ISSN: 0123-3033 – ISSN-e: 2027-8284
INGENIERIA Y COMPETITIVIDAD
REVISTA CIENTÍFICA Y TECNOLÓGICA
Vol. 23 No. 1-2021 - DOI: 10.25100/iyc.23i1.7337

INGENIERÍA CIVIL

Optimización de armaduras espaciales de acero utilizando algoritmos genéticos auto-adaptados: una primera aproximación

CIVIL ENGINEERING

An approximation to the use of self-adaptive genetic algorithms in weight optimization of 3-D steel trusses

Jesús D. Villalba-Morales[1§], Sebastián Ramírez-Echeverry[1]

[1]Pontificia Universidad Javeriana, Facultad de Ingeniería, Departamento de Ingeniería Civil, Bogotá, Colombia

[§]*sebastian-ramirez@javeriana.edu.co, :jesus.villalba@javeriana.edu.co*

Recibido: 18 de marzo de 2020 – **Aceptado**: 3 de noviembre de 2020

Abstract

In the past few decades structural optimization through metaheuristics has gain recognition in the scientific community; non the less, to guarantee good results we require a good selection of metaheuristic's parameters. In this paper we propose a multi-chromosome genetic algorithm with self-adaptive parameters, to optimize steel trusses in a three-dimensional space. The design variable are the sections assign to each truss element of the structure. The optimization objective is the minimize the weight of the structure, considering the displacement y maximum stress as constrains. The propose algorithm was applied to the optimization of two trusses, obtaining designs that had a 35% less weight than the initial designs and comparable to results obtained in other papers. However, the adaptation of the parameters allows a more robust optimization process when analyzing different types of trusses and eliminates the initial runs of the optimization algorithm required to calibrate the initial parameters.

Keywords: *design of truss structures, genetic algorithms, parameters self-adaptation, special coding, structural optimization.*

Resumen

En las últimas décadas, la optimización estructural mediante metaheurísticas ganó acogida en la comunidad científica; sin embargo, para garantizar buenos resultados se requiere una correcta selección de los parámetros de la metaheurísticas. En este trabajo se propone un algoritmo genético multi-cromosoma auto-adaptado para optimizar armaduras de acero tridimensionales. Las variables de diseño corresponden a las secciones asignadas a cada elemento en la armadura. El objetivo es la minimización del peso de la armadura, considerando desplazamientos y esfuerzos máximos como restricciones. El algoritmo propuesto se aplicó a la optimización de dos armaduras, produciendo diseños que pesan hasta un 35% menos que el mejor diseño inicial y son valores comparables al resultado obtenidos en otros trabajos. No obstante, la adaptación de los parámetros permite mayor robustez cuando se desea optimizar diferentes tipos de armadura y evita las ejecuciones del algoritmo de optimización que son necesarias para la calibración de sus parámetros.

Palabras clave: *algoritmos genéticos, auto-adaptación de parámetros, diseño de armaduras tridimensionales, optimización de estructuras, representaciones especiales.*

.

1. Introduction

On a civil engineering project, a structural designer must guarantee that the structure must have an adequate behavior regarding the resistance and service, cost-effective and complies with the architectonic and construction requirements. To archive this, the engineer should go through a "manual process" that modifies iteratively the initial structural configuration, considering the stablished design codes and the experience of the designer. Typically, few iterations are made during the design process, therefor, the following question is valid: does the quantity of iterations made guaranties that the structural configuration is the lowest cost? A way to improve the quality of the design (in terms of cost-efficiency) consist in the automatization of the design process through the realization of the denominated Structural Optimization, which is divided in the following categories [(1,2):] I) element section optimization, II) material optimization, III) shape optimization and IV) topologic optimization.

The solution to the optimization problem requires the selection of a numerical technique capable of solving the current problem, in this case being the metaheuristics a group of optimization algorithms that has good performance [(3-7).] The metaheuristic is an approximate optimization algorithm used in the efficient determination of "adequate" solutions of difficult and complex problems in science and engineering, but it cannot guaranty an optimal solution [(8)]. One of the most used metaheuristics is the genetic algorithm technique (GAs), which is based in the Darwin's theory of natural selection and evolution of species. Various authors in literature have introduced modifications either to the structure of the algorithm or the representation of solutions with the objective of increasing the algorithm performance in the solution of the problem of optimizing trusses. One of them consist in the definition of group of elements assigned with the same cross-section [(3)] to reduce the variable of the problem (therefor, reducing the search space of the solution). Considering the computational cost, the usage of solution representation schemes different from that of the binary codification could reduce significantly the time needed for the optimization of structure with great numbers of variables [(4)]. Additionally, improvements in the convergence process can be achieved by changing the way on how the violation of constrains is handled [(5)], the usage of initial populations generated heuristically [(6)] and the modification of the genetic operators (like in the case of the usage of a probabilistic operator for the selection [(7)]), among other strategies. Non the less, the effect that the genetic parameters have over the optimization process of trusses have not been studied in depth, being that the values assumed on these parameters could affect de performance of the GA.

Generally, the selected GA parameters in truss optimization correspond to the values found by the researches that generated the best results after testing various parameter combinations. Non the less, there are 3 aspects that could indicate that the previous procedure is not the most adequate:

- The GA should be run for every parameter combination, which increases the number of evaluations of the objective function (at the same time, each evaluation of the objective function requires a structural analysis).
- A unique combination of genetic parameter is used for all analyzed trusses; although, there is no guarantee that the search space would be the same por each structure analyzed.
- The values are kept constant throughout the search process, but the dynamicity of the process could require assigning different values in function of the stages in which the optimization process is.

Therefore, it would be recommended the usage of GAs with low levels of dependency from the parameters used, being that it could be possible the application of control techniques for the parameters, like the adaptive or self-adaptation [9]. The present paper introduces a methodology for the three-dimensional steel trusses based on genetic algorithms, but that does not require the definition of genetic parameters from the user. In this manner the proposed methodology could be applied directly to any truss (under the constrain presented in this paper) without being necessary a previous calibration of the genetic paraments. Finally, from the obtained results it can be determined if the search space is unimodal or multimodal.

1.1. Genetic Algorithms

The GAs are a type of optimization algorithm proposed by John Holland on 1975 [10], the GAs evolve a population of possible solution through mathematical operations based on the principle of reproduction and survival of the fittest of Darwin.

The application of a GA to an optimization problem requires the definition of a representation scheme (binary, real, combination, among others) of the possible solutions. A possible solution is codified through a vector called chromosome, with each of the design variable corresponding to a section of this vector (gen). Then, it must be defined the genetic operators used in the transformation process (evolution) of the population and the genetic parameters that condition the application of the operators. The size of the population must be defined by the user.

The iterative process starts with either the heuristic or random generation of the initial population. The random generation requires that the process of generating random numbers is defined between the range of values of each variable, while the heuristic form implies that the knowledge of the characteristics of the problem must be used in the generation of the individuals of the population to accelerate the search process. Then, the genetic operations are applied in a consecutive manor: I) Selection, II) Reproduction and III) Mutation. The *Selection* choses the best individuals of the population considering the value of their fitness (measured in terms of the objective function). In the *Reproduction* o *Crossover* two new solutions are generated from the combination of two individuals chosen by the selection process. Finally, the *Mutation* alters randomly the characteristics of some individuals with the aim of introducing diversity to the population. The frequency on which the last two operators occurs must be limited through parameters know as *Crossover Rate* and *Mutation Rate.* Additionally, we can implement *Elitism* to preserve the best individual between two successive generations. The application of the previous operations creates a new population of individuals that is expected to be better than their predecessor. This process is repeated iteratively until a convergence criterion is reached (example, maximum number of generations). Given that the algorithm is stochastic, it is required that it must be executed multiple times before giving an answer to the user.

2. Methodology

In figure 1 the flowchart shows the proposed methodology for the optimization of the structural designs of the trusses**.** The process starts by reading the initial data of the trusses (node location, element description and materials, loads, etc.). Next, the initial population of solutions is generated, where each of the individuals corresponds to a determined structural design, by assigning randomly to each element (or group of elements) a cross-section from an available set of options. Then a finite element analysis is run for each structural configuration to determine the reactions of the structure (in terms of element's stresses and displacements in the nodes) under the effect of solicited loads. Given the previous information the following steps carry out: I) Evaluation of the self-weight of each individual

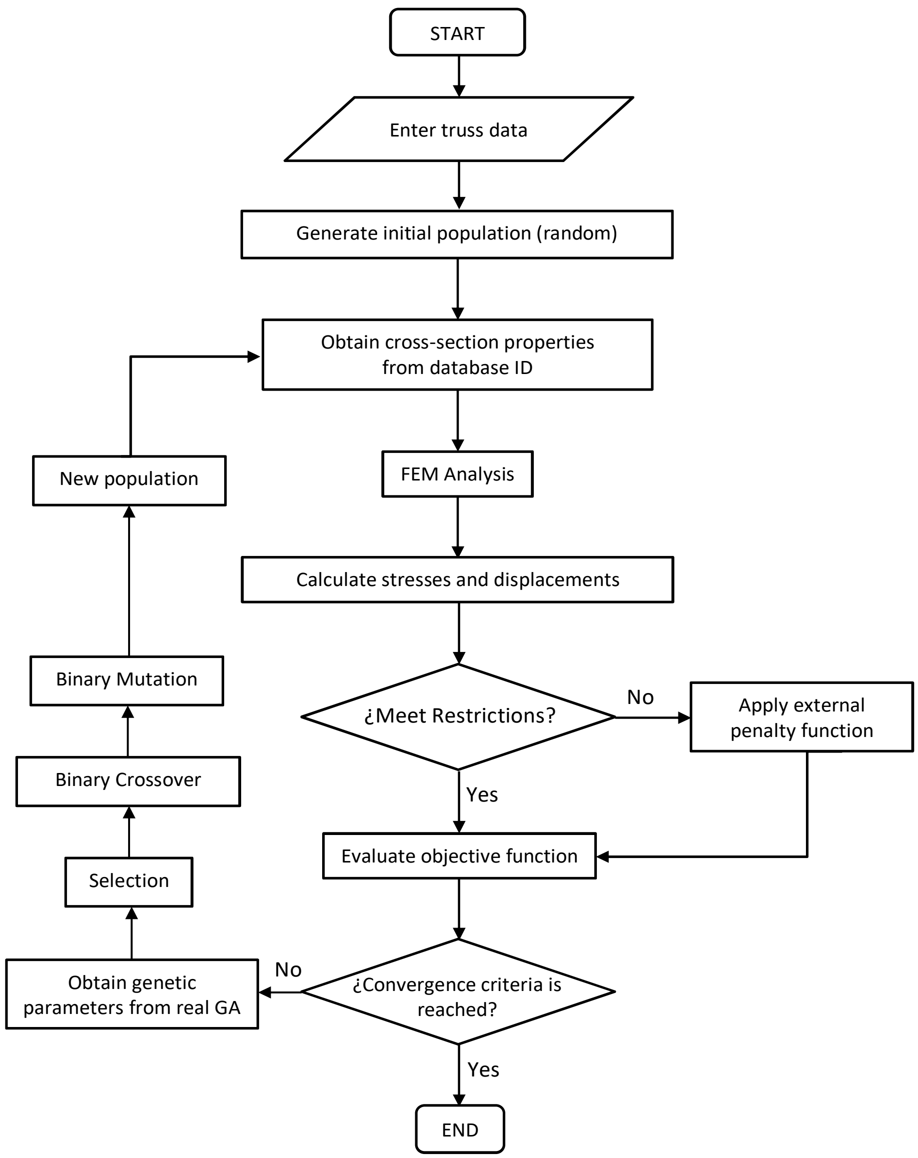

Figure 1. *General flowchart for the 3D optimization of trusses*

(structure), II) Verification of the constrain set in section 2.1, to establish if the design is viable or not, and III) Penalty of the cost of the solution following the criteria given in section 2.3, if the structural configuration violates any constrain. On this part, the criteria to stop the algorithm are evaluated, which are: I) a maximum number of allowed iterations and II) specific quantity of iteration where the *fitness* of the best induvial does not considerably change. If one of the criteria does not comply, a new solution is created through the operators detailed in section 2.2; on the contrary, the algorithm stops it execution and shows most cost-effective structural design found.

2.1 Problem Formulation for the Structural Optimization.

In this section the characteristics of the problem in study are defined in terms of design variables and parameters, objective function and constrains. In that sense, the variable of the design problem corresponds to the properties of the cross sections of the elements (area), meanwhile the properties of the steel and the geometric configuration are kept constant during the evolutionary process. The objective function is configured in terms of the self-weight of the structure *(W)*, shown in Ec.1, and the constrains established in Ec. 2 to 4:

Minimize: $W = \rho \sum_{i=1}^{ng} l_i A_i \quad i = 1, 2, \ldots ng$ Ec.1

Subject to: $\delta_j(A_i) \leq \delta_{max} \quad j = 1, 2, \ldots nj$ Ec.2

$\sigma_k(A_i) \leq \sigma_{adm} \quad k = 1, 2, \ldots nm$ Ec.3

$A_{il} \leq A_i \leq A_{iu}$ Ec.4

being, ρ the unit weight of steel (0.28 lb/in3), l_i and A_i the length and the cross-section area the element's in group i, respectively, ng number of groups, δ_j node displacement, j, δ_{max} maximum allowed displacement, nj number of nodes, σ_k maximum stress on the element k, σ_{adm} maximum allowed stress on the element and nm number of elements in the structure. A_{iu} y A_{il} are the minimum and maximum values that the variables can take. Due to constructive constrains, it is not possible to assign an unique cross-section to each individual element in the structure, because of this, groups of elements are assigned with the same cross-section, producing one design variable per group. The reduction in quantity of design variables also leads to a reduction in the search space to consider less possible solutions.

The algorithm was implemented on the integrated development environment Eclipse®, version Juno Service Release 2 complimented with JDK 1.7. (Java Development Kit), that includes the tools to develop, debug and monitor JAVA® applications.

It is important to consider that the mathematical formulation defined in the paper was used with the objective of comparing the performance of the methodology proposed with other algorithm in the literature. Non the less, the model of physical constrain is limited due to the exclusion of important phenomena in the steel structural design like the susceptivity to buckling.

2.2 Configuration of Multi-chromosome Genetic Algorithm.

A multi-chromosome GA (MCGA) is proposed to find the solution to the optimization problem established in section 2.1. The algorithm uses an especial representation for the individuals (figure 2), which are constituted by two chromosomes: (I) one *binary chromosome* with Gray codification to represent the solution of the problem, in other words, a combination of sections and II) a *real chromosome* that allows to self-adapt the genetic parameters during the evolutionary process. Said representation allows to simultaneously establish adequate values for the cross-section of the structure's elements and the genetic parameters required in each stage of