Nomenclature

a	Design variable
$b(x)$	Bump function
d_k^i	The kth output of the ith sample
f_s	Standard sigmoid transfer function
$g(x)$	Blending function
h	Perturbation step size

k	The kth component
$\dot{m}$	Mass flow rate
p	Static pressure
s_{gen}	The generated entropy
s_o	Local pitch
s_p	The pitch displacement
t	Time step
$\overline{u}$	Total mean of the objective functions
v_i	Current velocity
w_i	Weight vector
x_j	Input vector
y_k^i	Corresponding approximated output value
y_i^*	Value predicted by the RSM
R	Correlation matrix
ACO	Ant Colony Optimization
ANN	Artificial Neural Network
ANOVA	Analysis of Variation
BPNN	Back Propagation Neural Network
CCD	Central Composite Design
CFD	Computational Fluid Dynamics
DE	Differential Evolution
DNS	Direct Numerical Simulation
DOE	Design of Experiment
DIV	Dynamic inertia weight and elite velocity
EA	Evolutionary Algorithm
ES	Evolution Strategy
FE	Finite Element
FMDA	Fuzzy Multiple Discriminant Analysis
GA	Genetic Algorithms
GEK	Gradient-Enhanced Kriging
LES	Large Eddy Simulation
LHD	Latin Hypercube Design
I	Object function
MAM	Midrange Approximation Method
MGPSO	Metamodel Guided Particle Swarm Optimization
MLMS	Moving Least-Squares Method and Sensitivity
MLP	Multilayer Perceptron
MODE	Multi-Objective Design Exploration
MOGA	Multi-Objective Genetic Algorithm
NASA	National Aeronautics and Space Administration
NPGA	Niches-Pareto Genetic Algorithm
NSGA	Non-dominated Sorting Genetic Algorithm
NSGA-II	Non-dominated Sorting Genetic Algorithm II
NURBS	Non-Uniform Rational B-Spine
N_O	Number of outputs
N_S	Number of samples
P	Number of coefficients
PAES	Pareto Archived Evolution Strategy
PG	Particle Generator
POD	Proper Orthogonal Decomposition
PRESS	Predicted residual error sum of squares
PS	Pressure surface
PSO	Particle Swarm Optimization
P_{best}	The best position
P_1^t	Inlet total pressure
P_2^t	Outlet total pressure
RANS	Reynolds-Averaged Navier-Stokes
RBDO	Reliability-Based Design Optimization
RBF	Radial Basis Function
RBNN	Radial Basis Neural Networks
R^2	Determination statistic
R_{adj}^2	The adjusted determination statistic
SA	Simulating Annealing
SOM	Self-Organizing Map
SS	Suction Surface

SST	Shear Stress Transport
SVM	Support vector machine
SVR	Support vector regression
S_{best}	The best swarm experienced
TAF	Transformation of Algorithms in Fortran
TRAF3D	3D Flow Solver
T	Temperature
UQ	Uncertainty Quantification
U	Flow variable vector
V	Flow velocity
2D	Two-Dimensional
3D	Three-Dimensional
λ	Adjoint factor
σ	Root mean square error
σ_E	Response due to the error
σ_T	Total response
σ_R	Response due to the model
$\overline{\sigma}$	The mean variance of the objective functions
τ	Clearance size
γ	Blade row stagger angle
β	Regression coefficients
$\overline{\beta}$	The actual mass-averaged flow turning
$\overline{\beta}_o$	The reference mass-averaged flow turning of the turbine blade
ε	Total error
η	Isentropic efficiency
Λ	Selected coefficient

Subscripts

1	Inlet of control volume
2	Outlet of control volume
adjusted	Adjusted
baseline	Baseline point
best	Best
design	Design point
E	Error
gen	Generated
i	*i*th
j	*j*th
k	kth
o	Outlet
p	Pitch
R	Response
s	Sample
T	Total

Superscripts

$\bullet$	Derivative
*	Predicted value
$+$	Non-dimensional
$\bar{}$	Averaged
$\hat{}$	Predicted value
i	*i*th
t	Total
T	Transpose
2	Square
-1	Inverse

References

[1] L. He, P. Shan, Three-dimensional aerodynamic optimization for axial-flow compressors based on the inverse design and the aerodynamic parameters, ASME J. Turbomach. 134 (2012) 031004.

[2] D. Bonaiuti, M. Zangeneh, On the coupling of inverse design and optimization techniques for the multiobjective, multipoint design of turbomachinery blades, ASME J. Turbomach. 131 (2009) 021014.

[3] M. Zangeneh, A compressible three-dimensional design method for radial and mixed flow turbomachinery blades, Int. J. Numer. Methods Fluid 13 (1991) 599–624.

[4] J.H. Kim, J.H. Choi, K.Y. Kim, Design optimization of a centrifugal compressor impeller using radial basis neural network method, in: Proc. ASME Turbo Expo, 2009 (ASME Paper GT2009–59666).

[5] A. Bartold, J. Franz, Optimization of a centrifugal impeller using evolutionary algorithms, in: Proc. ASME Turbo Expo, 2008 (ASME Paper GT2008–50805).

[6] M. Arabnia, G. Wahid, A strategy for multi-point shape optimization of turbine stages in three-dimensional flow, in: Proc. ASME Turbo Expo, 2009 (ASME Paper GT2009–59708).
[7] B. Chen, X.L. Gao, X. Yuan, Aerodynamic 3-dimensional optimal design by NURBS of a certain stage turbine blades, J. Power Eng. 26 (2006) 201–206.
[8] T. Sonoda, Y. Yamaguchi, T. Arima, M. Olhofer, B. Sendhoff, H.A. Schreiber, Advanced high turning compressor airfoils for low reynolds number condition: Part I—design and optimization, in: Proc. ASME Turbo Expo, 2003 (ASME Paper GT2003–38458).
[9] T. Sonoda, Y. Yamaguchi, T. Arima, M. Olhofer, B. Sendhoff, H.A. Schreiber, Advanced high turning compressor airfoils for low reynolds number condition: Part II—experimental and numerical analysis, in: Proc. ASME Turbo Expo, 2003 (ASME Paper GT2003–38477).
[10] G. Briasco, D. Bruna, C. Cravero, A NURBS-based optimization tool for axial compressor cascades at design and off-design conditions, in: Proc. ASME Turbo Expo, 2008 (ASME Paper GT2008–50622).
[11] A. Arnone, D. Bonaiuti, A. Focacci, R. Pacciani, G.A.S. Del, E. Spano, Parametric optimization of a high-lift turbine vane, in: Proc. ASME Turbo Expo, 2004 (ASME Paper GT 2004–54308).
[12] F. Sieverding, B. Ribi, M. Casey, M. Meyer, Design of industrial axial compressor blade sections for optimal range and performance, in: Proc. ASME Turbo Expo, 2003 (ASME Paper GT2003–38036).
[13] E. Benini, A. Tourlidakis, Design Optimization of Vaned Diffusers of Centrifugal Compressors using Genetic Algorithms, 2001 (AIAA Paper AIAA-2001-2583).
[14] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (2000) 55–61.
[15] W.C. Carpenter, Effect of Design Selection on Response Surface Performance, 1993 (NASA Contractor Report 4520).
[16] M.R.G. Meireles, P.E.M. Almeida, M.G. Simões, A comprehensive review for industrial applicability of artificial neural networks, IEEE Trans. Ind. Electron. 50 (2003) 585–601.
[17] X.D. Wang, C. Hirsch, S. Kang, C. Lacor, Multi-objective optimization of turbomachinery using improved NSGA-II and approximation model, Comput. Methods Appl. Mech. Eng. 200 (2011) 883–895.
[18] M. Bellman, J. Straccia, B. Morgan, K. Maschmeyer, R. Agarwal, Improving Genetic Algorithm Efficiency with an Artificial Neural Network for Optimization of Low Reynolds Number Airfoils, 2009 (AIAA Paper AIAA 2009-1096).
[19] K. Ghorbanian, G. Mohammad, Axial compressor performance map prediction using artificial neural network, in: Proc. ASME Turbo Expo, 2007 (ASME Paper GT 2007–27165).
[20] S. Pierret, R.A. Van Den Braembussche, Turbomachinery blade design using a Navier-Stokes solver and artificial neural network, in: Proc. ASME Turbo Expo, 1998 (ASME Paper 98-GT-004).
[21] M.M. Rai, K.M. Nateri, Aerodynamic design using neural networks, AIAA J. 38 (2000) 173–182.
[22] D.C. Montgomery, R.H. Myers, Response Surface Methodology: Process and Product Optimization Using Designed Experiments, A Wiley-Interscience Publications, 1995.
[23] N.V. Queipo, R.T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, P.K. Tucker, Surrogate-based analysis and optimization, Prog. Aerosp. Sci. 41 (2005) 1–28.
[24] T.W. Simpson, J.D. Poplinski, P.N. Koch, J.K. Allen, Metamodels for computer-based engineering design: survey and recommendations, Eng. Comput. 17 (2001) 129–150.
[25] J.I. Madsen, W. Shyy, R.T. Haftka, Response surface techniques for diffuser shape optimization, AIAA J. 38 (2000) 1512–1518.
[26] R.H. Myers, D.C. Montgomery, C.M. Anderson-Cook, Response Surface Methodology: Process and Product Optimization Using Designed Experiments, John Wiley & Sons, 2016.
[27] T.W. Simpson, T.M. Mauery, J.J. Korte, F. Mistree, Kriging models for global approximation in simulation-based multidisciplinary design optimization, AIAA J. 39 (2001) 2233–2241.
[28] J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn, Design and analysis of computer experiments, Stat. Sci. 4 (1989) 409–423.
[29] D.D. Cox, S. John, A statistical method for global optimization, in: IEEE International Conference, October 1992.
[30] M.W. Trosset, V. Torczon, Numerical Optimization Using Computer Experiments, 1997 (NASA Report NASA CR-201724).
[31] J.R. Koehler, A.B. Owen, Computer experiments, Handb. Stat. 13 (1996) 261–308.
[32] C. Currin, T. Mitchell, M. Morris, D. Yivisaker, Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments, J. Am. Stat. Assoc. 86 (1991) 953–963.
[33] X.F. Wang, G. Xi, Z.H. Wang, Aerodynamic optimization design of centrifugal compressor's impeller with Kriging model, Proc. Inst. Mech. Eng. Part A J. Power Energy 220 (2006) 589–597.
[34] C. Voß, M. Aulich, B. Kaplan, E. Nicke, Automated multiobjective optimization in axial compressor blade design, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–90420).
[35] M. Aulich, U. Siller, High-dimensional constrained multiobjective optimization of a fan stage, in: Proc. ASME Turbo Expo, 2011 (ASME Paper GT 2011–45618).
[36] W. Li, S. Padula, Approximation methods for conceptual design of complex systems, in: Eleventh International Conference on Approximation Theory, 2004.
[37] L.E. Zerpa, N.V. Queipo, S. Pintos, J.L. Salager, An optimization methodology of alkaline surfactant polymer flooding processes using field scale numerical simulation and multiple surrogates, J. Petrol. Sci. Eng. 47 (2005) 197–208.
[38] T. Goel, R.T. Haftka, W. Shyy, N.V. Queipo, Ensemble of surrogates, Struct. Multidiscip. Optim. 33 (2007) 199–216.
[39] R. Jin, C. Wei, T.W. Simpson, Comparative studies of metamodelling techniques under multiple modelling criteria, Struct. Multidiscip. Optim. 23 (2001) 1–13.
[40] A. Samad, K.Y. Kim, T. Goel, R.T. Haftka, Multiple surrogate modeling for axial compressor blade shape optimization, J. Propuls. Power 24 (2008) 301–310.
[41] N.X. Chen, H.W. Zhang, Q. Xu, W.G. Huang, Application of simple gradient-based method and multi-section blade parameterization technique to aerodynamic design optimization of a 3D transonic single rotor compressor, in: Proc. ASME Turbo Expo, 2009 (ASME Paper GT 2009–59734).
[42] N. Chen, H. Zhang, F. Ning, Y. Xu, W. Huang, An effective turbine blade parameterization and aerodynamic optimization procedure using an improved response surface method, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–90104).
[43] G.G. Wang, Adaptive response surface method using inherited Latin hypercube design points, Trans. ASME J. Mech. Des. 125 (2003) 210–220.
[44] R. Unal, R.A. Lepsch, M.L. McMillin, Response surface model building and multidisciplinary optimization using D-optimal designs, in: Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 1998.
[45] T.J. Mitchell, An algorithm for the construction of "D-optimal" experimental designs, Technometrics 16 (1974) 203–210.
[46] A.A. Giunta, V. Balabanov, D. Haim, B. Grossman, W.H. Mason, L.T. Watson, R.T. Haftka, Multidisciplinary optimization of a supersonic transport using design of experiments theory and response surface modeling, Aeronaut. J. 101 (1997) 347–356.
[47] R. Unal, R.A. Lepsch, W. Engelund, D.O. Stanley, Approximation Model Building and Multidisciplinary Design Optimization Using Response Surface Methods, 1996 (AIAA Paper AIAA-96–404).
[48] G. Taguchi, Y. Yoshiko, Taguchi Methods: Design of Experiments, in: Amer Supplier Inst, 1993.
[49] A.B. Owen, Orthogonal arrays for computer experiments, integration and visualization, Stat. Sin. 4 (1992) 439–452.
[50] T. Boxin, Orthogonal array-based Latin hypercubes, J. Am. Stat. Assoc. 88 (1993) 1392–1397.
[51] J.S. Park, Optimal Latin-hypercube designs for computer experiments, J. Stat. Plan. Inferace 39 (1994) 95–111.
[52] W. Yi, H. Huang, W. Han, Design optimization of transonic compressor rotor using CFD and genetic algorithm, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–90155).
[53] A. Samad, K.Y. Kim, Shape optimization of an axial compressor blade by multi-objective genetic algorithm, Proc. Inst. Mech. Eng. Part A J. Power Energy 222 (2008) 599–611.
[54] J.H. Kim, K.Y. Kim, K.J. Choi, Design optimization of circumferential casing grooves for a transonic axial compressor to enhance stall margin, in: Proc. ASME Turbo Expo, 2010 (ASME Paper GT 2010–22396).
[55] J.H. Kim, J.W. Kim, K.Y. Kim, Axial-flow ventilation fan design through multi-objective optimization to enhance aerodynamic performance, ASME J. Fluid. Eng. 133 (2011) 101101.
[56] Y.S. Lian, M.S. Liou, Multiobjective optimization using coupled response surface model and evolutionary algorithm, AIAA J. 43 (2005) 1316–1325.
[57] Y.S. Lian, N.H. Kim, Reliability-based design optimization of a transonic compressor, AIAA J. 44 (2006) 368–375.
[58] K. Yu, X. Yang, Z. Yue, Aerodynamic and heat transfer design optimization of internally cooling turbine blade based different surrogate models, Struct. Multidiscip. Optim. 44 (2011) 75–83.
[59] D. Pasquale, P. Giacomo, R. Stefano, Optimization of turbomachinery flow surfaces applying a CFD-based throughflow method, ASME J. Turbomach. 136 (2014) 031013.
[60] G.G. Wang, S. Shan, Review of metamodeling techniques in support of engineering design optimization, Trans. ASME J. Mech. Des. 129 (2007) 370–380.
[61] A. Demeulenaere, L. Alban, H. Charles, Application of multipoint optimization to the design of turbomachinery blades, in: Proc. ASME Turbo Expo, 2004 (ASME Paper GT 2004–53110).
[62] T. Mengistu, G. Wahid, Aerodynamic optimization of turbomachinery blades using evolutionary methods and ANN-based surrogate models, Optim. Eng. 9 (2008) 239–255.
[63] S. Derakhshan, M. Bijan, N. Ahmad, The comparison of incomplete sensitivities and Genetic algorithms applications in 3D radial turbomachinery blade optimization, Comput. Fluid 39 (2010) 2022–2029.
[64] A. Huppertz, P.M. Flassing, R.J. Flassing, M. Swoboda, Knowledge based 2d blade design using multi-objective aerodynamic optimization and a neural network, in: Proc. ASME Turbo Expo, 2007 (ASME Paper GT 2007-28204).
[65] J.H. Kim, J.H. Chio, K.Y. Kim, Surrogate modeling for optimization of a centrifugal compressor impeller, Int. J. Fluid Mach. Syst. 3 (2010) 29–38.
[66] C.M. Jang, K.Y. Kim, Optimization of a stator blade using response surface method in a single-stage transonic axial compressor, Proc. Inst. Mech. Eng. Part A J. Power Energy 219 (2005) 595–603.
[67] T. Goel, R. Vaidyanathan, R.T. Haftka, Response surface approximation of Pareto optimal front in multi-objective optimization, Comput. Methods Appl. Mech. Eng. 196 (2007) 879–893.
[68] C. Kim, K.K. Choi, Reliability-based design optimization using response surface method with prediction interval estimation, Trans. ASME J. Mech. Des. 130 (2008) 121401.
[69] C. Kim, S. Wang, K.K. Choi, Efficient response surface modeling by using moving least-squares method and sensitivity, AIAA J. 43 (2005) 2404–2411.

[70] N. Papila, W. Shyy, L. Griffin, D.J. Dorney, Shape optimization of supersonic turbines using global approximation methods, J. Propuls. Power 18 (2002) 509–518.
[71] D. Buche, G. Gianfranco, S. Peter, Automated design optimization of compressor blades for stationary, large-scale turbomachinery, in: Proc. ASME Turbo Expo, 2003 (ASME Paper GT 2003–38421).
[72] F. Wallin, L.E. Eriksson, Response surface-based transition duct shape optimization, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–90978).
[74] K.C. Giannakoglou, Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence, Prog. Aerosp. Sci. 38 (2002) 43–76.
[75] M.A. Trigg, G.R. Tubby, A.G. Sheard, Automatic genetic optimization approach to 2D blade profile design for steam turbines, in: Proc. ASME Turbo Expo, 1997 (ASME Paper 97-GT-392).
[76] M.L. Shelton, B.A. Gregory, S.H. Lamson, H.L. Moses, R.L. Doughty, T. Kiss, Optimization of a transonic turbine airfoil using artificial intelligence, CFD and cascade testing, in: Proc. ASME Turbo Expo, 1993 (ASME Paper 93-GT-161).
[77] J. Li, Z. Feng, J. Chang, Z. Shen, Aerodynamic optimum design of transonic turbine cascades using genetic algorithms, J. Therm. Sci. 6 (1997) 111–116.
[78] S.M.H. Mahmood, G.M. Turner, K. Siddappaji, Flow characteristics of an optimized axial compressor rotor using smooth design parameters, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–57028).
[79] S. Shahpar, A comparative study of optimization methods for aerodynamic design of turbomachinery blades, in: Proc. ASME Turbo Expo, 2000 (ASME Paper 2000-GT-0523).
[80] K. Deb, A. Pratap, S. Agarwal, T.A.M.T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197.
[81] Z. Chi, H. Liu, S. Zang, Multi-objective optimization of the impingement-film cooling structure of a HPT endwall using conjugate heat transfer CFD, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56559).
[82] C. Poloni, A. Giurgevish, L. Onesti, V. Pediroda, Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics, Comput. Methods Appl. Mech. Eng. 186 (2000) 403–420.
[83] E. Benini, Three-dimensional multi-objective design optimization of a transonic compressor rotor, J. Propuls. Power 20 (2004) 559–565.
[84] O.I. Oksuz, A. Sinan, S.K. Mehmet, Aerodynamic optimization of turbomachinery cascades using Euler/boundary-layer coupled genetic algorithms, J. Propuls. Power 18 (2002) 652–657.
[85] D.E. Goldberg, Genetic and evolutionary algorithms come of age, Comm. ACM 37 (1994) 113–120.
[86] Fine™/design3d User Guide, NUMECA, 2007.
[87] M.A. Trigg, G.R. Tubby, A.G. Sheard, Automatic genetic optimization approach to two-dimensional blade profile design for steam turbines, ASME J. Turbomach. 121 (1999) 11–17.
[88] C. Kong, S. Kho, J. Ki, Component map generation of a gas turbine using genetic algorithms, ASME J. Eng. Gas. Turbines Power 128 (2006) 92–96.
[89] B.H. Dennis, G.S. Dulikravich, Z.X. Han, Optimization of turbomachinery airfoils with a genetic/sequential-quadratic-programming algorithm, J. Propuls. Power 17 (2001) 1123–1128.
[91] K. Ashihara, G. Akira, Turbomachinery blade design using 3-D inverse design method, CFD and optimization algorithm, in: Proc. ASME Turbo Expo, 2001 (ASME Paper 2001-GT-0358).
[92] J. Li, N. Satofuka, Optimization design of a compressor cascade airfoil using a Navier-Stokes solver and genetic algorithms, Proc. Inst. Mech. Eng. Part A J. Power Energy 216 (2002) 195–202.
[93] A. Demeulenaere, A. Purwanto, A. Ligout, C. Hirsch, R. Dijkers, F. Visser, Design and optimization of an industrial pump: application of genetic algorithms and neural network, in: Proc. ASME Turbo Expo, 2005 (ASME Paper FEDSM2005–77487).
[94] T. Ning, C. Gu, X. Li, T. Liu, Three-dimensional aerodynamic optimization of a multi-stage axial compressor, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT2016–57626).
[95] C.M. Fonseca, J.F. Peter, Genetic algorithms for multiobjective optimization: formulation discussion and generalization, ICGA 93 (1993) 416–423.
[96] K.A. DeJong, Analysis of the Behavior of a Class of Genetic Adaptive Systems (pH.D. thesis), University of Michigan, Ann Arbor, 1975.
[97] N. Srinivasan, K. Deb, Multi-objective function optimisation using non-dominated sorting genetic algorithm, Evol. Comput. 2 (1994) 221–248.
[98] J. Horn, N. Nafpliotis, D.E. Goldberg, A niched Pareto genetic algorithm for multiobjective optimization, in: IEEE World Congress on Computational Intelligence, Proceedings of the First IEEE Conference on IEEE, 1994.
[99] C.M. Fonseca, J.F. Peter, An overview of evolutionary algorithms in multiobjective optimization, Evol. Comput. 3 (1995) 1–16.
[100] C.A.C. Coello, An updated survey of evolutionary multiobjective optimization techniques: state of the art and future trends, in: Proceedings of the Congress on Evolutionary Computation, 1999.
[101] Y. Lian, A. Oyama, M.S. Liou, Progress in design optimization using evolutionary algorithms for aerodynamic problems, Prog. Aerosp. Sci. 46 (2010) 199–223.
[102] A. Oyama, M.S. Liou, S. Obayashi, Transonic axial-flow blade optimization: evolutionary algorithms/three-dimensional Navier-Stokes solver, J. Propuls. Power 20 (2004) 612–619.
[103] A. Oyama, M.S. Liou, Multi-objective optimization of rocket engine pumps using evolutionary algorithm, J. Propuls. Power 18 (2002) 528–535.
[104] Y. Lian, M.S. Liou, A. Oyama, An enhanced evolutionary algorithm with a surrogate model, in: Proceedings of Genetic and Evolutionary Computation Conference, Seattle, WA, 2004.
[105] C.A.C. Coello, Evolutionary multi-objective optimization: a historical view of the field, IEEE Comput. Intell. Mag. 1 (2006) 28–36.
[106] B.H. Dennis, I.N. Egorov, Z.X. Han, Multi-objective optimization of turbomachinery cascades for minimum loss, maximum loading, and maximum gap-to-chord ratio, Int. J. Turbo Jet Engines 18 (2001) 201–210.
[107] F. Muyl, D. Laurent, H. Vincent, Hybrid method for aerodynamic shape optimization in automotive industry, Comput. Fluid 33 (2004) 849–858.
[108] Y.S. Ong, B.N. Prasanth, J.K. Andrew, Evolutionary optimization of computationally expensive problems via surrogate modeling, AIAA J. 41 (2003) 687–696.
[109] O. Baysal, E.E. Mohamed, Aerodynamic design optimization using sensitivity analysis and computational fluid dynamics, AIAA J. 30 (1992) 718–725.
[110] D.X. Wang, L. He, Adjoint aerodynamic design optimization for blades in multistage turbomachines—part I: methodology and verification, ASME J. Turbomach. 132 (2010) 021011.
[111] D.X. Wang, L. He, Adjoint aerodynamic design optimization for blades in multistage turbomachines—part II: validation and application, ASME J. Turbomach. 132 (2010) 021012.
[112] J. Luo, J. Xiong, F. Liu, I. McBean, Three-dimensional aerodynamic design optimization of a turbine blade by using an adjoint method, ASME J. Turbomach. 133 (2011) 011026.
[113] D.X. Wang, L. He, Concurrent blade aerodynamic-aero-elastic design optimization using adjoint method, ASME J. Turbomach. 133 (2011) 011021.
[114] J. Yang, J. Luo, J. Xiong, F. Liu, Aerodynamic design optimization of the last stage of a multi-stage compressor by using an adjoint method, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT2016–56893).
[115] C. Ma, X. Su, X. Yuan, Adjoint-based unsteady aerodynamic optimization of a transonic turbine stage, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56885).
[116] K. Arens, P. Rentrop, S.O. Stoll, U. Wever, An adjoint approach to optimal design of turbine blades, Appl. Numer. Math. 53 (2005) 93–105.
[117] Y. Li, D. Yang, Z. Feng, Inverse problem in aerodynamic shape design of turbomachinery blades, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–91135).
[118] S. Yang, H.Y. Wu, F. Liu, H.M. Tsai, Aerodynamic Design of Cascades by Using an Adjoint Equation Method, 2003. AIAA Paper.
[119] H.Y. Wu, F. Liu, H.M. Tsai, Aerodynamic design of turbine blades using an adjoint equation method, in: 43rd AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2005.
[120] R. Florea, C.H. Kenneth, Sensitivity analysis of unsteady inviscid flow through turbomachinery cascades, AIAA J. 39 (2001) 1047–1056.
[121] J.P. Thomas, C.H. Kenneth, H.D. Earl, Discrete adjoint approach for modeling unsteady aerodynamic design sensitivities, AIAA J. 43 (2005) 1931–1936.
[122] H. Li, L. Song, Y. Li, Z. Feng, 2D viscous aerodynamic shape design optimization for turbine blades based on adjoint method, ASME J. Turbomach. 133 (2011) 031014.
[123] B. Walther, N. Siva, Optimum shape design for multirow turbomachinery configurations using a discrete adjoint approach and an efficient radial basis function deformation scheme for complex multiblock grids, ASME J. Turbomach. 137 (2015) 081006.
[124] D.I. Papadimitriou, K.C. Giannakoglou, Compressor blade optimization using a continuous adjoint formulation, in: Proc. ASME Turbo Expo, 2006 (ASME Paper GT 2006–90466).
[125] R. Rai, I.C. Matthew, Qualitative and quantitative sequential sampling, in: Proc. ASME Comput. Inform. Eng, 2006 (ASME Paper DETC2006–99178).
[126] D.A. Romero, C.H. Amon, S. Finger, On adaptive sampling for single and multi-response Bayesian surrogate models, in: Proc. ASME Comput. Inform. Eng, 2006 (ASME Paper DETC2006–99210).
[127] J. Peter, M. Marcelet, Comparison of surrogate models for turbomachinery design, WSEAS Trans. Fluid Mech. 3 (2008) 10–17.
[128] D. Büche, M. Sibylle, K. Petros, Self-adaptation for multi-objective evolutionary algorithms, in: International Conference on Evolutionary Multi-criterion Optimization, 2003.
[129] J. Knowles, J.H. Evan, Multiobjective optimization on a budget of 250 evaluations, in: International Conference on Evolutionary Multi-criterion Optimization, 2005.
[130] S. Bleuler, M. Laumanns, L. Thiele, E. Zitzler, PISA—a platform and programming language independent interface for search algorithms, in: International Conference on Evolutionary Multi-criterion Optimization, 2003.
[131] R.C. Purshouse, On the Evolutionary Optimisation of Many Objectives (pH.D. thesis), University of Sheffield, Sheffield, UK, 2003.
[132] S. Lee, D.H. Rhee, K. Yee, Optimal arrangement of the film cooling holes considering the manufacturing tolerance for high pressure turbine nozzle, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–57973).
[133] O. Baysal, E.E. Mohamed, Aerodynamic sensitivity analysis methods for the compressible Euler equations, ASME J. Fluid. Eng. 113 (1991) 681–688.
[134] A. Jameson, Aerodynamic design via control theory, J. Sci. Comput. 3 (1988) 233–260.
[135] M.B. Giles, A.P. Niles, An introduction to the adjoint approach to design, ERCOFTAC 65 (2000) 393–415.
[136] L. Ji, W. Li, Y. Tian, W. Yi, J. Chen, Multi-stage turbomachinery blades optimization design using adjoint method and thin shear-layer NS equations, in: Proc. ASME Turbo Expo, 2012 (ASME Paper GT 2012–68537).
[137] C. Frey, H.P. Kersken, D. Nurnberger, The discrete adjoint of a turbomachinery RANS solver, in: Proc. ASME Turbo Expo, 2009 (ASME Paper GT 2009–59062).
[138] B. Walther, N. Siva, Adjoint-based constrained aerodynamic shape optimization for multistage turbomachines, J. Propuls. Power 31 (2015) 1298–1319.
[139] B. Walther, N. Siva, An adjoint-based multi-point optimization method for robust turbomachinery design, in: Proc. ASME Turbo Expo, 2015 (ASME Paper GT 2015–44142).

[140] R.S. Bunker, The effects of manufacturing tolerances on gas turbine cooling, J. Turbomach. 131 (2009) 041018.
[141] S. Shahpar, C. Stefano, Aerodynamic optimization of high-pressure turbines for lean-burn combustion system, ASME J. Eng. Gas. Turbines Power 135 (2013) 055001.
[142] A.V. Kirill, V.K. Gennady, V.O. Kseniya, A.D. Roman, V.K. Dmitry, N.S. Yury, Robust optimization of the HPT blade cooling and aerodynamic efficiency, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56195).
[143] J.H. Page, R. Watson, Z. Ali, P.G. Tucker, P. Hield, Advances of turbomachinery design optimization, in: 53rd AIAA Aerospace Sciences Meeting Kissimmee, Florida, 2015.
[144] M.B. Giles, A.P. Niles, Improved Lift and Drag Estimates Using Adjoint Euler Equations, 1999 (AIAA paper AIAA-99–3293).
[145] J. Luo, F. Liu, I. McBean, Turbine blade row optimization through endwall contouring by an adjoint method, J. Propuls. Power 31 (2014) 505–518.
[146] L. Chen, J. Chen, Aerodynamic optimization design for high pressure turbines based on the adjoint approach, Chin. J. Aeronaut. 28 (2015) 757–769.
[147] J. Shinkyu, C. Kazuhisa, O. Shigeru, Data mining for aerodynamic design space, J. Aerosp. Comput. Inf. Comm. 2 (2005) 452–469.
[148] M. Manas, Graphical methods of multicriterial optimization, ZAMM 62 (1982) T375–T377.
[149] W.S. Cleveland, The Elements of Graphing Data, Wadsworth Advanced Books and Software, Monterey, CA, 1985.
[150] T. Tatsukawa, O. Akira, F. Kozo, Comparative study of data mining methods for aerodynamic multiobjective optimizations, in: 8th World Congress on Computational Mechanics and 5th European Congress on Computational Methods in Applied Sciences and Engineering, 2008.
[151] A.M. Geoffrion, S.D. James, A. Feinberg, An interactive approach for multi-criterion optimization, with an application to the operation of an academic department, Manag. Sci. 19 (1972) 357–368.
[152] T. Kohonen, M.R. Schroeder, T.S. Huang, S.O. Maps, Springer-Verlag New York, Inc., Secaucus, NJ, 2001.
[153] D. Yamashiro, Y. Tomohiro, F. Takeshi, Visualization Method of Pareto Solutions Reflecting Users' Subjectivity in Multi-objective Optimization Problems, PDPTA, 2007, pp. 746–752.
[154] A. Oyama, N. Taku, F. Kozo, Data mining of Pareto-optimal transonic airfoil shapes using proper orthogonal decomposition, J. Aircr. 47 (2010) 1756–1762.
[155] K. Chiba, O. Shigeru, Data mining for multidisciplinary design space of regional-jet wing, J. Aerosp. Comput. Inf. Comm. 4 (2007) 1019–1036.
[156] K. Sugimura, Design Optimization and Knowledge Mining for Turbomachinery, Tohoku University, Japan, 2009 (Ph.D. thesis).
[157] K. Shimoyama, K. Sugimura, S. Jeong, S. Obayashi, Performance map construction for a centrifugal diffuser with data mining techniques, J. Comput. Sci. Tech. 4 (2010) 36–50.
[158] S. Shahpar, A review of automatic optimization applications in aerodynamic design of turbomachinery components, in: CD Proceedings of the ERCOFTAC Conference in Design Optimization: Methods and Applications, Athens, Greece, 2004.
[159] L.V. Santana-Quintero, A.M. Alfredo, A. Carlos, A Review of Techniques for Handling Expensive Functions in Evolutionary Multi-objective Optimization, Springer, Berlin Heidelberg, 2010.
[160] S. Shahpar, High fidelity multi-stage design optimisation of multi-stage turbine blades using a Mid-range Approximate Method (MAM), in: 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, 2010.
[161] D. Wang, F. Omano, Y.S. Li, R. Wells, C. Hjalmarsson, L. Hedlund, T. Povey, FLOvane: a new approach for HP vane design, ASME J. Turbomach. 139 (2017) 061002.
[162] S. Shahpar, C. Stefano, P. Laurens, Automatic design optimisation of profiled endwalls including real geometrical effects to minimize turbine secondary flows, in: Proc. ASME Turbo Expo, 2014 (ASME Paper GT 2014–26628).
[163] Z. Giulio, B. Gabriel, B. Simon, Gradient-based adjoint and design of experiment CFD methodologies to improve the manufacturability of high pressure turbine blades, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56042).
[164] X. Tang, J. Luo, F. Liu, Adjoint-response surface method in aerodynamic shape optimization of turbomachinery blades, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56170).
[165] R.C. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995.
[166] Y. Shi, Particle swarm optimization: developments, applications and resources, Evol. Comput. 1 (2001) 81–86.
[167] R.C. Eberhart, Y. Shi, Comparing inertia weights and constriction factors in particle swarm optimization, Evol. Comput. 1 (2000) 84–88.
[168] M.S. Khurana, H. Winarto, A.K. Sinha, Application of Swarm Approach and Artificial Neural Networks for Airfoil Shape Optimization, 2008 (AIAA Paper AIAA 2008-5954).
[169] M. Pontani, B.A. Conway, Particle swarm optimization applied to space trajectories, J. Guid. Control. Dynam. 33 (2010) 1429–1441.
[170] Y. Shi, R.C. Eberhart, Parameter selection in particle swarm optimization, in: International Conference on Evolutionary Programming, Springer, Berlin Heidelberg, 1998.
[171] P.C. Fourie, A.A. Groenwold, The particle swarm optimization algorithm in size and shape optimization, Struct. Multidiscip. Optim. 23 (2002) 259–267.
[172] M. Clerc, J. Kennedy, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, IEEE Trans. Evol. Comput. 6 (2002) 58–73.
[173] B. Yang, Q. Xu, L. He, L.H. Zhao, C.G. Gu, P. Ren, A novel global optimization algorithm and its application to airfoil optimization, ASME J. Turbomach. 137 (2015) 041011.
[174] D. Chao, Z. Qiu, Particle swarm optimization algorithm based on the idea of simulated annealing, IJCNS 6 (2006) 152–157.
[175] X. Wu, Y. Wang, T. Zhang, An improved GAPSO hybrid programming algorithm, in: International Conference on Information Engineering and Computer Science, 2009.
[176] X. Xie, P. Wu, Research on the optimal combination of ACO parameters based on PSO, in: Networking and Digital Society (ICNDS), 2010 2nd International Conference on IEEE, 2010.
[177] A. Safari, K.H. Hajikolaei, H.G. Lemu, G.G. Wang, A high-dimensional model representation guided PSO methodology with application on compressor airfoil shape optimization, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016-56741).
[178] Y. Duan, W. Wu, Z. Fan, T. Chen, An introduction of aerodynamic shape optimization platform for compressor blade, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56861).
[179] N. Bahrani, Multidisciplinary Design Optimization of Turbomachinery Blade (Ph.D. thesis), University of Toronto, CA, 2015.
[180] I. Tsalicoglou, B. Phillipsen, Design of radial turbine meridional profiles using particle swarm optimization, in: 2nd International Conference on Engineering Optimization, 2010.
[181] Z. Song, B. Liu, X. Mao, X. Lu, Optimization of tandem blade based on improved particle swarm algorithm, in: Proc. ASME Turbo Expo, 2016 (ASME Paper GT 2016–56901).
[182] A. Ratnaweera, S.K. Halgamuge, H.C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, IEEE Trans. Evol. Comput. 8 (2004) 240–255.
[183] T.M. Blackwell, P. Bentley, Don't push me! Collision-avoiding swarms, in: Evolutionary Computation, Proceedings of the 2002 Congress on IEEE, 2002.
[184] X. Li, Adaptively choosing neighborhood bests using species in a particle swarm optimizer for multimodal function optimization, in: Genetic and Evolutionary Computation Conference, Springer, Berlin Heidelberg, 2004.
[185] J. Kennedy, R. Mendes, Population structure and particle swarm performance, in: Proceeding of the 2002 Congress on IEEE, 2002.
[186] P.N. Suganthan, Particle swarm optimizer with neighborhood operator, in: Proceedings of the Congress on Evolutionary Computation, Piscataway, 1999.
[187] W.J. Zhang, X.F. Xie, DEPSO: hybrid particle swarm with differential evolution operator, in: IEEE International Conference on Systems Man and Cybernetics, 2003.
[188] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680.
[189] T. Tong, Z.P. Feng, Multi-objective optimization design for transonic turbine cascades using simulated annealing algorithm, J. Hydrodyn. Ser. B Engl. Ed. 11 (1999) 20–25.
[190] W.T. Tiow, K.F.C. Yiu, M. Zangeneh, Application of simulated annealing to inverse design of transonic turbomachinery cascades, Proc. Inst. Mech. Eng. Part A J. Power Energy 216 (2002) 59–73.
[191] S. Pierret, R.A. Van den Braembussche, Turbomachinery blade design using a Navier-Stokes solver and artificial neural network, in: Proc. ASME Turbo Expo, 1998 (ASME Paper 98-GT-004).
[192] R.A. Rutenbar, Simulated annealing algorithms: an overview, IEEE Circ. Dev. Mag. 5 (1989) 19–26.
[193] F. Busetti, Simulated Annealing Overview. http://www.geocities.com/francorbusetti/saweb.pdf, (last accesses on October 1, 2016).
[194] R. Dwight, Z.H. Han, Efficient uncertainty quantification using gradient-enhanced kriging, in: 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference 17th AIAA/ASME/AHS Adaptive Structures Conference, Palm Springs, CA, May 2009.
[195] J.H. De Baar, R.P. Dwight, H. Bijl, Improvements to gradient-enhanced Kriging using a Bayesian interpretation, Int. J. Uncertain. Quantif. 4 (2014) 205.
[196] J.H. De Baar, Stochastic Surrogates for Measurements and Computer Models of Fluids (Ph.D. thesis), Delft University of Technology, Delft, Netherlands, 2014.
[197] K.R. Dalbey, Efficient and Robust Gradient Enhanced Kriging Emulators, Sandia Report, 2013 (SAND 2013-7022).
[198] H. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, V. Vapnik, Support vector regression machines, Adv. Neural Inf. Process. Syst. 9 (1997) 155–161.
[199] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Process Lett. 9 (1999) 293–300.
[200] K.Y. Chen, L.S. Chen, M.C. Chen, C.L. Lee, Using SVM based method for equipment fault detection in a thermal power plant, Comput. Ind. 62 (2011) 42–50.
[201] C. Kalathakis, C. Romesis, N. Aretakis, K. Mathioudakis, Fault diagnosis of thermal turbomachines using support vector machines (SVM), 2013 (AIAA Paper: ISABE-2013-1324).
[202] Z. Guo, L. Song, Z. Zhou, J. Li, Z. Feng, Multi-objective aerodynamic optimization design and data mining of a high pressure ratio centrifugal impeller, ASME J. Eng. Gas. Turbines Power 137 (2015) 092602.
[203] Y. Lian, M.S. Liou, Aerostructural optimization of a transonic compressor rotor, J. Propuls. Power 22 (2006) 880–888.
[204] J. Backhaus, M. Aulich, C. Frey, T. Lengyel, C. Voß, Gradient enhanced surrogate models based on adjoint CFD methods for the design of a counter rotating turbofan, in: Proc. ASME Turbo Expo, 2012 (ASME Paper GT 2012–69706).
[205] W. Yamazaki, M. Rumpfkeil, D. Mavriplis, Design Optimization Utilizing Gradient/Hessian Enhanced Surrogate Model, 2010 (AIAA Paper AIAA 2010-4363).
[206] Picture cited from website (https://ecfd.nlr.nl/explore-generic-models/).