International Journal of Current Engineering and Technology
ISSN 2277 - 4106

Available at http://inpressco.com/category/ijcet

Research Article

Optimum Design of Welded Steel Plate Girder using Genetic Algorithms

Vinay Agrawal[a], Vinay Chandwani[b*] and Arpit Porwal[a]

[a]Department of Civil Engineering, Malaviya National Institute of Technology Jaipur, Rajasthan, India
[b]Investigation Design & Research, Water Resources Unit, Government of Rajasthan, Jaipur, Rajasthan, India

Accepted 25 August 2013, Available online 01 October 2013, **Vol.3, No.4 (October 2013)**

Abstract

The design of structures whether reinforced concrete or steel are governed by code provisions. In structural engineering the primary goal of optimization lies in either minimizing the weight of the structure or reducing its cost, keeping in view the overall safety and serviceability aspects. Engineering design is influenced by several parameters and associated constraints hence while designing a structure numerous solutions are obtained, thereby making it difficult to choose an optimal solution. The essence of Darwin's principle of "Survival of the Fittest", has led to the development of genetic algorithms which are nowadays being used as global optimization tools for identifying optimal or near optimal solutions for a given problem. Genetic Algorithms based on the mechanism of natural evolution, consider multiple design solutions at a time and cleverly explore the solution space to achieve a global minima or maxima. Genetic Algorithms work well than Classical optimization algorithms as these do not require the gradient information for the function to be optimized. Genetic algorithm based optimization approach works well on discrete, continuous or mixed search spaces, thereby making it versatile for engineering optimisation. The paper presents the capability of genetic algorithms as a directed search technique for optimum design of welded plate girder governed by the mixed nature of design variables.

Keywords: *Welded Plate Girder, Genetic Algorithms, Structural Optimization, Objective Function, Mutation, Crossover.*

1. Introduction

Optimization in the field of engineering not only aims at feasible design but also caters to the requirements of the design objective. The ultimate goal of optimization is to maximize the benefits. It can be defined as theory of finding the best solution from a collection of alternatives by means of maximizing or minimizing a certain mathematical function. The process of optimization lies at the root of engineering, since the classical function of the engineer is to design new, better, more efficient and less expensive systems as well as to devise plans and procedures for the improved operation of existing systems (Ravindran A. *et al*, 2006). Structural optimization problems are characterized by various objective and constraint functions, which are generally non-linear functions of the design variables (Lagaros N.D. *et al*, 2002). The nature and type of such problems have posed challenges and has led to the development of computational intelligence.

In the field of computational intelligence, the natural phenomenon is used for developing tools for solving the problems which are normally difficult to be solved using traditional means of computing. It can described as the simulation and emulation of nature by means of computing aimed at creating patterns, forms, behaviors, and organisms that (do not necessarily) resemble 'life-as-we-know-it' (Castro L.N., 2007).

Evolutionary Computation is a branch of Computational Intelligence whose roots lie in the principle of natural evolution. Based on the Darwin's principle "Survival of the fittest", the organisms which are more capable of modifying themselves according to the changing environments, will have more decedents or off springs in the future. Evolutionary algorithms are a class of non-gradient population-based algorithms used in many areas of engineering optimization (Hare W. *et al, 2013*). One of the evolutionary computation techniques is the Genetic Algorithms (GA). GAs are considered directed search algorithms based on the mechanics of biological evolution (Akerker and Sajja, 2010). GAs can rapidly identify discrete regions within a large search space to concentrate searching for the optimum solution (Rafiq M.Y., 1995).

The central theme of research on GA has been the robustness, the balance between efficiency and efficacy necessary for survival in many different environments (Goldberg D.E., 2009). From the initial population, a new population is produced by eliminating the weaker population. Hence as we proceed from one generation to the next and so forth, the quality of the solution increases. It is an iterative procedure maintaining a population of structures that are candidate solutions to specific domain challenges (Oke S.A., 2008). The solutions to problems

*Corresponding author: **VinayChandwani**

encountered in optimization are generated by GA using the evolution inspired techniques, such as reproduction, mutation and crossover.

Structural Engineering being a creative field involves designing of structures which should be resistant to applied forces and fulfill the serviceability requirements. Also in the era of limited resources the economics of the structure is nowadays being addressed while designing a structure. In simple terms engineering optimization can be defined as a procedure to maximize the benefits keeping in view the design requirements. The optimization problems related to engineering field are governed by various design variables. The design variables are either continuous, discrete or mixed nature. Also the constraints employed in engineering problems are of a complex nature. The engineers thus have to choose among the bulk of design outcomes to arrive at a structurally and economically feasible design. The combined effect increases the complexity of optimization in the engineering field. The GA provides the necessary solution yielding the global minimum or maximum values of multidimensional and complicated functions (Michalewicz Z., 1992).Almost all conventional optimization techniques search from a single point but GAs always operate on a whole population of points (strings) i.e., GA uses population of solutions rather than a single solution for searching (Sivanandan and Deepa, 2008).

In case of steel structures, the optimization problem is centered on the reducing overall weight of the structure because this reduces the amount of material used for construction and eventually reduces the overall construction cost. The primary goal of optimization in steel structures is to choose the minimum weight structure among the potential design outcomes. GA being a stochastic search tool is able to locate the optimal solution which can satisfy the code provisions and caters to the objective of minimum weight. The present research paper deals with the optimal design of Steel Plate Girders using GA.

2. Literature Review

Classical optimization algorithms are based on steepest gradient descent approach and are designed for continuous nature of variables. On the other hand GA based optimization approach can work well on discrete, continuous or mixed search spaces. Most of the engineering optimization problems require discrete variables. Discrete optimization of structures using GAs has been performed by many researchers including Jenkins (1991, 1992), Rajeev and Krishnamoorthy (1992, 1998), Koumousis and Arsenis (1994), Lin and Haleja (1992), Wu and Chow (1995), Camp *et al* (1998), Erbatur *et al* (2000) and Lee and Ahn (2003).

Razani and Goble (1966) were the first to attempt cost optimization of steel girders. Holt and Heithecker (1969) studied the minimum weight design of symmetrical welded plate girders without web stiffeners. Annamalai *et al* (1972) studied cost optimization of simply supported, arbitrarily loaded, welded plate girders with transverse stiffeners. Anderson and Chong (1986) presented the minimum cost design of homogeneous and hybrid stiffened steel plate girders.

The first application of GA for structural engineering was carried out by Goldberg and Samtani (1986). Minimum cost design of composite continuous welded plate girders were presented by them. Ghanem, H.M.F. *et al* (2003) used the GA optimization technique for behavior and strength of built up plate girders subjected to localized edge loading in the plane of the web. Fu, K. *et al* (2005) used GA with elitism for optimum design of welded steel plate girder used for a single-span bridge and a two span continuous bridge.

3. Genetic Algorithm as Optimization Tool

The real world problems generally have many feasible solutions. The difficulty in choosing the best solution among the feasible alternatives has compelled the human civilization to learn from nature. Analogous to "Survival of the Fittest", the GAs intelligently searches the solution space and brings forth the best potential solution by eliminating the weaker one. Due to their parallel nature, GAs are able to produce multiple solutions to the problem governed by various parameters at a time. The merit of feasibility, or how good the potential solution solves the given problem, is being introduced in a form of the fitness function which then acts as an evolutionary guide (Stanković T. *et al*, 2006). According to Goldberg (2005), fitness function can be described as "some measure of profit, utility, or goodness that we want to maximize." When GAs are used for optimization problems, the fitness function is used to evaluate the degree of closeness between a typical solution and an ideal solution. GA acts as good optimization tools as long as the fitness function is properly defined to solve the problem. Matous *et al* (2000) have shown that even a simple GA can find its applicability in the design of engineering structures, where traditional gradient schemes become unacceptably expensive as they require an execution for a large number of different points to increase the chance in locating the global optimum, or cannot be used at all due to discontinuous nature of the objective function.

In optimization problems the different potential solutions to a problem are compared and contrasted to choose the best quality solution. Due to its parallel nature the GA searches the solution space in all directions at a time for the best value of the fitness function thereby reducing the computational effort and the possibility of getting trapped in local minima. The search space is gradually narrowed down thereby improving the probability of finding an optimal solution. Application of GA for the study of optimization techniques has achieved interest worldwide because of their appropriateness in determining the global optimum. A number of studies done in the past clearly reveal the various advantages of GA for the optimization of structural design process and also have concluded that GA has successfully proven to be an effective tool in determining the optimal value for the design problems that involves too many constraints and variables.

3. Parameters of Genetic Algorithm

GAs are population based stochastic search techniques which rely on determination of an objective or fitness function for the possible solutions to survive or reproduce. Each possible solution is termed as a “chromosome”. The GA maintains a population of “chromosomes”. During each iteration of GA the chromosomes are altered using the tools of biological evolution viz., selection, mutation and crossover. As Holland and Miller (1991) note, this three-part process of reproduction, crossover, and mutation may seem to be nothing more than a random search algorithm which retains the best potential solutions. The value of the fitness function evaluates the extent to which the best solution will be represented in the next search process. Mutation is traditionally seen as a “background” operator (Booker, 1987 & DeJong, 1985). It adds random diversity in the populations and helps the algorithm to attain a larger exploratory space. Too high mutation rate increases the search space to a level that convergence or finding global optima becomes a difficult issue. Whereas a lower mutation rate drastically reduces the search space and eventually leads GA to get stuck in a local optimum. When used sparingly with reproduction and crossover, it is an insurance policy against premature loss of important notions (Goldberg D.E., 2009). Schaffer *et al* (1989) studied the effects of some control parameters of GA on its performance. The computational results reveal that the selection and mutation operators together are very significant in the behavior of GA. Certain genetic algorithm parameter settings viz., population size, generations, crossover type, crossover rate, mutation types and mutation rate have been suggested by Dejong and Spears (1990). In the present paper an initial population size comprising of 20 chromosomes is selected with mutation rate and crossover fraction as 1 % and 0.9 respectively.

4. Steel Plate Girders and their Optimal Design

Plate Girders are constructed by riveting or welding the plates to form an I-section. Usually Plate Girders are used for heavy loads and large spans. Since they are constructed using plates, hence these can be fabricated or tailor made to suit the designed load. This in a way offers greater freedom and convenience to the designer to choose from the various sections of the plates available in the market. The optimal design of plate girders is governed by serviceability, flexural strength, shear strength and above all the weight of the structure. The optimization of steel structures can thus be formulated as a weight minimization problem keeping in view the serviceability, flexural and shear strength aspects as suggested by the design codes. The plate girders comprise primarily of two flange plates and one web plate. The components of a simple plate girder are shown in Fig. 1. The plates are available in the market having discrete thicknesses. This makes the job of the designer all the more difficult, because the thickness and size of plates are to be chosen to satisfy the design code provisions and to minimize the overall weight of the

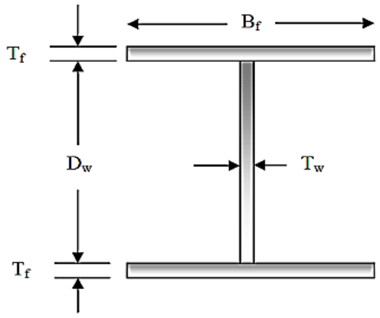

Fig 1 Components of a Plate Girder

structure. The complexity of the problem wherein a number of solutions are evolved can be solved using GA by defining an objective function along with the imposed constraints. Fig 2 shows the flow chart of Structural design optimization using GA.

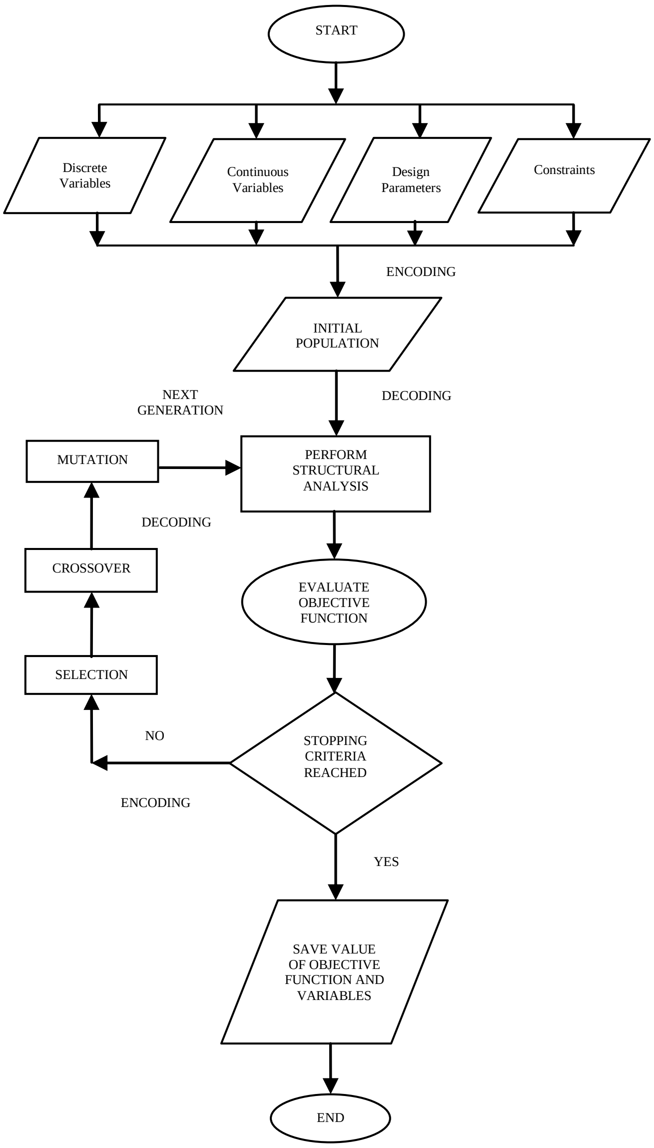

Fig 2 Flow Chart of Structural Design Optimization using Genetic Algorithm

The objective function can be defined as

$$Minimize\ f(x) = [2\ (T_f B_f) + (D_w T_w)]\ 7850\ L/10^6 \qquad (1)$$

Where f(x) is the weight of the plate girder in kN, T_f and B_f is the thickness of flange and width of flange, T_w and D_w are the thickness of web and depth of web respectively in mm and L is the length of the girder in metres. Once objective function has been formulated, the design constraints and design parameters have to be imposed as code provisions. For design of steel structures IS 800:2007 design code has been referred. The design parameters primarily comprise of strength to resist bending moment, shear force and the serviceability criteria and design constraints include thickness of web, flange and depth of girder.

The various design parameters and constraints employed are:

(a) Factored design moment M at any section due to external loads must be less than the designed bending strength of the section $M \leq M_d$ (2)
Design bending strength $M_d = \beta_b Z_p f_y/\gamma_{m0}$ (3)
Where β_b = 1.0 for plastic and compact sections,
Z_p = plastic section modulii of the cross-section,
f_y = yield stress of the material, and
γ_{m0} = partial safety factor.
For a uniformly distributed load factored design bending moment at mid section $M = w_f L^2/8$ (4)
Where w_f is the factored load and L is the length of the plate girder.

(b) Factored design shear force V due to external loads shall satisfy $V \leq V_d$ (5)
Where V_d is the design strength given by
$V_d = V_n/\gamma_{m0}$ (6)
Where V_n is the nominal plastic shear strength given by $V_p = A_v f_{yw}/\sqrt{3}$ (7)
A_v = shear area and f_{yw} = yield strength of web.

(c) Optimum depth of plate girder: $d \geq k\ (M/f_y)^{0.33}$ (8)
k= 5 for riveted girder.

(d) Check for serviceability: $d/T_w \leq 200\varepsilon$ (9)
Where $\varepsilon = \sqrt{(250/f_y)}$

(e) To avoid flange buckling: $d/T_w \leq 345\varepsilon_f^2$ (10)
Where $\varepsilon_f = \sqrt{(250/f_{yf})}$, f_{yf} is the yield stress of the compression flange.

(f) To resist shear buckling: $d/T_w > 67\varepsilon$ (11)

(g) Limiting width to thickness ratio: $B_f/T_f < 9.4\varepsilon$ (12)

The variables in the constraints e.g. depth of web D_w, width of flange B_f, thickness of web T_w and thickness of flange T_f are discrete in nature as these are dependent on certain sizes of steel sections available in the market. An optimum design of plate girder envisages the use of the discrete variable to arrive at a safe and economically feasible section. For evaluating the usefulness of GA in arriving at an optimal design of plate girder, the results of GA are compared with conventional design. The results are shown in Table 1. A graphical comparison of the results is shown in Fig. 3 and Fig. 4.

Table 1 Comparison of weight of plate girder obtained using Genetic Algorithm and conventional design.

Length (m)	5.0		10.0	
Loading (kN/m)	Weight (kN)		Weight (kN)	
	GA	Conventional Design	GA	Conventional Design
10	86.25	94.20	486.70	549.50
15	117.75	120.49	588.75	612.30
20	149.15	152.09	659.40	690.80
25	149.15	157.00	769.30	800.70
30	180.55	174.66	890.97	930.22
35	188.40	194.28	993.02	1012.65
40	227.65	235.50	1012.65	1032.27
45	227.65	243.35	1138.25	1177.50
50	235.50	266.90	1287.40	1310.95
55	266.90	298.30	1310.95	1358.05

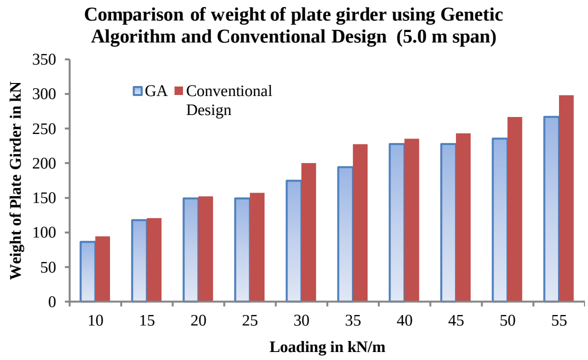

Fig 3 Plate Girder Weight comparison for 5.0 m span

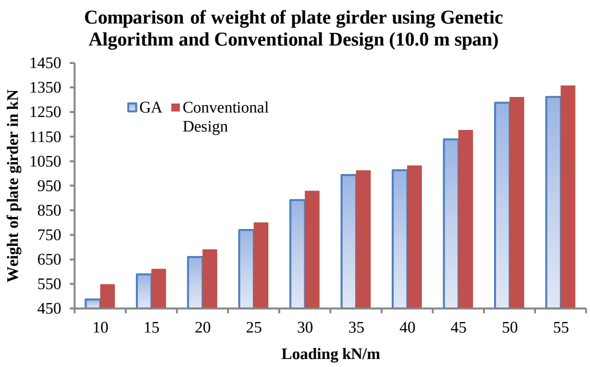

Fig 4 Plate Girder Weight comparison for 10.0 m span

Conclusions

Limited resources and time constraints have compelled humans to search for nature inspired computational tools. Computational intelligence uses the computational power of the modern day computers to solve problems which are difficult to solve using traditional means of computing. The research paper has presented the use of one of the computational intelligence tool, the Genetic Algorithm, in solving the problems encountered in structural engineering.

The structural design is influenced by various design parameters and constraints imposed by the design codes. Moreover the discrete sizes of the steel sections available in the market, makes the optimal design of structural members very complex. The results show that the GA has a convincing promise of finding the optimal solution to a problem influenced by complex parameters. GA through its random search inspired by the theory of evolution is able to find the optimal section of the structural member conforming to the code requirements. The optimized weights of the plate girder obtained through GA technique are considerably less in comparison to those evaluated using conventional design. A minimum of 8.5 % and a maximum of 10.5 % reduction in weight of plate girder are encountered when the design is done using GA. Thus holistic incorporation of GA in structural design, will not only lead to economical and structurally feasible design but offers a designer a flexible and very simple automated design optimization technique which can be used readily for solving complex problems.

References

Akerker, R., Sajja, P. (2010), Knowledge- Based Systems, Jones and Bartlett Publishers, LLC, ISBN-13: 978-0-7637-7647-3, ISBN-10: 0-7637-7647-5.

Anderson, K.E., Chong, K.P. (1986), Least cost computer-aided design of steel girders, *Engineering Journal*, AISC, Fourth Quarter, 151-156.

Annamalai, N., Lewis, A., Goldberg, J. (1972), Cost optimization of welded plate girders, 98 (ST10), 2235-2246.

Camp, C.V., Pezeshk, S., Cao, G. (1998), Optimized Design of Two-dimensional Structures using a Genetic Algorithm, *Journal of Structural Engineering*, 124(5),551-559.

Castro, L.N. (2007), Fundamentals of natural computing: an overview, *Physics of Life Reviews*, 4, 1-36.

DeJong, K.A., Spears, W.M. (1990), An Analysis of the Interacting Roles of Population Size and Crossover in Genetic Algorithms, *Proceedings First Workshop Parallel Problem Solving from Nature*, Springer-Verlag, Berlin, 38-47.

Erbatur, F., Hasancebi, O., Tutuncil, I., Kihc, H. (2000), Optimal design of planar and space structures with genetic algorithms, *Computers and Structures*, 74, 209-224.

Fu, K., Zhai, Y., Zhou, S. (2005), Optimum Design of Welded Steel Plate Girder Bridges Using a Genetic Algorithm with Elitism, *Journal of Bridge Engineering*, 10(3), 291–301.

Ghanem, H.M.F, Ibrahim, A.A.M., Diwan, M.A.F., Shawki, K.M. (2003), Limit analysis of ultimate strength of I sections subjected to edge loading using Genetic Algorithms, *Alexandria Engineering Journal*, 42(1), 53-63.

Goldberg, D.E. (2009), Genetic algorithms in search, optimization and machine learning, Fourth Edition, Pearson Education Inc.

Goldberg, D.E., Samtani, M.P. (1986), Engineering optimization via genetic algorithm, *Proc., 9th Conf. Electronic Computation*, ASCE, 471-482.

Hare, W., Nutini, J., Tesfamariam, S. (2013), A survey of non-gradient optimization methods in structural engineering, *Advances in Engineering Software*, 59, 19-28.

Holland, J. H., Miller, J.H. (1991), Artificial Adaptive Agents in Economic Theory, *American Economic Review*, 81, 365-370.

Holt, E., Heithecker, G. (1969), Minimum weight proportions for steel girders, *J. Struct.Div*, ASCE, 2205-2217.

IS 800:2007, General Construction in Steel – Code of Practice, Third Revision, Bureau of Indian Standards.

Jenkins, W.M. (1991), Structural optimization with genetic algorithm, *Journal of Structural Engineering*, 69(24), 418-422.

Jenkins, W.M. (1992), Plane frame optimum design environment based on genetic algorithm, *Journal of Structural Engineering*, 118(11), 3103-3112.

Koumousis, V.K., Arsenis, S.J. (1994), Genetic algorithms in a multi-criterion optimal detailing of reinforced concrete members, *Advances in structural optimization*, CIVIL-COMP Ltd, Edinburgh, Scotland, 233-240.

Lagaros, N.D., Papadrakakis, M., Kokossalakis, G. (2002), Structural optimization using evolutionary algorithms, *Computers and Structures*, 80, 571-589.

Lee, C., Ahn, J. (2003), Flexural design of reinforced concrete frames by genetic algorithms, *Journal of Structural Engineering*, 129, 762-774.

Lin, C.Y., Haleja, P. (1992), Genetic algorithms in optimization problems with discrete and integer design variables, *Eng. Optimization*, 19(4), 309-327.

Matous, K., Leps, M., Zeman, J., Sejnoha, M. (2000), Applying genetic algorithms to selected topics commonly encountered in engineering practice, *Computer Methods in Applied Mechanics and Engineering*, 190, 1629-1650.

Michalewicz, Z. (1992), Genetic Algorithms + Data Structures = Evolution programs, Springer-Verlag, Berlin

Oke, S.A. (2008), A Literature Review on Artificial Intelligence, *International Journal of Information and Management Sciences*, 19(4), 535-570.

Rafiq, M.Y. (1995). "Genetic Algorithms in optimum design, capacity check and final detailing of reinforced concrete columns." Transactions on Built Environment, 13, WIT Press.

Rajeev, S., Krishnamoorthy, C.S. (1992), Discrete optimization of structures using genetic algorithms, *Journal of Structural Engineering*, 118(5), 1233-1250.

Rajeev, S., Krishnamoorthy, C.S. (1998), Genetic algorithm-based methodology for design optimization of reinforced concrete frames, *Computer-Aided Civil Infrastruct. Eng.*, 13, 63-74.

Ravindran, A., Ragsdell, K.M., Reklaitis, G.V. (2006), Engineering Optimization: Methods and Applications, John Wiley & Sons Inc, Hoboken, New Jersey, USA.

Razani, R., Goble, G.G. (1966), Optimum design of constant depth plate girders, *J. Struct. Div*, ASCE, 92, ST 2.

Sivanandan, S.N., Deepa, S.N. (2008), Introduction to Genetic Algorithms, Springer-Verlag, Berlin, Heidelberg.

Stankovic, T., Stosic, M., Marjanovic, D., Evolutionary Algorithms in Design, *International Design Conference, DESIGN 2006*, Durbrovnik, Croatia, May 15-18, 2006.

Wu, S.J., Chow, P.T. (1995), Steady-state genetic algorithms for discrete optimization of trusses, *Computers and Structures*, 56(6), 979-991.