OPTIMIZATION OF ROTATING DISKS VIA GENETIC ALGORITHM

Francesco Scaramuzzino
Second University of Naples, Aversa, Italy

Luigi Iuspa
University of Naples "Federico II" ,Italy

Introduction

Structural optimization methodology represents, in aerospace research, a consolidated and almost a mandatory tool to analyze some critical problems generated by conflicting project parameters or severe requirements needed.

As known, the optimization problem is set, in mathematical terms, as:

Find the minimum [maximum] of an objective function F(X) with $X = (x_1, x_2, .., x_n)$ subject to the following constraints:

$$x_{j\,min} < x_j < x_{j\,max} \quad (j = 1, n) \tag{1}$$

$$g_{k\,min} < g_k(X) < g_{k\,max} \quad (k = 1, m) \tag{2}$$

Where the m functions g_k allowing functional constraints for design are usually named as "state functions".

The traditional methods developed for solving this problem, as the simplexum method, gradient and penalty function methods, Powell's method and others are generally suffering from important limitations with relative lack of robustness and/or efficiency[1][2][3][4].

In fact, each of these methods stops as soon as a minimum is found, without the possibility to distinguish among local and global minima.

Additionally, severe limitations for objective and state functions are set.

These must be often continuous, together with their derivatives up to some orders.

For these reasons the use of Genetic Algorithm based procedures is a more and more diffuse practice in optimization problems.

With an absolutely different approach, emulating natural laws of biological evolution, Genetic Algorithms permit to overcome the above-mentioned limitations and offer high probability of convergence to the absolute minimum also if discrete and/or multiconnected domain based mathematical functions are used[7].

The principal limitation of Genetic Algorithms, expecially if used within F.E.M. or C.F.D. methods, is an excessive amount of computing resources, due to high number of loops needed.

To make this problem worse, there is often no possibility to use a such type of approximate model for objective or state functions, because a lack of accuracy or again, highly time consuming activity happens, with loss of utility.

The present paper, analyzing a specific optimization example, shows a general Genetic Algorithm based methodology, for Turbomachinery design and F.E.M. rotating disks analysis.

In many problems of rotating disks design, some difficulty occurs when traditional optimization methods are used.

An example is objective or state function which represents a quantity that may change geometric location from loop to loop.

This happens,for example, when in shape optimization process maximum stress is used as state variable.

In such case, as project parameters change, the stress peak position will be modified, generating a poor state variable approximation.

For this situation, the preferred solution is to define some key locations, monitoring stress in topic elements, despite of the difficulty to reveal and to choose enough of these points.

Of course, this solution is not possible for a location dependent objective function which must be naturally unique.

Another typical situation where some troubles occur, when traditional algorithms are applied, is using equality relations in constraints equations.

This happens, for example, when a 2D axial-symmetric F.E.M. model is used to simulate an highly cyclic-symmetric configured geometry, (presence of ovalized holes and/or 2D modeled blades) and an equivalent distribution of element properties, according to an experimentally determined foundamental frequency, must be found.

In this case, two state variables for the same quantity (frequency) are needed, with dummy, wide enough ranges for smooth approximation, and braketed on the desired value.

Otherwise, using one state variable with a very small range centered on the target value, a function approximation could be not realized.

The negative effect that occurs using this trick is a great number of infeasible generated solution and a low speed convergence process.

The use of a Genetic Algorithm in these critical situations is a very suitable solution, because the high number of analyzed configurations is balanced by dramatic loss of efficiency of traditional algorithms and moreover the superior capability of Genetic Algorithms in finding absolute optimum is used.

The Genetic Algorithm Optimization

The suggested methodology is illustrated with an example concerning the optimization design of an aeronautical turbine.

The turbomachinery element object of study, is an A.P.U. (Auxiliary Power Unit) turbine equipping an high performance aircraft.

The desired objective is to maximize the low-cycle fatigue life.

To achieve this goal, the objective function has been defined as the radial stress peak in the disk subject to centrifugal load[5].

Project variables are geometric parameters defined in critical areas, with modifying profile shape capability.

The moment of Inertia Iyy (with y symmetry axis) has been defined as state variable to satisfy severe start-up requirements.

Parametric Geometry

In Figures 1 and 2 respectively, the axial-symmetric turbine shape and a detail of parametric regions of interest for optimization are showed.

The stress peak to be minimized is found near at two specific sectors on "Back-Face" profile,respectively defined as "Upper Back-Face" and "Bottom Back Face".

In these two zones of the disk, if also small shape modifications are made, a sensible and fast stress alteration takes place.

To reach a better high gradient stress control on Back Face, the shapes have been modeled with cubic spline curves, because with this type of geometric entity, smooth profiles, according to the scope of optimization and best stress distribution are easily obtained.

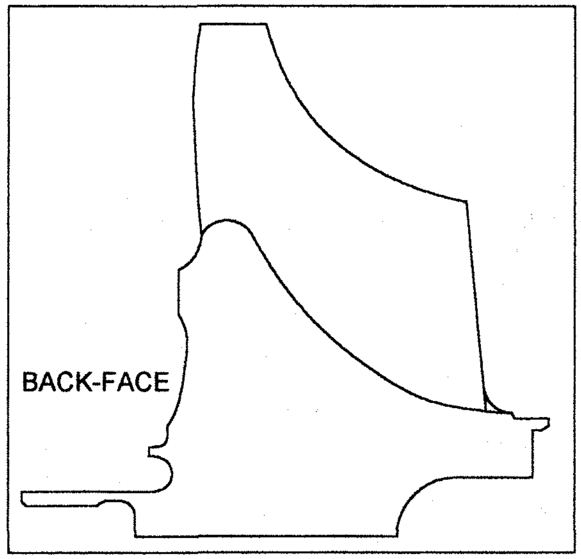

Figure 1
A.P.U. Turbine axial-symmetric profile

The Upper Back-Face spline is formed by six control points.

First and last point are a boundary for the parametric profile, so they are fixed (not parametric) entities.

To define a geometry, four control points must be set.

This is done by defining eight point coordinates in oblique axis system with origin onto A point, X axis toward AB chord direction and Y axis parallel to turbine axis.

For convenience, X is defined as radial direction and Y as axial. The eight coordinates of parametric points must be organized into an ordered and hierarchical system.

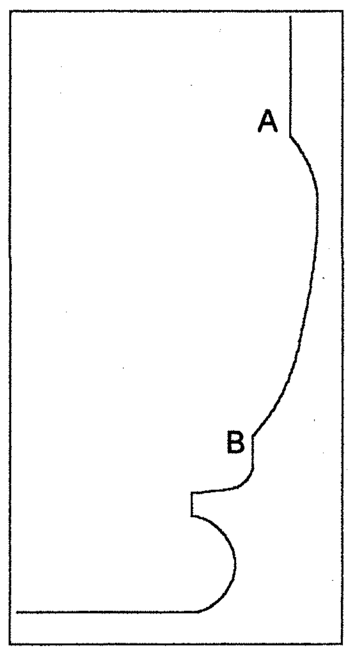

Figure 2:Upper Back-Face parametric profile (AB curve)

In fact, the connected spline points forms an ordered list; if these points are made parametric, the radial coordinate of a Pi point is needed to be not greater than next or smaller than previous P_j point coordinate.

Therefore Xi coordinates must be defined under the following constraints:

$$x_{i-1} < x_i < x_{i+1} \quad i = 1,4 \tag{3}$$

$$\sum_{i=0.4} (x_{i+1} - x_i) = x_5 = \left|\overline{AB}\right| \tag{4}$$

If the one of the above conditions cannot be verified, the spline would be trasformed into a laced curve with one or more coincident knots and therefore totally different than the needed one.

According to the last, a definition of radial design variables as adimensional quantities is required because if defined as relative increments, a functional dependency condition would be taken.

The problem is solved by breaking the AB chord with a middle point P1 guided by adimensional and normalized variable X1.

The last of two segments obtained with the cut, is broken too with X2 parameter, generating P2 point and so on, until the last division is made (see figure 3.A).

Radial geometric coordinates of spline base points therefore, are expressed by a recursive form

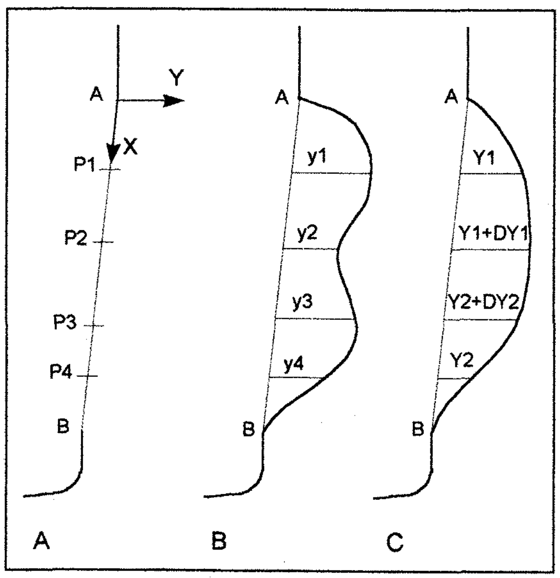

Figure 3-A-B-C

as follows:

$$\begin{aligned} x_0 &= 0 \\ x_1 &= LX_1 \\ x_2 &= (L - x_1)X_2 + x_1 \\ x_3 &= (L - x_2)X_3 + x_2 \\ x_4 &= (L - x_3)X_4 + x_3 \\ x5 &= L \end{aligned} \tag{5}$$

where $x_1,..x_5$ are the radial geometric locations of spline control points, $X_1,..,X_5$ are the radial project variables with a normalized range [0,1] and L is the AB chord length.

Also if no specific design limitation is really needed for axial coordinates, these are conveniently conditioned, so generated splines cannot have concavity toward outer side.

This is done to avoid any clearly useless configuration for the examined problem, with the resulting low convergence speed (see figure 3.B).

Constraints for axial varables are made, for a interior point coordinates, by using relative increment parameters regard to exterior ones as showed in the following expressions:

$$\begin{aligned} y_0 &= 0 \\ y_1 &= Y_1 \\ y_2 &= Y_1 + DY_1 \\ y3 &= Y_2 + DY_2 \\ y_4 &= Y_2 \\ y_5 &= 0 \end{aligned} \tag{6}$$

So, the axial parameters projects are the quantities:Y1, DY1, DY2 and Y2 (see figure 3.C).

Bottom Back-Face

Shape profile in the Bottom Back-Face is again an high gradient stress region.

In this zone, a nearly-elliptical shape profile is the only one allowed by assembly and functional limitations, thus it was modeled with a one variable parametric geometry (Y6 parameter), which consists of three based points spline with both extreme points fixed and a parametric middle point allowing curve eccentricity (see figure 4).

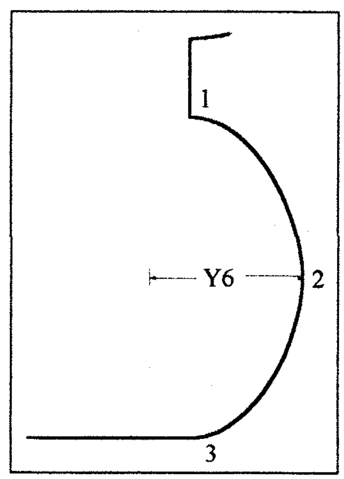

Figure 4: Bottom Back-Face parametric profile

Mesh Generation

To achieve high quality results of an optimization procedure, some fundamental requirements for meshing operations are needed.

In first place, each generated mesh must have no more geometric configuration, distortion (principally shape and aspect ratio) than amount permitted by the working ranges of element form functions[6](In this case F.E.M. model is meshed with axial-symmetric, iso-parametric quad elements).

At same time, it is necessary to assure a constant mesh density with configuration variations and to avoid time consuming activity, if in a specific non parametric zone an exceeding mesh than analysis requirements is found, a coarse one must be set.

Therefore an algorithmically generated mesh has been used for F.E.M. modelling.

To avoid shape distorctions, the contiguous patches to the parametric profiles are generated via homotethic duplicates of spline profiles, with two selectable modalities: the one defines homotethic centre using two special coordinate systems located onto A and B extreme spline points; the last uses a varying homotethic centre according to slope of tangency at ending of spline points (see figure 3).

For each patch built near the parametric spline profile, the number of mesh elements along the radial direction is chosen, for each loop, according to the interesting arc length and the desired (constant) mesh density.

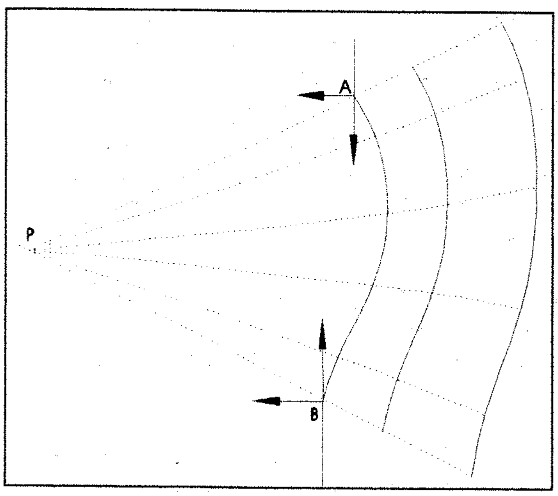

Figure 5: Homothetic built patches

For this one a calibration operation, allowing punctual stress analysis into critical zones of the disk, was preliminary performed.

The mesh built algorithm is finally completed with a sub-procedure making selective thinning out for zones far by those of interest, where a course mesh is only needed.

Alternatively this sub-procedure can make a substructure grouping for non-parametric regions.

Genetic Algorithm Definitions

The tecnique of Genetic Algorithms is a totally different procedure than those usually employed for "traditional" optimization methods.

Reaching objective function optimum is done, in this case, with a simulation of typical evolution process of biological systems, when put in competition in a bounded environment[7].

Time succession of various generations of individuals, brings about a refining of mechanisms which are used to achieve a specific purpose, until a solution or a set of optimal solutions are reached.

Definitions

Terms which are often used to describe the previous analogy are shortly mentioned.

A possible solution formed by the n-vector of design variables values is defined as "individual" and the whole design parameter set forms the "chromosomical" structure.

String expressed into the alphabet allowing the more extensive coding (usually binary), for project variables, represents the "genetic" structure of an individual and the single digits (bits) are defined "genes".

A group of a usually fixed quantity of individuals forms a "population".

Temporary configurations of a population during the evolutioning process represent "generations" of individuals.

The degree of adaptation, that is the conforming of a generic string to the examining problem, which therefore is the environmental context, is measured with a fitness index.

Basic working

The evolutioning mechanics are based on the following phases:

Strings belonging to a population are chosen to form pairs with a fitness index proportional probability.

With a specific "Crossover" operator, the genetic patrimony of strings forming pairs are used to create two new offspring individuals, with blended genetic information inside.

Nextly, a "Mutation" operator is used to obtain some randomly generated modifications for genes forming strings, with the purpose to increase variety into the population.

A "Clonation" operator is finally used to copy strings according to their fitness value into next generations.

The use of these three operators allows new generations of an assigned starting population to be formed.

Fitness, which is a objective function value correlated quantity, is observed to become better during the generation progress, according to a near-exponential law.

At the end of process, the final generation bit string is a nearly-optimum coded solution for the given problem.

Genetic Algorithm Implementation

To examine the above-mentioned problem, a Fortran genetic algorithm based procedure has been created and interfaced with F.E.M. Ansys program.

The optimization algorithm is formed by the following parts:

1) A starting, zero order randomly generated population is created, with N binary strings obtained by coding, scaling and merging project variable values of generic attempting vectors.

Possible meaningful individuals, or known feasible solutions for the problem, may be added, at this time, with the same modality , to the initial set.

For each individual of population, real values assumed by design variables are achieved from the scaled and codified ones with fixed resolution, from genetic strings.

The scaling mechanism is illustrated below.

$$V_{real}(i) = \frac{(x_{\max}(i) - x_{\min}(i))V_{scaled}(i)}{2^r - 1} + x_{\min}(i) \qquad (7)$$

Xmin(i) and Xmax(i) are the lower and upper limits of project variable i.

Vscaled (i) and Vreal(i) are the scaled and real values of project variable i.

r is the number of bits used for codifying.

Note that the 2^r-1 term is just the codifying resolution used for the xi variable.

2) For each individual of population built in this way, the fitness function is evaluated.

For this case the fitness is expressed by:

$$\begin{aligned} &fit(X) = V_{\max} - Rs(X) \\ &Rs(X) < V_{\max} \quad (\forall \text{ X}) \\ &\text{with } X = (x_1, x_2, .., x_9) \end{aligned} \qquad (8)$$

Vmax is a number much larger than the expected value of radial stress; Rs(X) is the objective function, or the radial stress for the X project variables vector based configuration.

This is done because Genetic Algorithms always try to maximize internal fitness value while for this situation, the minimum of the objective function is needed.

Objective function is modified with a simple penalty function to keep account of constraints related by state variable.

In this case the single used costraint variable is defined as the Moment of Inertia Ixx:

$$fit(X) = (V_{\max} - Rs(X))\frac{1}{k}$$

$$\begin{aligned} &k = \left(\frac{I - I_{\min}}{I_{\max} - I_{\min}}\right)^2 && (\text{I} < \text{I}_{\min}) \\ &k = 1 && (\text{I}_{\min} \le I \le I_{\max}) \\ &k = \left(\frac{I - I_{\max}}{I_{\max} - I_{\min}}\right)^2 && (\text{I} > \text{I}_{\max}) \end{aligned} \qquad (9)$$

Where I is the actual value of Moment of Inertia Ixx; Imin and Imax are the extremes of the allowed variation range and k is a penalty coefficient.

This penalty is applied when any limit violation of state variable Ixx range occurs, even if only the upper limit may be really overcome.

To avoid infeasible solutions generated by very slight violations of penalty functions, a bit smaller value for state function upper limit is used than the allowed one.

3) A population string sorting is made with the fitness used as index.

This operation is optimized through a pointer list structure.

4) a number M of pairs of individuals is chosen with a random mechanism which returns a larger probability as higher is the fitness expressed.

Fitness Scaling

The selection procedure for mating is completed with a fitness scaling mechanism, to avoid both a premature convergence, for initial generations, due to few much higher fitness provided strings that population mean and, near the end of iterative process, a fitness levelling, due to mean fitness increasing, with a modification of the evolution process in a random walk into the definition domain occurs.

The scaling model used is linear with raw-fitness pre-processing, to avoid negative values as follows:

$$F' = aF + b \qquad (10)$$

with a and b obtained by the following conditions:

$$F'_{average} = F_{average}$$
$$F'_{max} = hF'_{average} \quad (11)$$
$$\text{with } 1.2 < h < 2$$

(see figure 6)

To amplify fitness scaling effects, a choice between two different mating methods has been provided.

First allows no repeating of pairs built with the same parents; on the contrary the second one is able to do it.

With the first possibility a much more exchange of genetic knowledge present into a population is allowed.

In fact, a generic string with a more greater fitness value than that expressed by current population mean, will not be able to create arbitrary numbers of pairs together other strings with the same characteristics, but must necessarily mate

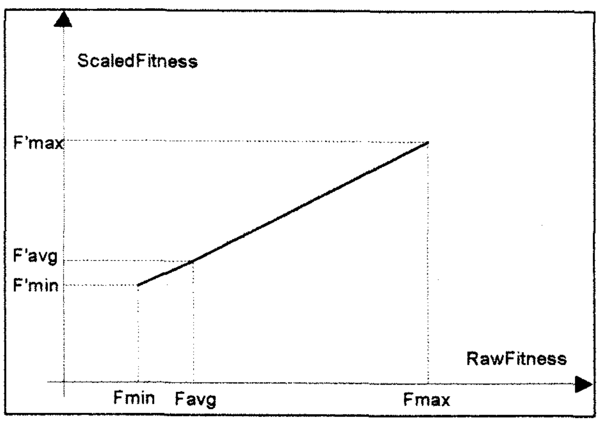

Figure 5:Fitness scaling

with others with a lower fitness value

On the contrary, using the second coupling mode, a genetic structure exchange between best strings is allowed.

5) A number of 2*M new individuals are generated with a crossover operator beginning from selected pairs.

Crossover Modality

Crossover operator has been provided with two distinct working modalities: cut and bit-to-bit crossover.

First mode makes a cut into the same position of two binary strings of coded parents and obtains two offspring individuals by sub-string swapping.

Second makes a random selection between genes (bits) of parent strings.

In both cases two complementary offsprings regard genetic knowledge are obtained (see Table 1 and 2).

1	0	1	1	0	1	0	1	parent1
0	1	1	0	0	1	0	0	parent2

1	0	1	0	0	1	0	0	offspring1
0	1	1	1	0	1	0	1	offspring2

Table 1: cut crossover example

1	0	1	1	0	1	0	1	parent1
0	1	1	0	0	1	0	0	parent2

1	1	1	1	0	1	0	0	offspring1
0	0	1	0	0	1	0	1	offspring2

Table 2: bit-to-bit crossover example

6) Mutation operator is applied with random inversion of bit values forming offspring strings.

The number of mutation is controlled by a specific probability index parameter.

7) New individuals are arranged behind the others forming the previous generation.

8) A new iteration is made returning to step 2.

For subsequent iterations than the zero order, an extended sorting is made with the N+2*M population individuals but only the best N are retained.

This is really an alternative way to apply clonation operator.

In this mode strings are not directly cloned into next generation, according to a fitness proportioned probability, but are "bequeathed" until they become obsolete and so removed.

Envelope Parameters

Genetic Algorithm has been provided of a simple envelope mechanism to modify dynamically the working parameters when optimization is in progress.

Domain formed by the number of generations of the optimization process is subdivided into four intervals through five control points.

For each of them it is possible to define crossover modality, pair choosing mode, scaling fitness parameters and mutation index.

These quantities change for every interval with a linear law (constant for discrete values).

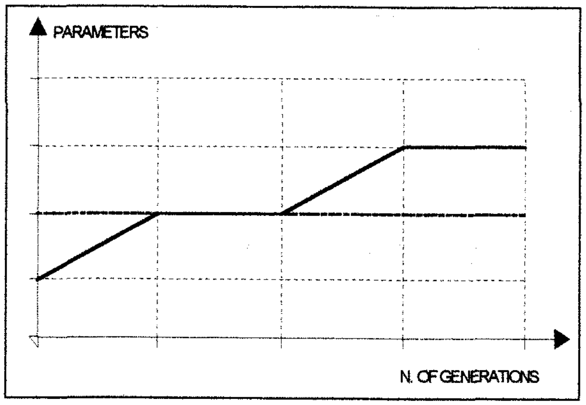

Figure7: Optimization Parameters Envelope

Optimization Procedure

Optimization procedure has been executed according to preliminary considerations with the following parameters:

N. of Individuals=30
N. of Pairs=15
N. of Generations=28

Variables	Min	Max	Bit Depth
X1	0.15	0.6	16
X2	0.15	0.8	16
X3	0.15	0.8	16
X4	0.15	0.8	16
Y1	1E-6	2.0	16
DY1	1E-6	1.0	16
Y2	1E-6	1.0	16
DY2	1E-6	1.0	16
Y6	3.1	3.6	16

Table 3: Project variables data

Section	0	1	2	3	4
Mutation Index	.01	.015	.015	.025	.045
Choose Pairs*	2	2	2	1	1
Fitness Scaling (h)	1.5	1.5	2	2	2
Crossover Type**	1	1	2	2	2

* With/without repetition= 1/2
** Cut=1; Bit to bit=2

Table 4: envelope parameters

This choice was made because a very small population size must be used, due to high resource consuming F.E.M. activity required to evaluate objective and state functions.

To maximize process efficiency and avoid as possible a premature convergence, the above-mentioned parameters were chosen with the purpose to involve, in a first time, the most part of genetic knowledge present into the population, avoiding to allow best individuals.

So, a limited possibility of mutation is allowed; cut crossover is applied to keep the genetic schemata integrity as much as possible and a severe fitness compress scaling is done.

On the contrary, near the end of iterative process, best strings become naturally advantaged and an extensive use of mutation operator is made to scan far sub-spaces of project variables domain.

So bit-to-bit crossover, amplified fitness scaling, pairs formed by same individuals and a progressive increase of mutation index are made possible.

Analysis Results

At the end of the iterative process the project variable vector obtained by decoding the best string of last generation, represents the best design set connected with the lower radial stress value according to the requirements about the Moment of Inertia.

Analysis results showed a great improvement of radial stress peak value than the starting configuration.

The stress peak was reduced of 11.24% with a value of 0.8411 times than the initial set, and a iso-stress distribution close to Back-Face parametric profiles was also obtained.

The Moment of Inertia has been found greater of 1.003 times than initial value, with a bit relative increment of 0.87%

In figure 8 the genetic algorithm fitness trend is showed.

A very interesting point is to compare these results with others obtained by the use of the penalty function based algorithm built into the Ansys program.

Due to above-mentioned problems a first optimization had realized a reduction of 0.92 times only than starting stress value.

To obtain comparable results, the use of two further sequential optimization processes was needed.

For each of them, the default starting random approximation sets were replaced with the best design set and some feasible solution sets obtained with a closeness to the optimum based selection of the previous optimization process, with the purpose to improve efficiency and convergence speed.

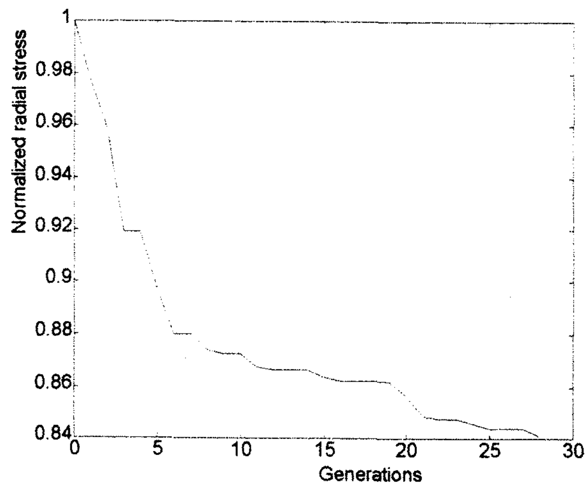

Fig. 8 Genetic Optimization Time History

In addition to these actions, an increase of penalty effects has been applied.

Despite these actions, a very high number of iterations, absolutely comparable with those generated by genetic algorithm procedure occurred and a worst best design set was reached.

A radial stress peak value of 0.889 times only than the starting one was reached with a moment of inertia 1.009 times greater.

In figure 9 a comparison between the final configuration obtained with genetic algorithm (continous line) and initial profile (dashed line) is showed.

Finally, with the aid of experimental local stress-strain approach[8] theory based diagram for assigned material and thermal condition and using the stress peak reduced value obtained with the genetic algorithm procedure, a new nearly 100% increased value of low-cycle fatigue life was determined.

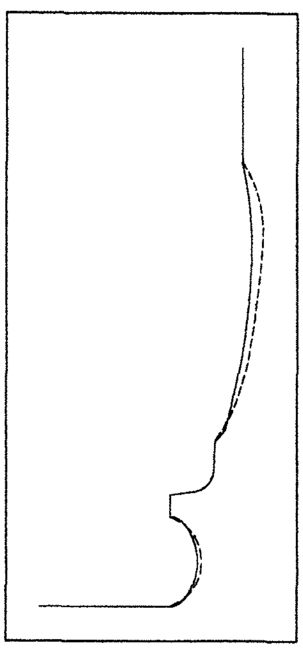

Fig. 9: Best Design Profile

Conclusions

In the present paper some problems involving design optimization of rotating disks have been discussed.

The advantage of using, in typically critical analysis, a genetic algorithm based optimization procedure was showed.

In particular, an example of an high performance A.P.U. turbine disk optimization to maximize low-cycle fatigue life with varying profile shape into critical zones was showed.

For the purpose a Genetic Algorithm based procedure interfaced with a F.E.M. analysis code and a specific algorithmically controlled adaptive mesh was created.

Genetic Algorithm was made to work well as possible with a very small size of populations and generations number.

At the end of analysis procedure, a final configuration allowing a radial stress peak reduction more than 11% and a relative increasing of low-cycle fatigue life almost of 100% was reached.

For comparision, the same analysis made with conventional optimization tools showed high inefficiency of non-genetic procedures to obtain comparable results when used for critical situations without a massive intervention of an expert user.

On the other hand, the objective difficulty to use Genetic Algorithm and high time consuming codes as F.E.M. or C.F.D. programs together, when an efficient method for approximate objective and state function is not available is also highlighted.

REFERENCE

(1) R Fletcher "Pratical methods of optimization" 2nd. Ed. John Wiley & Sons, 1987.

(2) A.J. Morris "Foundations of structural optimization: a unified approach" Ed. John Wiley & Sons, 1982.

(3) G. Dantzig "Linear programming and extensions" Princeton University press, 1963.

(4) M.J.D. Powell "An efficient method of finding the minimum of a function of several variables without calculating derivatives" Computer Journal, 1964, vol.7 n.4, pag. 303-307.

(4) K. Loffler "Die Berechnung von rotierendenSwcheiben und Schalen" Ed. Springer-Verlag, 1961

(5) S.S. Manson "Thermal stress and low-cycle fatugue" Ed. McGraw-Hill, 1966.

(6) O.C. Zienkiewicz "The finite element method" Ed. McGraw-Hill, 1977.

(7) D.E. Goldberg "Genetic algorithms in search, optimization and machine learning" Addison-Wesley, 1989.

(8) J.A. Collins "Failure of materials in mechanical designs" Wiley & Sons, 1992.