Optimizing the Number of Airfoils in Turbine Design Using Genetic Algorithms

José M. Chaquet[1,⋆], Enrique J. Carmona[2], and Roque Corral[1]

[1] Technology and Methods Dep., Industria de TurboPropulsores S.A.
Avda. de Castilla 2, 28830, San Fernando de Henares, Madrid, Spain
Tel.: +34 91 2079147, Fax: +34 91 2079411
[2] Dpto. de Inteligencia Artificial, Escuela Técnica Superior de Ingenería Informática,
Universidad Nacional de Eduación a Distancia, Madrid, Spain
jose.chaquet@itp.es, ecarmona@dia.uned.es, roque.corral@itp.es

Abstract. A method for optimizing the number of airfoils of a turbine design is presented. The optimization consists of reducing the total number of airfoils meanwhile a set of geometric, aerodynamic and acoustic noise restrictions are fulfilled. It is described how is possible to reduce the problem degrees of freedom to just one variable per row. Due to the characteristics of the problem, a standard Genetic Algorithm has been used. As a case study, a real aeronautical Low Pressure Turbine design of 6 stages has been optimized.

1 Introduction

A turbine of a gas turbine engine is a device that extracts work from a pressured gas stream. It is normally made up of three modules, called HPT, IPT and LPT (*High*, *Intermediate* and *Low Pressure Turbine*). The extraction of work from the fluid is done by means of several aerodynamic surfaces called *airfoils* which are placed in an annular way forming *rows*. A turbine *stage* is formed by two consecutive rows, called *stator* and *rotor*. Stator airfoils are called *vanes*, meanwhile rotor airfoils are called *blades*. The stator is attached to the casing and directs the flow towards the rotor, meanwhile the rotor transmits the power to the turbine *shaft*. The number of airfoils of a row is called *NumberOff*.

The design process of an aeronautical turbine is a very challenging task. A LPT can contribute with one third to the total weight and with up to 15% to the total cost [1]. A lot of different constraints must be taken into account when designing the LPT airfoils and usually the final decision on the optimum particular configuration requires a trade-off among different requirements.

In this work it is presented a method for optimizing the NumberOffs of a turbine. The optimization consists of reducing the total number of airfoils meanwhile a set of geometric, acoustic and aerodynamic restrictions are fulfilled. It will be demonstrated that is possible to reduce the problem *Degrees of Freedom*

⋆ Corresponding author.

N. García-Pedrajas et al. (Eds.): IEA/AIE 2010, Part III, LNAI 6098, pp. 169–178, 2010.

(DoF) to just the NumberOff for each row. The approach adopted to solve the optimization problem uses a Genetic Algorithm (GA).

There are several applications in the literature that make use of GAs in the design process of gas turbine components, like the control system unit [2,9], blade cooling holes [3,5,6], the combustor [4,10], rotor system [7] or the 2D and 3D design of airfoils [8,11]. Some applications use standard GA [4,8,11], while others implement specific GA, as Multi-objective GA (MOGA) which evolves a Pareto-optimal solution [2,7,9,10]. In some methods the initial strategy involves the identification of high performance (HP) regions of conceptual design spaces and the extraction of relevant information regarding the characteristics of the solutions within these regions [3,5]. HP regions are rapidly identified using the COGA approach (*Cluster-oriented* GA). Another special GA used is called GAANT, which is based upon ant colony concepts and genetic algorithms [5]. Other possible approach is a Generalized Regression GA (GRGA) which explores the relationship among the variables of the solutions belonging to any continues portion of the Pareto front using non-linear multivariable regression analysis [6].

In this paper, first a description of the problem to solve is presented (section 2). It will be shown how to reduce the problem DoF. Then the GA approach will be described (section 3) and the results obtained in the optimization of a real 6 stage aeronautical gas turbine are presented (section 4). Finally, the conclusions and future works are given (section 5).

2 Problem Description

The problem consists of the turbine total number of airfoils minimization for a given flow-path (Fig. 1a) and aerodynamic exit angles. The minimization process has to fulfil a set of aerodynamic, acoustic and geometric restrictions that may be reduced to a set of explicit analytical expressions. As a consequence, both the objective function and the restrictions are extremely fast to evaluate.

In order to parametrize the problem, a simplified geometry will be used approximating each row by a rectangle (Fig. 1b). For a turbine of M number of rows, each row is defined with only 5 parameters: NumberOff (N_i), gap (g_i), chord (c_i), mean radius (R_i) and span (S_i) where i goes from 1 to M. The mean radius is the distance of the middle point of the row to the turbine axis. It is also needed one global variable, L, which is the total axial length of the turbine. The turbine inner and outer annuli are supposed to be optimized in an outer loop and in this exercise are kept constant. Therefore the mean radius and the span of all the rows are constant. NumberOffs, gaps and chords will be modified in order to find optimum feasible configurations. Gaps for row i is the distance between the trailing edge of row i and the leading edge of next row $i+1$, or the exit station for the last row. The initial gap, g_0, is defined as the distance between the inlet station and the leading edge of first the row (figure 1b).

The geometric constraints are defined with the following parameters for each row: maximum aspect ratio (MA_i), minimum pitch to chord ratio (mPC_i), maximum pitch to chord ratio (MPC_i), minimum gap (mG_i), minimum gap to chord

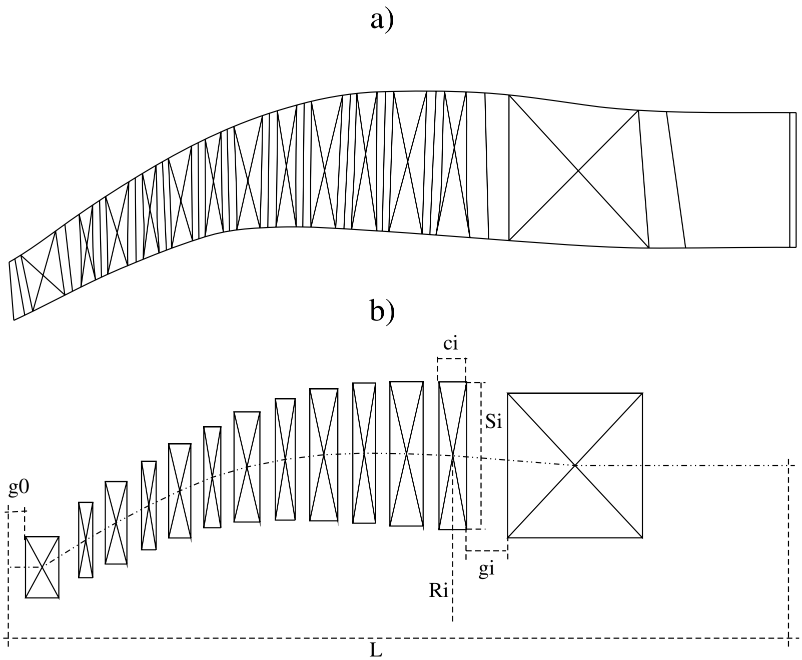

Fig. 1. Real geometry (a) and simplified one (b)

ratio (mGC_i), maximum gap to chord ratio (MGC_i), maximum NumberOff (MN_i) and the NumberOff for each package (P_i). The maximum aspect ratio should be limited by mechanic and flutter response. The pitch to chord ratio is limited in order to maintain Zweiffel coefficient bounded. Gaps are bounded in order to avoid mechanical interferences and by noise restrictions. The package parameter imposes that the NumberOff must be multiple of P_i. For the inlet gap g_0, two constraints are given for bounding its value between a minimum and a maximum value: mG_0 and MG_0.

It is well known that one way of reducing the generation of noise associated to pure tones is to force that the NumberOffs ratio for two consecutive rows lies within some specific intervals [1]. When the NumberOff ratio fulfills this conditions, the acoustic wave amplitudes decrease with the axial distance, the stage is said to be *cut-off* and the perturbations do not propagate outside the turbine. The cut-off condition depend as well on the flow variables, but in our problem these are assumed to remain constant. Noise constraints are given by four parameters: α_i, β_i, γ_i and δ_i. These parameters define two intervals $[\alpha_i, \beta_i]$ and $[\gamma_i, \delta_i]$ where the ratio of NumberOff of row i and row $i+1$ must be located. Normally $0 \leq \alpha_i \leq \beta_i \leq 1 \leq \gamma_i \leq \delta_i$. When both ranges are used, the configuration is called Mixed cut-off. For *Direct* cut-off mode, $[\alpha_i, \beta_i]$ interval is chosen for even rows and $[\gamma_i, \delta_i]$ for odd rows, therefore it will be more vanes than blades. The opposite is chosen for *Reverse* cut-off mode.

Putting all together, the mathematical problem consist of finding the M positive integer numbers N_i and the M positive real numbers c_i and g_i which fulfill the following constraints for $i \in [1, M]$

$$\frac{S_i}{c_i} \leq MA_i \;, \tag{1}$$

$$mPC_i \leq \frac{2 \cdot \pi \cdot R_i}{N_i \cdot c_i} \leq MPC_i \;, \tag{2}$$

$$mG_i \leq g_i \;, \tag{3}$$

$$mGC_i \leq \frac{g_i}{c_i} \leq MGC_i \;, \tag{4}$$

$$N_i \leq MN_i \;, \tag{5}$$

$$N_i \% P_i = 0 \;, \tag{6}$$

$$mG_0 \leq g_0 \equiv L - \sum_{i=1}^{M} (c_i + g_i) \leq MG_0 \;, \tag{7}$$

$$if (i \neq M \;\&\; Mixed) \; \frac{N_i}{N_{i+1}} \in [\alpha_i, \beta_i] \cup [\gamma_i, \delta_i] \;, \tag{8}$$

$$if \, \{i \neq M \;\&\; (Direct \,\&\, i\%2 = 0) \; or \; (Reverse \,\&\, i\%2 = 1)\} \; \frac{N_i}{N_{i+1}} \in [\alpha_i, \beta_i] \;, \tag{9}$$

$$if \, \{i \neq M \;\&\; (Direct \,\&\, i\%2 = 1) \; or \, (Reverse \,\&\, i\%2 = 0)\} \; \frac{N_i}{N_{i+1}} \in [\gamma_i, \delta_i] \;. \tag{10}$$

In equation (6) the symbol % means the remainder of integer division. Equation (7) computes the first gap g_0 and it imposes that g_0 must be in between mG_0 and MG_0.

Taking into account the three parameters for each row (N_i, c_i and g_i), there are $3M$ DoF. We will show that the problem may be reduced to that of finding the M DoF associated to the number of airfoils for each individual row.

Fig. 2a displays Gap-Chord space. The shaded region represents g_i and c_i feasible pairs where constraints (3) and (4) are represented. Points A and A' have the same chord, but A' has the minimum possible gap. The same happens with points B and B'. If point A is feasible regarding to all constraints except those in equation (7), point A' will be feasible as well. But point A' could be considered better than A because gives more room to other rows to increase their gaps and chords. For that reason and regardless of other considerations, gaps will be set to the minimum for a given chord:

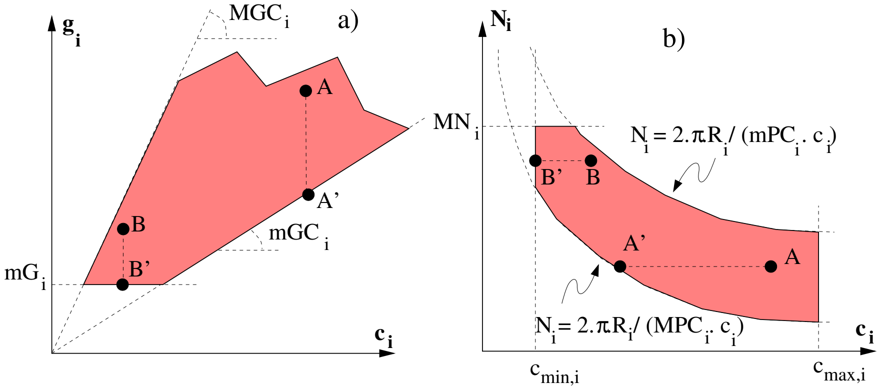

Fig. 2. Feasible domain spaces (shaded area): Gap-Chord (a) and NumberOff-Chord (b)

$$g_i \equiv g_i(c_i) = \max\left(mG_i, mGC_i \cdot c_i\right) \ . \tag{11}$$

In Fig. 2b a typical NumberOff-Chord diagram for a specific row is plotted. The shaded region represents c_i and N_i feasible pairs without taking into account the Noise constraints. Using constraints (1), (2), (3) and (4) we obtain:

$$c_{min,i} = \max\left(\frac{S_i}{MA_i}, \frac{mG_i}{MGC_i}, \frac{2 \cdot \pi \cdot R_i}{MN_i \cdot MPC_i}\right) \ . \tag{12}$$

Using (7), (11) and (12) expressions it is possible to compute for each row a maximum chord considering that the rest of rows have their minimum chords. It is important to notice that this limit is not absolute, but it depends on the rest the turbine rows. It is an upper limit, but will decrease if at least one row has his chord larger than his minimum chord.

$$c_{max,i} = L - \sum_{j=1, j\neq i}^{M} \left(c_{min,j} + g_j(c_{min,j})\right) - mg_0 \ . \tag{13}$$

From Fig. 2b we can argue that if point A is feasible regarding all constraints, point A' will be feasible as well. A' is considered better because it has the minimum chord for a given NumberOff. Smaller chords give more room to other rows. The same consideration may be done for points B and B'. Then, given a NumberOff, the optimum chord can be chosen using the following expression:

$$c_i \equiv c_i(N_i) = \max\left(c_{min,i}, \frac{2 \cdot \pi \cdot R_i}{N_i \cdot MPC_i}\right) \ . \tag{14}$$

Finally, the range of N_i to explore may be derived from expressions (2), (5) and (12):

$$N_{max,i} = floor_{P_i}\left(\min\left(MN_i, \frac{2 \cdot \pi \cdot R_i}{mPC_i \cdot c_{min,i}}\right)\right) \ , \tag{15}$$

$$N_{min,i} = ceil_{P_i}\left(\min\left(N_{max,i}, \frac{2 \cdot \pi \cdot R_i}{MPC_i \cdot c_{min,i}}\right)\right) . \quad (16)$$

In these expression $floor_{P_i}()$ function is the largest integer value not greater than the argument and multiple of P_i. Function $ceil_{P_i}()$ is the smallest integer value not less than the argument and multiple of P_i.

Summarizing, it has been demonstrated than we can reduce the problem to only M DoF, which will be the number of airfoils of every row, N_i. Once N_i is known, the c_i value can be calculated from expression (14) and, in turn, the g_i value can be derived from expression (11).

3 Genetic Algorithm

Once it has been demonstrated that each design configuration is determined by a set of NumberOffs, it could be possible to perform an exhaustive search computing all the possible configurations. Using expressions (15) and (16), the number of configurations to explore will be $\prod_{i=1}^{M} \left(N_{max,i} - N_{min,i}\right)/P_i$. As it will be showed in section 4, huge numbers appear in real problems.

Due to the multiple restrictions, it is difficult to define a continuous and derivable optimization function. Therefore methods based in gradient of the optimization function are not recommended. Several characteristics of the problem makes appropriate the use of a Genetic Algorithm (GA). First of all, it is easy to transform a *Constrained Optimization Problem* (COP) into a *Free Optimization Problem* (FOP). Secondly, the optimization function is not necessary to be continuous. Thirdly, the solution codification is made easily using a numeric vector.

The first step for defining a GA is to link the *real* world to the GA world. Objects forming possible solutions within the original problem context are referred to as *phenotypes*, while their encoding are called *genotypes*. In our problem, the phenotypes are vectors of natural numbers with the NumberOff for each row. Each NumberOff only can change in the range given by (15) and (16). The encoding of each genotype is a bit string for each NumberOff. The number of bit strings will be M, one for each row. Each bit string is called a *gene*. Each gene may have a different number of bits

$$B_i = ceil_1\left[\left(\ln\frac{N_{max,i} - N_{min,i}}{P_i}\right)/\ln 2\right] . \quad (17)$$

The way of decoding the genotype to the phenotype consists of obtaining the integer number n_i from the bit string. A Gray coding is used instead of the usual binary coding because of its advantages, described extensively in the literature [12]. Then, the NumberOff will be $N_i = N_{min,i} + n_i \cdot P_i$. Gaps and chords are obtained using the expressions (11) and (14). Maybe it will be necessary to modify the gaps in order to fulfill restriction (7).

As it was mentioned in section 2, equation (11) does not take into account the constraint (7). A repairing process may be necessary if, in obtaining the phenotype, g_0 does not meet that constraint. If $mG_0 \leq g_0 \leq MG_0$ the fixing is not necessary. On the other hand, if $g_0 \leq mG_0$ it is not possible to repair and the individual receives a high penalty in its fitness. If $g_0 > MG_0$ a repairing process is needed. The repairing process is made in the phenotypic space and this consists of distributing the amount $\triangle g = g_0 - MG_0$ among the the rest of the gaps maintaining the constraints $g_i \leq MGC_i \cdot c_i$.

The next step in the design process of the GA is to choose the *fitness* function. The role of the fitness function is to represent the requirements to be optimized. The fitness function implemented transforms our initial COP into a FOP. With the representation of individuals adopted, all the constraints are satisfied but (7), (8), (9) and (10). Being M the number of rows, a penalty function F_R is defined as 0 for individuals placed in the feasible regions and negative values increasing exponentially in the following way

$$F_R = \begin{cases} 1 - \exp\left(\lambda \frac{mG_0 - g_0}{L}\right) + \sum_{i=1}^{M} F_i & if\ g_0 < mG_0 \\ \sum_{i=1}^{M} F_i & if\ mG_0 \leq g_0 \leq MG_0 \\ 1 - \exp\left(\lambda \frac{g_0 - MG_0}{L}\right) + \sum_{i=1}^{M} F_i & if\ MG_0 < g_0 \end{cases}, \tag{18}$$

where g_0 is computed using expression (7). The value of constant parameter λ is used for modulating the exponential decreasing in the unfeasible regions. Its value is taken experimentally and does not have a big influence in the performance of the algorithm. F_i deals with the noise restrictions depending of the cut-off mode. For instance, for Direct cut-off mode and $i\%2 = 0$ or Reverse cut-off mode and $i\%2 = 1$:

$$F_i = \begin{cases} 1 - \exp\left[\lambda\left(\alpha_i - \frac{N_i}{N_{i+1}}\right)\right] & if\ \frac{N_i}{N_{i+1}} < \alpha_i \\ 0 & if\ \alpha_i \leq \frac{N_i}{N_{i+1}} \leq \beta_i \\ 1 - \exp\left[\lambda\left(\frac{N_i}{N_{i+1}} - \beta_i\right)\right] & if\ \beta_i < \frac{N_i}{N_{i+1}} \end{cases}. \tag{19}$$

In order to perform an simple optimization among the feasible individuals, the fitness functions is defined:

$$F_N = \begin{cases} F_R & if\ F_R < 0 \\ 1/\sum_{i=1}^{N} N_i & if\ F_R \geq 0 \end{cases}. \tag{20}$$

GA uses a population of possible solutions. The parent selection mechanism implemented here is the *tournament method*, i.e. k individuals with replacement are chosen randomly from the population and the final individual chosen will be the best of these k in terms of their fitness value. Once the parents have been selected, there is a recombination probability p_r which sets if the offspring of two parents are just a copy of the parents or a real recombination is produced. Two recombination methods have been implemented: one point crossover over the genes (called *gene crossover*) and one point crossover over the bits for each gene (*bit crossover*). Another parameter that control the algorithm is the mutation probability p_m. After doing the crossover of the parents, the offspring is mutated. The mutation is done in each gene swapping a random bit. A generational model is used, so for each generation all parents are changed by their offspring. It has been implemented the feature of elitism, swapping the worst individual after the mutation operator is applied by the best individual of the previous generation.

The initialization is made by taking a random representation of possible solutions from the design space and carrying out fitness evaluations on all the individuals.

4 Case Study and Experimental Results

The GA described in the previous section has been applied to an aeronautical LPT made up of 6 stages and an *Outlet Guide Vane* (OGV), which makes a total of 13 rows (Fig. 1). The total number of airfoils of this turbine is 1486 and its axial length is 1.225 meters. The restrictions imposed has been the same used by the design team, so the study case must be considered a real one. Only the Reverse cut-off condition is considered feasible.

Using the expressions (15) and (16), it is possible to compute the possibilities of each NumberOff. The number of possible values varies greatly between each row. The minimum number is 1 possibility for the first and last row, and the maximum value is 63 for the fourth row. Multiplying all the possibilities we obtain a total number of configurations to explore of $2.8 \cdot 10^{18}$. If we use a exhaustive search and consider that each configuration was evaluated in 10^{-6} seconds, the computing time would be 73117 years. So an exhaustive search can not be used in this case.

The parameters of the GA have been chosen by trial and error. The best results are obtained for a population of $5 \cdot 10^4$ individuals, 50 generations, $k = 5$, $p_r = 0.8$, $p_m = 0.01$, gene crossover and elitism. Due to the stochastic nature of the GA, the algorithm has been run 5 times. The average total NumberOff achieved has been 1461.4, with a standard deviation of 1.96. The best total NumberOff has been 1460. The average time needed has been 2 minutes and 22 seconds using a 2.40 GHz Intel Core Duo machine with a 4 GB of RAM memory and a operative system Linux openSUSE 10.3.

Fig. 3a shows the NumberOff for each row of the original design and the new ones obtained by the best run of the GA. Several configurations with the same total NumberOff of 1460 have been found. All these solutions are indistinguishable by the algorithm. As is clear from the interpretation of equation (20), the evolving process involves two phases: an initial stage ($F_R < 0$) dedicated to satisfying the constraints, and a second stage ($F_R \geq 0$) devoted to minimizing the total number of airfoils once the constraints are satisfied. This can be seen in Fig. 3b, where the evolution of the best fitness of the population in each run is plotted against the generation number. Fig. 3c is an enlarged view of Fig. 3b and shows the border between these two stages. More specifically, 15 to 19 generations are needed for having at least one individual that meets all the restrictions. In Fig. 3d it is showed the new real geometry in meridional plane obtained by the GA.

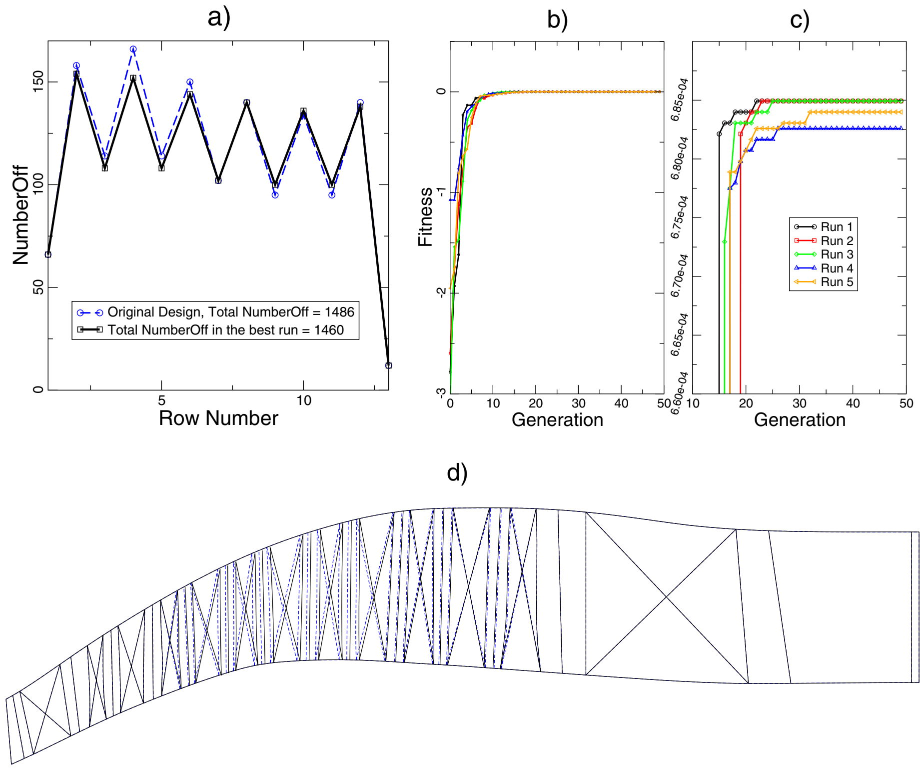

Fig. 3. Summary of experimental results: NumberOffs achieved in the best run (a), progress plots (b) and (c), where (c) shows a magnified region of (b), and optimized real geometry (continuous line) compared with the original one (dashed line) (d)

Notice that the algorithm has changed the chord and the gap of some rows in order to fulfill the constraints.

5 Conclusions and Future Work

A GA has been applied to perform the airfoil number optimization of a LPT gas turbine fulfilling a set of realistic restrictions. The turbine model used as input to the GA corresponds to the final design of a turbine made from a standard design methodology. The algorithm has reduced the total number of airfoils in 1.74%. The improvement is not bigger because the input to the GA is an already optimized final design using the same set of restrictions, so it was close to the optimum. However, from the standpoint of the real turbine design, the improvement is not negligible. With the appropriate parameters for the GA, the optimization process found the same configuration with a very low dispersion (standard deviation in the total NumberOff less than 2 blades). The time consuming of the algorithm is low, despite of dealing with a high population number.

As a future work it would be possible to consider more ambitious goals. For example, the total number of airfoils is a rough estimation of the weight and efficiency of the LPT since other factors such as the chords, spans and thickness are not taken into consideration. A more refine model should consider the trade-off between efficiency and weight to include other optimization functions.

References

1. de la Calzada, P.: Aerothermodynamic Design of Low Pressure Turbines. Aero-engine design: from state of the art turbofans towards innovative architecture, Brussels, Belgium, March 3-7. Von Karman Institute Lecture Series LS-2008-04 (2008)
2. Chipperfield, A., Flemming, P.: Multiobjective Gas Turbine Engine Controller Design Using Genetic Algorithms. IEEE Transactions on Industrial Electronics 43(5), 583–587 (1996)
3. Parmee, I.C., Bonham, C.R.: Towards the Support of Innovative Conceptual Design Through Interactive Designer/Evolutinary Computing Strategies. Artificial Intelligence for Engineering Design, Analysis and Manufacturing 14(1), 3–16 (2000)
4. Rogero, J.M., Tiwari, A., Munaux, O., Rubini, P.A., Roy, R., Jared, G.: Applications of evolutionary algorithms for solving real-life design optimisation problems. In: 6th Int. Conf. Parallel Problem Solving from Nature, Paris, France (September 2000)
5. Parmee, I.C.: Evolutionary Computing Strategies for Preliminary Design Search and Exploration. In: Proceedings US United Engineering Foundation's 'Optimisation in Industry' Conference, Tuscany, Italy, 2001. Springer, London (2002)
6. Roy, R., Tiwari, A., Corbett, J.: Designing a Turbine Blade Cooling System Using a Generalised Regression Genetic Algorithm. CIRP Annals 52(1), 415–418 (2003)
7. Angantyr, A., Aidanpaa, J.O.: A Pareto-Based Genetic Algorithm Search Approach to Handle Damped Natural Frequency Constraints in Turbo Generator Rotor System Design. J. Eng. Gas Turbines Power 126, 619 (2004), doi:10.1115/1.1760529
8. Ahuja, V., Hosangadi, A.: Design Optimization of Complex Flowfields Using Evolutionary Algorithms and Hybrid Unstructured CFD. In: 17th AIAA Computational Fluid Dynamics Conference, Toronto, Ontario, Canada, June 6-9 (2005)
9. Parrilla, M., Aranda, J., Dormido-Canto, S.: Parallel Evolutionary Computation; Application of an EA to Controller Design. In: Mira, J., Álvarez, J.R. (eds.) IWINAC 2005. LNCS, vol. 3562, pp. 153–162. Springer, Heidelberg (2005)
10. Elliott, L., Ingham, D.B., Kyne, A.G., Mera, N.S., Pourkashanian, M., Wilson, C.W.: A Novel Approach to Mechanism Reduction Optimization for an Aviation Fuel/Air Reaction Mechanism Using a Genetic Algorithm. J. Eng. Gas Turbines Power 128, 255 (2006), doi:10.1115/1.2131887
11. Varvill, R., Paniagua, G., Kato, H., Thatcher, M.: Design and Testing of the Contra-Rotating Turbine for the Scimitar precooled Mach 5 Cruise Engine. In: IAC-08-C4.5.3 (2008)
12. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Springer, Heidelberg (2007)