Ingeniería y Ciencia, ISSN 1794-9165
Volumen 2, número 4, páginas 29-41, septiembre de 2006

Optimal reinforcing of reticular structures[1]

Manuel Julio García[2] and Juan Santiago Mejía[3]

Recepción: 01 de diciembre de 2005 — Aceptación: 30 de mayo de 2006
Se aceptan comentarios y/o discusiones al artículo

Resumen

Este artículo presenta una aplicación de Algoritmos Genéticos (GA) y Análisis por Elementos Finitos (FEA) a la solución de un problema de optimización estructural en estructuras reticulares plásticas. Optimización estructural es usada para modificar la forma original colocando refuerzos en posiciones óptimas. Como resultado se obtuvo una reducción en el esfuerzo máximo de 14,70 % para una estructura cuyo volumen original aumento en 8,36 %. Este procedimiento soluciona el problema de optimización estructural ajustando el molde original y evitando la manufactura de un nuevo molde.

Palabras claves: optimización estructural, estructuras reticulares, estructuras reforzadas, algoritmos genéticos.

Abstract

This article presents an application of Genetic Algorithms (GA) and Finite Element Analysis (FEA) to solve a structural optimisation problem on reticular plastic structures. Structural optimisation is used to modify the original shape by placing reinforcements at optimum locations. As a result, a reduction in

[1] This is an extended version of the article presented at the nineteenth Canadian Congress on Applied Mechanics, Calgary 2003.

[2] Ph.D. Aeronautical Engineering, mgarcia@eafit.edu.co, associate professor, Department of Mechanical Engineering, EAFIT University.

[3] BSc. Mechanical Engineering, jsmejia@uiuc.edu, Research Assistant, Department of Industrial and Enterprise Systems, University of Illinois at Urbana-Champaign, IL USA.

the maximum stress by 14,70% for a structure with a final volume increase of 8,36% was achieved. This procedure solves the structural optimisation problem by adjusting the original mold and thereby avoiding the re-construction of a new one.

Key words: structural optimisation, reticular structures, reinforced structures, genetic algorithms.

1 Introduction

Structural optimisation is an expanding field for companies competing for better products or improved ways to manufacture them. Among the design problems, using reinforcement to repair structures is a solution used in practise in different fields such as Aeronautical Engineering [1] and Civil Engineering [2]. The main problem is the location of and shape of the reinforcements. Optimisation techniques has been applied to a number of these cases. In [3] a gradient base optimisation method is used to optimally locate metallic welded reinforcements on flat plates. The method presented reduces the stress concentrations on the plate allowing the plate thickness to be reduced and resulting in considerable material savings. In [4] a genetic algorithm is used to solve a composite reinforcement problem on metallic plate. The reinforcement consist of a composite patch that is bonded to the plate. Their optimisation variables are the shape of the patch and the ply orientations of the composite.

K. Jármai et al use different optimisation techniques like leap-frog, LFOPC, Dynamic–Q, ETOPC, and particle swarm to improve the design of a cylindrical orthogonally stiffened shell member of an offshore fixed platform truss [5]. The design variables considered in the optimisation are the shell thickness as well as the dimensions of and numbers of stiffeners. Besides structural constraints the objective function also accounts for the manufacturing process. This study shows that significant cost savings can be achieved by orthogonal stiffening, since it allows for considerable reduction of the shell thickness.

In contrast to the cases described above, the present study is centred on the modification of already constructed plastic molds in order to produced reinforced plastic structures. This work applies to reticular structures and uses Genetic Algorithms (GA) to search for the optimum layout.

Genetic algorithms have become a popular computational methodology to solve problems involving the search for an optimum value. Due to their broad spectrum of application they are used in various science and engineering fields. Instead of using traditional strategies like calculus-based methods or enumerative methods, GA are stochastic search techniques based on the natural selection strategy where the fittest individuals survive over generations [6]. While traditional strategies are not very robust for complex problems, GA behave very well in this kind of situation [7]. Genetic Algorithms start with a population of random individuals, each one representing a possible solution to the specific problem. Individuals are usually codified as bit-strings called *chromosomes*. The fitness of the individual is a function of the chromosome and represents the objective function of the optimisation problem [8]. Every new generation is obtained through the selection of the fittest individuals and by applying crossover and mutation operators [9].

2 Description of the Problem

Reticular structures are presented in many plastic items, for example: milk crates, plastic boxes, and several kinds of platforms, as shown in figure (1).

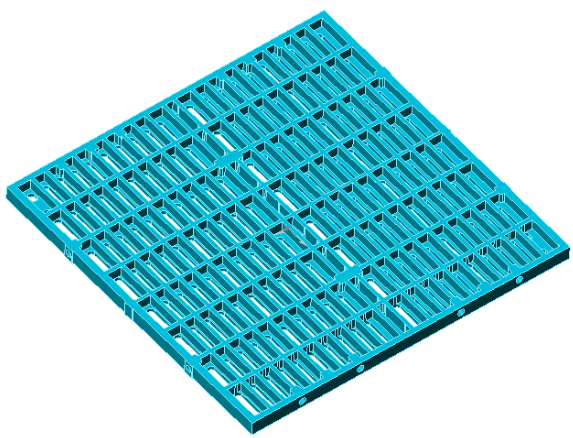


Figure 1: Reticular structure example

Many of these structures can present structural failures due to several factors including load conditions not considered during the design process. In order to improve the performance of these structures, a reinforced version of the original mold must be obtained. This re-manufacture of the original avoids manufacturing a new mold. A reinforcement can be obtained by adding material among the grids of the structure. This can be easily manufactured by grooving material from the original mold. Figure (2) shows an example of a reinforcement at one of the grid holes. The question that arises is: ¿where should be located the reinforcements in order to reduce at the maximum the stress and add a minimum amount of material?

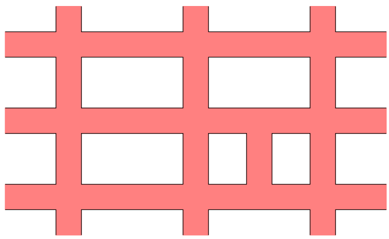


Figure 2: Schematic representation of a reinforcement at one of the grid holes of the reticular structure

3 Design Variables

As the geometry of the reticular structure can be simplified by a grid structure, each hole of the grid where a reinforcement can be placed is numbered and represents a design variable. In this way, a chromosome representing a reinforced structure is given by $s = (x_0, x_1, \ldots, x_n)$ where x_i can take values of one or zero depending on the existence or not of a reinforcement at position i of the grid.

The fitness or objective function is given by

$$\text{fitness}(s) = \alpha \left(\frac{\sigma(s_0)}{\sigma(s)} \right) + \beta \left(\frac{V(s_0)}{V(s)} \right),$$

where $\sigma(s_0)$ represents the maximum Von Mises stress of the original structure and $\sigma(s)$ is the maximum stress of the reinforced s structure. In the same way

$V(s_0)$ is the original volume and $V(s)$ the volume of the reinforced s structure. The variables α and β are weighted factors that control the incidence of stress and volume in the process.

4 Algorithm Structure

In order to evaluate the objective function, a computer program was written. It consists of four main modules: The first computes the geometry and boundary conditions of the structure given the bit-string s representing a reinforced structure. The output of this module is an FEA script. The second module performes a finite element analysis to compute the maximum Von Mises stress (beams elements) starting from a FEA script in BATCH mode. The third module computes the volume of the structure. These modules are then linked with a simple standard Genetic Algorithm code [7] [10] and the fitness function to complete the whole program as shown in algorithm (1).

The conversion between a chromosome and a coherent geometric model is accomplished, generating a matrix of points depending on the number of rows, columns, original dimensions of the structure, and holes. Knowing the relationships between the different generated points, the matrix is traversed in directions x and y linking related points with segments, having as criteria the index and neighbourhood positions of the generated points. The filling of each generated cell k with a reinforcement depends on the bit string position k in the chromosome. If a bit is equal to 1, a segment is generated between the corresponding nodes of each cell. An example of this procedure can be seen in figure (3). Once the geometry is constructed, the constraints and loads are applied to the model so the FEA script can be executed in order to obtain results.

4.1 Input and Output Parameters

The whole algorithm works on the following input parameters: weighted factors α for stress and β for volume on the fitness function, population size (number of individuals for each generation) n, number of generations N, crossover factor c and mutation factor m. These parameters determine the way the algorithm evolves as the optimal solution of the problem.

Algorithm 1 Genetic Algorithm

Main algorithm

Read input parameters from file (α, β, n, N, m, c)
Evaluate $\sigma(s_0)$ and $V(s_0)$
Initialise type chromosome *(Length, Objective Function)*
Initialise simple GA *(chromosome)*
Initialise simple GA parameters (n, N, m, c)
Evolve_simple_GA
Flush Result *()*
End Main

Evolve_simple_GA

$g \leftarrow 0$
Population $P(g)$
Evaluate_Fitness $P(g)$
while ($g <= final\, N$) **do**
  Recombine $P(g)$ to yield Population $F(g)$
  Evaluate_Fitness $F(g)$
  Select Population $P(g+1)$ from $P(g)$ and $F(g)$
  $g \leftarrow g+1$
end while
End

Evaluate_Fitness_Function

Generate geometry and FEA script for Individual s
Eval $\sigma(s)$
Eval individual $V(s)$
Eval fitness$(s) = \alpha \left(\frac{\sigma(s_0)}{\sigma(s)}\right) + \beta \left(\frac{V(s_0)}{V(s)}\right)$
End

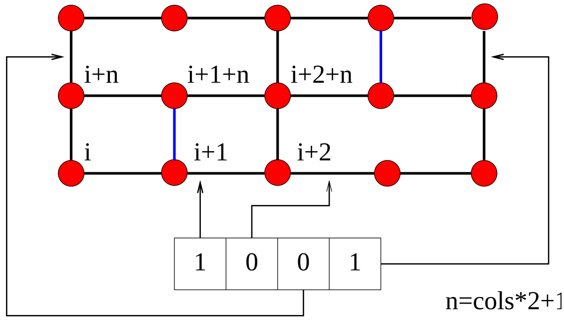


Figure 3: Reinforcement example from chromosome

Most of those parameters are selected by intuition and there is no strong supporting theory except for the selection of the population size. An estimate for the population size on GA requires the existence of a good amount of raw building blocks [7]. The building block hypothesis is based on the existence of schemata, a concept developed by Goldberg [7]. The difficulty of applying this theory to the present situation is due to the impossibility of selecting a matching schema that is desirable as a trait in individuals of the initial population. The final criteria for choosing the population size was based on several trials.

Once the algorithms reaches the ending criteria (number of generations N), it will flush the best individual for the given parameters, determining the reduced stress percentage σ_r, the increased volume percentage V_i, and the total number of reinforcement beams b. An example of the FEA result is shown in figure (4)

4.2 Finite Element Selection

The three–dimensional structure was simplified with beam elements. The finite element type for the FEA analysis is a 3–D beam uniaxial element with tension, compression, torsion, and bending capabilities. The element has six degrees of freedom at each node: translations in the nodal x, y, and z directions and rotations about the nodal x, y, and z axes. Stress stiffening and large deflection capabilities are included.

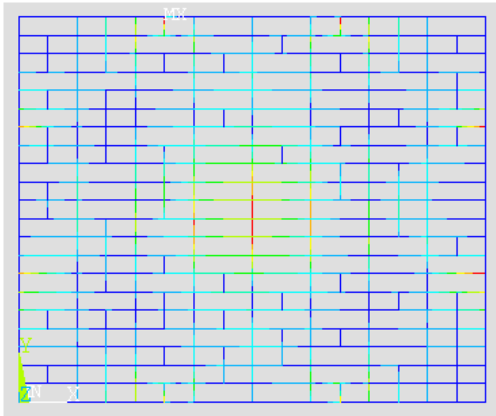


Figure 4: Von Mises stress of a reinforced structure

The reasons for choosing this type of beam element are its capabilities for supporting pressures on specific faces of the beam, the speed of analysis, and the accuracy of results for the problem that is being handled. Due to better time performance, beam elements where preferred over shell or brick elements. The FEA package for the analysis is a common commercial software with the capacity of running in batch mode without graphic user interface (GUI).

5 Results

The structure was supported from its four corners. In order to simulate a vertical load, equivalent to the weight of an object, a pressure was applied to the center of the structure as is shown in figure (5). The structure was made of injected polypropylene. The three–dimensional structure was simplified with beam elements as mentioned before. Due to a better time performance, beam elements where preferred over shell or brick elements.

Many different evolutionary parameters were used to test the program and the results for some of the best individuals can be seen in table (1). For the best individuals, weighed factors α of 0,6; 0,7 and 0,9; and β factors of 0,4; 0,3 and 0,1 were combined in different cases with populations of 12, 30

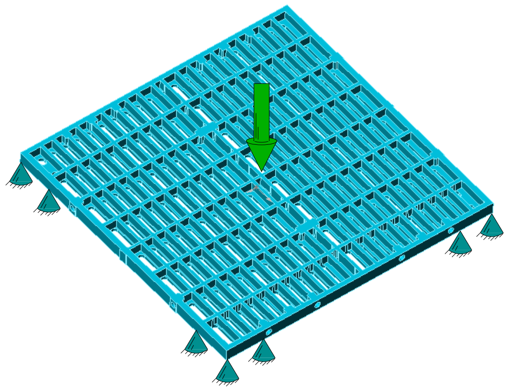


Figure 5: Boundary conditions of a reticular structure used in the analysis

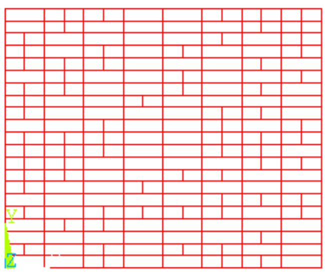
and 40. The number of generations was mantained constant at 100. Evolutionary operators of mutation m were considered in the range from 0,01 to 0,05 and crossover operator c in the range from 0,4 to 0,6.

Table 1: Results

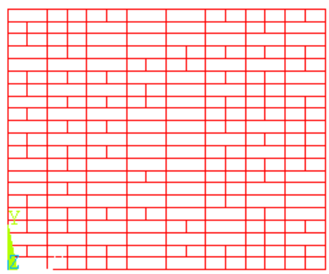
α	β	n	N	m	c	σ_r	V_i	b
0,7	0,3	12	100	0,01	0,5	13,88	7,98	77
0,7	0,3	12	100	0,01	0,4	10,48	8,17	79
0,8	0,2	40	100	0,01	0,5	13,97	7,79	75
0,9	0,1	30	100	0,05	0,6	13,42	8,55	83
0,9	0,1	30	100	0,03	0,6	12,59	7,79	75
0,7	0,3	30	100	0,04	0,5	11,30	7,31	70
0,9	0,1	30	100	0,04	0,6	14,33	8,55	83
0,6	0,4	40	100	0,03	0,5	14,70	8,36	81

From the input configurations and the output results, it can be concluded that small populations as well as larger ones can converge to a good solution. However, it was observed that populations converge faster to the solution when a large sample (large population) is used. Therefore, genetic operations will play an important but secondary role in the convergence of the problem because their performance is strongly affected by the initial random sample space. Large populations, even with the noise from the genetic operators, will have a higher chance to allocate better individuals. The best individual was obtained from a population size of 40 individuals, the highest population from all tested configurations. Greater population size could be tested, but computational resources were needed but not available.

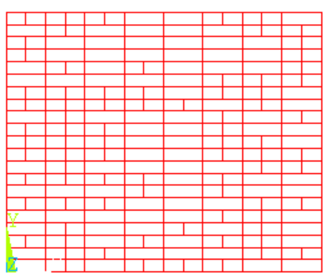
From the results, three individuals shown in figures 6(a), 6(b) and 6(c), arise as the best results between all the different tried configurations of the parameters. It can be seen in these figures that the center zone of the structures where the load is present, does not allow reinforcement. Therefore the candidate space for beam reinforcement is outside the centre where the load is applied.



(a) α 0,8; β 0,2; n 40; N 100; m 0,01; c 0,5 and output results σ_r 13,97; V_i 7,79; b 75



(b) α 0,9; β 0,1; n 30; N 100; m 0,04; c 0,6; and output results σ_r 14,33; V_i 8,55; b 83



(c) α 0,6; β 0,4; n 40; N 100; m 0,03; c 0,5 and output results σ_r 14,70; V_i 8,36; b 81

Figure 6: Different result obtained with different initial parameters

Some intuitive solutions were proposed and compared with the optimal solutions from the GA. The intuitive solution proposed to reinforce the structure by adding beams in the 4 center columns of holes as shown in figure 7. The comparison between solutions shows that the intuitive solution did not reduce the stress. Even worse, it increased the maximum stress in the order of 30%, and increasing the volume in the order of 8%. The intuitive solution is not always near the optimal solution.

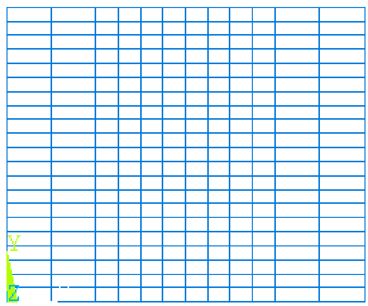
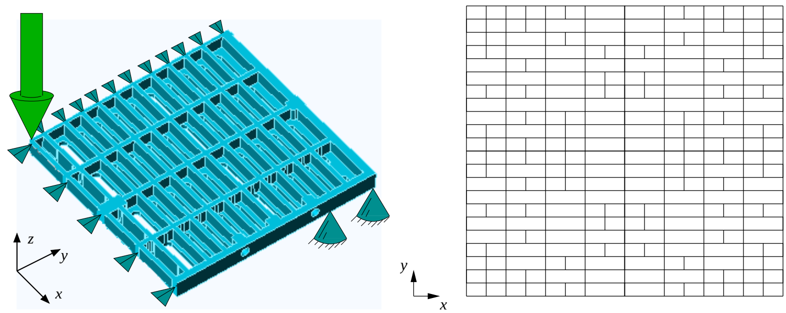


Figure 7: Intuitive solution with reinforcement in the four center columns of holes

Results produced an acceptable configuration of a reinforced structure consisting of 81 reinforcements located as shown in figure (6(a)). It reduced the initial maximum stress by 14,70% at a cost of increasing the volume by 8,36%.

6 Symmetry

The structure obtained by this method is not symmetric, thus reflecting the nature of the search algorithm. Given that the structure and loads are symmetric, it is possible that the solution found is near optimum. This problem can be handled by reducing the structural problem to a cuarter of its domain and apply simmetry boundary condition, see figure (8). Symmetry boundary condition is equivalent to restrict the movement in x and y directions for this case.



(a) (b)

Figure 8: Symmetry boundary condition for the plate and its solution

7 Conclusions

The combination of GA and FEA represents a powerful combination to solve engineering problems of complex systems. It was successfully applied to obtain the optimum reinforcements configuration of a plastic structure reducing the maximum stress of the original structure by 14,70% of its initial value and an increase of volume by 8,36%.

The GA does not allow beam reinforcement in the loaded zone of the structure, leaving the not loaded area for this purpose. Knowing about this can be useful to predict the performance of the algorithm and anticipate final solution.

In spite of the symmetry of the problem the Genetic Algorithm produces non symmetric solutions. However a way to enforce symmetry in the solutions is by reducing the structural problem to its minimum repetitive unit, in this case a quarter of structure, and apply the optimisation method. As a result faster and symmetric solutions can be obtained.

The right combination of initial genetic and evolution operators values can increase the result of the genetic search in a very significant way and therefore, further investigation should be undertaken. Also, because only single load conditions were used in this study, a multiple load condition could lead to a different configuration of the structure.

This is an excellent example of what kind of problems can be solved by GA. The solution can be easily codified and handled with GA characteristics and operators. As well, GA performs well on complex problems such as this, where other classic methods like calculus based and exhaustive methods may have difficulty due to the robustness of the problem.

References

[1] Baker A. *Bonded composite repair of fatigue-cracked primary aircraft structure*, Composite Structures, **47**, 431–443 (1999).

[2] Hollaway L and Leeming M. *Strengthening of reinforced concrete structures.* CRC Press, 1999.

[3] Manuel García. *Optimally reinforced plates.* MSc. Thesis, Universidad de los Andes, Bogotá, Colombia, 1990.

[4] Jean-Denis Mathias, Xavier Balandraud and Michel Grediac. *Applying a genetic algorithm to the optimization of composite pathches*, Computers and Structures, **84**, 823–834 (2006).

[5] K. Jarmai, J.A. Snyman and J. Farkas. *Minimum cost design of a welded orthogonally stiffened cylindrical shell*, Computers and Structures, **84**, 787–797 (2006).

[6] Mitsuo Gen and Runwei Cheng. *Genetic Algorithms and Engineering Design.* John Wiley & Sons, first edition, 1997.

[7] David E. Goldberg. *Genetic Algorithms in Search Optimization and Machine Learning.* Addison-Wesley Longman Publishing Co., first edition, 1989.

[8] Melanie Mitchell. *An Introduction to Genetic Algorithms.* The MIT Press, second edition, 1996.

[9] Stephanie Forrest. *Genetic algorithms: Principles of natural selection applied to computation*, Computing in Science, **261**, 872–878 (1993).

[10] Matthew Wall. *GAlib: A C++ Library of Genetic Algorithm Components.* Mechanical Engineering Department, MIT, 1996.